



UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO

FACULTAD DE CIENCIAS

SIMULACIÓN DE MAPAS DINÁMICOS DE PAULI
DE UN QUBIT EN COMPUTADORAS CUÁNTICAS

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

FÍSICO

P R E S E N T A :

TOMÁS RICARDO BASILE ÁLVAREZ

TUTOR:

DR CARLOS FRANCISCO PINEDA ZORRILLA



CIUDAD UNIVERSITARIA, CDMX, 2023



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas

Tesis Digitales

Restricciones de uso

DERECHOS RESERVADOS ©

PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Hoja de datos del jurado

1. Datos del alumno

Basile
Álvarez
Tomás Ricardo
55 4354 0104
Universidad Nacional Autónoma de México
Facultad de Ciencias
Física
316617194

2. Datos del tutor

Dr.
Carlos Francisco
Pineda
Zorrilla

3. Datos del sinodal 1

Dr.
Alfred Barry
U'Ren
Cortés

4. Datos del sinodal 2

Dr.
Daniel
Finkelstein
Shapiro

5. Datos del sinodal 3

Dr.
Daniel
Sahagún
Sánchez

6. Datos del sinodal 4

Dr.
Santiago Francisco
Caballero
Benítez

7. Datos del trabajo escrito

Simulación de Mapas Dinámicos
de Pauli de un Qubit en Computadoras Cuánticas
76 p
2023

Tengo tiempo para saber si lo que sueño concluye en algo.
-Luis Alberto Spinetta

Dedicatoria

A a mis padres Ángeles y Roberto, a quienes les agradezco enormemente su apoyo incondicional, su compañía, consejos y ayuda. A mis hermanos Andrés, Martín, Pablo y Ale por acompañarme y hacer del día a día algo más ligero y divertido. A mis abuelos, que aunque ya no se encuentren conmigo, siempre me apoyaron y espero se sientan orgullosos de mí.

A mis amigos de la secundaria y preparatoria, que me han acompañado por muchos años: Andru, Max, Pedro, Alex, Daniel, Mariana, Simonee, Rebe, Moni, Ely, Zully, Montse, Pau, Katia, Aranza y muchos más. A mis amigos de la carrera Rebeca y León, que me apoyaron mucho durante este reto.

Finalmente, a mi novia Jessie, quien me acompañó, escuchó, aconsejó y me dio infinitas razones para ser feliz durante estos años.

Agradecimientos

Gracias a la UNAM por la formación académica recibida y por el ejemplo que brindaron durante la carrera los profesores, especialmente Ángel Sánchez Cecilio, Alfredo López, Patricia Goldstein, Paola Arévalo, Luis Gómez, Daniel Ávila, Erick Leonardo Patiño, Roxana del Castillo, Jorge Fujioka y Daniel Sahagún. Agradezco también a la Preparatoria LaSalle por la formación ofrecida y los excelentes profesores Juan José Sánchez, Martha Borrego, Circe González, Aarón Domínguez y especialmente a Cuauhtli Martínez por darme el primer verdadero acercamiento a la física. Gracias a todos por su labor y el apoyo brindado.

Gracias a mi asesor, el Dr. Carlos Pineda, por su apoyo, paciencia, recomendaciones, por invitarme a su grupo y por enseñarme cómo se realiza la investigación.

Finalmente, gracias al apoyo del proyecto PAPIIT AG101421, que facilitó el desarrollo de esta tesis.

Resumen

En este trabajo estudiamos la dinámica de sistemas cuánticos abiertos y en particular nos fijamos en un tipo de proceso sobre un qubit conocido como mapa dinámico de Pauli. Estos procesos aplican al qubit una combinación convexa de las matrices de Pauli que depende de un parámetro continuo que denotamos por p .

Hemos propuesto un algoritmo simple que permite implementar estos mapas de Pauli sobre un qubit en una computadora cuántica. Encontramos que un mapa de Pauli arbitrario se puede realizar usando este algoritmo, el cual incluye dos qubits de ancilla y requiere de tres compuertas de rotación que dependen del parámetro p . Luego nos fijamos en las condiciones necesarias para poder implementar un mapa dinámico usando solamente una rotación parametrizada en lugar de tres y encontramos que todos estos mapas tienen una forma matemática sencilla de describir. Finalmente, realizamos estos algoritmos en sistemas cuánticos reales utilizando las computadoras cuánticas de IBM.

Índice general

Índice de figuras	xI
1. Introducción	1
2. Marco teórico	3
2.1. Postulados de la mecánica cuántica	3
2.2. Qubit	4
2.3. Operaciones	5
2.4. Sistemas de dos qubits	9
2.5. Circuitos cuánticos	9
2.5.1. Compuertas sobre múltiples qubits	10
2.5.1.1. Compuertas Controladas	11
2.5.1.2. Compuertas generales sobre múltiples qubits	12
2.5.2. Aclaración sobre la notación	12
2.6. Matrices de densidad	14
2.6.1. Representación en la Esfera de Bloch	16
2.6.2. Evolución temporal de la matriz de densidad	18
2.6.3. Postulados en términos de matrices de densidad	19
2.7. Sistema cuántico abierto	20
2.7.1. Evolución de un sistema cuántico abierto	20
2.7.2. Representación de Kraus	21
2.7.3. Matriz de Choi	25
3. Canales de Pauli	27
3.1. Definiciones y propiedades	27
3.2. Circuito para un canal de Pauli	33
4. Mapas dinámicos de Pauli	35
4.1. Definición y ejemplos	35
4.2. Implementar un mapa dinámico en un circuito	38
4.2.1. Circuito utilizando una sola puerta parametrizada	38
4.2.2. Circuito completo del mapa dinámico	43
4.2.3. Ejemplos	44

ÍNDICE GENERAL

4.3. Conjunto de mapas dinámicos de Pauli simulables	49
4.3.1. Ejemplos	52
5. Simulación en una Computadora Cuántica	57
5.1. Introducción	57
5.2. Definiciones	59
5.3. Ejecución de los circuitos en computadoras cuánticas	61
5.3.1. Análisis de los circuitos para el mapa depolarizador y el parabólico σ_3	67
6. Conclusiones	71
Bibliografía	73

Índice de figuras

2.1. Compuertas de un qubit.	10
2.2. Ejemplo de un circuito.	10
2.3. Ejemplo de circuito con una compuerta de 2 qubits.	11
2.4. Compuerta σ_2 sobre el qubit 0 controlada por los qubits 1 y 2.	11
2.5. Compuerta CNOT.	11
2.6. Compuerta CPHASE.	12
2.7. Compuerta U aplicada a dos qubits.	12
2.8. Representación de un sistema y el ambiente.	20
2.9. Efecto del canal BF (con $p = 0.7$) sobre la esfera de Bloch.	22
2.10. Efecto del canal PF (con $p = 0.7$) sobre la esfera de Bloch.	23
2.11. Efecto del canal BPF (con $p = 0.7$) sobre la esfera de Bloch.	23
2.12. Efecto del canal depolarizador (con $p = 0.5$) sobre la esfera de Bloch.	24
2.13. Efecto del canal AD (con $\gamma = 0.8$) sobre la esfera de Bloch.	25
3.1. Tetraedro en el espacio de las lambdas de canales de Pauli.	32
3.2. Propuesta de circuito para aplicar un canal de Pauli.	33
3.3. Circuito para crear el estado de 2 qubits $\sqrt{\alpha_0} 0\rangle 0\rangle + \sqrt{\alpha_1} 0\rangle 1\rangle + \sqrt{\alpha_2} 1\rangle 0\rangle + \sqrt{\alpha_3} 1\rangle 1\rangle$	34
3.4. Circuito para aplicar un canal de Pauli.	34
4.1. Curvas de los mapas BF, PF, BFP, depolarizador y parabólico σ_3 . Cada mapa dinámico forma una curva de canales dibujados sobre el tetraedro de canales de Pauli.	37
4.2. Circuito para implementar un mapa dinámico de Pauli.	38
4.3. Circuito general de dos qubits con una rotación parametrizada.	40
4.4. Circuito para crear la curva de estados de dos qubits.	42
4.5. Circuito para implementar un mapa dinámico de Pauli sobre el qubit q_0 usando una rotación parametrizada.	44
4.6. Mapa dinámico $\varepsilon_p(\rho) = (1 - p)^3\rho + p\sigma_1\rho\sigma_1 + (2p - 3p^2 + p^3)\sigma_2\rho\sigma_2$, que es un ejemplo de un mapa que no se puede implementar con un circuito de una rotación parametrizada.	50
4.7. Ejemplo 1 de mapa dinámico de Pauli que empieza en la identidad y se puede construir con una sola compuerta parametrizada.	54

4.8. Ejemplo 2 de mapa dinámico de Pauli que empieza en la identidad y se puede construir con una sola compuerta parametrizada	55
4.9. Ejemplo 3 de mapa dinámico de Pauli que empieza en la identidad y se puede construir con una sola compuerta parametrizada	55
5.1. Circuitos para realizar QST para un qubit.	61
5.2. Fidelidad de los circuitos que implementan canales de Pauli para distintos canales en el tetraedro realizados en el procesador ibmq-lima.	64
5.3. Fidelidad de los circuitos que implementan canales de Pauli para distintos canales en el tetraedro realizados en ibmq-qasm-simulator (que simula ser una computadora cuántica sin errores).	65
5.4. Fidelidad de los circuitos que implementan canales de Pauli para distintos canales en el tetraedro realizados en ibmq-qasm-simulator con un error de medición de $q = 0.07$	66
5.5. Fidelidad de los circuitos para los mapas dinámicos depolarizador y parabólico σ_3	68

Introducción

A principios del siglo XX, el estudio de sistemas cuánticos significó una revolución en la comprensión de la física. En las últimas décadas se ha desarrollado una nueva revolución en la física debido a la posibilidad de manipular y controlar directamente estos sistemas cuánticos. Entre las invenciones más importantes en esta nueva revolución se encuentran las computadoras cuánticas, que aprovechan las propiedades únicas de los sistemas cuánticos como la superposición y entrelazamiento para realizar operaciones de manera que puede ser exponencialmente más eficiente que computadoras clásicas [1, 2].

En el contexto de esta revolución emerge la disciplina de la información cuántica como intersección de la física cuántica y las ciencias de la computación. Esta disciplina surge de la realización de que cualquier tipo de información siempre se encuentra codificada en el estado de un sistema físico. Se conoce entonces como información cuántica cuando esta información se representa, procesa y comunica por medio de sistemas cuánticos [3]. En este trabajo se estudia uno de los sistemas cuánticos más sencillos y que es central en la investigación de la información cuántica, el qubit. El qubit es una versión cuántica del bit clásico, pues se define como un sistema cuántico que tiene solamente dos niveles [4] y se utiliza como la base de muchos modelos de computadoras cuánticas actuales [1].

Los sistemas cuánticos suelen interactuar con sus alrededores, perdiendo información hacia el ambiente y cambiando su dinámica respecto a como se comportarían en aislamiento [5]. A estos sistemas se les conoce como sistemas cuánticos abiertos y sus conceptos básicos se presenta en el marco teórico de este trabajo. Entender esta interacción es útil para construir procesadores cuánticos [4], pues los qubits de las computadoras cuánticas no están completamente aislados, lo que produce errores en las operaciones realizadas [1].

Para profundizar en estos conceptos, se estudian los *canales cuánticos*, que son procesos a través de los cuales pueden transformarse los sistemas cuánticos abiertos [5]. En particular, para un qubit hay un tipo especial de canales cuánticos conocidos como *canales de Pauli*, que consisten en aplicar una combinación convexa de matrices de Pauli al qubit [6].

Los canales de Pauli nos son de interés por varias razones. Para empezar, de entre

todos los canales de un qubit, los canales de Pauli son los más sencillos de estudiar y con varias propiedades interesantes que veremos en este trabajo. Además, estos canales son muy usados en información cuántica ya que los errores que tiene un qubit por su interacción con el ambiente se pueden modelar utilizándolos [7] e incluso son un modelo práctico para analizar los métodos de corrección de errores [8].

En el capítulo 3 se propone un algoritmo nuevo y sencillo para implementar un canal de Pauli arbitrario sobre un qubit en una computadora cuántica que utiliza dos qubits adicionales (conocidos como qubits de ancilla). Posteriormente, en el capítulo 5 aprovechamos que IBM tiene computadoras cuánticas disponibles para el uso del público en general y así simulamos este algoritmo para varios canales de Pauli distintos, y determinamos la fidelidad de esta implementación.

Más allá de los canales de Pauli, que son procesos discretos, se pueden definir también transformaciones continuas de un qubit, conocidas como mapas dinámicos de Pauli. Para estas transformaciones se usa un parámetro real, de tal forma que para cada valor de este parámetro se aplica algún canal de Pauli y así se tiene una evolución continua del qubit. Entonces, se adapta el algoritmo de canales de Pauli mencionado antes, añadiendo una dependencia en el parámetro a las compuertas cuánticas para que así se implemente un mapa dinámico de Pauli. Además, se hace especial énfasis en encontrar las condiciones que debe de cumplir un mapa dinámico de Pauli para que se pueda implementar con solamente una compuerta de rotación que dependa del parámetro, ya que de esta forma, dicha rotación funciona como una perilla que nos permite desarrollar el mapa dinámico. Estas condiciones son encontradas en el capítulo 4 y sirven para caracterizar a los mapas dinámicos de Pauli que se pueden implementar de esta manera.

Finalmente, se implementa este algoritmo para los mapas dinámicos depolarizador y el parabólico σ_3 en las computadoras cuánticas de IBM. Además, se calcula la fidelidad con la que la implementación se acerca a realizar los mapas definidos teóricamente y se muestran y discuten los resultados en el capítulo 5.

Es importante mencionar que el uso de computadoras cuánticas como herramientas para simular sistemas cuánticos fue propuesto por Feynman [9] y es una de las razones por las que empezó el estudio de esta disciplina. Luego, se han propuesto varios algoritmos cuánticos para simular sistemas cuánticos, ya sea sistemas cerrados descritos por un hamiltoniano [10, 11, 12] o para sistemas abiertos descritos por la ecuación de Lindblad [13]. Incluso se han realizado simulaciones en computadoras cuánticas para sistemas específicos como una cadena de espines [14] o el rotor pateado cuántico [15].

Es en este contexto que en este trabajo se propone un algoritmo nuevo para otro tipo de dinámica cuántica bien conocida y sencilla, que son los canales de Pauli. Además, se encuentra el circuito parametrizado para mapas dinámicos de Pauli y se demuestra el resultado sobre cuáles de estos circuitos se pueden implementar con sólo una rotación que dependa del parámetro.

Marco teórico

En esta sección repasamos algunos de los conceptos básicos de mecánica cuántica que serán necesarios para el resto del texto. El objetivo es cubrir la teoría necesaria para poder introducir el concepto de canales de Pauli en el siguiente capítulo.

Empezaremos hablando sobre los postulados de la mecánica cuántica para sistemas cerrados e introduciremos el sistema cuántico central para este trabajo, que es el qubit. Posteriormente, explicaremos en qué consisten los circuitos cuánticos, que son la representación diagramática de los algoritmos que se pueden realizar en computadoras cuánticas. Para ello, introduciremos las operaciones que se pueden realizar sobre un qubit, entre las que destacan las matrices de Pauli.

Más adelante, estudiaremos lo que sucede en sistemas cuánticos que pueden interactuar con sus alrededores, conocidos como sistemas cuánticos abiertos, ya que es en estos sistemas en los que se definen los canales de Pauli. Para estudiar estos sistemas será de gran utilidad el concepto de matriz de densidad, que se usa para representar los matemáticamente. Finalmente, introduciremos el concepto de canales cuánticos y daremos algunos de los resultados más importantes sobre ellos.

2.1. Postulados de la mecánica cuántica

Empezamos enumerando los postulados en los que se basa la mecánica cuántica en la manera en que se enuncian en [4].

- **Postulado 1:** Todo sistema físico tiene asociado un espacio vectorial \mathcal{H} complejo, completo y con producto interno (es decir, un espacio de Hilbert). El estado del sistema está determinado totalmente por su vector de estado, que es un vector de norma uno $|\psi\rangle$ en este espacio de Hilbert.
- **Postulado 2:** La medición de un sistema cuántico está descrita por una colección de operadores $\{M_m\}$, donde el índice m se refiere al resultado que puede obtenerse al realizar la medición. Si el estado del sistema cuántico es $|\psi\rangle$ antes de la medición,

la probabilidad de obtener el resultado m está dada por

$$p(m) = \langle \psi | M_m^\dagger M_m | \psi \rangle, \quad (2.1)$$

y el estado del sistema tras la medición será

$$\frac{M_m | \psi \rangle}{\sqrt{\langle \psi | M_m^\dagger M_m | \psi \rangle}}.$$

Además, los operadores de medición deben de satisfacer la ecuación $\sum_m M_m^\dagger M_m = I$, con I la identidad.

- **Postulado 3:** La evolución temporal de un estado $|\psi\rangle$ en un sistema cuántico cerrado está dada por la ecuación de Schrödinger,

$$\frac{d}{dt} |\psi(t)\rangle = -i\hbar \hat{H} |\psi(t)\rangle, \quad (2.2)$$

donde \hat{H} es el operador asociado al hamiltoniano del sistema. La solución a esta ecuación para un sistema que a $t = 0$ se encuentra en el estado inicial $|\psi(0)\rangle$ es

$$|\psi(t)\rangle = \hat{T} \exp \left[-\frac{i}{\hbar} \int_0^t \hat{H}(\tau) d\tau \right] |\psi(0)\rangle, \quad (2.3)$$

con \hat{T} el operador de ordenamiento temporal. Como el operador $\hat{T} \exp \left[-\frac{i}{\hbar} \int_0^t \hat{H}(\tau) d\tau \right]$ es unitario, se puede entonces reescribir este postulado indicando que la evolución temporal del estado de un sistema cerrado está dada por

$$|\psi(t)\rangle = \hat{U}(t) |\psi(0)\rangle, \quad (2.4)$$

donde $\hat{U}(t) : \mathcal{H} \rightarrow \mathcal{H}$ es un operador lineal unitario.

- **Postulado 4:** El espacio de Hilbert de un sistema físico compuesto es el producto tensorial de los espacios de cada componente $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \cdots \otimes \mathcal{H}_N$.

2.2. Qubit

Para este trabajo, será de especial importancia un sistema cuántico conocido como **qubit** que definimos a continuación [4].

Definición 1 Qubit: Un qubit es un sistema cuántico que tiene dos estados ortonormales, los cuales denotamos por $|0\rangle$ y $|1\rangle$. Como un sistema cuántico puede encontrarse en una superposición de estados, en general un qubit estará en un estado de la forma

$$|\psi\rangle = a|0\rangle + b|1\rangle,$$

donde $a, b \in \mathbb{C}$ y como está normalizado, cumplen que $|a|^2 + |b|^2 = 1$. Por lo tanto, el estado de un qubit se puede representar como un vector de \mathbb{C}^2 como sigue,

$$|\psi\rangle = a|0\rangle + b|1\rangle = \begin{pmatrix} a \\ b \end{pmatrix}.$$

Por el postulado 2, cuando medimos un qubit (respecto a la base $\{|0\rangle, |1\rangle\}$), su estado colapsará a uno de los estados base $|0\rangle$ o $|1\rangle$ con probabilidades $|a|^2$ y $|b|^2$ respectivamente.

Producto interno: Los estados $|0\rangle$ y $|1\rangle$ son ortonormales, por lo que si tenemos dos qubits $|\psi_1\rangle = a_1|0\rangle + b_1|1\rangle$ y $|\psi_2\rangle = a_2|0\rangle + b_2|1\rangle$, su producto interno es

$$\begin{aligned} \langle\psi_1|\psi_2\rangle &= (a_1^*\langle 0| + b_1^*\langle 1|)(a_2|0\rangle + b_2|1\rangle) \\ &= a_1^*a_2 + b_1^*b_2. \end{aligned}$$

2.3. Operaciones

En la sección anterior definimos el concepto de qubit y los estados en los que se pueden encontrar; ahora veremos las operaciones más comunes que se pueden aplicar a un qubit para convertir su estado y pasarlo de un estado inicial a un estado final. Para ser válidas físicamente, estas operaciones deben de cumplir con ser lineales y unitarias (para que preserven la condición de normalización del qubit). Como los qubits se pueden representar en forma de un vector de \mathbb{C}^2 , y las operaciones son lineales, una operación en este espacio se puede representar como una matriz de 2×2 .

Entonces, el espacio de todas las operaciones que pueden actuar sobre un qubit es equivalente al espacio de matrices unitarias de 2×2 con entradas en \mathbb{C} . Por ejemplo, si A es una matriz de este tipo, la transformación que realiza sobre un estado inicial arbitrario $|\psi_i\rangle$ es:

$$|\psi_i\rangle \rightarrow A|\psi_i\rangle.$$

Dentro de este espacio, dados dos operadores A y B , definimos su producto interno como

$$\langle A|B\rangle = \text{tr}(A^\dagger B), \tag{2.5}$$

donde la traza de un operador, por ejemplo A , se define como

$$\text{tr}(A) = \sum_i \langle i|A|i\rangle,$$

con $\{|i\rangle\}$ una base ortonormal arbitraria del sistema al que se aplica la operación.

Veremos primero un grupo de operadores que son especialmente importantes y serán en los que más nos fijaremos en este trabajo, los operadores de Pauli.

Definición 2 Operadores de Pauli: En el espacio de sistemas de un solo qubit existe una familia de operadores especialmente importantes conocidos como **operadores de Pauli** y son:

$$\sigma_0 := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \sigma_1 := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \sigma_2 := \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \sigma_3 := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Podemos ver el efecto de estas transformaciones sobre un qubit arbitrario $|\psi\rangle = a|0\rangle + b|1\rangle$,

$$\begin{aligned} \sigma_0|\psi\rangle &= |\psi\rangle, \\ \sigma_1|\psi\rangle &= \sigma_1(a|0\rangle + b|1\rangle) = b|0\rangle + a|1\rangle, \\ \sigma_2|\psi\rangle &= \sigma_2(a|0\rangle + b|1\rangle) = -ib|0\rangle + ia|1\rangle, \\ \sigma_3|\psi\rangle &= \sigma_3(a|0\rangle + b|1\rangle) = a|0\rangle - b|1\rangle. \end{aligned}$$

Vemos que el operador σ_1 tiene el efecto de intercambiar $|0\rangle$ y $|1\rangle$, por lo que a esta transformación se le llama **Bit Flip (BF)**. El operador σ_3 tiene el efecto de multiplicar por menos la amplitud de $|1\rangle$ (por lo que cambia la fase relativa del qubit) y se llama **Phase Flip (PF)**. El operador σ_2 , que hace ambas cosas y multiplica por i , se llama **Bit Phase Flip (BPF)**. Finalmente, el operador σ_0 es simplemente la identidad.

Bases: Como los operadores de Pauli son unitarios, sabemos que tienen eigenvectores que forman una base del espacio de qubits, escribimos ahora estas bases de eigenvectores para cada operador de Pauli.

- σ_1) Los eigenvectores son:

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

cuyos eigenvalores son 1 y -1 respectivamente.

- σ_2) Los eigenvectores son:

$$|R\rangle = \frac{|0\rangle + i|1\rangle}{\sqrt{2}}, \quad |L\rangle = \frac{|0\rangle - i|1\rangle}{\sqrt{2}},$$

cuyos eigenvalores son 1 y -1 respectivamente.

- σ_3) Los eigenvectores son:

$$|0\rangle, |1\rangle,$$

cuyos eigenvalores son 1 y -1 respectivamente.

Enlistamos ahora algunas propiedades importantes de los operadores de Pauli (en las ecuaciones siguientes, j, k son índices que toman los valores $0, 1, 2, 3$).

- $\sigma_j^\dagger = \sigma_j$, lo que quiere decir que los operadores de Pauli son hermitianos.
- $\sigma_j^2 = \sigma_0$, por lo que junto con la propiedad anterior, concluimos que $\sigma_j^\dagger \sigma_j = \sigma_0$, lo que significa que los operadores son unitarios.
- $\text{tr}(\sigma_j) = 0$, excepto para σ_0 .
- $\text{tr}(\sigma_j \sigma_k) = 2\delta_{jk}$, esta propiedad, junto con la de ser hermitianos, implica que los operadores de Pauli son ortogonales, pues cumplen que $\langle \sigma_j | \sigma_k \rangle := \text{tr}(\sigma_j^\dagger \sigma_k) = \text{tr}(\sigma_j \sigma_k) = 2\delta_{jk}$. En particular, tendremos que los operadores

$$\left\{ \frac{1}{\sqrt{2}}\sigma_0, \frac{1}{\sqrt{2}}\sigma_1, \frac{1}{\sqrt{2}}\sigma_2, \frac{1}{\sqrt{2}}\sigma_3 \right\}$$

son ortonormales (porque ahora la norma de cada uno sería 1 en vez de 2).

- $[\sigma_j, \sigma_k] = 2i\epsilon_{jkl}\sigma_l$, donde ϵ_{jkl} es el símbolo de Levi-Civita y se está usando la notación de suma de Einstein.
- Finalmente, se puede probar que los operadores de Pauli cumplen

$$\sigma_j \sigma_k \sigma_j = B_{jk} \sigma_k, \quad (2.6)$$

donde no se está usando la notación de Einstein y el elemento B_{jk} vale 1 cuando $j = k$ o alguno de los índices j, k son 0 y vale -1 en caso contrario.

Otras operaciones: También hay un par de operaciones adicionales sobre un qubit que son importantes y que describimos a continuación.

- **Hadamard:** Matricialmente el operador de Hadamard se define como

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

Algo importante de esta operación es que se puede utilizar para crear los eigenvectores de σ_1 a partir de $|0\rangle$ y $|1\rangle$, pues

$$H|0\rangle = |+\rangle, \quad H|1\rangle = |-\rangle. \quad (2.7)$$

- **Operador S:** El operador S (también conocido como la raíz cuadrada de σ_3 , ya que cumple que $S^2 = \sigma_3$) se define como

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

Es importante notar que a diferencia de las demás transformaciones que hemos estudiado, ésta no cumple ser su propio inverso. El inverso de S (que es también el conjugado porque es una matriz unitaria) es

$$S^\dagger = \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix}.$$

Este operador junto con el de Hadamard se pueden usar para convertir la base $\{|0\rangle, |1\rangle\}$ en la base $\{|R\rangle, |L\rangle\}$ del operador σ_2 , ya que

$$SH|0\rangle = |R\rangle, \quad SH|1\rangle = |L\rangle. \quad (2.8)$$

- **Rotación σ_1 :** Es un operador que depende de un parámetro θ y se define como

$$R_{\sigma_1}(\theta) := e^{-i\theta\sigma_1/2} = \sigma_0 \cos \frac{\theta}{2} - i\sigma_1 \sin \frac{\theta}{2} = \begin{pmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix}.$$

- **Rotación σ_2 :** Es un operador que depende de un parámetro θ y se define como

$$R_{\sigma_2}(\theta) := e^{-i\theta\sigma_2/2} = \sigma_0 \cos \frac{\theta}{2} - i\sigma_2 \sin \frac{\theta}{2} = \begin{pmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix}.$$

- **Rotación σ_3 :** Es un operador que depende de un parámetro θ y se define como

$$R_{\sigma_3}(\theta) := e^{-i\theta\sigma_3/2} = \sigma_0 \cos \frac{\theta}{2} - i\sigma_3 \sin \frac{\theta}{2} = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}.$$

- **Rotación $\sigma_{\hat{n}}$:** Es un operador que generaliza las rotaciones anteriores. Para un vector de norma uno $\hat{n} = (n_1, n_2, n_3)$, este operador se define como sigue

$$R_{\sigma_{\hat{n}}}(\theta) = e^{-i\frac{\theta}{2}(n_1\sigma_1 + n_2\sigma_2 + n_3\sigma_3)} = \sigma_0 \cos \frac{\theta}{2} - i\sigma_1 n_1 \sin \frac{\theta}{2} - i\sigma_2 n_2 \sin \frac{\theta}{2} - i\sigma_3 n_3 \sin \frac{\theta}{2}.$$

Vemos que las matrices σ_1 , σ_2 y σ_3 son casos especiales de las matrices de rotación (salvo por una fase global), pues:

- $\sigma_1 = iR_{\sigma_1}(\pi)$,
- $\sigma_2 = iR_{\sigma_2}(\pi)$,
- $\sigma_3 = iR_{\sigma_3}(\pi)$.

La propiedad más importante de los operadores de rotación es que cualquier operador unitario U en el espacio de un qubit se puede factorizar en términos de las rotaciones como se indica en el siguiente teorema.

Teorema 1 *Para cualquier operador unitario U en el espacio de un qubit, existen números reales $\alpha, \beta, \gamma, \delta$ tales que*

$$U = e^{i\alpha} R_{\sigma_3}(\beta) R_{\sigma_2}(\gamma) R_{\sigma_3}(\delta).$$

El enunciado y su demostración se encuentra en el teorema 4.1 de [4] y se detalla también en [16].

2.4. Sistemas de dos qubits

Un sistema de dos qubits tiene cuatro estados ortogonales que se pueden formar como el producto tensorial de los estados $|0\rangle$ y $|1\rangle$ de cada uno de los qubits individuales y por tanto son $|00\rangle, |01\rangle, |10\rangle, |11\rangle$, donde la primera entrada indica el estado del primer qubit y la segunda la del segundo. Un estado arbitrario de un par de qubits es una combinación lineal de estos estados y se puede representar con un vector de 4 dimensiones

$$|\psi\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle = \begin{pmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{pmatrix}, \quad (2.9)$$

donde $a_{00}, a_{01}, a_{10}, a_{11}$ son números complejos. Al medir ambos qubits en este estado $|\psi\rangle$ en la base computacional, obtendremos necesariamente alguno de los estados de la base, y la probabilidad de obtener uno es igual a la norma cuadrada de su coeficiente en la expansión 2.9. Por ejemplo, la probabilidad de obtener $|00\rangle$ al medir ambos qubits en esta base es $|a_{00}|^2$. Por lo tanto, los coeficientes deben de cumplir con una condición de normalización

$$|a_{00}|^2 + |a_{01}|^2 + |a_{10}|^2 + |a_{11}|^2 = 1.$$

Si tenemos dos qubits que se encuentran en estados $|\psi_a\rangle = \begin{pmatrix} a_0 \\ a_1 \end{pmatrix}$ y $|\psi_b\rangle = \begin{pmatrix} b_0 \\ b_1 \end{pmatrix}$, el estado colectivo de ambos es el producto de Kronecker de estos vectores,

$$|\psi_{ba}\rangle = |\psi_b\rangle \otimes |\psi_a\rangle = \begin{pmatrix} b_0 \cdot \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} \\ b_1 \cdot \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} b_0 a_0 \\ b_0 a_1 \\ b_1 a_0 \\ b_1 a_1 \end{pmatrix}.$$

Esta notación e ideas se pueden generalizar a sistemas de más qubits, sin embargo, no será necesario para el resto de esta tesis, por lo que no lo haremos.

2.5. Circuitos cuánticos

Un **circuito cuántico** es la representación de un algoritmo que consiste en aplicar operaciones en cierto orden deseado a un sistema de uno o varios qubits. Estos algoritmos se crean con el objetivo de ser implementados con ayuda de una computadora cuántica.

En un circuito cuántico, cada qubit se representa con una línea horizontal y la evolución de los qubits se lee de izquierda a derecha. Se añaden símbolos llamados “compuertas” que representan las operaciones que se realizan sobre los qubits en nuestro algoritmo. En la figura 2.1 se muestran las compuertas correspondientes a las operaciones que hemos estudiado.

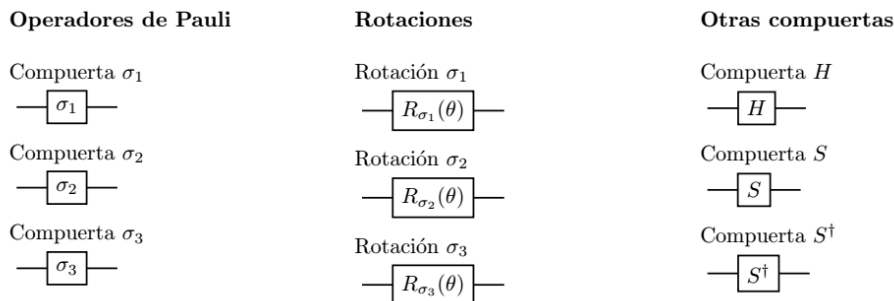


Figura 2.1: Compuertas de un qubit.

Usando estas compuertas se pueden construir circuitos cuánticos para realizar las operaciones que deseamos sobre los qubits. Por ejemplo, en la figura 2.2 se muestra un circuito sobre tres qubits que aplica una operación σ_1 seguida de una S^\dagger al primer qubit, una operación H seguida de una S al segundo y una rotación R_{σ_2} de ángulo $\pi/2$ al tercero.

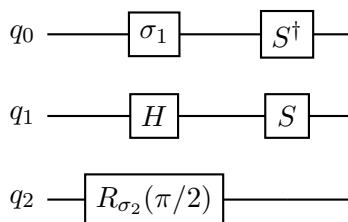


Figura 2.2: Ejemplo de un circuito.

Algo importante a tomar en cuenta es que los qubits siempre inician en el estado $|0\rangle$, a menos que se indique lo contrario al inicio del circuito cuántico.

2.5.1. Compuertas sobre múltiples qubits

Además de las operaciones sobre un qubit, también existen compuertas que actúan sobre múltiples qubits a la vez. Una operación sobre múltiples qubits también se puede representar como una matriz, con la diferencia que ahora la dimensión de la matriz será de $2^n \times 2^n$ cuando actúa sobre n qubits (ya que un estado de n qubits es un elemento de un espacio de Hilbert de 2^n dimensiones).

Para representar a las compuertas que actúan sobre múltiples qubits en un circuito, se usan figuras similares a las de compuertas de un qubit pero que abarcan a todos los qubits sobre los que están actuando. Por ejemplo, en la figura 2.3 se muestra un circuito en el que el primer qubit se le aplica una rotación R_{σ_1} de ángulo $\pi/4$ y a los qubits q_1, q_2 se les aplica una compuerta de dos qubits arbitraria que llamamos U .

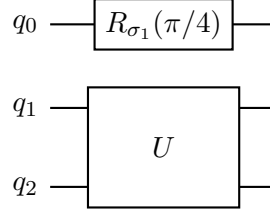


Figura 2.3: Ejemplo de circuito con una compuerta de 2 qubits.

2.5.1.1. Compuertas Controladas

Un conjunto especial de compuertas de múltiples qubits son las compuertas controladas. Las compuertas controladas son aquéllas que se aplican a ciertos qubits (llamados objetivo) solamente si se cumple la condición de que otros qubits (llamados controles) tengan cierto valor. Por ejemplo, en la figura 2.4 se muestra una compuerta σ_2 que se aplica sobre el qubit q_0 bajo la condición de que el qubit q_1 se encuentre en el estado $|1\rangle$ (indicado por un punto relleno) y que el qubit q_2 se encuentre en el estado $|0\rangle$ (indicado por el punto blanco). Claro que como los qubits de control usualmente están en un estado superpuesto de $|0\rangle$ y $|1\rangle$, entonces la compuerta controlada estará en una superposición de aplicarse y no aplicarse sobre el qubit objetivo.

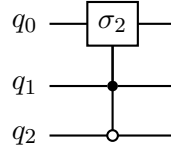


Figura 2.4: Compuerta σ_2 sobre el qubit 0 controlada por los qubits 1 y 2.

Una de las compuertas más importantes es la compuerta σ_1 controlada. Es decir, una compuerta que aplica el operador σ_1 a un qubit si se cumple la condición de que otro qubit (o varios otros qubits) se encuentre en cierto estado particular. Para estas compuertas, en lugar de dibujar una caja con la σ_1 sobre el qubit objetivo, usualmente se sigue una notación algo distinta, usando un símbolo de \oplus para denotar la operación σ_1 . Por ejemplo en la figura 2.5 se muestra una compuerta σ_1 controlada (conocida también como compuerta CNOT) que se aplica al qubit objetivo q_1 y tiene como qubit de control a q_0 (se activa la compuerta cuando q_0 está en $|1\rangle$).

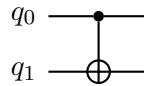


Figura 2.5: Compuerta CNOT.

Finalmente, cuando se usan compuertas σ_3 controladas, la notación también cambia

un poco. En lugar de dibujar una caja con una σ_3 indicando la compuerta σ_3 , se ponen puntos en todos los qubits involucrados (control y objetivo). Esto porque resulta que el estado final tras pasar por una de estas compuertas no depende de cuál qubit es el control y cuál el objetivo. Por ejemplo, en la figura 2.6 se muestra una compuerta σ_3 controlada (también conocida como compuerta CPHASE) sobre 2 qubits.

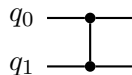


Figura 2.6: Compuerta CPHASE.

2.5.1.2. Compuertas generales sobre múltiples qubits

No todas las compuertas que actúan sobre múltiples qubits son compuertas controladas. Como se mencionó antes, también podemos tener una compuerta U en general que actúa sobre n qubits y que se representa por una matriz unitaria de dimensiones $2^n \times 2^n$. Por ejemplo, si tenemos una matriz unitaria arbitraria U de dimensión 4×4 , podemos dibujar en un circuito una compuerta que aplique esta matriz a dos qubits como en la figura 2.7.

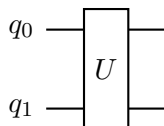


Figura 2.7: Compuerta U aplicada a dos qubits.

En general, al implementar una compuerta de varios qubits en una computadora cuántica, la computadora no la puede aplicar directamente, sino que es necesario descomponerla usando el siguiente teorema.

Teorema 2 *Cualquier operación U sobre varios qubits se puede descomponer en compuertas CNOT junto con compuertas unitarias de un solo qubit.*

El enunciado y su demostración se encuentran en la sección 4.5.2 de [4].

2.5.2. Aclaración sobre la notación

Para construir los circuitos cuánticos que revisaremos, usaremos el software Qiskit creado por IBM [17]. Con dicho software se pueden crear y manipular circuitos cuánticos e incluso se pueden ejecutar en verdaderas computadoras cuánticas o en simuladores y estudiar los resultados que se obtienen.

Para utilizar Qiskit hay que tomar en cuenta que maneja una convención distinta a la que se usa en algunos de los libros de texto para representar estados de múltiples qubits. Si tenemos un circuito con múltiples qubits q_0, q_1, \dots, q_n (numerados de arriba a abajo en el circuito), Qiskit representará el estado de múltiples qubits como $|q_n q_{n-1} \dots q_1 q_0\rangle$, convención que se conoce bajo el nombre de “**little-endian**”. Por otro lado, en algunos textos de computación cuántica los qubits se ordenan al revés, pues este mismo estado se representaría como $|q_0 q_1 \dots q_n\rangle$, convención conocida como “**big-endian**”. Esta diferencia puede causar confusión en la forma en que escribimos estados de múltiples qubits y también en la forma matricial de las compuertas que actúan sobre estos estados.

Por ejemplo, consideremos de nuevo el circuito de la figura 2.5, que consta solamente de una compuerta CNOT que tiene como control al qubit q_0 y actúa sobre q_1 . Esta compuerta aplica un operador σ_1 al qubit q_1 (que tiene el efecto de intercambiar $|0\rangle$ por $|1\rangle$ y viceversa) cuando q_0 se encuentra en $|1\rangle$. Si seguimos la convención de little-endian (en la que el estado de dos qubits se escribe como $|q_1 q_0\rangle$), eso significa que el efecto de esta compuerta sobre los estados base es

$$|00\rangle \rightarrow |00\rangle, \quad |01\rangle \rightarrow |11\rangle, \quad |10\rangle \rightarrow |10\rangle, \quad |11\rangle \rightarrow |01\rangle.$$

O bien, en notación vectorial, según la convención de little-endian tenemos que

$$\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

Por lo tanto, según esta convención, el operador tiene la siguiente representación matricial

$$CNOT_{\text{little-endian}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

Por otro lado, si analizamos el mismo circuito pero con la convención de big-endian (en la que se ordena como $|q_0 q_1\rangle$), tendremos que el efecto del circuito es

$$|00\rangle \rightarrow |00\rangle, \quad |01\rangle \rightarrow |01\rangle, \quad |10\rangle \rightarrow |11\rangle, \quad |11\rangle \rightarrow |10\rangle.$$

O bien, en notación vectorial,

$$\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}.$$

Entonces, usando esta notación, la matriz que representa al circuito sería

$$\text{CNOT}_{\text{big-endian}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Por lo tanto, vemos que la matriz es distinta según la convención que usemos. Sin embargo, físicamente la operación es la misma (el qubit q_1 cambia su estado de $|0\rangle$ a $|1\rangle$ y viceversa siempre y cuando el qubit q_0 se encuentre en el estado $|1\rangle$) a pesar de la notación.

En este trabajo **usaremos siempre la convención big-endian (donde los estados de múltiples qubits se escriben como $|q_0q_1q_2\cdots q_n\rangle$)** para todos los estados y circuitos que hagamos. Sin embargo, cuando implementemos circuitos en Qiskit habrá que tener cuidado, ya que si queremos crear un operador en un circuito, habrá que escribir su matriz con la convención little-endian. También habrá que tomar en cuenta que los resultados que nos da el programa siguen esta convención.

2.6. Matrices de densidad

Ahora nos fijaremos en la forma en que se suelen representar matemáticamente los sistemas cuánticos abiertos que estudiaremos más adelante. Para hacerlo, estudiaremos el concepto de matriz de densidad, la cual es una forma más general de describir sistemas cuánticos que el lenguaje de vectores de estado y que será más necesaria para sistemas abiertos.

La definición de la matriz de densidad surge de la necesidad de caracterizar sistemas cuánticos en los que no sabemos con certeza el estado en que se encuentra nuestro sistema de interés (algo común en escenarios experimentales). Se definen de la siguiente forma:

Definición 3 Matriz de densidad: Si el sistema que estudiamos se encuentra en alguno de los estados $\{|\psi_1\rangle, |\psi_2\rangle, \dots\}$ con probabilidades $\{p_1, p_2, \dots\}$ de encontrarse en cada uno (que deben de cumplir $\sum_i p_i = 1$ por ser probabilidades), entonces la matriz de densidad que lo representa es

$$\rho := \sum_i p_i |\psi_i\rangle\langle\psi_i|. \quad (2.10)$$

Definición 4 Estados puros y mixtos: Se dice que un sistema cuántico se encuentra en un estado puro si conocemos con certeza que su estado es $|\psi\rangle$. En el contexto de las

matrices de densidad, tendríamos que sólo hay un estado posible y entonces la matriz sería

$$\rho = |\psi\rangle\langle\psi|.$$

Por otro lado, un estado mixto se define simplemente como un estado que no es puro.

Es decir, el estado de un sistema mixto se representa usando una matriz, la cual es una suma de proyecciones $|\psi_i\rangle\langle\psi_i|$, cada una pesada con una probabilidad p_i . Las probabilidades naturalmente deben de cumplir que $p_i \geq 0$ y que $\sum_i p_i = 1$. Por ejemplo, para el caso de un qubit, las matrices de densidad tienen dimensión 2×2 y en general para n qubits son matrices de $2^n \times 2^n$. Para cuantificar qué tan puro o mixto es un estado ρ se usa el concepto de pureza, que se define a continuación.

Definición 5 Pureza: Dado un estado cuántico representado por la matriz de densidad ρ , su pureza es una cantidad escalar definida como [18]:

$$\mathcal{P}(\rho) := \text{tr}(\rho^2). \quad (2.11)$$

Se puede probar que la pureza de un estado normalizado satisface que $\frac{1}{d} \leq \mathcal{P}(\rho) \leq 1$, donde d es la dimensión del espacio de Hilbert en el que se define el estado [18]. Además, se puede corroborar que los estados que definimos como puros en la definición 4 tienen pureza igual a 1. Esto es debido a que dichos estados tienen la forma $\rho_{\text{puro}} = |\psi\rangle\langle\psi|$ y entonces su pureza es

$$\mathcal{P}(\rho_{\text{puro}}) = \text{tr}(\rho_{\text{puro}}^2) = \text{tr}(|\psi\rangle\langle\psi|\psi\rangle\langle\psi|) = \text{tr}(|\psi\rangle\langle\psi|) = 1.$$

Por otro lado, los estados mixtos tienen una pureza menor que 1 y el límite inferior en la pureza se alcanza cuando el estado es de la forma $\rho = \frac{1}{d} \sum_{i=1}^d |\psi_i\rangle\langle\psi_i| = \frac{1}{d} I_d$, con $|\psi_i\rangle$ una base ortonormal del espacio de Hilbert [4]. Este estado se conoce como *estado completamente mixto* y corresponde con la situación en la que se tiene la menor información posible sobre el sistema, ya que según la ecuación 2.10, esta matriz de densidad describe una mezcla en la que todos los estados $|\psi_i\rangle$ de la base ortonormal tienen la misma probabilidad $1/d$ en la mezcla.

Enlistamos ahora algunas propiedades de las matrices de densidad [19]:

- **Traza unitaria:** Las matrices de densidad cumplen que

$$\text{tr}(\rho) = 1. \quad (2.12)$$

Esta propiedad se puede comprobar sencillamente con el siguiente cálculo,

$$\text{tr}(\rho) = \text{tr}\left(\sum_i p_i |\psi_i\rangle\langle\psi_i|\right) = \sum_i p_i \text{tr}(|\psi_i\rangle\langle\psi_i|) = \sum_i p_i = 1,$$

donde se utilizó que la traza de un proyector $\text{tr}(|\psi_i\rangle\langle\psi_i|)$ es igual a 1.

- **Hermiticidad:** La matriz de densidad ρ es hermitiana. La demostración se puede hacer como sigue,

$$\rho^\dagger = \left[\sum_i p_i |\psi_i\rangle\langle\psi_i| \right]^\dagger = \sum_i p_i^\dagger |\psi_i\rangle\langle\psi_i| = \sum_i p_i |\psi_i\rangle\langle\psi_i| = \rho, \quad (2.13)$$

donde utilizamos que las probabilidades p_i son reales y que los proyectores $|\psi_i\rangle\langle\psi_i|$ son hermitianos.

- **Semidefinida Positiva:** Para todos los vectores $|v\rangle \in \mathcal{H}$, la matriz de densidad ρ cumple que $\langle v|\rho|v\rangle \geq 0$. Se puede probar como sigue,

$$\langle v|\rho|v\rangle = \langle v| \left[\sum_i p_i |\psi_i\rangle\langle\psi_i| \right] |v\rangle = \sum_i p_i \langle v|\psi_i\rangle\langle\psi_i|v\rangle = \sum_i p_i |\langle\psi_i|v\rangle|^2 \geq 0. \quad (2.14)$$

Esta propiedad se suele escribir como $\rho \geq 0$.

Aplicar un operador a una matriz de densidad: Por último estudiemos qué le pasa a la matriz de densidad cuando aplicamos un operador al sistema. Como hemos visto antes, si tenemos un operador unitario \mathcal{O} (que podría ser $\sigma_1, \sigma_2, \sigma_3, H, S$, etc.) y lo aplicamos al vector de estado de un qubit (digamos $|\psi\rangle$), el resultado se indica como $\mathcal{O}|\psi\rangle$. Para matrices de densidad, aplicar el operador \mathcal{O} tiene como efecto lo siguiente:

$$\begin{aligned} \text{Aplicar } \mathcal{O} \text{ a } \rho &= \text{Aplicar } \mathcal{O} \text{ a } \sum_i p_i |\psi_i\rangle\langle\psi_i|, \text{ donde escribimos la definición de } \rho. \\ &= \sum_i p_i \mathcal{O}|\psi_i\rangle\langle\psi_i|\mathcal{O}^\dagger, \text{ aplicamos } \mathcal{O} \text{ a los vectores de estado.} \\ &= \mathcal{O} \sum_i p_i |\psi_i\rangle\langle\psi_i|\mathcal{O}^\dagger \\ &= \mathcal{O}\rho\mathcal{O}^\dagger. \end{aligned}$$

Es decir, **aplicar un operador \mathcal{O} a una matriz de densidad ρ tiene el efecto siguiente:**

$$\rho \rightarrow \mathcal{O}\rho\mathcal{O}^\dagger. \quad (2.15)$$

2.6.1. Representación en la Esfera de Bloch

Veremos ahora una forma de representar geoméricamente a las matrices de densidad de un qubit, que como hemos visto, son matrices de dimensión de 2×2 . Por lo tanto, considerando que las matrices $\left\{ \frac{1}{\sqrt{2}}\sigma_0, \frac{1}{\sqrt{2}}\sigma_1, \frac{1}{\sqrt{2}}\sigma_2, \frac{1}{\sqrt{2}}\sigma_3 \right\}$ son una base ortonormal

de este espacio de matrices, podemos escribir ρ de la siguiente forma:

$$\begin{aligned}\rho &= \left\langle \frac{1}{\sqrt{2}}\sigma_0 \middle| \rho \right\rangle \frac{1}{\sqrt{2}}\sigma_0 + \left\langle \frac{1}{\sqrt{2}}\sigma_1 \middle| \rho \right\rangle \frac{1}{\sqrt{2}}\sigma_1 + \left\langle \frac{1}{\sqrt{2}}\sigma_2 \middle| \rho \right\rangle \frac{1}{\sqrt{2}}\sigma_2 + \left\langle \frac{1}{\sqrt{2}}\sigma_3 \middle| \rho \right\rangle \frac{1}{\sqrt{2}}\sigma_3 \\ &= \text{tr} \left(\frac{1}{\sqrt{2}}\sigma_0^\dagger \rho \right) \frac{1}{\sqrt{2}}\sigma_0 + \text{tr} \left(\frac{1}{\sqrt{2}}\sigma_1^\dagger \rho \right) \frac{1}{\sqrt{2}}\sigma_1 + \text{tr} \left(\frac{1}{\sqrt{2}}\sigma_2^\dagger \rho \right) \frac{1}{\sqrt{2}}\sigma_2 + \text{tr} \left(\frac{1}{\sqrt{2}}\sigma_3^\dagger \rho \right) \frac{1}{\sqrt{2}}\sigma_3 \\ &= \frac{1}{2} \text{tr}(\rho)\sigma_0 + \frac{1}{2} \text{tr}(\sigma_1\rho)\sigma_1 + \frac{1}{2} \text{tr}(\sigma_2\rho)\sigma_2 + \frac{1}{2} \text{tr}(\sigma_3\rho)\sigma_3 \\ &= \frac{\sigma_0 + \text{tr}(\sigma_1\rho)\sigma_1 + \text{tr}(\sigma_2\rho)\sigma_2 + \text{tr}(\sigma_3\rho)\sigma_3}{2}.\end{aligned}$$

Luego, podemos usar el siguiente teorema para llegar a la representación de Bloch.

Teorema 3 *Para una matriz de densidad ρ se cumple que $\text{tr}(\sigma_1\rho), \text{tr}(\sigma_2\rho), \text{tr}(\sigma_3\rho)$ son números reales y que $\text{tr}(\sigma_1\rho)^2 + \text{tr}(\sigma_2\rho)^2 + \text{tr}(\sigma_3\rho)^2 \leq 1$. Por lo tanto, concluimos que cualquier operador de densidad ρ de un qubit se puede escribir como*

$$\rho = \frac{\sigma_0 + r_{\sigma_1}\sigma_1 + r_{\sigma_2}\sigma_2 + r_{\sigma_3}\sigma_3}{2} = \frac{1}{2} \begin{pmatrix} 1 + r_{\sigma_3} & r_{\sigma_1} - ir_{\sigma_2} \\ r_{\sigma_1} + ir_{\sigma_2} & 1 - r_{\sigma_3} \end{pmatrix}, \quad (2.16)$$

donde $\vec{r} = (r_{\sigma_1}, r_{\sigma_2}, r_{\sigma_3}) := (\text{tr}(\sigma_1\rho), \text{tr}(\sigma_2\rho), \text{tr}(\sigma_3\rho))$ es un vector que cumple $|\vec{r}| \leq 1$.

Además, si ρ es un estado mixto, se cumple $|\vec{r}| < 1$ y si es puro se cumple $|\vec{r}| = 1$.

La demostración de este resultado se encuentra en la sección III.F de [5].

Es decir, cada matriz de densidad se puede representar de manera única con un vector en una esfera unitaria, con las matrices de densidad de estados puros en la superficie y los mixtos en el interior. Por ejemplo, se puede comprobar que el vector $(0, 0, 1)$ (el polo norte de la esfera) corresponde al estado puro $|0\rangle\langle 0|$, pues en ese caso

$$\rho = \frac{\sigma_0 + 0\sigma_1 + 0\sigma_2 + 1\sigma_3}{2} = \frac{\sigma_0 + \sigma_3}{2} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix},$$

que es la representación matricial de $|0\rangle\langle 0|$. De forma similar, el polo sur $(0, 0, -1)$ corresponde al estado $|1\rangle\langle 1|$. Finalmente, el origen representa al estado

$$\rho = \frac{\sigma_0 + 0\sigma_1 + 0\sigma_2 + 0\sigma_3}{2} = \frac{\sigma_0}{2},$$

que es el estado completamente mixto.

Luego, los operadores que hemos visto hasta ahora se pueden interpretar geométricamente, pues la acción que tienen sobre una matriz de densidad se puede contemplar al analizar como se transforma el vector de Bloch correspondiente en la esfera de Bloch. A continuación se mencionan los efectos que tienen sobre la esfera de Bloch algunos de los operadores que hemos estudiado:

- **Operador σ_1 :** Rota la esfera de Bloch π radianes alrededor del eje σ_1 .
- **Operador σ_2 :** Rota la esfera de Bloch π radianes alrededor del eje σ_2 .
- **Operador σ_3 :** Rota la esfera de Bloch π radianes alrededor del eje σ_3 .
- **Rotación σ_1 ($R_{\sigma_1}(\theta)$):** Rota la esfera de Bloch θ radianes alrededor del eje σ_1 .
- **Rotación σ_2 ($R_{\sigma_2}(\theta)$):** Rota la esfera de Bloch θ radianes alrededor del eje σ_2 .
- **Rotación σ_3 ($R_{\sigma_3}(\theta)$):** Rota la esfera de Bloch θ radianes alrededor del eje σ_3 .
- **Rotación $\sigma_{\hat{n}}$ ($R_{\sigma_{\hat{n}}}(\theta)$):** Rota la esfera de Bloch θ radianes alrededor del eje \hat{n} .

Vemos que esto explica por qué los operadores de rotación reciben ese nombre. Podemos revisar cómo se comprobaría la última de estas afirmaciones, la cual es la más general y tiene como casos especiales a las otras. Iniciamos con una matriz de densidad ρ que represente a un qubit, y la escribimos como $\rho = \frac{\sigma_0 + r_{\sigma_1}\sigma_1 + r_{\sigma_2}\sigma_2 + r_{\sigma_3}\sigma_3}{2}$, por lo que su vector de Bloch es $\vec{r} = (r_{\sigma_1}, r_{\sigma_2}, r_{\sigma_3})$. Aplicarle el operador $R_{\sigma_{\hat{n}}}$ a esta matriz tiene como efecto convertirla en $\rho' = R_{\sigma_{\hat{n}}}\rho R_{\sigma_{\hat{n}}}^\dagger$. Luego, las componentes del vector de Bloch de la matriz de densidad final ρ' son

$$r'_j = \text{tr}(\sigma_j \rho') = \text{tr}(\sigma_j R_{\sigma_{\hat{n}}} \rho R_{\sigma_{\hat{n}}}^\dagger).$$

Usando la expresión para $R_{\sigma_{\hat{n}}}$ y propiedades de los operadores de Pauli se obtiene que [20]:

$$r'_j = \sum_k A_{jk} r_k,$$

donde

$$A_{jk} = \delta_{jk} \cos \theta + (1 - \cos \theta) n_j n_k + \epsilon_{jkl} n_l \sin \theta,$$

con δ la delta de Kronecker y ϵ el tensor de Levi Civita. Esto implica que el vector de Bloch \vec{r} se transforma de acuerdo a la matriz A_{jk} , que se puede revisar que es una matriz ortogonal correspondiente a una rotación de ángulo θ alrededor del eje \vec{n} . El cálculo detallado se puede encontrar en [21].

2.6.2. Evolución temporal de la matriz de densidad

Estudiaremos ahora la evolución temporal de la matriz de densidad de un sistema cuántico *cerrado* (en una sección posterior veremos cómo evolucionan los sistemas cuánticos abiertos que nos interesan para este trabajo). Recordemos que por el postulado 3 de la mecánica cuántica, un sistema cuántico cerrado que empieza en el estado puro representado por el vector $|\psi(0)\rangle$ evoluciona con el tiempo según la regla

$$|\psi(t)\rangle = \hat{U}(t)|\psi(0)\rangle,$$

donde $\hat{U}(t) = \hat{T} \exp \left[-\frac{i}{\hbar} \int_0^t \hat{H}(\tau) d\tau \right]$, con \hat{H} el operador Hamiltoniano. Podemos generalizar esta ecuación a estados mixtos, en los que el estado está representado por una matriz de densidad $\rho(t)$ y sabemos por la ecuación 2.15 que el efecto de aplicar $\hat{U}(t)$ es

$$\rho(t) = \hat{U}(t)\rho(0)\hat{U}^\dagger(t). \quad (2.17)$$

2.6.3. Postulados en términos de matrices de densidad

Con los resultados que hemos obtenido, podemos volver a escribir los postulados de la mecánica cuántica pero ahora en términos de matrices de densidad, por lo que serán aplicables a estados mixtos [5].

- **Postulado 1:** El espacio de estados posibles que puede tener un sistema físico es un espacio de Hilbert \mathcal{H} compuesto por operadores lineales ρ tales que $\text{tr}(\rho) = 1$ y $\rho \geq 0$. El producto interno en este espacio está definido por $\langle \rho, \eta \rangle := \text{tr}(\rho^\dagger \eta)$ para dos operadores $\rho, \eta \in \mathcal{H}$.
- **Postulado 2:** Cualquier observable \mathcal{O} del sistema tiene asociado un operador lineal y hermitiano $\hat{\mathcal{O}}$. El valor esperado de este observable para un sistema en el estado descrito por el operador de densidad ρ es

$$\langle \hat{\mathcal{O}} \rangle_\rho = \text{tr}(\rho \hat{\mathcal{O}}).$$

Para justificar que esta ecuación es correcta, recordamos que el valor esperado para un estado puro $|\psi_i\rangle$ es $\langle \psi_i | \hat{\mathcal{O}} | \psi_i \rangle$. Pero la matriz de densidad ρ representa una combinación de estados puros $|\psi_i\rangle$ con probabilidades p_i , por lo que para calcular el valor esperado de \mathcal{O} , tendremos que multiplicar cada valor esperado $\langle \psi_i | \hat{\mathcal{O}} | \psi_i \rangle$ por su probabilidad p_i y sumarmas todas,

$$\begin{aligned} \langle \hat{\mathcal{O}} \rangle_\rho &= \sum_i p_i \langle \psi_i | \hat{\mathcal{O}} | \psi_i \rangle = \sum_i p_i \text{Tr}(\hat{\mathcal{O}} |\psi_i\rangle \langle \psi_i|) \\ &= \text{tr}(\hat{\mathcal{O}} \sum_i p_i |\psi_i\rangle \langle \psi_i|) = \text{tr}(\hat{\mathcal{O}} \rho) = \text{tr}(\rho \hat{\mathcal{O}}). \end{aligned}$$

- **Postulado 3:** Los operadores de densidad de un sistema cerrado evolucionan según la ecuación

$$\rho(t) = \hat{U}(t)\rho(0)\hat{U}^\dagger(t), \quad (2.18)$$

$$\text{donde } \hat{U}(t) = \hat{T} \exp \left[-\frac{i}{\hbar} \int_0^t \hat{H}(\tau) d\tau \right].$$

- **Postulado 4:** El espacio de Hilbert de un sistema físico compuesto es el producto tensorial de los espacios de cada componente $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \cdots \otimes \mathcal{H}_N$.

2.7. Sistema cuántico abierto

En este trabajo estudiaremos sistemas cuánticos abiertos, que se definen como sistemas cuánticos que pueden interactuar con un sistema externo conocido como el ambiente (figura 2.8) y que se representan matemáticamente usando matrices de densidad. El universo se encuentra dividido en el sistema principal (el sistema que nos interesa) más el ambiente y en un sistema abierto ambas partes pueden interactuar entre sí [19].

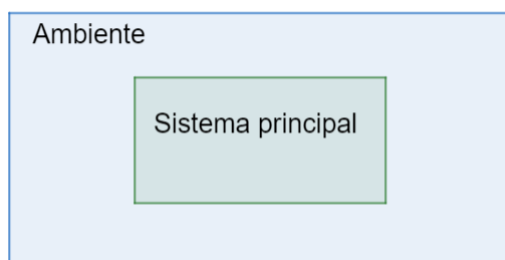


Figura 2.8: Representación de un sistema y el ambiente.

Un posible ejemplo de un sistema cuántico abierto podría ser un átomo con dos niveles de energía que tiene la posibilidad de decaer del estado alto al bajo a través de la emisión de un fotón. En este caso el átomo es nuestro sistema de interés y es un sistema abierto porque puede perder energía e interactuar con el ambiente al emitir un fotón.

2.7.1. Evolución de un sistema cuántico abierto

Estudiaremos ahora la evolución de sistemas cuánticos abiertos, cuya dificultad radica en que no podemos utilizar el postulado 3, el cual es sólo aplicable para sistemas cerrados. Sin embargo, sabemos que el sistema principal que estamos estudiando se representa por una matriz de densidad ρ . Después de que suceda algún proceso en el sistema, la matriz de densidad va a cambiar de acuerdo a algún operador ε (que toma matrices de densidad y las convierte en otras matrices de densidad). Tras el proceso, la matriz de densidad se transforma en general como

$$\rho \rightarrow \varepsilon(\rho).$$

De entre todos los procesos de este tipo, nos interesará un subconjunto en particular conocido como **canales cuánticos**, que se define a continuación.

Definición 6 Canal cuántico: Es un proceso que toma una matriz de densidad ρ y la convierte en otra matriz de densidad $\rho' = \varepsilon(\rho)$ y que cumple las siguientes propiedades [3]:

- ε es un operador lineal.
- ε preserva la traza: Se cumple $\text{tr}(\varepsilon(\rho)) = \text{tr}(\rho)$ para toda matriz de densidad ρ .
- ε es un operador completamente positivo.

Para entender este concepto, es necesario antes definir lo que se quiere decir por operador completamente positivo. Para hacerlo, se define primero el concepto de operador positivo.

Definición 7 Operador Positivo: ε es un operador positivo si $\rho \geq 0$ implica que $\varepsilon(\rho) \geq 0$, donde ≥ 0 significa que la matriz es semidefinida positiva, como se define en la ecuación 2.14.

Con esta definición se define ahora sí el concepto de operador completamente positivo.

Definición 8 Operador completamente positivo: ε es un operador completamente positivo si para todo $n \in \mathbb{N}$, $\varepsilon \otimes I_n$ es positivo.

Es importante tomar en cuenta que aunque todos los operadores completamente positivos son automáticamente operadores positivos, lo inverso no es cierto.

Retomando la definición de canal cuántico, podemos ver la razón por la que se pide que ε preserve la traza y sea positivo. Esto se hace para que si ρ es una matriz de densidad, entonces $\varepsilon(\rho)$ siga cumpliendo con las condiciones necesarias para ser también una matriz de densidad (tener traza igual a 1 y ser positiva). Sin embargo, se pide la condición más fuerte de que ε sea completamente positivo en vez de solamente positivo para que si iniciamos con un estado ρ de un sistema compuesto por dos partes P y Q y aplicamos el operador $\varepsilon \otimes I_Q$ (con I_Q el operador identidad en Q), entonces el resultado $\varepsilon \otimes I_Q(\rho)$ sea positivo.

2.7.2. Representación de Kraus

Ahora veremos la forma matemática que tienen los operadores ε que se usan para la evolución de un sistema cuántico abierto. Para ello tenemos el siguiente teorema [19].

Teorema 4 Teorema de Choi-Kraus: Un operador ε es lineal, preserva traza y es completamente positivo si y sólo si se puede escribir como

$$\varepsilon(\rho) = \sum_i E_i \rho E_i^\dagger, \quad (2.19)$$

donde E_i son operadores lineales sobre el espacio del sistema, llamados **operadores de Kraus** y cumplen que $\sum_i E_i^\dagger E_i = \sigma_0$ (esto último es necesario para que ε preserve la traza).

El enunciado y la demostración se puede encontrar en [19].

Esta representación del canal cuántico se conoce como **representación de Kraus**. En el caso en que ρ corresponde a un qubit, las E_k son matrices de 2×2 (pues actúan sobre el espacio de un qubit). A continuación presentamos algunos ejemplos de estos canales juntos con el efecto que tienen sobre la esfera de Bloch.

- **Ejemplo 1. Canal Bit Flip (BF):** Es una transformación que con una probabilidad p intercambia los estados $|0\rangle$ y $|1\rangle$ del qubit (lo que se hace aplicando la matriz σ_1) y con probabilidad $1-p$ no hace nada. El efecto de esta transformación en la representación de Kraus se escribe como

$$\varepsilon(\rho) = (1-p)\rho + p \sigma_1 \rho \sigma_1,$$

con $p \in [0, 1]$. Es decir, los operadores de Kraus son $E_0 = \sqrt{1-p}\sigma_0$ y $E_1 = \sqrt{p}\sigma_1$. Este canal se puede usar para representar la situación física en que con cierta probabilidad se intercambien los estados $|0\rangle$ y $|1\rangle$, ya sea por interacción con el ambiente o por errores en la medición.

Recordando que las matrices de densidad se pueden representar como vectores en la esfera de Bloch, podemos estudiar este canal cuántico observando su efecto sobre la esfera de Bloch. En la figura 2.9, a la izquierda se muestra la esfera de Bloch que representa a todas las matrices de densidad y a la derecha se muestra la forma en que se transforma la esfera de Bloch tras aplicar el canal BF (con $p = 0.7$).

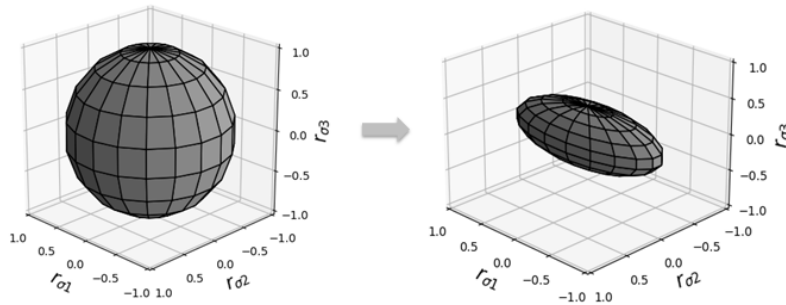


Figura 2.9: Efecto del canal BF (con $p = 0.7$) sobre la esfera de Bloch.

- **Ejemplo 2. Canal Phase Flip (PF):** Transformación que con una probabilidad p aplica la matriz σ_3 sobre el qubit y con probabilidad $1-p$ no hace nada. El

efecto de este operador se ve en la representación de Kraus como

$$\varepsilon(\rho) = (1 - p)\rho + p \sigma_3 \rho \sigma_3,$$

con $p \in [0, 1]$, por lo que los operadores de Kraus correspondientes son $E_0 = \sqrt{1 - p}\sigma_0$ y $E_1 = \sqrt{p}\sigma_3$. El efecto sobre la esfera de Bloch se puede ver en la figura 2.10, en la que $p = 0.7$.

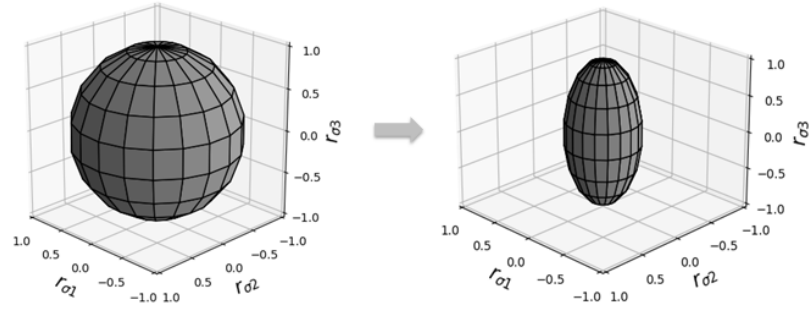


Figura 2.10: Efecto del canal PF (con $p = 0.7$) sobre la esfera de Bloch.

- **Ejemplo 3. Canal Bit-Phase Flip (BPF):** Transformación que con una probabilidad p aplica la matriz σ_2 sobre el qubit y con probabilidad $1 - p$ no hace nada. El efecto de este operador se ve en la representación de Kraus como

$$\varepsilon(\rho) = (1 - p)\rho + p \sigma_2 \rho \sigma_2,$$

con $p \in [0, 1]$, por lo que los operadores de Kraus correspondientes son $E_0 = \sqrt{1 - p}\sigma_0$ y $E_1 = \sqrt{p}\sigma_2$. El efecto sobre la esfera de Bloch se puede ver en la figura 2.11, en la que $p = 0.7$.

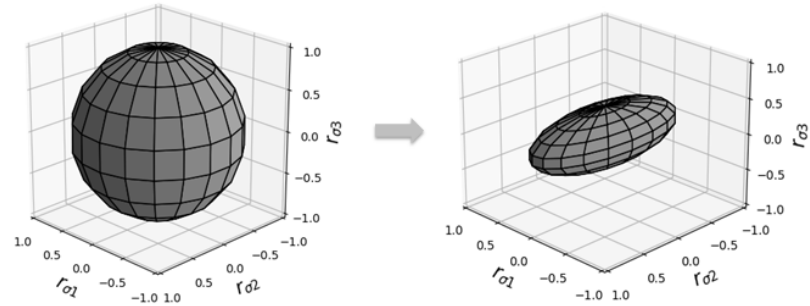


Figura 2.11: Efecto del canal BPF (con $p = 0.7$) sobre la esfera de Bloch.

- **Ejemplo 4. Canal depolarizador:** Este canal tiene una probabilidad p de depolarizar al qubit (es decir, convertirlo en el estado completamente mixto $\sigma_0/2$)

y una probabilidad $1 - p$ de no realizar nada. Entonces, el estado del qubit tras este canal es

$$\epsilon(\rho) = \frac{p\sigma_0}{2} + (1 - p)\rho,$$

con $p \in [0, 1]$. Sin embargo, esta transformación no está escrita en la forma de operadores de Kraus. Para que tenga esta forma se puede utilizar que para todo ρ , $\frac{\sigma_0}{2} = \frac{\rho + \sigma_1\rho\sigma_1 + \sigma_2\rho\sigma_2 + \sigma_3\rho\sigma_3}{4}$ y entonces tenemos que el canal se puede escribir como

$$\epsilon(\rho) = \left(1 - \frac{3p}{4}\right)\rho + \frac{p}{4}(\sigma_1\rho\sigma_1 + \sigma_2\rho\sigma_2 + \sigma_3\rho\sigma_3).$$

Por lo tanto, los operadores de Kraus son $E_0 = \sqrt{1 - \frac{3p}{4}}\sigma_0$, $E_1 = \sqrt{p/2}\sigma_1$, $E_2 = \sqrt{p/2}\sigma_2$ y $E_3 = \sqrt{p/2}\sigma_3$. El efecto de esta transformación sobre la esfera de Bloch se puede ver en la figura 2.12, en la que se aprecia que todos los estados puros se transforman en estados mixtos y se acercan al origen. Es por ello que este canal representa la situación física en la que debido a la interacción con el ambiente se pierde información sobre el estado del qubit y éste se acerca al estado completamente mixto.

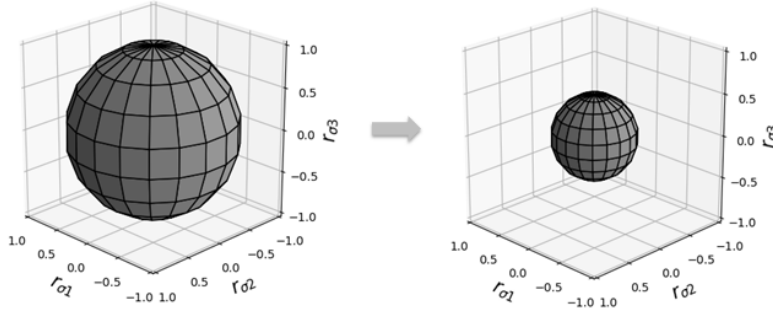


Figura 2.12: Efecto del canal depolarizador (con $p = 0.5$) sobre la esfera de Bloch.

- **Ejemplo 5. Canal Amplitude Damping (AD):** El canal de AD se utiliza para representar procesos en los que el sistema principal pierde energía. La representación de Kraus de esta transformación es

$$\epsilon(\rho) = E_0\rho E_0^\dagger + E_1\rho E_1^\dagger,$$

donde $E_0 = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{pmatrix}$ y $E_1 = \begin{pmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{pmatrix}$, con γ un parámetro que caracteriza al canal AD. El efecto del canal AD sobre la esfera de Bloch se puede ver en la figura 2.13.

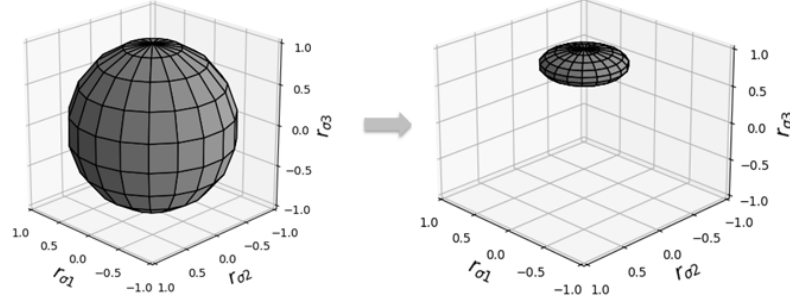


Figura 2.13: Efecto del canal AD (con $\gamma = 0.8$) sobre la esfera de Bloch.

Nótese como toda la esfera se aplasta hacia el polo norte, que es el punto que representa a la matriz de densidad del estado puro $|0\rangle$. Lo anterior corresponde con el significado de amplitude damping de quitar energía a los estados (considerando que el estado base es $|0\rangle$ y el excitado es $|1\rangle$).

Mencionamos ahora algunas propiedades importantes de la representación de Kraus [4].

- **Propiedad 1 (Libertad unitaria):** Supongamos que tenemos dos canales ε y ς que tienen como operadores de Kraus a $\{E_1, E_2, \dots, E_m\}$ y $\{F_1, F_2, \dots, F_n\}$ respectivamente. Si alguna de las listas de operadores es de menor longitud que la otra, podemos agregarle operadores cero de tal forma que $m = n$. Luego, $\varepsilon = \varsigma$ si y sólo si existen números complejos u_{ij} tales que $E_i = \sum_j u_{ij} F_j$ y u_{ij} es una matriz unitaria de $m \times m$.
- **Propiedad 2:** Todos los canales cuánticos ε en un sistema con un espacio de Hilbert de dimensión d se pueden generar con una representación de Kraus de a lo sumo d^2 elementos, es decir,

$$\varepsilon(\rho) = \sum_{k=1}^M E_k \rho E_k^\dagger,$$

con $1 \leq M \leq d^2$.

En el caso de un qubit, el espacio de Hilbert del qubit tiene dimensión $d = 2$, por lo que este teorema nos dice que todo operador sobre un qubit se puede escribir con a lo sumo 4 operadores de Kraus.

2.7.3. Matriz de Choi

Un canal cuántico también se puede representar de una forma alternativa a los operadores de Kraus, usando una matriz conocida como matriz de Choi, la cual contiene toda la información del canal y la definimos a continuación.

Definición 9 Matriz de Choi: La matriz de Choi es una forma de representar en una matriz a un canal cuántico. Si tenemos un canal cuántico ε , la matriz de Choi que representa a este canal se define como [5]:

$$C_\varepsilon = \sum_{i,j=0}^{n-1} |i\rangle\langle j| \otimes \varepsilon[|i\rangle\langle j|],$$

donde $|i\rangle, |j\rangle$ son vectores de una base ortonormal del sistema que estamos estudiando, que tiene dimensión n .

De esta forma, a cada uno de los canales que hemos estudiado hasta el momento se le puede asociar una matriz que lo representa. Como se puede ver, para calcular esta matriz es necesario evaluar el canal cuántico ε en cada uno de los operadores $|i\rangle\langle j|$ (que al representarlos matricialmente en la base del sistema, forman una matriz de $n \times n$ con un 1 en la posición ij y 0 en todas las demás). Es decir, para obtener la matriz de Choi asociada a un canal cuántico, es necesario conocer el efecto del canal cuántico sobre todas las matrices $|i\rangle\langle j|$, pues esto da la información suficiente para determinar completamente al canal.

Una de las mayores utilidades de la matriz de Choi es que nos permite determinar fácilmente si un canal ε es completamente positivo o no, con lo cual podemos saber si es físicamente válido. Esta utilidad queda clara con el teorema 5.

Teorema 5 Teorema de Choi: Un canal ε es completamente positivo si y sólo si su matriz de Choi C_ε es positiva semi-definida.

La demostración de este teorema se puede encontrar en [22].

Con ayuda de este teorema, si queremos comprobar si un canal es completamente positivo, sólo hace falta calcular su matriz de Choi y comprobar si es positiva semi-definida o no.

Canales de Pauli

En este capítulo se definirá el concepto de canales de Pauli para sistemas de un qubit. Los canales de Pauli son un conjunto de canales cuánticos especialmente importantes debido a su simplicidad y a su uso para modelar errores en computadoras cuánticas [7].

Después de definirlos y analizar algunas de sus propiedades principales, se propondrá un circuito para simular en una computadora cuántica cualquier canal de Pauli sobre un qubit.

3.1. Definiciones y propiedades

Definición 10 *Canal de Pauli:* Es un canal cuántico que tiene la forma

$$\varepsilon(\rho) = \alpha_0 \sigma_0 \rho \sigma_0 + \alpha_1 \sigma_1 \rho \sigma_1 + \alpha_2 \sigma_2 \rho \sigma_2 + \alpha_3 \sigma_3 \rho \sigma_3, \quad (3.1)$$

con α_i números reales. Por lo tanto, los operadores de Kraus de este canal son $E_0 = \sqrt{\alpha_0} \sigma_0$, $E_1 = \sqrt{\alpha_1} \sigma_1$, $E_2 = \sqrt{\alpha_2} \sigma_2$, $E_3 = \sqrt{\alpha_3} \sigma_3$ [23].

Es decir, en los canales de Pauli, los operadores de Kraus son proporcionales a las matrices de Pauli. Esto hace que sean fáciles de estudiar, ya que las propiedades de las matrices de Pauli son bien conocidas. Se puede revisar que muchos de los ejemplos de canales cuánticos de un qubit del capítulo anterior son de Pauli, en particular lo son los canales BF, PF, BFP y depolarizador.

Veremos un par de condiciones que deben de cumplir los coeficientes $\alpha_0, \alpha_1, \alpha_2, \alpha_3$ para que el canal de Pauli cumpla con los requerimientos necesarios para ser considerado un canal cuántico según la definición 6 (ser completamente positivo y preservar la traza):

1. Para que el canal de Pauli sea completamente positivo, se debe de cumplir que

$$\alpha_i \geq 0, \quad (3.2)$$

3. CANALES DE PAULI

para todo $i \in \{0, 1, 2, 3\}$. Para comprobar esto, primero obtenemos la matriz de Choi del canal de Pauli, que se obtiene como

$$C_\varepsilon = \sum_{i,j=0}^1 |i\rangle\langle j| \otimes \varepsilon(|i\rangle\langle j|).$$

Esta matriz es fácil de calcular usando la definición de ε y da como resultado

$$C_\varepsilon = \begin{pmatrix} \alpha_0 + \alpha_3 & 0 & 0 & \alpha_0 - \alpha_3 \\ 0 & \alpha_1 + \alpha_2 & \alpha_1 - \alpha_2 & 0 \\ 0 & \alpha_1 - \alpha_2 & \alpha_1 + \alpha_2 & 0 \\ \alpha_0 - \alpha_3 & 0 & 0 & \alpha_0 + \alpha_3 \end{pmatrix}.$$

Luego, según el teorema 5, el canal ε es positivo si y sólo si esta matriz C_ε es positiva semi-definida. Comprobar que es positiva semi-definida se puede hacer calculando sus valores propios y exigiendo que no sean negativos (ya podemos asegurar que van a ser reales, ya que claramente la matriz es hermitiana). Los valores propios de la matriz se pueden calcular sencillamente y resultan ser $2\alpha_0, 2\alpha_1, 2\alpha_2, 2\alpha_3$. Por lo tanto, exigir que estos valores propios sean mayor o iguales a cero implica que $\alpha_i \geq 0$ para todo i y ésta es la condición necesaria para que ε sea completamente positivo.

2. Para que el canal de Pauli preserve la traza, el teorema 4 indica que los operadores de Kraus deben de cumplir $\sum_k E_k^\dagger E_k = \sigma_0$. Es fácil comprobar que esta condición implica que

$$\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 = 1, \quad (3.3)$$

pues usando los operadores de Kraus del canal de Pauli, se tiene que

$$\begin{aligned} \sigma_0 &= E_0^\dagger E_0 + E_1^\dagger E_1 + E_2^\dagger E_2 + E_3^\dagger E_3 \\ &= (\sqrt{\alpha_0}\sigma_0)^\dagger (\sqrt{\alpha_0}\sigma_0) + (\sqrt{\alpha_1}\sigma_1)^\dagger (\sqrt{\alpha_1}\sigma_1) + (\sqrt{\alpha_2}\sigma_2)^\dagger (\sqrt{\alpha_2}\sigma_2) + (\sqrt{\alpha_3}\sigma_3)^\dagger (\sqrt{\alpha_3}\sigma_3) \\ &= \alpha_0\sigma_0^2 + \alpha_1\sigma_1^2 + \alpha_2\sigma_2^2 + \alpha_3\sigma_3^2 \\ &= \alpha_0\sigma_0 + \alpha_1\sigma_0 + \alpha_2\sigma_0 + \alpha_3\sigma_0 \\ &= (\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3)\sigma_0. \end{aligned}$$

Esto implica que $\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 = 1$, por lo que esta condición se debe de cumplir para que el canal preserve la traza.

Se le puede dar una interpretación física al resultado de aplicar el canal de Pauli ε a un estado inicial ρ . Notamos que este resultado se ve como una mezcla estadística de los estados $\sigma_i\rho\sigma_i$, cada uno pesado por una probabilidad α_i . Es decir, el canal de Pauli consiste en aplicar cada uno de los operadores σ_i sobre la matriz de densidad ρ con una probabilidad α_i . Con esta interpretación, la condición $\alpha_i \geq 0$ indica que las

probabilidades tienen que ser positivas y la condición $\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 = 1$ indica que deben de sumar una probabilidad total de 1.

Por otro lado, además de la representación usando operadores de Kraus, los canales de Pauli se pueden entender por su efecto sobre la esfera de Bloch. Este efecto resulta ser fácil de describir geoméricamente y se demuestra en el teorema 6.

Teorema 6 Efecto de un canal de Pauli sobre la esfera de Bloch: Si \vec{r} es el vector de Bloch asociado a una matriz de densidad ρ y \vec{r}' es el vector correspondiente a la matriz de densidad $\varepsilon(\rho)$ (con ε un canal de Pauli como el de la ecuación 3.1), entonces los vectores se relacionan como

$$\vec{r}' = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \vec{r},$$

con

$$\begin{aligned} \lambda_1 &= 1 - 2\alpha_2 - 2\alpha_3, \\ \lambda_2 &= 1 - 2\alpha_1 - 2\alpha_3, \\ \lambda_3 &= 1 - 2\alpha_1 - 2\alpha_2. \end{aligned} \tag{3.4}$$

Es decir, el efecto de un canal de Pauli sobre un vector de Bloch es aplicarle una matriz diagonal. Esta matriz diagonal multiplica cada componente r_{σ_i} de \vec{r} por un número λ_i , por lo que el efecto sobre la esfera de Bloch es de escalar la i -ésima dirección por el valor λ_i .

Demostración: Empezamos con una matriz de densidad $\rho = \frac{\sigma_0 + r_{\sigma_1}\sigma_1 + r_{\sigma_2}\sigma_2 + r_{\sigma_3}\sigma_3}{2}$, por lo que su vector de Bloch es $\vec{r} = (r_{\sigma_1}, r_{\sigma_2}, r_{\sigma_3})$. Esta matriz se puede escribir como

$$\rho = \sum_{k=0}^3 \frac{r_{\sigma_k}}{2} \sigma_k,$$

donde se define $r_{\sigma_0} = 1$ para poder usar la notación de suma. Luego, le aplicamos el canal de Pauli a esta matriz de densidad para obtener $\varepsilon(\rho)$:

$$\begin{aligned} \varepsilon(\rho) &= \sum_{j=0}^3 \alpha_j \sigma_j \rho \sigma_j \\ &= \frac{1}{2} \sum_{j=0}^3 \sum_{k=0}^3 \alpha_j r_{\sigma_k} \sigma_j \sigma_k \sigma_j, \quad \text{donde sustituimos la expresión de } \rho. \end{aligned}$$

3. CANALES DE PAULI

Luego, usamos la ecuación 2.6, que dice que $\sigma_j \sigma_k \sigma_j = B_{jk} \sigma_k$ con $B_{jk} = 1$ si $j = k$ o alguno de los índices es 0 y $B_{jk} = -1$ en caso contrario y entonces nos queda:

$$\begin{aligned}\varepsilon(\rho) &= \frac{1}{2} \sum_{j=0}^3 \sum_{k=0}^3 \alpha_j r_{\sigma_k} B_{jk} \sigma_k \\ &= \sum_{k=0}^3 \left(\sum_{j=0}^3 \frac{\alpha_j B_{jk} r_{\sigma_k}}{2} \right) \sigma_k\end{aligned}$$

Esta última expresión ya tiene la forma de la representación de Bloch de $\varepsilon(\rho)$, que es

$$\varepsilon(\rho) = \frac{\sigma_0 + r'_{\sigma_1} \sigma_1 + r'_{\sigma_2} \sigma_2 + r'_{\sigma_3} \sigma_3}{2}, \text{ y entonces concluimos que } r'_{\sigma_k} = \sum_{j=0}^3 \alpha_j B_{jk} r_{\sigma_k}.$$

Por lo tanto, usando la definición de B_{jk} , concluimos que el vector de Bloch se transformó en:

$$\begin{aligned}r'_{\sigma_1} &= (\alpha_0 + \alpha_1 - \alpha_2 - \alpha_3) r_{\sigma_1} = (1 - 2\alpha_2 - 2\alpha_3) r_{\sigma_1}, \\ r'_{\sigma_2} &= (\alpha_0 - \alpha_1 + \alpha_2 - \alpha_3) r_{\sigma_2} = (1 - 2\alpha_1 - 2\alpha_3) r_{\sigma_2}, \\ r'_{\sigma_3} &= (\alpha_0 - \alpha_1 - \alpha_2 + \alpha_3) r_{\sigma_3} = (1 - 2\alpha_1 - 2\alpha_2) r_{\sigma_3},\end{aligned}$$

donde al final se usa que $\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 = 1$, por lo que $\alpha_0 = 1 - \alpha_1 - \alpha_2 - \alpha_3$ ■.

Más allá de la interpretación geométrica sobre la esfera de Bloch que tienen los números $(\lambda_1, \lambda_2, \lambda_3)$, también son valores propios de ε (que junto con el valor $\lambda_0 := 1$, forman los 4 valores propios de ε). Además, los vectores propios correspondientes son las matrices de Pauli, es decir, se cumple que $\varepsilon(\sigma_i) = \lambda_i \sigma_i$ para todo $i \in \{0, 1, 2, 3\}$. Se puede comprobar esto sencillamente al aplicar ε a cada una de las matrices de Pauli usando la propiedad 2.6:

$$\begin{aligned}\varepsilon(\sigma_0) &= \sum_{j=0}^3 \alpha_j \sigma_j \sigma_0 \sigma_j = \alpha_0 \sigma_0 + \alpha_1 \sigma_0 + \alpha_2 \sigma_0 + \alpha_3 \sigma_0 = \sigma_0, \\ \varepsilon(\sigma_1) &= \sum_{j=0}^3 \alpha_j \sigma_j \sigma_1 \sigma_j = \alpha_0 \sigma_1 + \alpha_1 \sigma_1 - \alpha_2 \sigma_1 - \alpha_3 \sigma_1 = \lambda_1 \sigma_1, \\ \varepsilon(\sigma_2) &= \sum_{j=0}^3 \alpha_j \sigma_j \sigma_2 \sigma_j = \alpha_0 \sigma_2 - \alpha_1 \sigma_2 + \alpha_2 \sigma_2 - \alpha_3 \sigma_2 = \lambda_2 \sigma_2, \\ \varepsilon(\sigma_3) &= \sum_{j=0}^3 \alpha_j \sigma_j \sigma_3 \sigma_j = \alpha_0 \sigma_3 - \alpha_1 \sigma_3 - \alpha_2 \sigma_3 + \alpha_3 \sigma_3 = \lambda_3 \sigma_3.\end{aligned}$$

Por lo tanto, los coeficientes $(\lambda_1, \lambda_2, \lambda_3)$ se pueden interpretar de dos formas distintas: como factores de escala de la esfera de Bloch al aplicar el canal de Pauli o bien como valores propios de ε .

Tetraedro de Canales de Pauli

Nos interesa ahora conocer un poco más la representación de un canal de Pauli usando el punto de vista de los coeficientes $(\lambda_1, \lambda_2, \lambda_3)$. En particular, veremos ahora que los números $(\lambda_1, \lambda_2, \lambda_3)$ no se pueden escoger arbitrariamente para definir un canal de Pauli, sino que hay ciertas condiciones que deben de cumplir. Estas condiciones se muestran en el teorema 7 y nos llevarán a caracterizar al conjunto de posibles canales de Pauli de forma geométrica.

Teorema 7 *Para que un canal de Pauli sea válido, los valores de $\lambda_1, \lambda_2, \lambda_3$ deben de ser reales y cumplir que:*

$$1 + \lambda_i - \lambda_j - \lambda_k \geq 0,$$

$$1 + \lambda_1 + \lambda_2 + \lambda_3 \geq 0,$$

para $i, j, k \in \{1, 2, 3\}$ distintas [24].

Demostración: Partiendo de las expresiones de λ_j en función de α_i vistas la ecuación 3.4, queda claro que las λ_j son reales, pues así lo son las α_i . Además, se pueden encontrar las ecuaciones inversas en las que α_i se expresan en términos de λ_j como:

$$\alpha_1 = \frac{1}{4}(1 + \lambda_1 - \lambda_2 - \lambda_3),$$

$$\alpha_2 = \frac{1}{4}(1 - \lambda_1 + \lambda_2 - \lambda_3),$$

$$\alpha_3 = \frac{1}{4}(1 - \lambda_1 - \lambda_2 + \lambda_3).$$

Luego, para tener un canal de Pauli, como se indica en la ecuación 3.2, se debe de cumplir que $\alpha_i \geq 0$, lo cual da lugar a las condiciones $1 + \lambda_i - \lambda_j - \lambda_k \geq 0$. Por otro lado, si usamos ahora la condición 3.3, $\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 = 1$, para definir α_0 , y expresamos la desigualdad $\alpha_0 \geq 0$, llegamos a que $1 + \lambda_1 + \lambda_2 + \lambda_3 \geq 0$. ■.

Si definimos un espacio de 3 dimensiones con ejes $\lambda_1, \lambda_2, \lambda_3$, las desigualdades encontradas en el teorema anterior definen el interior de un tetraedro en este espacio. Cada punto dentro de este tetraedro corresponde con valores de $(\lambda_1, \lambda_2, \lambda_3)$ válidos que identifican a un canal de Pauli en particular, como se ve en la figura 3.1. Por ejemplo, tenemos los siguientes puntos importantes en el tetraedro:

- $(\lambda_1, \lambda_2, \lambda_3) = (1, 1, 1)$: Corresponde al canal identidad, pues la esfera de Bloch no es escalada en ninguna dirección.
- $(\lambda_1, \lambda_2, \lambda_3) = (1, -1, -1)$: Este canal refleja el eje σ_2 y el eje σ_3 de la esfera de Bloch, lo que es equivalente a una rotación de π radianes alrededor del eje σ_1 . Por ello, este punto corresponde a aplicar la compuerta σ_1 al estado y lo denotamos con una σ_1 en el tetraedro.

3. CANALES DE PAULI

- $(\lambda_1, \lambda_2, \lambda_3) = (-1, 1, -1)$: Este canal refleja el eje σ_1 y el eje σ_3 de la esfera de Bloch, lo que corresponde con una rotación de π radianes alrededor del eje σ_2 . Por ello, este punto representa aplicar la compuerta σ_2 al estado y lo denotamos con una σ_2 en el tetraedro.
- $(\lambda_1, \lambda_2, \lambda_3) = (-1, -1, 1)$: Este canal refleja el eje σ_1 y el eje σ_2 de la esfera de Bloch, lo que es equivalente a una rotación de π radianes alrededor del eje σ_3 . Por ello, este punto corresponde a aplicar la compuerta σ_3 al estado y lo denotamos con una σ_3 en el tetraedro.
- $(\lambda_1, \lambda_2, \lambda_3) = (0, 0, 0)$: Este canal comprime al 0 todas las direcciones de la esfera de Bloch, por lo que manda a cualquier estado inicial al estado totalmente mixto y se llama canal totalmente depolarizador.

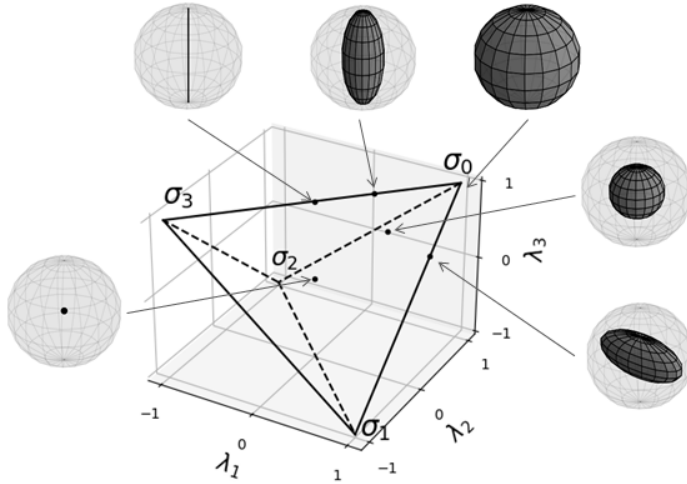


Figura 3.1: Tetraedro en el espacio de las lambdas de canales de Pauli.

En la figura 3.1 se puede apreciar el tetraedro formado en el espacio con ejes $(\lambda_1, \lambda_2, \lambda_3)$. Como se dijo antes, cada punto dentro de este tetraedro corresponde a un canal de Pauli distinto y en la figura se indica el efecto que tienen algunos de estos canales sobre la esfera de Bloch.

En conclusión, tenemos dos representaciones equivalentes de los canales de Pauli. La primera se obtiene dando los valores de $(\alpha_0, \alpha_1, \alpha_2, \alpha_3)$ que surgen de la representación de Kraus del canal y deben de cumplir que $\alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 = 1$ y que $\alpha_i \geq 0$. La segunda representación se consigue usando $(\lambda_1, \lambda_2, \lambda_3)$, se relaciona con el efecto sobre la esfera de Bloch y debe de cumplir que el valor de las lambdas se encuentre dentro del tetraedro.

3.2. Circuito para un canal de Pauli

El objetivo de esta sección es encontrar una forma de simular en una computadora cuántica cualquier canal de Pauli sobre un qubit. Para ello buscamos un circuito cuántico que toma como entrada un qubit q_0 en un estado arbitrario ρ y crea el estado $\varepsilon(\rho)$ sobre él (con ε el canal de Pauli).

Para proponer el circuito que haga esto, es necesario recordar que un canal de Pauli

$$\varepsilon(\rho) = \alpha_0 \sigma_0 \rho \sigma_0 + \alpha_1 \sigma_1 \rho \sigma_1 + \alpha_2 \sigma_2 \rho \sigma_2 + \alpha_3 \sigma_3 \rho \sigma_3,$$

$$\text{con } \alpha_0 + \alpha_1 + \alpha_2 + \alpha_3 = 1, \alpha_i \geq 0,$$

se puede interpretar físicamente como una operación que crea la mezcla de estados $\sigma_j \rho \sigma_j$, cada uno con probabilidad α_j . Tomando eso en cuenta, es fácil proponer un circuito que aplica el canal de Pauli sobre un qubit. Para hacerlo, será necesario agregar dos qubits adicionales llamados qubits de ancilla, que se usarán para crear las probabilidades α_i . La propuesta para el circuito se muestra en la figura 3.2.

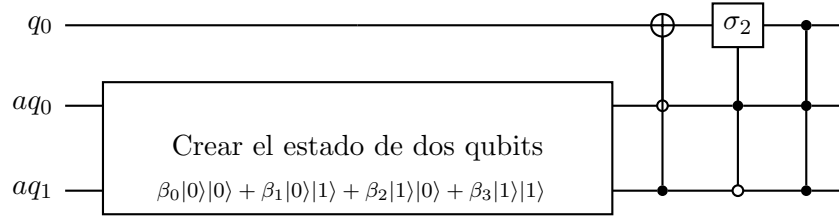


Figura 3.2: Propuesta de circuito para aplicar un canal de Pauli.

El qubit q_0 o qubit principal es el sistema que nos interesa y los otros dos qubits aq_0, aq_1 son los qubits de ancilla que se utilizan como ayuda para aplicar el canal a q_0 . El circuito es sencillo y consta de dos partes, primero crea el estado $\beta_0|0\rangle|0\rangle + \beta_1|0\rangle|1\rangle + \beta_2|1\rangle|0\rangle + \beta_3|1\rangle|1\rangle$ sobre los dos qubits de ancilla, donde β_j se define como cualquier número complejo tal que

$$|\beta_j|^2 = \alpha_j,$$

para $j = 0, 1, 2, 3$. Notar que con esta definición, β_j debe tener norma $\sqrt{\alpha_j}$ y puede tener cualquier fase. Después de crear este estado, se aplican las compuertas σ_1, σ_2 y σ_3 a q_0 controladas por los qubits de ancilla.

Para ver que el circuito funciona como esperaríamos, notamos que con probabilidad $|\beta_0|^2 = \alpha_0$ los qubits de ancilla estarán en $|0\rangle|0\rangle$ y por tanto no se aplicará ninguna de las compuertas controladas a q_0 . Por otro lado, con probabilidad $|\beta_1|^2 = \alpha_1$ los qubits de ancilla estarán en estado $|0\rangle|1\rangle$ y entonces se le aplica σ_1 a q_0 . Similarmente, con probabilidad $|\beta_2|^2 = \alpha_2$ se le aplica σ_2 y con probabilidad $|\beta_3|^2 = \alpha_3$ se le aplica σ_3 . Por ello, el circuito cumple con la interpretación estadística que tienen los canales de Pauli de aplicar σ_i con probabilidad α_i y efectivamente aplica ε sobre el estado del qubit q_0 .

3. CANALES DE PAULI

Por lo tanto, simular un canal de Pauli se reduce ahora al problema de crear el estado

$$\beta_0|0\rangle|0\rangle + \beta_1|0\rangle|1\rangle + \beta_2|1\rangle|0\rangle + \beta_3|1\rangle|1\rangle$$

sobre los qubits de ancilla. Entonces, de ahora en adelante nos enfocaremos en cómo crear este estado de dos qubits.

Hay muchas formas de crear un estado arbitrario de dos qubits utilizando rotaciones y otras compuertas. En particular, una manera de crear este estado para el caso en que β_j se escogen reales y entonces simplemente $\beta_j = \sqrt{\alpha_j}$ se propone en [25] y se presenta a continuación en la figura 3.3.

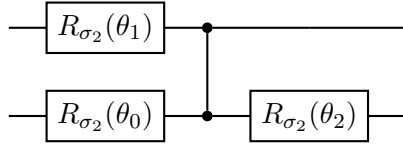


Figura 3.3: Circuito para crear el estado de 2 qubits $\sqrt{\alpha_0}|0\rangle|0\rangle + \sqrt{\alpha_1}|0\rangle|1\rangle + \sqrt{\alpha_2}|1\rangle|0\rangle + \sqrt{\alpha_3}|1\rangle|1\rangle$.

Los ángulos $\theta_0, \theta_1, \theta_2$ de las rotaciones están dados por:

$$\begin{aligned} \theta_0 &= \arg(\sqrt{\alpha_0} + i\sqrt{\alpha_1}) - \arg(\sqrt{\alpha_2} + i\sqrt{\alpha_3}), \\ \theta_1 &= 2 \arccos(\sqrt{\alpha_0 + \alpha_1}), \\ \theta_2 &= \arg(\sqrt{\alpha_0} + i\sqrt{\alpha_1}) + \arg(\sqrt{\alpha_2} + i\sqrt{\alpha_3}). \end{aligned} \tag{3.5}$$

Por lo tanto, ya teniendo la forma de crear el estado de dos qubits, el circuito que implementa un canal de Pauli es el de la figura 3.4.

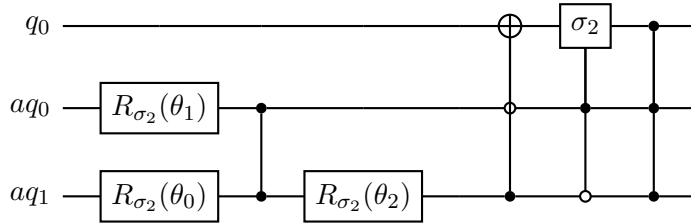


Figura 3.4: Circuito para aplicar un canal de Pauli.

Mapas dinámicos de Pauli

En este capítulo estudiaremos el concepto de mapas dinámicos de Pauli, que son curvas parametrizadas dibujadas dentro del espacio de canales de Pauli. Primero definiremos este concepto y veremos algunos ejemplos particulares. Posteriormente veremos cómo simular estos mapas dinámicos usando un circuito cuántico e impondremos la condición sobre el circuito de que sólo contenga una compuerta parametrizada.

En el teorema 9 qué curvas de estados se pueden crear usando sólo una rotación parametrizada. El resultado de este teorema es general e independiente de los canales de Pauli, pues nos habla en general sobre qué tipo de operaciones se pueden implementar con una sola rotación parametrizada. Luego se utilizará este resultado de vuelta en los mapas dinámicos de Pauli para ver cuáles cumplen con las condiciones para ser implementados de esta forma.

4.1. Definición y ejemplos

Definición 11 *Mapa dinámico de Pauli (para un qubit): Es una familia de canales de Pauli parametrizada por un parámetro $p \in [a, b] \subset \mathbb{R}$. Es decir, canales de la forma:*

$$\epsilon_p(\rho) = \alpha_0(p)\sigma_0\rho\sigma_0 + \alpha_1(p)\sigma_1\rho\sigma_1 + \alpha_2(p)\sigma_2\rho\sigma_2 + \alpha_3(p)\sigma_3\rho\sigma_3. \quad (4.1)$$

Para cada valor de p tenemos un canal de Pauli ϵ_p y el mapa dinámico consiste en el conjunto de todos estos canales conforme variamos p . Además, debe de cumplir que el mapa correspondiente a $p = a$ es la identidad, es decir $\epsilon_a(\rho) = \rho$.

Es decir, al variar p en un mapa dinámico de Pauli, se dibuja una curva en 4 dimensiones en el espacio de las alphas $(\alpha_0(p), \alpha_1(p), \alpha_2(p), \alpha_3(p))$. Sin embargo, para poder visualizar al mapa, pasamos al espacio de las lambdas y dibujamos la curva correspondiente $(\lambda_1(p), \lambda_2(p), \lambda_3(p))$ dentro del tetraedro de posibles canales.

Ejemplos

Vemos ahora algunos ejemplos de mapas dinámicos de Pauli. En realidad ya estudiamos varios de estos en la sección 2.7.2, pero en esos ejemplos pensábamos en el valor del parámetro p como algo fijo. Ahora para los mapas dinámicos de Pauli pensaremos en toda la familia de operadores conforme variamos el valor de p .

- **Ejemplo 1 Mapa BF:** Es el mapa

$$\epsilon_p(\rho) = (1 - p)\rho + p\sigma_1\rho\sigma_1, \quad (4.2)$$

con $p \in [0, 1]$. Cuando $p = 0$, el canal es simplemente la identidad (pues tenemos $\epsilon_0(\rho) = \rho$) y cuando $p = 1$, el canal es $\epsilon_1(\rho) = \sigma_1\rho\sigma_1$ que es el canal BF completo, pues intercambia los estados $|0\rangle$ y $|1\rangle$. Es decir, conforme vamos aumentando p desde 0 hasta 1, el estado ρ empieza como está y poco a poco va aumentando la probabilidad de intercambiar $|0\rangle$ y $|1\rangle$, terminando el intercambio cuando $p = 1$.

Este mapa se puede ver también en el espacio de las lambdas:

$$\begin{aligned} \lambda_1(p) &= 1 - 2\alpha_2(p) - 2\alpha_3(p) = 1 - 2(0) - 2(0) = 1, \\ \lambda_2(p) &= 1 - 2\alpha_1(p) - 2\alpha_3(p) = 1 - 2(p) - 2(0) = 1 - 2p, \\ \lambda_3(p) &= 1 - 2\alpha_1(p) - 2\alpha_2(p) = 1 - 2(p) - 2(0) = 1 - 2p. \end{aligned}$$

Entonces, conforme variamos p desde 0 hasta 1, $(\lambda_1(p), \lambda_2(p), \lambda_3(p))$ dibuja una curva en el espacio de las lambdas que se puede ver en la figura 4.1, vemos que el mapa empieza en la identidad y se acerca hacia la esquina σ_1 .

Otros ejemplos similares son el mapa PF y el BPF, que se definen de igual manera que éste pero con σ_3 o con σ_2 en vez de σ_1 . Las curvas dibujadas sobre el tetraedro correspondientes a estos mapas se muestran también en la figura 4.1.

- **Ejemplo 2 (Mapa depolarizador):** Es un conjunto de canales cuánticos parametrizados por $p \in [0, 1]$ de la siguiente forma,

$$\varepsilon_p(\rho) = (1 - 3p/4)\sigma_0\rho\sigma_0 + (p/4)\sigma_1\rho\sigma_1 + (p/4)\sigma_2\rho\sigma_2 + (p/4)\sigma_3\rho\sigma_3. \quad (4.3)$$

Para cada valor de p tenemos un canal depolarizador distinto; con $p = 0$ el canal identidad y con $p = 1$ el canal totalmente depolarizador. Es decir, el estado ρ empieza como está cuando $p = 0$ y se va depolarizando conforme aumentamos p . Podemos escribir este mapa en el espacio de las lambdas como:

$$\begin{aligned} \lambda_1(p) &= 1 - 2\alpha_2(p) - 2\alpha_3(p) = 1 - 2\frac{p}{4} - 2\frac{p}{4} = 1 - p, \\ \lambda_2(p) &= 1 - 2\alpha_1(p) - 2\alpha_3(p) = 1 - 2\frac{p}{4} - 2\frac{p}{4} = 1 - p, \\ \lambda_3(p) &= 1 - 2\alpha_1(p) - 2\alpha_2(p) = 1 - 2\frac{p}{4} - 2\frac{p}{4} = 1 - p. \end{aligned}$$

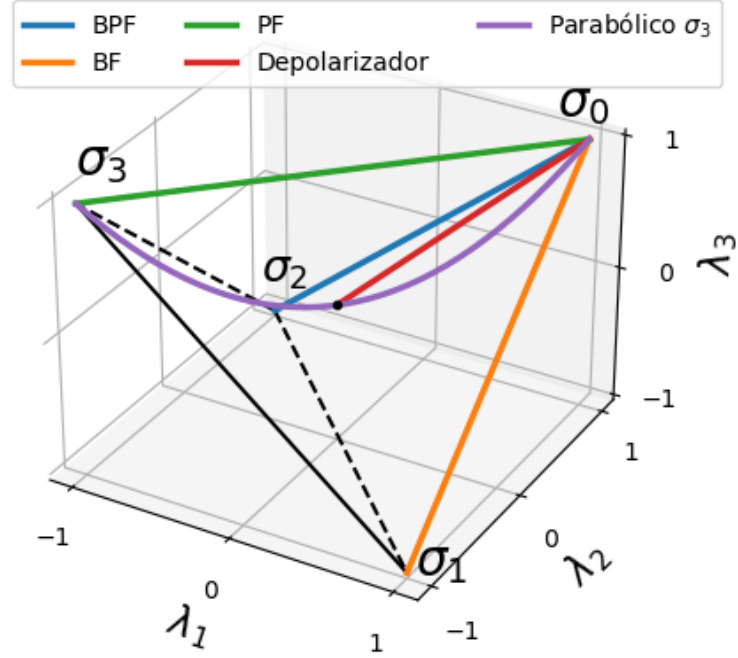


Figura 4.1: Curvas de los mapas BF, PF, BFP, depolarizador y parabólico σ_3 . Cada mapa dinámico forma una curva de canales dibujados sobre el tetraedro de canales de Pauli.

La curva que dibujan estas lambdas conforme variamos p se muestra en la figura 4.1. Se puede notar que el mapa traza una curva que empieza en la identidad y se acerca hacia el origen del tetraedro (correspondiente con el canal totalmente depolarizador).

- **Mapa parabólico σ_3 :** Definimos este mapa dinámico como

$$\varepsilon_p(\rho) = \frac{1}{4}(1-p)^2\rho + \frac{1}{4}(1-p^2)\sigma_1\rho\sigma_1 + \frac{1}{4}(1-p^2)\sigma_2\rho\sigma_2 + \frac{1}{4}(1+p)^2\sigma_3\rho\sigma_3,$$

con $p \in [-1, 1]$, por lo que las lambdas son

$$\begin{aligned}\lambda_1(p) &= 1 - 2\alpha_2(p) - 2\alpha_3(p) = -p, \\ \lambda_2(p) &= 1 - 2\alpha_1(p) - 2\alpha_3(p) = -p, \\ \lambda_3(p) &= 1 - 2\alpha_1(p) - 2\alpha_2(p) = p^2.\end{aligned}$$

La curva que dibujan estas lambdas conforme variamos p se muestra en la figura 4.1 y se puede notar que es una parábola que une la esquina σ_0 con la esquina σ_3 . Se pueden definir mapas análogos para σ_2 y σ_1 .

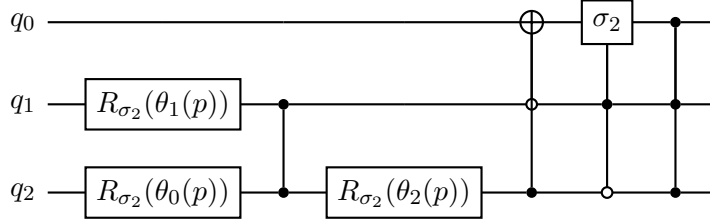


Figura 4.2: Circuito para implementar un mapa dinámico de Pauli.

4.2. Implementar un mapa dinámico en un circuito

En esta sección estudiaremos una forma de implementar mapas dinámicos de Pauli de un qubit en circuitos cuánticos. Veremos que estos circuitos en general requieren tres rotaciones parametrizadas (es decir, utilizan tres rotaciones que dependen del parámetro p del mapa dinámico). Posteriormente, buscaremos mapas que se puedan implementar en un circuito de forma más conveniente, utilizando una sola rotación parametrizada en lugar de tres.

Comenzamos desarrollando un circuito general para ejecutar cualquier mapa dinámico de Pauli de un qubit. Para hacerlo, recordamos que en la sección 3.2, se vio que un canal de Pauli de la forma

$$\epsilon_p(\rho) = \alpha_0 \sigma_0 \rho \sigma_0 + \alpha_1 \sigma_1 \rho \sigma_1 + \alpha_2 \sigma_2 \rho \sigma_2 + \alpha_3 \sigma_3 \rho \sigma_3$$

se puede implementar usando el circuito de la figura 3.4, en el cual se utilizan tres ángulos $\theta_0, \theta_1, \theta_2$ que dependen de los coeficientes $\alpha_0, \alpha_1, \alpha_2, \alpha_3$ que definen al canal.

Nos podemos basar en este circuito y generalizarlo para ejecutar mapas dinámicos de Pauli. Lo que caracteriza a los mapas dinámicos es que están parametrizados por una variable p , es decir, los coeficientes $\alpha_0, \alpha_1, \alpha_2, \alpha_3$ dependen de p . Por lo tanto, podemos implementar un mapa dinámico en un circuito cuántico de la misma forma que en la figura 3.4, con la diferencia que ahora los ángulos $\theta_0, \theta_1, \theta_2$ también dependerán de p . Tomando esto en cuenta, el circuito para un mapa dinámico es el de la figura 4.2.

Para cada valor que pueda tomar el parámetro p en el mapa dinámico, este circuito ejecuta el canal ϵ_p sobre el qubit principal q_0 siguiendo la misma lógica que el de la figura 3.4. Al igual que en la sección 3.2, los ángulos $\theta_0(p), \theta_1(p), \theta_2(p)$ se consiguen a partir de $\alpha_0(p), \alpha_1(p), \alpha_2(p), \alpha_3(p)$ según la ecuación 3.5.

4.2.1. Circuito utilizando una sola puerta parametrizada

Encontramos en la sección anterior que el circuito 4.2 puede implementar cualquier mapa de Pauli ϵ_p sobre un qubit y que para hacerlo se requieren tres rotaciones que dependen del valor de p . Es decir, si queremos simular el mapa ϵ_p en una computadora cuántica, para cada valor de p tendremos que ajustar los ángulos $\theta_0(p), \theta_1(p), \theta_2(p)$ de las

tres rotaciones. Sin embargo, modificar los tres ángulos conforme varía el parámetro p puede no ser del todo conveniente. Sería más conveniente poder construir un circuito con una sola rotación parametrizada por p , para que de esta forma sea necesario modificar un solo ángulo para recorrer todos los canales del mapa dinámico. La ventaja de hacerlo de esta forma es que el ángulo que depende de p funcionaría como un control o una perilla, que al ir cambiándolo, modifica al circuito cuántico para que implemente cada uno de los canales contenidos en el mapa dinámico.

Sin embargo, no necesariamente todos los mapas dinámicos de Pauli se podrán ejecutar con un circuito con una sola rotación parametrizada. En esta sección estudiaremos los mapas que sí se pueden implementar de esta manera y veremos algunos ejemplos. Para hacerlo, primero recordemos de la sección 3.2, que para implementar un mapa dinámico de Pauli en un circuito, se empieza creando el estado

$$|\beta(p)\rangle := \beta_0(p)|00\rangle + \beta_1(p)|01\rangle + \beta_2(p)|10\rangle + \beta_3(p)|11\rangle \quad (4.4)$$

sobre los dos qubits de ancilla, donde $\beta_j(p)$ son números complejos tales que $|\beta_j(p)|^2 = \alpha_j(p)$. Entonces, para saber qué mapas dinámicos se pueden realizar con una rotación parametrizada, el problema se reduce a identificar los casos en que sea posible crear esta curva de estados de 2 qubits usando una sola rotación que dependa de p , lo cual se determina en el teorema 9. Para poder establecer esto, primero concretamos lo que significa tener un circuito con una sola rotación parametrizada en la definición 12.

Definición 12 *Circuito con una sola rotación parametrizada:* Decimos que un circuito es un circuito con una sola rotación parametrizada por el parámetro p si solamente hay una compuerta de un qubit en el circuito que depende del valor de p (que puede ser controlada o no por otro qubit) y dicha compuerta es una rotación con un eje de rotación arbitrario.

En realidad, no es necesario considerar rotaciones con un eje de rotación arbitrario, pues una rotación respecto al eje \hat{n} parametrizada por p se puede convertir en una rotación respecto al eje σ_3 sin necesidad de introducir otras compuertas que dependan de p . Para comprobar esto, consideramos que la rotación respecto al eje \hat{n} parametrizada por p en general tiene la forma $R_{\hat{n}}(s)$, donde s es alguna función de p y $\hat{n} = (n_1, n_2, n_3)$ es el eje de rotación. La dirección del eje de rotación \hat{n} se encuentra en la esfera unitaria y por tanto se puede escribir como $(n_1, n_2, n_3) = (\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta)$ para ángulos ϕ y θ fijos para cierto \hat{n} . Habiendo definido estos ángulos, se demuestra que

$$R_{\hat{n}}(s) = R_{\sigma_3}(\phi)R_{\sigma_2}(\theta)R_{\sigma_3}(s)R_{\sigma_2}(-\theta)R_{\sigma_3}(-\phi). \quad (4.5)$$

Este resultado se puede comprobar usando las expresiones de las rotaciones R_{σ_2} , R_{σ_3} y $R_{\hat{n}}$ definidas en la sección 2.3 y desarrollando los productos; este desarrollo se hace explícitamente en [21].

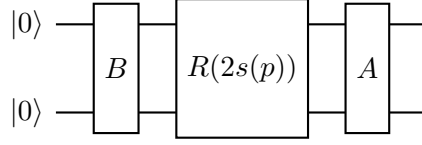
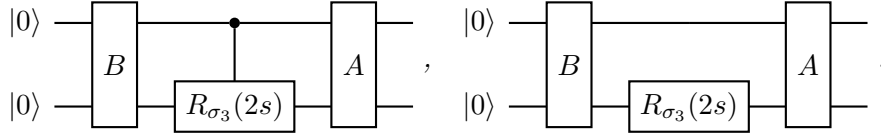


Figura 4.3: Circuito general de dos qubits con una rotación parametrizada.

Entonces, como los ángulos θ, ϕ no dependen del parámetro p de la rotación, cualquier circuito con una sola rotación parametrizada alrededor del eje \hat{n} se puede convertir en un circuito con una sola rotación parametrizada alrededor del eje σ_3 . Por lo tanto, al considerar circuitos con una sola rotación parametrizada, solamente hace falta considerar rotaciones alrededor del eje σ_3 .

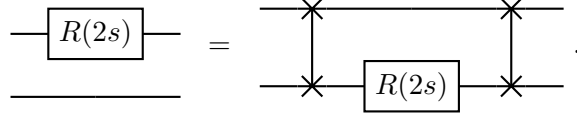
Regresamos entonces al problema de encontrar las curvas de estados de dos qubits de la forma $|\beta(p)\rangle$ (definida en 4.4) que se pueden crear con un circuito con una sola rotación parametrizada. Para ello, usando que se puede considerar que la rotación es respecto al eje σ_3 , en el teorema 8 veremos que los circuitos de dos qubits con una sola rotación parametrizada tienen una forma particular. Teniendo la forma particular de estos circuitos, podremos encontrar las curvas de estados de dos qubits que se pueden crear con ellos.

Teorema 8 *Un circuito de dos qubits con una rotación parametrizada por p se puede reescribir para que tenga la forma de alguno de los dos circuitos:*



Demostración: Asumamos que tenemos un circuito con una sola rotación R parametrizada por p (R representa cualquier rotación simple o controlada actuando sobre alguno de los qubits). Dicho circuito siempre se conseguirá aplicando primero alguna compuerta unitaria B (que no depende de p), seguida por la rotación R con un ángulo que depende de p , que por conveniencia, escribiremos como $2s(p)$ y finalmente una compuerta unitaria A (que no depende de p), como se ve en la figura 4.3. Vamos a ver que sin importar cuál sea la rotación R , este circuito siempre se puede reescribir modificando las compuertas A y B (de modo que sigan sin depender de p), de tal forma que la rotación sea una R_{σ_3} sobre el segundo qubit. Lo haremos considerando lo siguiente:

- Si R es una rotación que actúa sobre el primer qubit (controlada o simple), siempre podemos modificarla para que actúe sobre el segundo usando compuertas swap (las cuales intercambian dos qubits) usando la siguiente relación:



Luego incorporamos estas compuertas swap dentro de las compuertas A y B , con lo cual ya tenemos un circuito con la forma de la figura 4.3 pero con la compuerta actuando sobre el segundo qubit.

- Si la rotación es respecto a un eje \hat{n} , puede reescribirse el circuito para que sea respecto a σ_3 usando la relación 4.5:

$$- \boxed{R_{\hat{n}}(2s)} - = - \boxed{R_{\sigma_3}(-\phi)} - \boxed{R_{\sigma_2}(-\theta)} - \boxed{R_{\sigma_3}(2s)} - \boxed{R_{\sigma_2}(\theta)} - \boxed{R_{\sigma_3}(\phi)} - ,$$

donde los ángulos θ, ϕ se definen como antes a partir de \hat{n} de tal forma que $(n_1, n_2, n_3) = (\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta)$ y por lo tanto no dependen de p . Luego, las rotaciones a la izquierda de $R_{\sigma_3}(2s)$ se pueden incorporar a B y las que están a la derecha se incorporan a A . De esa forma, modificando los operadores A y B (pero de tal forma que no dependan de p), podemos convertir la rotación respecto a \hat{n} en una respecto a σ_3 .

Por lo tanto, un circuito de dos qubits con una rotación parametrizada por p se puede reescribir de tal forma que la rotación sea σ_3 (controlada o no) y que actúe sobre el segundo qubit. ■

Como tenemos ya que un circuito de dos qubits con una sola rotación parametrizada tiene la forma de alguno de los circuitos del teorema 8, podemos ahora sí encontrar el tipo de curvas de estados de dos qubits que se pueden crear con estos circuitos, lo que se detalla en el teorema 9.

Teorema 9 Una curva de estados de dos qubits $|\eta(p)\rangle = \eta_0(p)|00\rangle + \eta_1(p)|01\rangle + \eta_2(p)|10\rangle + \eta_3(p)|11\rangle$ se puede construir usando un circuito con una sola rotación parametrizada si y sólo si $\eta_j(p)$ se puede escribir como:

$$\eta_j(p) = c_j + a_j e^{is} + b_j e^{-is},$$

donde $s = s(p)$ es alguna función de p y $\vec{a} = (a_0, a_1, a_2, a_3)$, $\vec{b} = (b_0, b_1, b_2, b_3)$, $\vec{c} = (c_0, c_1, c_2, c_3)$ son vectores ortogonales en \mathbb{C}^4 tales que $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$.

Demostración: Por el teorema 8, sabemos las dos formas posibles que puede tener un circuito con una sola rotación parametrizada. Si se trata del circuito en el que la

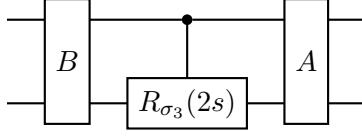


Figura 4.4: Circuito para crear la curva de estados de dos qubits.

rotación es controlada, entonces haciendo el producto de matrices de las compuertas aplicado al estado inicial $|00\rangle$, vemos que el circuito crea la curva de estados:

$$|\eta(p)\rangle = A\vec{z} + A\vec{y}e^{-is} + A\vec{w}e^{is},$$

donde $\vec{z} = (B_{00}, B_{10}, 0, 0)^T$, $\vec{y} = (0, 0, B_{20}, 0)^T$ y $\vec{w} = (0, 0, 0, B_{30})^T$, con $(B_{00}, B_{10}, B_{20}, B_{30})^T$ la primera columna de la matriz B . Esto se puede reescribir como $\vec{c} + \vec{b}e^{-is} + \vec{a}e^{is}$, con $\vec{c} = A\vec{z}$, $\vec{a} = A\vec{w}$, $\vec{b} = A\vec{y}$. Además, se puede ver que $\vec{z} \cdot \vec{w} = \vec{z} \cdot \vec{y} = \vec{w} \cdot \vec{y} = 0$, $|\vec{w}|^2 + |\vec{y}|^2 + |\vec{z}|^2 = 1$ y como A es unitaria, eso implica que $\vec{c} \cdot \vec{a} = \vec{c} \cdot \vec{b} = \vec{a} \cdot \vec{b} = 0$ y $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$.

Alternativamente, si el circuito del teorema 8 es el que tiene la rotación simple, entonces haciendo el producto de matrices se llega a que el estado creado es:

$$|\eta(p)\rangle = A\vec{y}e^{-is} + A\vec{w}e^{is},$$

donde $\vec{y} = (B_{00}, 0, B_{20}, 0)^T$ y $\vec{w} = (0, B_{10}, 0, B_{30})^T$. Este estado se puede reescribir como $\vec{a}e^{is} + \vec{b}e^{-is}$ con $\vec{b} = A\vec{y}$ y $\vec{a} = A\vec{w}$ y es fácil comprobar que $|\vec{w}|^2 + |\vec{y}|^2 = 1$ y que $\vec{w} \cdot \vec{y} = 0$. Luego, como A es unitaria, concluimos que $|\vec{a}|^2 + |\vec{b}|^2 = 1$ y $\vec{a} \cdot \vec{b} = 0$. Por lo que en este caso se llega también a la expresión $\vec{c} + \vec{a}e^{is} + \vec{b}e^{-is}$ (con $\vec{c} = 0$), y se cumple que $|\vec{c}|^2 + |\vec{a}|^2 + |\vec{b}|^2 = 1$ y $\vec{a} \cdot \vec{c} = \vec{b} \cdot \vec{c} = \vec{a} \cdot \vec{b} = 0$.

Por lo tanto, en cualquier caso concluimos que el estado final creado tiene la forma:

$$|\eta(p)\rangle = \vec{c} + \vec{a}e^{is} + \vec{b}e^{-is},$$

con $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$, $\vec{a} \cdot \vec{c} = \vec{b} \cdot \vec{c} = \vec{a} \cdot \vec{b} = 0$.

Regreso: Veremos ahora que si se tienen números c_i, a_i, b_i tales que cumplen con las condiciones mencionadas en el teorema, entonces se puede construir la curva de estados $|\eta(p)\rangle = \vec{c} + \vec{a}e^{is} + \vec{b}e^{-is}$ usando una sola rotación parametrizada.

Para probarlo, veremos que se puede crear la curva de estados $\vec{c} + \vec{a}e^{is} + \vec{b}e^{-is}$ con el circuito de la figura 4.4. En dicho circuito, las matrices se definen como:

$$B = \begin{pmatrix} 0 & * & * & * \\ |\vec{c}| & * & * & * \\ |\vec{b}| & * & * & * \\ |\vec{a}| & * & * & * \end{pmatrix}, \quad A = \begin{pmatrix} * & c_0/|\vec{c}| & b_0/|\vec{b}| & a_0/|\vec{a}| \\ * & c_1/|\vec{c}| & b_1/|\vec{b}| & a_1/|\vec{a}| \\ * & c_2/|\vec{c}| & b_2/|\vec{b}| & a_2/|\vec{a}| \\ * & c_3/|\vec{c}| & b_3/|\vec{b}| & a_3/|\vec{a}| \end{pmatrix}.$$

Los asteriscos representan entradas de las matrices que se pueden definir arbitrariamente, siempre y cuando las matrices sean unitarias (para que representen compuertas). Por lo tanto, primero tenemos que comprobar que estas matrices propuestas son válidas, para lo que debe de ser posible completarlas de forma que sean unitarias. Para la matriz B , basta revisar que la columna que está escrita tiene norma 1, lo cual se cumple porque $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$. La matriz A también se puede completar de forma que sea unitaria. Para ello, vemos que cada columna escrita tiene norma 1, lo cual está claro porque las columnas corresponden a los vectores $\vec{a}, \vec{b}, \vec{c}$ normalizados y además, estas columnas son ortogonales, ya que $\vec{a}, \vec{b}, \vec{c}$ lo son.

Ya sabiendo que A y B son unitarias, comprobamos ahora que el circuito 4.4 efectivamente crea el estado esperado. Para hacerlo, simplemente hay que realizar el producto de matrices de las compuertas del circuito y aplicarlo al estado inicial $|00\rangle = (1, 0, 0, 0)^T$. Con un cálculo sencillo usando las matrices propuestas, se obtiene que el resultado es

$$|\eta(p)\rangle = \vec{c} + \vec{a}e^{is} + \vec{b}e^{-is}. \quad \blacksquare$$

Nota: Para el regreso del teorema, si \vec{c}, \vec{a} o \vec{b} son el vector $\vec{0}$, entonces la matriz A no está definida (ya que alguna de sus columnas tiene una división por cero). Sin embargo, lo único que hay que hacer en estos casos es omitir estas columnas de A y reemplazarlas por cualquier vector que haga que A sea unitaria.

Concluimos entonces que todas las curvas de estados que se puedan escribir de la forma $\vec{c} + e^{is}\vec{a} + e^{-is}\vec{b}$ (cumpliendo con las condiciones mencionadas en el teorema) se pueden crear en un circuito con una sola rotación parametrizada y, además, que son las únicas que se pueden crear así.

Entonces, regresando a la implementación de un mapa dinámico con un circuito de una rotación parametrizada, requeríamos crear el estado $|\beta(p)\rangle$ (definido de tal manera que $|\beta_i(p)|^2 = \alpha_i(p)$ como se menciona en la ecuación 4.4). Ahora sabemos que para crearlo con una sola rotación parametrizada, es necesario que tenga la forma $|\beta(p)\rangle = \vec{c} + \vec{a}e^{is} + \vec{b}e^{-is}$, con las condiciones sobre los vectores definidas en el teorema 9.

Aunque las condiciones sobre la curva de estados parecen muy restrictivas, en la sección 4.2.3 veremos que muchos mapas dinámicos las cumplen y por tanto los podremos simular usando una rotación parametrizada.

4.2.2. Circuito completo del mapa dinámico

En conclusión, el mapa dinámico

$$\varepsilon_p(\rho) = \alpha_0(p)\rho + \alpha_1(p)\sigma_1\rho\sigma_1 + \alpha_2(p)\sigma_2\rho\sigma_2 + \alpha_3(p)\sigma_3\rho\sigma_3,$$

se puede implementar con una rotación parametrizada siempre y cuando la curva de estados de 2 qubits

$$|\beta(p)\rangle = (\beta_0(p), \beta_1(p), \beta_2(p), \beta_3(p)),$$

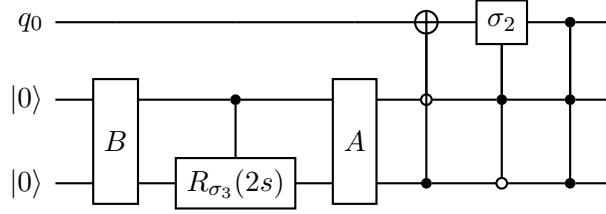


Figura 4.5: Circuito para implementar un mapa dinámico de Pauli sobre el qubit q_0 usando una rotación parametrizada.

cumpla las condiciones del teorema 9, con $|\beta_j(p)|^2 = \alpha_j(p)$. Lo que quiere decir que

$$\beta_j(p) = c_j + a_j e^{is} + b_j e^{-is},$$

con s alguna función de p y $\vec{a}, \vec{b}, \vec{c}$ vectores ortogonales en \mathbb{C}^4 tales que $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$. Dados estos vectores $\vec{a}, \vec{b}, \vec{c}$ y la función $s(p)$, entonces la forma de las $\alpha_j(p)$ del mapa tiene que ser:

$$\begin{aligned} \alpha_j(p) &= |\beta_j(p)|^2 \\ &= |c_j|^2 + |a_j|^2 + |b_j|^2 + (a_j c_j^* + c_j b_j^*) e^{is} + (b_j c_j^* + c_j a_j^*) e^{-is} + a_j b_j^* e^{2is} + a_j^* b_j e^{-2is}. \end{aligned} \quad (4.6)$$

Una vez creado el estado $|\beta(p)\rangle$ sobre los qubits de ancilla usando una rotación parametrizada, se puede usar el resto del circuito de la figura 4.2 para realizar el mapa dinámico sobre el qubit principal. Entonces, el circuito completo que implementa el mapa dinámico de Pauli sobre un qubit es el de la figura 4.5.

4.2.3. Ejemplos

En esta sección volvemos a estudiar algunos de los ejemplos de mapas dinámicos vistos antes y demostramos que cumplen con las condiciones del teorema 9 para poder ser implementados con una sola rotación parametrizada. Además, en cada uno de los ejemplos encontramos las expresiones de las matrices A y B que se tienen que usar en el circuito 4.5.

- **Mapa PF:** El mapa dinámico está dado por

$$\varepsilon_p(\rho) = (1 - p)\rho + p\sigma_3\rho\sigma_3,$$

con $p \in [0, 1]$. Por lo tanto, para implementarlo en un circuito cuántico, necesitamos crear la curva de estados $|\beta(p)\rangle$ sobre los qubits de ancilla, donde $|\beta_0(p)|^2 = (1 - p)$, $|\beta_1(p)|^2 = 0$, $|\beta_2(p)|^2 = 0$, $|\beta_3(p)|^2 = p$. En particular,

veremos que se puede crear la curva de estados cuando se toman las β_j reales, es decir, la curva:

$$|\beta(p)\rangle = (\sqrt{1-p}, 0, 0, \sqrt{p}).$$

Esta curva de estados se puede reescribir como

$$\begin{aligned} |\beta(p)\rangle &= \cos(s)(1, 0, 0, 0) + \sin(s)(0, 0, 0, 1) \\ &= e^{is} \left(\frac{1}{2}, 0, 0, \frac{-i}{2} \right) + e^{-is} \left(\frac{1}{2}, 0, 0, \frac{i}{2} \right), \end{aligned}$$

donde $s = \arcsin(\sqrt{p})$. Por lo tanto, se satisface la hipótesis del teorema 9 con los vectores $\vec{a} = (\frac{1}{2}, 0, 0, \frac{-i}{2})$, $\vec{b} = (\frac{1}{2}, 0, 0, \frac{i}{2})$ y $\vec{c} = \vec{0}$, que se puede ver que son ortogonales y cumplen $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$. Entonces, según el teorema 9, las matrices A y B para construir el circuito están dadas por:

$$\begin{aligned} A &= \begin{pmatrix} * & * & b_0/|\vec{b}| & a_0/|\vec{a}| \\ * & * & b_1/|\vec{b}| & a_1/|\vec{a}| \\ * & * & b_2/|\vec{b}| & a_2/|\vec{a}| \\ * & * & b_3/|\vec{b}| & a_3/|\vec{a}| \end{pmatrix} = \begin{pmatrix} * & * & 1/\sqrt{2} & 1/\sqrt{2} \\ * & * & 0 & 0 \\ * & * & 0 & 0 \\ * & * & i/\sqrt{2} & -i/\sqrt{2} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 & 0 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 & 0 \\ 0 & 0 & i/\sqrt{2} & -i/\sqrt{2} \end{pmatrix}, \\ B &= \begin{pmatrix} 0 & * & * & * \\ |\vec{c}| & * & * & * \\ |\vec{b}| & * & * & * \\ |\vec{a}| & * & * & * \end{pmatrix} = \begin{pmatrix} 0 & * & * & * \\ 0 & * & * & * \\ 1/\sqrt{2} & * & * & * \\ 1/\sqrt{2} & * & * & * \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 & 0 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 & 0 \end{pmatrix}, \end{aligned}$$

donde al final se completaron los asteriscos asegurando que A y B sean unitarias. Luego, el circuito que implementa este canal sobre el qubit principal es el de la figura 4.5 con las matrices A y B definidas de esta manera. Además, los mapas BF y BFP también se pueden implementar con un circuito de una rotación parametrizada de forma análoga.

- **Mapa depolarizador:** El mapa dinámico depolarizador está dado por

$$\varepsilon_p(\rho) = (1 - 3p/4)\sigma_0\rho\sigma_0 + (p/4)\sigma_1\rho\sigma_1 + (p/4)\sigma_2\rho\sigma_2 + (p/4)\sigma_3\rho\sigma_3, \quad (4.7)$$

con $p \in [0, 1]$. Al igual que en el ejemplo anterior, para ejecutarlo en un circuito con una rotación parametrizada, necesitamos crear la curva de estados $|\beta(p)\rangle$ sobre los qubits de ancilla, donde $|\beta_0(p)|^2 = (1 - 3p/4)$, $|\beta_1(p)|^2 = p/4$, $|\beta_2(p)|^2 = p/4$, $|\beta_3(p)|^2 = p/4$. En particular, veremos que se puede crear el estado con β_j reales, que es:

$$|\beta(p)\rangle = (\sqrt{1-3p/4}, \sqrt{p/4}, \sqrt{p/4}, \sqrt{p/4}).$$

Después de analizar esta expresión, notamos que se puede escribir como

$$\begin{aligned} |\beta(p)\rangle &= \cos s (1, 0, 0, 0) + \sin s (0, 1/\sqrt{3}, 1/\sqrt{3}, 1/\sqrt{3}) \\ &= e^{is} \left(\frac{1}{2}, \frac{-i}{2\sqrt{3}}, \frac{-i}{2\sqrt{3}}, \frac{-i}{2\sqrt{3}} \right) + e^{-is} \left(\frac{1}{2}, \frac{i}{2\sqrt{3}}, \frac{i}{2\sqrt{3}}, \frac{i}{2\sqrt{3}} \right), \end{aligned}$$

donde $s = \arcsin(\sqrt{3p/4})$. Por lo tanto, los vectores son $\vec{a} = \left(\frac{1}{2}, \frac{-i}{2\sqrt{3}}, \frac{-i}{2\sqrt{3}}, \frac{-i}{2\sqrt{3}} \right)$, $\vec{b} = \left(\frac{1}{2}, \frac{i}{2\sqrt{3}}, \frac{i}{2\sqrt{3}}, \frac{i}{2\sqrt{3}} \right)$, $\vec{c} = \vec{0}$ y se puede ver que cumplen con las condiciones del teorema 9. Entonces, sabemos que vamos a poder implementar el mapa en un circuito usando una sola compuerta parametrizada, donde las matrices A y B están dadas por:

$$A = \begin{pmatrix} * & * & b_0/|\vec{b}| & a_0/|\vec{a}| \\ * & * & b_1/|\vec{b}| & a_1/|\vec{a}| \\ * & * & b_2/|\vec{b}| & a_2/|\vec{a}| \\ * & * & b_3/|\vec{b}| & a_3/|\vec{a}| \end{pmatrix} = \begin{pmatrix} * & * & 1/\sqrt{2} & 1/\sqrt{2} \\ * & * & i/\sqrt{6} & -i/\sqrt{6} \\ * & * & i/\sqrt{6} & -i/\sqrt{6} \\ * & * & i/\sqrt{6} & -i/\sqrt{6} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{6} & -1/\sqrt{2} & i/\sqrt{6} & -i/\sqrt{6} \\ 1/\sqrt{6} & 1/\sqrt{2} & i/\sqrt{6} & -i/\sqrt{6} \\ -\sqrt{2/3} & 0 & i/\sqrt{6} & -i/\sqrt{6} \end{pmatrix}, \quad (4.8)$$

$$B = \begin{pmatrix} 0 & * & * & * \\ |\vec{c}| & * & * & * \\ |\vec{b}| & * & * & * \\ |\vec{a}| & * & * & * \end{pmatrix} = \begin{pmatrix} 0 & * & * & * \\ 0 & * & * & * \\ 1/\sqrt{2} & * & * & * \\ 1/\sqrt{2} & * & * & * \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 & 0 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 & 0 \end{pmatrix}. \quad (4.9)$$

Al igual que en el ejemplo anterior, se reemplazaron los asteriscos de forma que A y B sean unitarias. Nuevamente, el circuito que ejecuta el mapa sobre un qubit se puede construir como en la figura 4.5 usando estas matrices. Más adelante construiremos el circuito y lo pondremos a prueba.

- **Mapa parabólico σ_3 :** Este mapa dinámico en la representación de Kraus está dado por

$$\epsilon_p(\rho) = \frac{1}{4}(1-p)^2\rho + \frac{1}{4}(1-p^2)\sigma_1\rho\sigma_1 + \frac{1}{4}(1-p^2)\sigma_2\rho\sigma_2 + \frac{1}{4}(1+p)^2\sigma_3\rho\sigma_3,$$

con $p \in [-1, 1]$. Al igual que en los dos ejemplos anteriores, para ver que se puede implementar, veremos que es posible crear la curva de estados:

$$|\beta(p)\rangle = \left(\frac{1}{2}(1-p), \frac{1}{2}\sqrt{1-p^2}, \frac{1}{2}\sqrt{1-p^2}, \frac{1}{2}(1+p) \right) \quad (4.10)$$

sobre los dos qubits de ancilla. Afortunadamente, esta curva se puede reescribir

como

$$\begin{aligned} |\beta(p)\rangle &= \left(\frac{1}{2}, 0, 0, \frac{1}{2}\right) + \cos(s) \left(0, \frac{1}{2}, \frac{1}{2}, 0\right) + \sin(s) \left(\frac{-1}{2}, 0, 0, \frac{1}{2}\right) \\ &= \left(\frac{1}{2}, 0, 0, \frac{1}{2}\right) + e^{is} \left(\frac{i}{4}, \frac{1}{4}, \frac{1}{4}, \frac{-i}{4}\right) + e^{-is} \left(\frac{-i}{4}, \frac{1}{4}, \frac{1}{4}, \frac{i}{4}\right), \end{aligned}$$

con $s = \arcsin(p)$. Por lo tanto, los vectores son $\vec{a} = (\frac{i}{4}, \frac{1}{4}, \frac{1}{4}, \frac{-i}{4})$, $\vec{b} = (\frac{-i}{4}, \frac{1}{4}, \frac{1}{4}, \frac{i}{4})$, $\vec{c} = (\frac{1}{2}, 0, 0, \frac{1}{2})$ y se puede ver que cumplen con las condiciones del teorema 9. Entonces, es posible hacer este mapa en un circuito usando una sola rotación parametrizada, con las matrices A y B definidas como:

$$A = \begin{pmatrix} * & c_0/|\vec{c}| & b_0/|\vec{b}| & a_0/|\vec{a}| \\ * & c_1/|\vec{c}| & b_1/|\vec{b}| & a_1/|\vec{a}| \\ * & c_2/|\vec{c}| & b_2/|\vec{b}| & a_2/|\vec{a}| \\ * & c_3/|\vec{c}| & b_3/|\vec{b}| & a_3/|\vec{a}| \end{pmatrix} = \begin{pmatrix} * & 1/\sqrt{2} & -i/2 & i/2 \\ * & 0 & 1/2 & 1/2 \\ * & 0 & 1/2 & 1/2 \\ * & 1/\sqrt{2} & i/2 & -i/2 \end{pmatrix} = \begin{pmatrix} 0 & 1/\sqrt{2} & -i/2 & i/2 \\ -1/\sqrt{2} & 0 & 1/2 & 1/2 \\ 1/\sqrt{2} & 0 & 1/2 & 1/2 \\ 0 & 1/\sqrt{2} & i/2 & -i/2 \end{pmatrix}, \quad (4.11)$$

$$B = \begin{pmatrix} 0 & * & * & * \\ |\vec{c}| & * & * & * \\ |\vec{b}| & * & * & * \\ |\vec{a}| & * & * & * \end{pmatrix} = \begin{pmatrix} 0 & * & * & * \\ 1/\sqrt{2} & * & * & * \\ 1/2 & * & * & * \\ 1/2 & * & * & * \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1/\sqrt{2} & 0 & 0 & -1/\sqrt{2} \\ 1/2 & 0 & 1/\sqrt{2} & 1/2 \\ 1/2 & 0 & -1/\sqrt{2} & 1/2 \end{pmatrix}. \quad (4.12)$$

Luego, el circuito que ejecuta al mapa sobre un qubit se puede construir como en la figura 4.5 usando estas matrices.

- **Mapa que no se puede implementar con una rotación parametrizada:** En este ejemplo revisamos un mapa dinámico en el cual no se cumplen las condiciones del teorema 9 y entonces no se puede realizar usando un circuito de una rotación parametrizada como el de la figura 4.5.

El mapa que consideramos está dado por:

$$\varepsilon_p(\rho) = (1-p)^3 \rho + p \sigma_1 \rho \sigma_1 + (2p - 3p^2 + p^3) \sigma_2 \rho \sigma_2,$$

con $p \in [0, 1]$. Para empezar, notamos que esta expresión efectivamente representa un canal de Pauli para todos los valores de p en $[0, 1]$, ya que $\alpha_0(p) = (1-p)^3$, $\alpha_1(p) = p$, $\alpha_2(p) = 2p - 3p^2 + p^3$, $\alpha_3(p) = 0$ cumplen con las condiciones de las ecuaciones 3.2 y 3.3.

Como hemos visto, para ejecutar este mapa necesitamos crear la curva de estados $|\beta(p)\rangle$ sobre los qubits de ancilla, donde $|\beta_j(p)|^2 = \alpha_j(p)$. De acuerdo al teorema 9, esta curva se puede crear con una rotación parametrizada solamente si existen

vectores $\vec{a}, \vec{b}, \vec{c}$ ortogonales que cumplen $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$ y una función $s(p)$ tales que:

$$|\beta(p)\rangle = \vec{c} + \vec{a}e^{is} + \vec{b}e^{-is}.$$

Es decir, considerando cada entrada por separado, se debe de cumplir que:

$$\beta_0(p) = c_0 + a_0e^{is} + b_0e^{-is} \quad (4.13)$$

$$\beta_1(p) = c_1 + a_1e^{is} + b_1e^{-is} \quad (4.14)$$

$$\beta_2(p) = c_2 + a_2e^{is} + b_2e^{-is} \quad (4.15)$$

$$\beta_3(p) = c_3 + a_3e^{is} + b_3e^{-is}, \quad (4.16)$$

lo cual vamos a probar que es imposible. Empezamos notando que como las funciones $\{1, e^{is}, e^{-is}\}$ son linealmente independientes y $\beta_3(p) = 0$ (ya que $\alpha_3(p) = 0$), la ecuación 4.16 implica que $c_3 = a_3 = b_3 = 0$, por lo que podemos ignorar esa ecuación. Ahora calculamos la norma cuadrada de cada lado de estas expresiones, con lo que obtenemos:

$$|\beta_0(p)|^2 = (1-p)^3 = \mu_0 + e^{-2is}\nu_0^* + e^{2is}\nu_0 + e^{-is}\kappa_0^* + e^{is}\kappa_0 \quad (4.17)$$

$$|\beta_1(p)|^2 = p = \mu_1 + e^{-2is}\nu_1^* + e^{2is}\nu_1 + e^{-is}\kappa_1^* + e^{is}\kappa_1 \quad (4.18)$$

$$|\beta_2(p)|^2 = 2p - 3p^2 + p^3 = \mu_2 + e^{-2is}\nu_2^* + e^{2is}\nu_2 + e^{-is}\kappa_2^* + e^{is}\kappa_2, \quad (4.19)$$

donde se define $\mu_j := |a_j|^2 + |b_j|^2 + |c_j|^2$, $\nu_j := a_j b_j^*$, $\kappa_j := b_j^* c_j + c_j^* a_j$. Entonces $\mu_0 + \mu_1 + \mu_2 = |\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2$, $\nu_0 + \nu_1 + \nu_2 = \vec{a} \cdot \vec{b}$ y $\kappa_0 + \kappa_1 + \kappa_2 = \vec{b} \cdot \vec{c} + \vec{c} \cdot \vec{a}$. Luego, como $\vec{a}, \vec{b}, \vec{c}$ tienen que ser ortogonales y satisfacen $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$, concluimos que se debe de cumplir:

$$\mu_0 + \mu_1 + \mu_2 = 1, \quad \nu_0 + \nu_1 + \nu_2 = 0, \quad \kappa_0 + \kappa_1 + \kappa_2 = 0, \quad (4.20)$$

Por otro lado, utilizando 4.18, llegamos a que:

$$1 - p = 1 - \mu_1 - e^{-2is}\nu_1^* - e^{2is}\nu_1 - e^{-is}\kappa_1^* - e^{is}\kappa_1, \quad (4.21)$$

y elevando al cubo nos queda:

$$(1-p)^3 = -e^{6is}\nu_1^3 + K(s), \quad (4.22)$$

donde $K(s)$ es una suma de términos proporcionales a $e^{-6is}, e^{-5is}, e^{-4is}, e^{-3is}, e^{-2is}, e^{-is}, 1, e^{is}, e^{2is}, e^{3is}, e^{4is}, e^{5is}$, pero ninguno proporcional a e^{6is} , ya que al elevar 4.21 al cubo, el único término con e^{6is} es $-e^{6is}\nu_1^3$.

Luego, en 4.22 y 4.17 tenemos dos expresiones de $(1-p)^3$, por lo que las podemos igualar. Como las funciones $\{e^{ims}\}_{m \in \mathbb{Z}}$ son linealmente independientes, al igualar las expresiones, se iguala cada coeficiente de la forma e^{ims} por separado. Entonces, como solamente 4.22 tiene un término proporcional a e^{6is} y 4.17 no, su coeficiente tiene que ser 0, por lo que $-\nu_1^3 = 0$, lo que implica que $\boxed{\nu_1 = 0}$.

Por lo tanto, la expresión 4.21 se simplifica como:

$$1 - p = 1 - \mu_1 - e^{-is} \kappa_1^* - e^{is} \kappa_1, \quad (4.23)$$

y al elevar al cubo, obtenemos ahora:

$$(1 - p)^3 = -e^{3is} \kappa_1^3 + G(s),$$

donde $G(s)$ es una suma de términos que son proporcionales a $e^{-3is}, e^{-2is}, e^{-is}, 1, e^{is}, e^{2is}$, pero ninguno proporcional a e^{3is} , ya que al elevar 4.23 al cubo, el único término con e^{3is} es $-e^{3is} \kappa_1^3$.

Luego, igualamos las expresiones de 4.23 y 4.17. Como las exponenciales son linealmente independientes, se pueden igualar término a término. En 4.17 no hay término proporcional a e^{3is} , mientras que en 4.23 sí aparece ese término y tiene coeficiente $-\kappa_1^3$. Entonces, al igualar las dos expresiones concluimos que $-\kappa_1^3 = 0 \Rightarrow \boxed{\kappa_1 = 0}$.

Finalmente, hemos mostrado entonces que $\nu_1 = 0$ y que $\kappa_1 = 0$, lo que implica que la relación 4.18 se reduce a simplemente:

$$p = \mu_1.$$

Pero esto es imposible, ya que μ_1 es constante que no depende del parámetro p . Por lo tanto, concluimos que este mapa no se puede implementar con un circuito de una rotación parametrizada, ya que suponer que se puede lleva a una contradicción. En la figura 4.6 se muestra el dibujo de este mapa dinámico en el tetraedro de canales de Pauli.

4.3. Conjunto de mapas dinámicos de Pauli simulables

En el teorema 9 encontramos condiciones que debe de cumplir un mapa dinámico de Pauli para poder implementarlo usando una sola rotación parametrizada. En esta sección vamos a estudiar el conjunto de todos los mapas dinámicos que cumplen con estas condiciones, con el objetivo de poder producir de forma metódica mapas que se puedan implementar de esta manera.

Para empezar, notamos que en realidad no todos los mapas que cumplen con las condiciones del teorema 9 son verdaderamente mapas dinámicos de Pauli según la definición 11. Esto porque falta considerar la condición de que el mapa dinámico debe de iniciar en la identidad (es decir, si el parámetro p varía sobre un intervalo $[a, b]$, se debe de cumplir que $\varepsilon_a(\rho) = \rho$). Hay muchas posibles operaciones que cumplen con el teorema 9 pero no pasan por la identidad y entonces técnicamente no son mapas dinámicos de Pauli.

Dicho esto, recordamos ahora que el teorema 9 dice que un mapa dinámico $\varepsilon_p(\rho) = \alpha_0(p)\rho + \alpha_1(p)\sigma_1\rho\sigma_1 + \alpha_2(p)\sigma_2\rho\sigma_2 + \alpha_3(p)\sigma_3\rho\sigma_3$ se puede implementar usando una sola

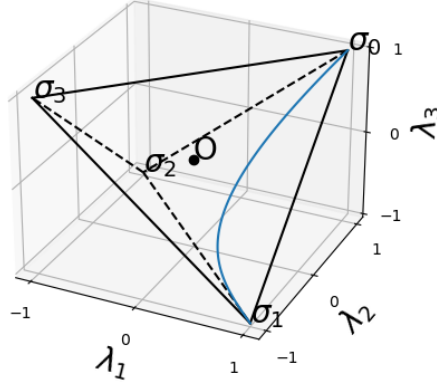


Figura 4.6: Mapa dinámico $\varepsilon_p(\rho) = (1-p)^3\rho + p\sigma_1\rho\sigma_1 + (2p-3p^2+p^3)\sigma_2\rho\sigma_2$, que es un ejemplo de un mapa que no se puede implementar con un circuito de una rotación parametrizada.

rotación parametrizada si se cumple que:

$$|\beta(p)\rangle = \vec{c} + e^{is}\vec{a} + e^{-is}\vec{b},$$

donde $|\beta_j(p)|^2 = \alpha_j(p)$, lo que significa que las $\alpha_j(p)$ del canal deben de tener la forma

$$\begin{aligned} \alpha_j(p) &= |\beta_j(p)|^2 \\ &= |c_j|^2 + |a_j|^2 + |b_j|^2 + (a_j c_j^* + c_j b_j^*)e^{is} + (b_j c_j^* + c_j a_j^*)e^{-is} + a_j b_j^* e^{2is} + a_j^* b_j e^{-2is}. \end{aligned} \quad (4.24)$$

Además, los vectores $\vec{a}, \vec{b}, \vec{c} \in \mathbb{C}^4$ son ortogonales, las normas $|\vec{a}|^2, |\vec{b}|^2, |\vec{c}|^2$ suman 1 y $s = s(p)$ es una función de p .

Para estudiar solamente los mapas que a parte de tener esta condición, inician en la identidad, agregaremos el requisito de que en $s = 0$ se cumpla que el canal correspondiente sea la identidad. Esto significa que se cumple que $\vec{c} + e^0\vec{a} + e^0\vec{b} = \vec{c} + \vec{a} + \vec{b} = (e^{i\theta}, 0, 0, 0)$ con $\theta \in [0, 2\pi]$, para que así en $s = 0$ el mapa dinámico tenga $\alpha_0 = 1, \alpha_1 = \alpha_2 = \alpha_3 = 0$ y entonces sea el canal identidad. Por lo tanto, hay que agregar esta condición adicional sobre los vectores $\vec{a}, \vec{b}, \vec{c}$.

Nota: Ya habíamos encontrado que el mapa parabólico σ_3 pasa por la identidad y sin embargo al escribirlo como $\vec{c} + e^{is}\vec{a} + e^{-is}\vec{b}$, no cumple con $\vec{c} + \vec{a} + \vec{b} = (e^{i\theta}, 0, 0, 0)$. Esto es por la parametrización particular que escogimos, pasaba por la identidad para un valor de s distinto a 0. Esto se puede arreglar con una reparametrización del tipo $s \rightarrow s + k$ con k una constante. Así que para estudiar el conjunto de los mapas que

pasan por la identidad, podemos fijarnos solamente en los que pasan cuando $s = 0$.

Entonces, juntando el teorema 9 con la nueva condición de que $\vec{c} + \vec{a} + \vec{b} = (e^{i\theta}, 0, 0, 0)$, concluimos que para elegir un mapa dinámico de Pauli que se pueda implementar con una rotación parametrizada, basta con escoger tres vectores $\vec{a}, \vec{b}, \vec{c}$ tales que:

- $\vec{a}, \vec{b}, \vec{c} \in \mathbb{C}^4$ son vectores ortogonales.
- Las normas cumplen que $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$.
- Se cumple que $\vec{c} + \vec{a} + \vec{b} = (e^{i\theta}, 0, 0, 0)$ con $\theta \in [0, 2\pi]$.

Para estudiar este conjunto de mapas dinámicos de forma sistemática, encontraremos una forma de escoger $\vec{a}, \vec{b}, \vec{c}$ tal que cumplan con todas estas condiciones. Los pasos para lograr esto se presentan a continuación:

1. El primer paso es escoger las normas de los vectores de tal manera que cumplan $|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2 = 1$. Notamos que esta condición significa que el vector $(|\vec{a}|, |\vec{b}|, |\vec{c}|) \in \mathbb{R}^3$ se encuentra en la esfera unitaria (y en particular en el octante positivo, ya que las normas son positivas o 0). Entonces, escoger estas normas se puede hacer eligiendo dos ángulos $\mu \in [0, \pi/2]$ y $\nu \in [0, \pi/2]$ y definiendo:

$$\begin{aligned} |\vec{a}| &= \sin \nu \cos \mu, \\ |\vec{b}| &= \sin \nu \sin \mu, \\ |\vec{c}| &= \cos \nu. \end{aligned}$$

2. Definir los vectores $\vec{a}' = (|\vec{a}|, 0, 0, 0)$, $\vec{b}' = (0, |\vec{b}|, 0, 0)$, $\vec{c}' = (0, 0, |\vec{c}|, 0)$. Notamos que estos vectores son ortogonales y cumplen que $|\vec{a}'| = |\vec{a}|$, $|\vec{b}'| = |\vec{b}|$, $|\vec{c}'| = |\vec{c}|$.
3. Escoger una matriz unitaria U tal que su primera fila sea $e^{i\theta}(|\vec{a}|, |\vec{b}|, |\vec{c}|, 0)$, con $\theta \in [0, 2\pi]$ y usar esta matriz para definir $\vec{a} = U\vec{a}'$, $\vec{b} = U\vec{b}'$, $\vec{c} = U\vec{c}'$.

De esta forma, como los vectores primados cumplen con las condiciones de norma y ortogonalidad, y U es unitaria, los vectores $\vec{a}, \vec{b}, \vec{c}$ también cumplirán estas condiciones. Además, que la primera fila de U sea $e^{i\theta}(|\vec{a}|, |\vec{b}|, |\vec{c}|, 0)$ asegura que $\vec{a} + \vec{b} + \vec{c} = (e^{i\theta}, 0, 0, 0)$, ya que:

$$\vec{a} + \vec{b} + \vec{c} = U(\vec{a}' + \vec{b}' + \vec{c}') = U(|\vec{a}|, |\vec{b}|, |\vec{c}|, 0)^T = (e^{i\theta}, 0, 0, 0)^T,$$

El último paso se debe a que al aplicar U al vector $(|\vec{a}|, |\vec{b}|, |\vec{c}|, 0)^T$, la primera entrada del resultado se obtiene con el producto de la primera fila de U por este vector. Por cómo escogimos la primera fila de U , esta primera entrada es $e^{i\theta}(|\vec{a}|^2 + |\vec{b}|^2 + |\vec{c}|^2) = e^{i\theta}$. Además, las demás entradas del resultado tienen que ser 0, ya que esa es la única forma de que el vector resultante tenga una norma de 1, lo cual se debe de cumplir ya que U preserva la norma y el vector $(|\vec{a}|, |\vec{b}|, |\vec{c}|, 0)^T$ tiene norma 1.

4. MAPAS DINÁMICOS DE PAULI

Entonces, al escoger una matriz U unitaria con la primera fila como se mencionó, nos aseguramos de conseguir vectores $\vec{a}, \vec{b}, \vec{c}$ que cumplan con todas las condiciones.

En conclusión, para elegir un mapa dinámico de Pauli que se pueda implementar con una rotación parametrizada, se tiene la libertad de escoger dos ángulos μ, ν para definir las normas de los vectores $\vec{a}, \vec{b}, \vec{c}$ y luego la libertad de escoger el ángulo θ y las 3 filas restantes de una matriz unitaria U cuya primera fila es $e^{i\theta}(|\vec{a}|, |\vec{b}|, |\vec{c}|, 0)$.

4.3.1. Ejemplos

En la sección 4.2.3 se mostró que el mapa PF cumple con las condiciones para ser implementado con una compuerta parametrizada. Esto se debe a que se probó que la curva de estados $|\beta(p)\rangle$ para este mapa satisface el teorema 9, pues tiene la forma:

$$|\beta(p)\rangle = \vec{a}e^{is} + \vec{b}e^{-is} + \vec{c},$$

con $\vec{a} = (\frac{1}{2}, 0, 0, \frac{-i}{2})$, $\vec{b} = (\frac{1}{2}, 0, 0, \frac{i}{2})$, $\vec{c} = \vec{0}$ y $s = \arcsin(\sqrt{p})$. Como este ejemplo cumple con las condiciones para ser simulado con una rotación parametrizada, deberíamos de poder encontrarlo en la forma en que se detalla en la sección anterior para alguna elección adecuada de los ángulos μ, ν, θ y la matriz U . Para comprobar esto, vemos que si escogemos los ángulos

$$\mu = \pi/4, \quad \nu = \pi/2, \quad \theta = 0,$$

y definimos la matriz U como

$$U = \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 & 0 \\ 0 & 0 & 1/\sqrt{2} & 1/\sqrt{2} \\ 0 & 0 & -1/\sqrt{2} & 1/\sqrt{2} \\ -i/\sqrt{2} & i/\sqrt{2} & 0 & 0 \end{pmatrix},$$

entonces los vectores $\vec{a}, \vec{b}, \vec{c}$ construidos con el procedimiento mencionado en la sección anterior quedan como $(\frac{1}{2}, 0, 0, \frac{-i}{2})$, $(\frac{1}{2}, 0, 0, \frac{i}{2})$, $\vec{0}$ respectivamente, lo que como se menciona antes, corresponde con un mapa PF.

Similarmente, hemos estudiado otros mapas de Pauli (BF, BPF, depolarizador y parabólico) que probamos que se pueden implementar con un circuito de una rotación parametrizada. Como se esperaba, estos otros mapas también se pueden conseguir con los pasos de la sección anterior para selecciones adecuadas de los ángulos μ, ν, θ y de la matriz unitaria U . Esto es de esperarse ya que con los pasos de la sección anterior se puede crear cualquier mapa de Pauli que se implementable con un circuito de una rotación parametrizada.

Más allá de estos ejemplos ya conocidos, la sección anterior describe una forma de encontrar infinitos ejemplos de mapas dinámicos que empiezan en la identidad y

que se pueden implementar con una rotación parametrizada. Para ello sólo hay que elegir ángulos μ, ν, θ , una matriz unitaria U y seguir los pasos de la sección anterior para conseguir a los vectores $\vec{a}, \vec{b}, \vec{c}$ que cumplen con las condiciones del teorema 9 y por tanto definen a un mapa de Pauli que se puede simular con una rotación parametrizada.

Con el objetivo de visualizar algunos otros mapas de Pauli que cumplan estas condiciones, se siguieron estos pasos con elecciones al azar de los ángulos μ, ν, θ y de la matriz U . Estas elecciones dieron lugar a vectores $\vec{a}, \vec{b}, \vec{c}$. Los resultados de hacer esto en tres ejemplos aleatorios fueron los siguientes:

1. Los ángulos escogidos en este caso fueron:

$$\mu = 0.865191 \quad , \quad \nu = 1.112356 \quad , \quad \theta = 1.827808,$$

luego, la matriz U se eligió al azar a partir de su primer renglón $e^{i\theta}(|\vec{a}\rangle, |\vec{b}\rangle, |\vec{c}\rangle, 0)$:

$$U = \begin{pmatrix} -0.148 + 0.562i & -0.173 + 0.660i & -0.112 + 0.428i & 0 \\ 0.0064 - 0.746i & 0.127 + 0.579i & -0.280 + 8.74 \times 10^{-2}i & 0.011 + 0.0039i \\ -0.199 - 0.236i & -0.374 + 0.201i & 0.838 - 3.33 \times 10^{-4}i & 0.148 + 0.030i \\ 0.066 + 0.040i & 0.026 - 0.043i & -0.126 + 0.013i & 0.988 + 0.055i \end{pmatrix}.$$

Finalmente, con esta matriz quedan definidos los vectores $\vec{a}, \vec{b}, \vec{c}$ según los pasos mencionados antes y son:

$$\begin{aligned} \vec{a} &= (-0.086 + 0.327i, 0.037 - 0.434i, -0.116 - 0.137i, 0.038 + 0.024i), \\ \vec{b} &= (-0.119 + 0.451i, 0.087 + 0.395i, -0.255 + 0.137i, 0.018 - 0.029i), \\ \vec{c} &= (-0.050 + 0.190i, -0.124 + 0.038i, 0.371 - 1.47 \times 10^{-4}i, -0.056 + 5.52 \times 10^{-3}i). \end{aligned}$$

Usando estos vectores, las α_j que definen al mapa dinámico de Pauli resultante están dadas por la ecuación 4.24. El mapa forma la curva de la figura 4.7 sobre el tetraedro y sabemos por construcción que se puede implementar con una sola rotación parametrizada.

2. Los ángulos escogidos al azar en este caso fueron:

$$\mu = 0.7176 \quad , \quad \nu = 1.0197 \quad , \quad \theta = 1.7498,$$

luego, partiendo del primer renglón $e^{i\theta}(|\vec{a}\rangle, |\vec{b}\rangle, |\vec{c}\rangle, 0)$, la matriz U se eligió al azar como:

$$U = \begin{pmatrix} -0.114 + 0.632i & -0.099 + 0.551i & -0.932 + 0.515i & 0 \\ -0.667 - 0.153i & 0.116 + 0.184i & 0.693 - 9.621 \times 10^{-3}i & -3.508 \times 10^{-3} + 0.060i \\ -0.099 + 0.224i & -0.041 - 0.526i & 0.165 + 0.288i & -0.234 - 0.705i \\ 0.197 + 0.146i & -0.568 - 0.196i & 0.366 + 0.031i & 0.662 + 0.076i \end{pmatrix}.$$

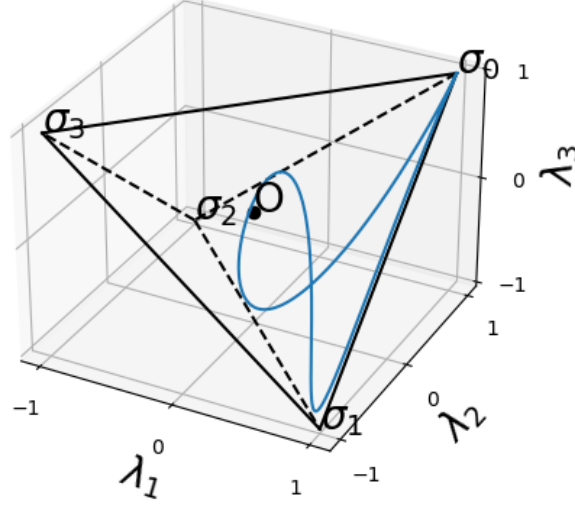


Figura 4.7: Ejemplo 1 de mapa dinámico de Pauli que empieza en la identidad y se puede construir con una sola compuerta parametrizada.

Finalmente, con esta matriz se definen los vectores $\vec{a}, \vec{b}, \vec{c}$ y quedan:

$$\begin{aligned}\vec{a} &= (-0.073 + 0.405i, -0.428 - 0.098i, -0.063 + 0.144i, 0.127 + 0.093i), \\ \vec{b} &= (-0.056 + 0.309i, 0.065 + 0.103i, -0.023 - 0.294i, -0.318 - 0.110i), \\ \vec{c} &= (-0.049 + 0.270i, 0.363 - 0.005i, 0.086 + 0.151i, 0.192 + 0.016i).\end{aligned}$$

Al igual que antes, las α_j del mapa dinámico de Pauli resultante están dadas por la ecuación 4.24 y el mapa forma la curva de la figura 4.8 sobre el tetraedro, la cual entonces se puede implementar con una sola rotación parametrizada.

3. Los ángulos escogidos en este caso fueron:

$$\mu = 1.3285 \quad , \quad \nu = 0.5938 \quad , \quad \theta = 0.5794,$$

luego, la matriz U se eligió al azar como:

$$U = \begin{pmatrix} 0.112 + 0.073i & 0.454 + 0.297i & 0.693 + 0.454i & 0 \\ -0.523 - 0.275i & -0.184 - 0.445i & 0.205 + 0.336i & -0.427 + 0.285i \\ -0.346 - 0.071i & 0.617 + 0.164i & -0.348 - 0.096i & 0.162 + 0.558i \\ 0.119 - 0.703i & -0.250 + 0.052i & 0.145 + 0.080i & 0.608 + 0.174i \end{pmatrix}.$$

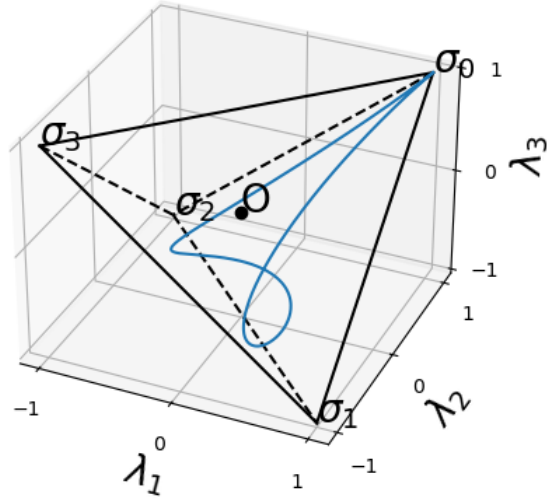


Figura 4.8: Ejemplo 2 de mapa dinámico de Pauli que empieza en la identidad y se puede construir con una sola compuerta parametrizada

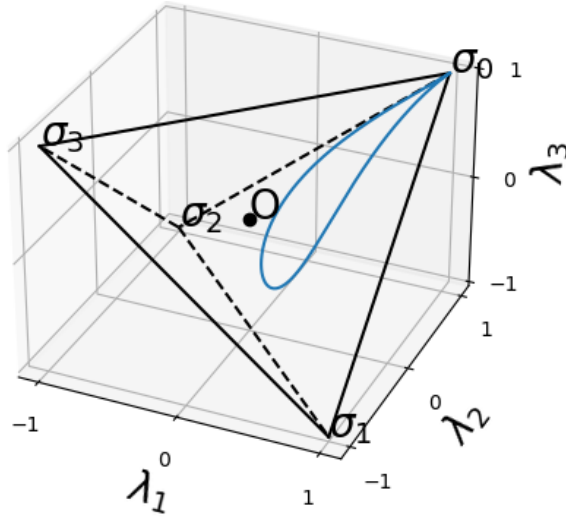


Figura 4.9: Ejemplo 3 de mapa dinámico de Pauli que empieza en la identidad y se puede construir con una sola compuerta parametrizada

Finalmente, con esta matriz se definen los vectores $\vec{a}, \vec{b}, \vec{c}$ y quedan:

$$\begin{aligned}\vec{a} &= (0.015 + 0.010i, -0.070 - 0.037i, -0.046 - 0.010i, 0.016 - 0.094i), \\ \vec{b} &= (0.247 + 0.162i, -0.100 - 0.242i, 0.335 + 0.089i, -0.136 + 0.028i), \\ \vec{c} &= (0.575 + 0.376i, 0.170 + 0.279i, -0.289 - 0.0792i, 0.120 + 0.066i).\end{aligned}$$

Nuevamente, el mapa dinámico de Pauli resultante está dado por las α_j como se consiguen en la ecuación 4.24 y forma la curva de la figura 4.9 sobre el tetraedro.

Simulación en una Computadora Cuántica

En este capítulo aprovecharemos que recientemente se abrió la posibilidad de mandar algoritmos cuánticos a computadoras cuánticas reales. En particular, IBM tiene a la disposición del público general varias computadoras cuánticas a las cuales se les pueden enviar circuitos y luego analizar los resultados obtenidos. Gracias a esta novedosa oportunidad, implementaremos en las computadoras de IBM algunos de los canales y mapas dinámicos de Pauli que definimos en el capítulo anterior.

En particular, mandaremos a las computadoras una selección de canales de Pauli elegidos del tetraedro y además varios de los canales que forman parte de los mapas dinámicos depolarizador y parabólico σ_3 . Luego, para cada uno de los canales realizados, encontraremos el proceso por el que pasó el qubit principal del circuito. Esto con la finalidad de comparar este proceso con el canal de Pauli que se quería implementar y cuantificar qué tan cerca se encuentra de realizar el canal definido teóricamente.

Para revisar esto, en cada canal implementado realizaremos un método conocido como Quantum Process Tomography, que definimos en la sección siguiente. En la forma particular en la que usaremos este método, nos servirá para conocer la matriz de Choi del proceso por el que pasa el qubit principal en la computadora cuántica. Luego podremos comparar esta matriz de Choi con la esperada teóricamente para el canal de Pauli y así cuantificar la fidelidad con la que la computadora realizó el canal.

Antes de entrar en detalle con esto, en la sección [5.1](#) damos un pequeño contexto sobre la computación cuántica y en específico sobre los procesadores cuánticos que tiene disponibles IBM y algunas de las cosas que se ha hecho con ellos.

5.1. Introducción

Como se mencionó en la introducción de esta tesis, en los últimos años la computación cuántica ha ganado mucha atención y ha generado grandes expectativas en la

comunidad científica y tecnológica. Esto es debido a que se pueden aprovechar propiedades intrínsecamente cuánticas como la superposición y el entrelazamiento para realizar ciertas operaciones de manera mucho más rápida y eficiente que en las computadoras clásicas [1, 2].

Aunque la tecnología de la computación cuántica todavía está en sus primeras etapas de desarrollo, se han realizado grandes avances en la construcción de procesadores cuánticos y en la comprensión de cómo aprovechar su capacidad para resolver problemas complejos. Grandes empresas como Google [26], IBM [27] y Microsoft [28], así como startups emergentes [29, 30, 31, 32] están invirtiendo en la investigación y el desarrollo de la computación cuántica, y se espera que en los próximos años veamos avances significativos.

Uno de los problemas más desafiantes en la fabricación de computadoras cuánticas es la construcción de qubits de alta fidelidad. Esto se debe a que si el qubit no está lo suficientemente aislado de su ambiente, sufre decoherencia, que es la pérdida de información cuántica hacia el ambiente [33]. Sin embargo, si el qubit está demasiado aislado de su ambiente, será difícil de modificar con operaciones cuánticas y de medir, ya que esto requiere interactuar con él. Algunas de las barreras para la construcción de computadoras cuánticas son construir qubits con tiempos de coherencia altos, que tengan errores pequeños al realizar operaciones o medir y que sean escalables (es decir, que se pueda aumentar el número de qubits fácilmente y sin incrementar las tasas de error).

Existen varias propuestas de sistemas físicos para que actúen como qubits. Estos sistemas deben de tener al menos dos niveles de energía (que correspondan con los estados que denotamos por $|0\rangle$ y $|1\rangle$) y además los qubits se deben de poder acoplar para poder tener sistemas de varios qubits. Más aún, se les deben de poder aplicar operaciones cuánticas y se deben de poder medir de manera fiable [4]. Las dos formas físicas más comunes de construir qubits son los qubits superconductores (en los que los qubits son implementados por circuitos eléctricos superconductores [34]) y los de iones atrapados (en los que los qubits son representados por los estados de iones atrapados en trampas electromagnéticas [35]).

En particular, IBM es una de las compañías líderes en la investigación y el desarrollo de la computación cuántica y utiliza qubits superconductores. En 2016 IBM presentó IBM Q System One, que es la primera computadora cuántica abierta al uso del público por medio de la nube [27]. Con esto, los usuarios pueden mandar circuitos cuánticos a cualquiera de los varios procesadores cuánticos que tiene disponibles IBM y recibir las mediciones obtenidas tras la realización del circuito en el procesador.

La compañía también ha lanzado una herramienta de software llamada Qiskit [17], la cual se puede descargar como una librería de Python. Qiskit permite construir circuitos cuánticos y prepararlos para ser mandados a uno de los procesadores de IBM.

Utilizando las computadoras cuánticas que ya están disponibles en el mercado como las de IBM, se han realizado varios algoritmos cuánticos de distintos tipos para simular sistemas cuánticos. Por ejemplo, en [36] se simulan cadenas de espines en un procesador de IBM de 27 qubits y se obtiene una fidelidad (concepto que se define en la

sección 5.3) de aproximadamente 0.98. Por otro lado, en [14] se simula un rotor pateado cuántico [15] en un procesador de 2 qubits y se obtienen fidelidades de aproximadamente 0.87. Continuando con este tipo de trabajos, en esta sección se simula el algoritmo particular propuesto en esta tesis para realizar canales de Pauli.

5.2. Definiciones

En esta sección definiremos algunos de los conceptos básicos para analizar circuitos cuánticos. En particular, definiremos los métodos de Quantum State Tomography y Quantum Process Tomography utilizados respectivamente para determinar estados y procesos cuánticos desconocidos. Antes de definir el método de Quantum Process Tomography (que es lo que en realidad usaremos), será necesario definir Quantum State Tomography, lo que hacemos a continuación.

Definición 13 Quantum State Tomography (QST): *Es un procedimiento para aproximar un estado cuántico desconocido a partir de realizar varias mediciones en distintas bases [4] (realizadas en varias copias de este estado, ya que al hacer una medición, perdemos el estado que queremos medir).*

A continuación se detalla cómo funciona el método de Quantum State Tomography (QST) para determinar el estado de un sistema de un qubit [4],[17] (aunque se puede generalizar a sistemas más complicados). Empezamos con un qubit que se encuentra en cierto estado desconocido ρ y nuestro objetivo es determinar ρ a partir de varias mediciones en distintas copias de este estado.

Para lograr este objetivo, empezamos recordando que el estado ρ del qubit se puede escribir en general como indica la ecuación 2.16:

$$\rho = \frac{\sigma_0 + \text{tr}(\sigma_1\rho)\sigma_1 + \text{tr}(\sigma_2\rho)\sigma_2 + \text{tr}(\sigma_3\rho)\sigma_3}{2},$$

por lo que si conocemos las trazas $\text{tr}(\sigma_1\rho)$, $\text{tr}(\sigma_2\rho)$, $\text{tr}(\sigma_3\rho)$, podemos determinar el estado ρ . Por lo tanto, el objetivo de QST será aproximar con precisión estas trazas para luego reconstruir el estado ρ que queremos conocer. Notamos que por el postulado 2 de operadores de densidad de la sección 2.6.3, sabemos que $\text{tr}(\sigma_1\rho)$ es simplemente el valor esperado $\langle\sigma_1\rangle$ para el estado ρ y similarmente $\text{tr}(\sigma_2\rho)$ y $\text{tr}(\sigma_3\rho)$ son los valores esperados $\langle\sigma_2\rangle$ y $\langle\sigma_3\rangle$ respectivamente. Por lo tanto, para encontrar ρ sólo será necesario aproximar los valores esperados de $\sigma_1, \sigma_2, \sigma_3$ en este estado.

Para obtener cada uno de estos valores esperados, será necesario preparar y medir ρ varias veces en una base distinta. En el circuito 5.1 se muestran las operaciones que hay que realizar para obtener cada valor esperado como se explica a continuación [17], [37]:

- $\langle\sigma_3\rangle$: Para aproximar este valor esperado lo que haremos es directamente medir al qubit en la base computacional $\{|0\rangle, |1\rangle\}$. Hecho esto, el estado final será $|0\rangle$ ó $|1\rangle$. En particular, dependiendo de cuál sea el estado ρ del qubit, tendremos una probabilidad $P(0)$ de obtener $|0\rangle$ y $P(1)$ de obtener $|1\rangle$.

Notamos ahora que $\{|0\rangle, |1\rangle\}$ es la base de eigenvectores de σ_3 , con eigenvalores 1 y -1 respectivamente. Esto quiere decir que el valor del observable σ_3 en el estado $|0\rangle$ es 1 y el valor en el estado $|1\rangle$ es -1 . El valor esperado de σ_3 consiste entonces en multiplicar 1 por la probabilidad de que el estado se encuentre en $|0\rangle$ (que definimos como $P(0)$) y sumarle el producto de -1 por la probabilidad de que el estado se encuentre en $|1\rangle$ (que definimos como $P(1)$). Por lo tanto, el valor esperado es $\langle\sigma_3\rangle = P(0) - P(1)$.

Dicho esto, es necesario aproximar $P(0)$ y $P(1)$ para poder obtener $\langle\sigma_3\rangle$. Para hacerlo, se puede ejecutar el circuito N veces y cada vez medir al qubit en la base computacional. Luego, $P(0)$ se aproxima como el número de veces que obtenemos $|0\rangle$ como resultado dividido por N y $P(1)$ como el número de veces que obtenemos $|1\rangle$ dividido por N . Con estos datos podemos calcular $\langle\sigma_3\rangle = P(0) - P(1)$ y conforme mayor número N de repeticiones realicemos, mejor se acercará nuestra aproximación al verdadero valor de $\langle\sigma_3\rangle$. El circuito para realizar esto es el circuito 3 de la figura 5.1 (que simplemente consiste en medir al qubit en la base computacional).

- $\langle\sigma_1\rangle$: Aproximar este valor requiere un paso adicional, pues la base de eigenvectores de σ_1 no es la base computacional, sino que es la base $\{|+\rangle, |-\rangle\}$, con eigenvalores 1 y -1 respectivamente.

Entonces, por la misma idea del inciso anterior, el valor esperado será $\langle\sigma_1\rangle = P(+)-P(-)$. Sin embargo, las mediciones en una computadora cuántica se suelen hacer en la base computacional y no en la base $\{|+\rangle, |-\rangle\}$. Por lo tanto, antes de medir el estado ρ del qubit, le aplicamos la compuerta de Hadamard H . Esta compuerta convierte a la base $\{|+\rangle, |-\rangle\}$ en la base computacional $\{|0\rangle, |1\rangle\}$ y tras aplicarla, la probabilidad de medir $|0\rangle$ será $P(+)$ y la de obtener $|1\rangle$ será $P(-)$.

El circuito que se usa para este valor esperado es el circuito 1 de la figura 5.1, que aplica una compuerta H antes de medir.

- $\langle\sigma_2\rangle$: Aproximar este valor es totalmente análogo al caso de $\langle\sigma_1\rangle$, con la diferencia que la base de eigenvectores de σ_2 es $\{|R\rangle, |L\rangle\}$ con eigenvalores 1 y -1 respectivamente.

Por lo tanto, en este caso el valor esperado es $P(R) - P(L)$ y para calcularlo haciendo mediciones en la base computacional es necesario antes aplicar las compuertas HS^\dagger al qubit, que transforman $\{|R\rangle, |L\rangle\}$ en $\{|0\rangle, |1\rangle\}$. El circuito para medir este valor esperado es el circuito 2 de la figura 5.1.

Por lo tanto, en total lo que tendremos que hacer para determinar el estado ρ que desconocemos es realizar los tres circuitos de la figura 5.1 para medir dicho estado en

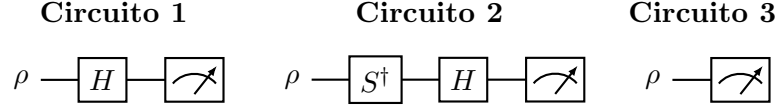


Figura 5.1: Circuitos para realizar QST para un qubit.

las tres distintas bases. En cada base se mide varias veces y así se aproximan los valores esperados de los operadores $\sigma_1, \sigma_2, \sigma_3$. Como se dijo antes, estos valores son las trazas $\text{tr}(\sigma_1\rho)$, $\text{tr}(\sigma_2\rho)$ y $\text{tr}(\sigma_3\rho)$ que podemos sustituir en la ecuación 2.16 para aproximar ρ .

Ahora que ya conocemos cómo determinar aproximadamente estados cuánticos desconocidos, introduciremos el concepto de Quantum Process Tomography, que se usa para determinar procesos cuánticos.

Definición 14 *Quantum Process Tomography (QPT)*: Dado un proceso cuántico que desconocemos, QPT es un algoritmo que permite determinarlo [38].

Nos interesará usar QPT para determinar procesos sobre un sistema de un qubit. Para estos sistemas, como se estudió en el marco teórico, la dinámica queda totalmente determinada por la matriz de Choi. Entonces, el objetivo de QPT en la forma que lo usaremos será encontrar una aproximación de la matriz de Choi que determina a la dinámica.

Notamos que para determinar completamente la transformación por la que pasa el qubit, es suficiente con determinar su efecto sobre solamente 4 estados iniciales. Esto debido a que con 4 estados se puede construir una base del espacio de las matrices de densidad de un qubit y si sabemos el efecto del circuito sobre esta base, podemos entonces conocer su efecto sobre cualquier matriz de densidad inicial y determinar completamente el proceso [38].

Por lo tanto, el procedimiento para realizar QPT sobre un proceso de un qubit consiste en inicializar el qubit en 4 estados distintos, que por su simplicidad escogeremos a $\{|0\rangle, |1\rangle, |+\rangle, |R\rangle\}$ (que son los estados que usa la función de QPT de qiskit [17]). Luego, para cada uno de estos estados aplicamos el proceso y hacemos QST sobre el resultado para determinar cómo se transformó el qubit en cada caso. De esta forma, conoceremos perfectamente el resultado de aplicar el proceso a cada uno de los estados iniciales y determinaremos así completamente el proceso.

5.3. Ejecución de los circuitos en computadoras cuánticas

En esta sección implementaremos en las computadoras cuánticas de IBM el circuito propuesto para simular un canal de Pauli. Empezamos recordando que un canal de

5. SIMULACIÓN EN UNA COMPUTADORA CUÁNTICA

Pauli sobre un qubit se define como

$$\varepsilon(\rho) = \alpha_0 \sigma_0 \rho \sigma_0 + \alpha_1 \sigma_1 \rho \sigma_1 + \alpha_2 \sigma_2 \rho \sigma_2 + \alpha_3 \sigma_3 \rho \sigma_3,$$

y se puede implementar sobre un qubit q_0 usando el circuito que se propone en la figura 3.4.

Lo que haremos será elegir una muestra de varios canales de Pauli y simular cada uno usando el circuito 3.4. Luego, al proceso le aplicaremos QPT sobre el primer qubit para encontrar la matriz de Choi correspondiente. Finalmente, compararemos la matriz de Choi encontrada por QPT con la matriz de Choi teórica que tiene el canal cuántico elegido.

La comparación de la matriz de Choi teórica con la encontrada por medio de QPT en la computadora cuántica se hace por medio del concepto de fidelidad de procesos, que es una medida que nos permite comparar qué tanto se parecen dos procesos cuánticos [4]. A continuación se define la fidelidad de estados y luego usamos esto para definir la fidelidad de procesos.

Definición 15 *Fidelidad de estados:* *Dados dos estados ρ y σ , la fidelidad entre ellos se define como [39]:*

$$F(\rho, \sigma) = \text{tr} \left(\sqrt{\rho^{1/2} \sigma \rho^{1/2}} \right)^2.$$

Esta cantidad es una forma de cuantificar la similitud entre los dos estados ρ y σ . La fidelidad entre ellos es un número entre 0 y 1 que alcanza el valor de 1 cuando los dos estados son iguales y toma valores más cercanos a 0 mientras más distintos sean los estados. Se puede probar que la fidelidad de estado cumple con las siguientes condiciones [39]:

1. $0 \leq F(\rho, \sigma) \leq 1$ con $F(\rho, \sigma) = 1$ si y sólo si $\rho = \sigma$.
2. $F(\rho, \sigma) = F(\sigma, \rho)$.

Para generalizar esta definición a procesos, haremos uso del isomorfismo de Choi-Jamiołkowski [40], [22]. Este isomorfismo nos permite convertir un canal cuántico ε en un estado ρ_ε (en un espacio con el doble de la dimensión del espacio en el que actúa ε) definido como:

$$\rho_\varepsilon = (I \otimes \varepsilon)[|\Omega\rangle\langle\Omega|],$$

donde I es el canal identidad y $|\Omega\rangle$ se define como el estado máximamente entrelazado: $\Omega = \frac{1}{\sqrt{\dim(\mathcal{H})}} \sum_i^{\dim(\mathcal{H})} |i\rangle|i\rangle$ (con \mathcal{H} el espacio de Hilbert sobre el que actúa el canal). Con este isomorfismo entre canales y estados, se puede definir la fidelidad de procesos como se muestra a continuación.

Definición 16 Fidelidad de Procesos: *Dados dos canales cuánticos ε y η , se define la fidelidad entre ellos como [4],[17]:*

$$F_{pro}(\varepsilon, \eta) = F(\rho_\varepsilon, \rho_\eta) = \text{tr} \left(\sqrt{\rho_\varepsilon^{1/2} \rho_\eta \rho_\varepsilon^{1/2}} \right)^2$$

A partir de las propiedades que cumple la fidelidad de estados, se puede ver que la de procesos cumple que [39],[4]:

1. $0 \leq F_{pro}(\varepsilon, \eta) \leq 1$ con $F_{pro}(\varepsilon, \eta) = 1$ si y sólo si $\varepsilon = \eta$.
2. $F_{pro}(\varepsilon, \eta) = F_{pro}(\eta, \varepsilon)$.

Similarmente al caso para estados, la fidelidad de procesos mide en una escala entre 0 y 1 qué tan parecidos son dos procesos, con 1 indicando que son exactamente iguales.

Definido esto, podemos ahora analizar en una computadora cuántica el circuito 3.4 para varios canales de Pauli distintos siguiendo los siguientes pasos:

1. Elegimos una muestra de varios puntos dentro del tetraedro de posibles canales de Pauli (en este caso se eligieron 249 puntos) y para cada uno construimos el circuito correspondiente de la figura 3.4.
2. Para cada canal, hacemos QPT sobre el qubit principal. El algoritmo para hacer QPT requiere de preparar el qubit en cada uno de 4 estados iniciales (en particular usaremos $\{|0\rangle, |1\rangle, |+\rangle, |R\rangle\}$). Luego, se aplica el circuito 3.4 para cada uno de estos estados iniciales y en cada caso se hace QST al resultado.

Esto significa que para cada canal hay que realizar un total de 12 circuitos en la computadora cuántica para conocer totalmente la dinámica del proceso. Estos 12 circuitos son creados en Qiskit y cada uno es realizado 20000 veces, para así encontrar una aproximación de la matriz de Choi del proceso.

3. Además, para cada uno de estos canales, calculamos también la matriz de Choi teórica a partir de la definición vista en el marco teórico $C = \sum_{ij} |i\rangle\langle j| \otimes \epsilon_p[|i\rangle\langle j|]$.
4. Finalmente, para cada canal se compara la matriz de Choi teórica con la obtenida al simular el proceso en la computadora cuántica y se calcula la fidelidad de procesos entre ellas.

Los resultados de este procedimiento se presentan en la figura 5.2, donde cada punto es uno de los canales de Pauli escogidos y su color representa la fidelidad del circuito respecto al canal que se espera reproducir. En este caso, se usó el procesador de IBM llamado ibmq-lima, que cuenta con 5 qubits [41].

Se puede apreciar que los canales cercanos a las aristas y en particular las esquinas del tetraedro tienen una fidelidad de alrededor de 0.5 y son los de más baja fidelidad.

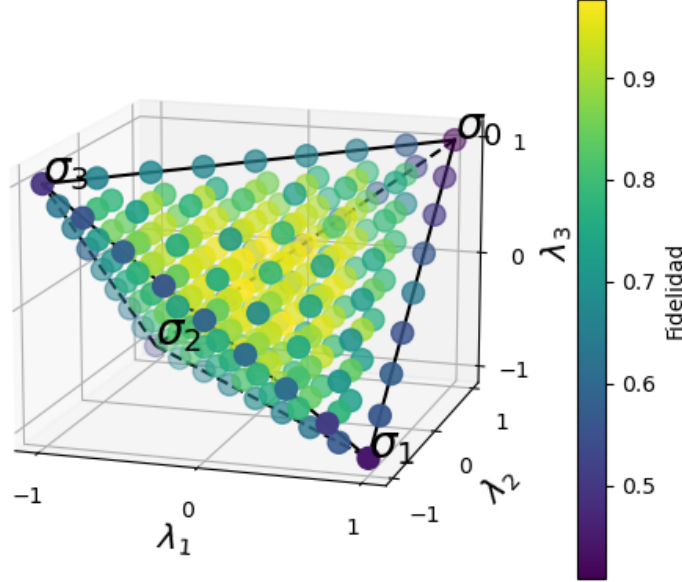


Figura 5.2: Fidelidad de los circuitos que implementan canales de Pauli para distintos canales en el tetraedro realizados en el procesador ibmq-lima.

Por otro lado, los canales más cercanos al origen del tetraedro tuvieron fidelidades mucho más altas, con muchos de ellos obteniendo fidelidades por arriba de 0.9.

Para intentar explicar este comportamiento, primero se realizaron los mismos pasos que antes pero en un simulador cuántico de Qiskit llamado `ibmq-qasm-simulator`. Cuando se pasa un circuito por este simulador, el circuito es realizado de forma clásica, lo que quiere decir que se toman las matrices correspondientes a cada una de las compuertas y se hace el producto matricial explícitamente para pasar del estado inicial de los qubits al final. Además, para simular la medición de un qubit en el estado $\alpha|0\rangle + \beta|1\rangle$ se escoge al azar uno de sus estados base $|0\rangle, |1\rangle$ con probabilidades $|\alpha|^2, |\beta|^2$ respectivamente, tal como sucedería cuánticamente.

Con este simulador se repitieron los pasos mencionados antes para simular los 249 canales de Pauli, obtener sus matriz de Choi del proceso y compararlas con las matrices teóricas por medio de la fidelidad de procesos. El resultado de esto se muestra en la figura 5.3, en donde se ve claramente que la fidelidad es casi perfecta. Las discrepancias en la fidelidad se deben a que el circuito se repite 20000 veces, pero una fidelidad de 1 requeriría repetirlo infinitas veces para que los resultados probabilísticos en las mediciones simuladas de los qubits sean iguales a lo teórico.

La figura 5.3 comprueba que el circuito implementa correctamente los canales de Pauli y entonces las fidelidades bajas en la figura 5.2 son un efecto de los errores a los que están sujetas las computadoras cuánticas.

Más aún, se puede dar una posible explicación sencilla de los errores que se tienen

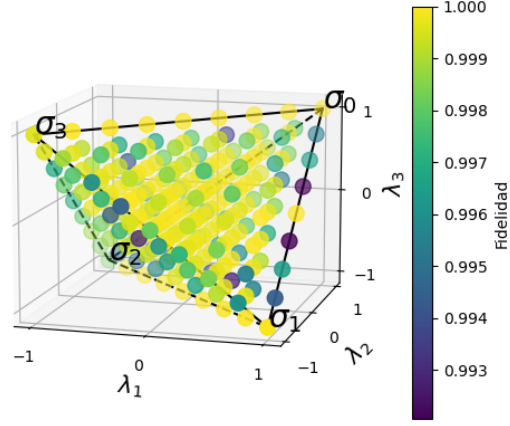


Figura 5.3: Fidelidad de los circuitos que implementan canales de Pauli para distintos canales en el tetraedro realizados en ibmq-qasm-simulator (que simula ser una computadora cuántica sin errores).

en la implementación en el procesador cuántico. Primero recordamos que el circuito propuesto para implementar un canal de Pauli (figura 3.2) requiere la creación del estado $|\beta\rangle = \beta_0|00\rangle + \beta_1|01\rangle + \beta_2|10\rangle + \beta_3|11\rangle$ sobre los qubits de ancilla y luego utilizar compuertas controladas por ellos para realizar operaciones sobre el qubit principal. Además, como después de realizar las compuertas controladas los qubits de ancilla no se vuelven a usar, se puede pensar que estos qubits son en realidad medidos antes de las compuertas controladas y luego se utiliza el resultado (que será $|00\rangle$, $|01\rangle$, $|10\rangle$, ó $|11\rangle$) para realizar las compuertas controladas. Esta equivalencia entre usar compuertas controladas o medir primero y usar compuertas controladas después se conoce como principio de la medida diferida [4].

Entonces, una posible fuente de los errores se puede deber a errores en la medición del estado de $|\beta\rangle$ de los dos qubits de ancilla. Los errores en la medición se pueden entender como la perturbación del estado de un sistema de qubits debido a ruido causado por la interacción con el ambiente durante la medición [42]. Los errores en la medición han sido tratados teóricamente [43, 44] y han sido estudiados en los procesadores de IBM, donde se menciona a la medición como una de las operaciones más propensas a errores [45].

En el libro de texto de Qiskit [42] se expone un modelo sencillo de este error. El modelo consiste en un proceso que con cierta probabilidad q voltear alguno de los qubits medidos. Por ejemplo, si el resultado de una medición sobre dos qubits fue $|11\rangle$, habrá una probabilidad $q(1-q)$ de que en realidad el estado resultante se cambie a $|01\rangle$ ó $|10\rangle$ y una probabilidad q^2 de que ambos qubits se vean afectados y el resultado sea $|00\rangle$.

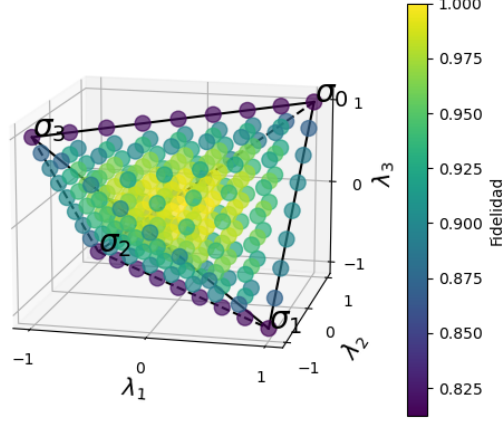


Figura 5.4: Fidelidad de los circuitos que implementan canales de Pauli para distintos canales en el tetraedro realizados en ibmq-qasm-simulator con un error de medición de $q = 0.07$.

Se puede agregar este modelo a la implementación en el simulador cuántico para ver cómo cambian las fidelidades al considerar este error. Entonces, se añadió este error cuando se miden los qubits de ancilla antes de aplicar las compuertas controladas del circuito de la figura 3.2 (cuando se encuentran en el estado $|\beta\rangle$). Por ejemplo, tras agregar este modelo con una probabilidad de error de $q = 0.07$ a la simulación (que se encuentra dentro del rango de probabilidades para errores de medición en las computadoras de IBM según [45]), se obtienen las fidelidades mostradas en la figura 5.4.

La figura 5.4 es cualitativamente similar a la de las fidelidades obtenidas utilizando el procesador ibmq-lima de la figura 5.2. Por lo tanto, podemos conjeturar que las fidelidades bajas en la implementación se deben, aunque sea en parte, a errores de medición. Por supuesto que también hay otros tipos de errores en la implementación de circuitos cuánticos, como pueden ser errores en la realización de compuertas o decoherencia debido a la interacción con el ambiente [4] y que disminuyen aún más la fidelidad.

Finalmente, podemos explicar por qué el error de medida sobre los qubits de ancilla da lugar a las fidelidades de la figura 5.4. Para ello, recordemos que antes de aplicar las compuertas controladas en el circuito, los qubits de ancilla se encuentran en el estado

$$|\beta\rangle = \beta_0|00\rangle + \beta_1|01\rangle + \beta_2|10\rangle + \beta_3|11\rangle,$$

donde $|\beta_j|^2 = \alpha_j$, con α_j los coeficientes que definen al canal de Pauli. Luego, recordamos que el canal en el centro del tetraedro (el canal completamente depolarizador) tiene como valores de α a

$$\alpha_1 = 0.25, \alpha_2 = 0.25, \alpha_3 = 0.25, \alpha_4 = 0.25.$$

Entonces, omitiendo fases (que no son importantes para el error de medición), el estado en los qubits de ancilla es

$$|\beta\rangle_{\text{depolarizador}} = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle.$$

Sin embargo, este estado cuántico no se ve realmente afectado por errores de medición. Esto se debe a que los cuatro estados base tienen la misma amplitud, por lo que los errores de cambiar un estado por otro tras la medición se cancelan. Por ejemplo, hay una probabilidad de 0.25 de medir $|11\rangle$, pero luego una probabilidad $q(1-q)$ de que el primer qubit se voltee y en realidad se reporte $|01\rangle$. Sin embargo, hay también una probabilidad de 0.25 de medir $|01\rangle$ y luego una probabilidad $q(1-q)$ de que se voltee el primer qubit y en realidad se reporte $|11\rangle$. Por esta simetría, el error de reportar $|01\rangle$ cuando en realidad se debería de medir $|11\rangle$ es igual al de reportar $|11\rangle$ cuando en realidad se debería de medir $|01\rangle$. Por lo tanto, los errores se cancelan y al final seguimos teniendo un 0.25 de probabilidad de medir cada uno de los estados base.

Más aún, los estados $|\beta\rangle$ correspondientes a canales de Pauli cercanos al origen del tetraedro se parecen a $|\beta_{\text{depolarizador}}\rangle$, pues también tienen las cuatro amplitudes β_j casi iguales entre sí. Entonces, para todos estos estados, los errores de medición casi se cancelarán de forma similar a como pasa en el completamente depolarizador, lo que hará que no disminuya significativamente la fidelidad. Es por ello que los canales cercanos al origen tienen fidelidades altas.

Por otro lado, si consideramos el canal identidad, que se encuentra en una de las esquinas del tetraedro, el estado que hay que crear sobre los qubits de ancilla es

$$|\beta\rangle_{\text{identidad}} = |00\rangle.$$

Este estado sí se verá fuertemente afectado por errores de medición, pues al medir hay una probabilidad $q(1-q)$ de que se voltee uno de los qubits y el estado se convierta en $|01\rangle$ ó $|10\rangle$ y una probabilidad q^2 de que ambos qubits se volteen. Por lo tanto, ahora los errores no se cancelan y las mediciones darán resultados muy distintos a los que se deberían de obtener. Por ello el canal identidad no tendrá una fidelidad alta y lo mismo se tiene para el resto de las esquinas del tetraedro. Esto explica las fidelidades bajas cerca de las esquinas del tetraedro.

5.3.1. Análisis de los circuitos para el mapa depolarizador y el parabólico σ_3

En esta sección analizaremos dos de los mapas dinámicos que vimos que se pueden implementar con una sola rotación parametrizada, en particular analizaremos el depolarizador y el parabólico σ_3 . Primero que nada, recordamos que estos mapas se definen como:

$$\begin{aligned}\epsilon_{\text{depo } p}(\rho) &= (1 - 3p/4)\rho + (p/4)\sigma_1\rho\sigma_1 + (p/4)\sigma_2\rho\sigma_2 + (p/4)\sigma_3\rho\sigma_3, \\ \epsilon_{\text{para } p}(\rho) &= \frac{1}{4}(1-p)^2\rho + \frac{1}{4}(1-p^2)\sigma_1\rho\sigma_1 + \frac{1}{4}(1-p^2)\sigma_2\rho\sigma_2 + \frac{1}{4}(1+p)^2\sigma_3\rho\sigma_3,\end{aligned}$$

5. SIMULACIÓN EN UNA COMPUTADORA CUÁNTICA

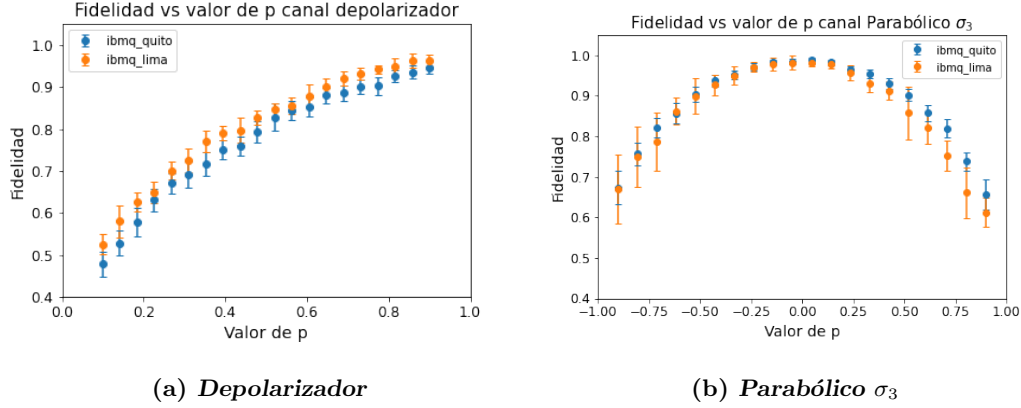


Figura 5.5: Fidelidad de los circuitos para los mapas dinámicos depolarizador y parabólico σ_3 .

donde el parámetro p se encuentra en $[0, 1]$ para el depolarizador y en $[-1, 1]$ para el parabólico σ_3 .

En la sección 4.2.3 concluimos que ambos mapas cumplen con las condiciones para poder implementarse en un circuito usando una sola rotación parametrizada y el circuito para realizarlos es el de la figura 4.5. Para el canal depolarizing encontramos que las matrices A , B y la expresión para s están dadas por 4.9 y para el canal parabólico están dadas por 4.12.

Para analizar estos mapas en una computadora cuántica, lo que haremos será construir el circuito 4.5 para varios valores distintos de p (desde $p = 0.1$ hasta $p = 0.95$ para el depolarizador y desde $p = -0.9$ hasta $p = 0.9$ para el parabólico). Para cada valor de p se realiza el circuito 20000 veces en dos computadoras (ibmq-lima e ibmq-quito) y se hace QPT . Luego, se calcula la fidelidad del proceso para cada valor de p en cada computadora respecto al proceso que teóricamente debería realizar el circuito. En la figura 5.5 se presentan los resultados de estas fidelidades para los dos canales. Para cada valor de p , se hicieron los pasos mencionados antes un total de 15 veces para tener en cuenta la incertidumbre de los resultados y con los datos se obtuvo la desviación estándar de la fidelidad, que se simboliza por las barras de error en las gráficas.

Para el mapa depolarizador se puede observar que las dos computadoras dan resultados similares. Para valores de p cercanos a 0 (que corresponden a un canal cercano a la identidad), la fidelidad es baja, con un valor de aproximadamente 0.5. Sin embargo, conforme aumenta el valor de p y los canales se acercan al totalmente depolarizador, la fidelidad aumenta a un hasta valores cercanos a 1.

Por otro lado, para el canal parabólico σ_3 , se puede apreciar también que las dos computadoras dan resultados muy similares y muestran una fidelidad que sigue una curva casi parabólica. Para valores de p cercanos a -1 (que es el canal identidad), la fidelidad es de aproximadamente 0.65. La fidelidad aumenta conforme p se acerca a 0

(correspondiente con el canal totalmente depolarizador) y finalmente vuelve a disminuir a aproximadamente 0.65 conforme p se acerca a 1 (que es el canal σ_z). Estos resultados concuerdan con lo visto en la figura 5.2, en la que se observa que la fidelidad es baja cerca de los vértices del tetraedro y aumenta considerablemente cerca del origen.

Conclusiones

En este trabajo se han estudiado los sistemas cuánticos abiertos y en particular los canales de Pauli para sistemas de un qubit. Se empezó por revisar que estos canales tienen la forma general $\varepsilon(\rho) = \sum \alpha_j \sigma_j \rho \sigma_j$ y las condiciones que deben de cumplir para ser físicamente válidos.

Se introdujeron teóricamente los canales de Pauli y revisaron algunos resultados notables, resaltando que estos canales se pueden representar de una forma muy sencilla, pues están caracterizados por tres variables reales que se obtienen a partir de $\{\alpha_j\}$. Adicionalmente, se mencionó la interpretación física de estas variables y se estudiaron algunos resultados conocidos, destacando que estas variables deben de encontrarse dentro de un tetraedro para que representen un canal válido.

En particular, este trabajo es una propuesta más del tipo de simulaciones de sistemas cuánticos que se pueden hacer con una computadora cuántica, pues se propuso un algoritmo original para implementar canales de Pauli en un circuito. Se mostró que cualquier canal de un qubit se puede realizar en un circuito cuántico con la ayuda de dos qubits de ancilla y un total de 3 rotaciones aplicadas sobre ellos. Una posible línea de investigación sería generalizar estos resultados a canales de Pauli de varios qubits o a canales cuánticos más generales para encontrar formas de realizarlos en computadoras cuánticas.

Luego se examinaron los mapas dinámicos de Pauli, y en particular en el teorema 9 se encontraron las condiciones para poder implementarlos en un circuito con una sola rotación parametrizada, que funcione así como una perilla que controla al mapa. Aunque las condiciones encontradas parecían muy restrictivas, se concluyó que en realidad no lo son tanto, pues se demostró que varios de los mapas dinámicos paradigmáticos como los *Bit Flip*, *Bit-Phase Flip*, *Phase Flip* y *Depolarizing* las cumplen.

Es importante mencionar que el teorema 9 habla en general sobre qué condiciones debe de cumplir una curva de estados de un qubit para que se pueda crear con un circuito de una sola compuerta parametrizada. En este trabajo sólo se estudian las consecuencias de esto sobre los mapas dinámicos de Pauli al encontrar cuáles se pueden implementar con esta condición sobre el circuito. Sin embargo, existe la posibilidad de usar estos resultados para circuitos con otros propósitos en los que también se quiera

6. CONCLUSIONES

minimizar la cantidad de compuertas parametrizadas.

Finalmente, en el último capítulo del trabajo se aprovechan las computadoras cuánticas de IBM para simular dos de estos mapas (el depolarizador y el parabólico σ_3) y varios canales de Pauli en sistemas reales. Se calculó la fidelidad con la que son realizados estos procesos en las computadoras cuánticas y se observó que la fidelidad es alta para canales cercanos al centro del tetraedro y baja para canales cerca de las esquinas. Esto se explicó como una consecuencia de las limitaciones y errores actuales de las computadoras cuánticas.

Bibliografía

- [1] Emily Grumbling and Mark Horowitz. *Quantum computing: progress and prospects*. The National Academies Press, 2019. [1](#), [58](#)
- [2] Jonathan P. Dowling and Gerard J. Milburn. Quantum technology: The second quantum revolution. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 361(1809):1655–1674, 2003. [1](#), [58](#)
- [3] Dan C. Marinescu and Marinescu. *Classical and Quantum Information*. Elsevier, 2012. [1](#), [20](#)
- [4] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2021. [1](#), [3](#), [4](#), [8](#), [12](#), [15](#), [25](#), [58](#), [59](#), [62](#), [63](#), [65](#), [66](#)
- [5] Daniel Lidar. *Lecture Notes on the Theory of Open Quantum Systems*. Departments of Electrical Engineering, Chemistry, and Physics and Astronomy Center for Quantum Information Science and Technology University of Southern California, Los Angeles, California 90089, 2015. [1](#), [17](#), [19](#), [26](#)
- [6] Ingemar Bengtsson and Karol Zyczkowski. *Geometry of Quantum States: An Introduction to Quantum Entanglement*. Cambridge University Press, 2006. [1](#)
- [7] Daniel Greenbaum and Zachary Dutton. Modeling coherent errors in quantum error correction. *Quantum Science and Technology*, 3(1):015007, Dec 2017. doi: 10.1088/2058-9565/aa9a06. [2](#), [27](#)
- [8] Flammia Steven and Wallman Joel. Efficient estimation of pauli channels. *ACM Transactions on Quantum Computing*, 1(3):1–32, 2020. [2](#)
- [9] Richard Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 1982. doi: <https://doi.org/10.1007/BF02650179>. [2](#)
- [10] S. Lloyd. Universal quantum simulators. *Science*, 1996. [2](#)
- [11] Childs A. M. et al. Berry, D.W. Simulating hamiltonian dynamics with a truncated taylor series. *Phys. Rev. Lett.*, 2015. [2](#)

- [12] Guang Hao Low and Isaac L. Chuang. Optimal hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.*, 118:010501, Jan 2017. doi: 10.1103/PhysRevLett.118.010501. URL <https://link.aps.org/doi/10.1103/PhysRevLett.118.010501>. 2
- [13] Richard Cleve and C. Wang. Efficient quantum algorithms for simulating lindblad evolution. In *International Colloquium on Automata, Languages and Programming*, 2016. 2
- [14] Amit Anand, Sanchit Srivastava, Sayan Gangopadhyay, and Shohini Ghose. Simulating quantum chaos on a quantum computer. 07 2021. 2, 59
- [15] F. Haake and M. Kuś. Kicked top. *Scholarpedia*, 5(11):10242, 2010. doi: 10.4249/scholarpedia.10242. revision #137061. 2, 59
- [16] Rebecca Roberts Chi-Kwong Li and Xiaoyan Yin. Decomposition of unitary matrices and quantum gates. *International Journal of Quantum Information*, 2012. doi: 10.1142/S0219749913500159. 8
- [17] Sajid Anis et al. Qiskit: An open-source framework for quantum computing, 2021. 12, 58, 59, 61, 63
- [18] Gregg Jaeger. *Quantum Information: An Overview*. Springer, Boston, MA 02215 USA, 2007. 15
- [19] Daniel Manzano. A short introduction to the lindblad master equation. *AIP Advances*, 10(2), 2020. 15, 20, 21, 22
- [20] Teiko Heinosaari and Mario Ziman. *The Mathematical Language of Quantum Theory: From Uncertainty to Entanglement*. Cambridge University Press, 2012. 18
- [21] Ian Glendinning. Rotations on the bloch sphere, 2010. 18, 39
- [22] Man-Duen Choi. Completely positive linear maps on complex matrices. *Linear Algebra and its Applications*, 10(3), 1975. 26, 62
- [23] Katarzyna Siudzińska. Geometry of pauli maps and pauli channels. *Physical Review A*, 100, 2019. doi: 10.1103/PhysRevA.100.062331. 27
- [24] David Davalos et al. “divisibility of qubit channels and dynamical maps. *Quantum*, 3:144, 2019. 31
- [25] Oscar Perdomo et al. Entanglement types for two-qubit states with real amplitudes. *Quantum Information Processing*, 20(3), 2021. 34
- [26] Google. Google quantum ai, 2023. URL <https://quantumai.google/>. 58

- [27] IBMQ. Ibm quantum system one, 2023. URL <https://research.ibm.com/interactive/system-one/>. 58
- [28] Microsoft Azure. Azure quantum, 2023. URL <https://azure.microsoft.com/en-us/solutions/quantum-computing/#overview>. 58
- [29] Rigetti. Rigetti think quantum, 2023. URL <https://www.rigetti.com/>. 58
- [30] Ion Q. Ion q, 2023. URL <https://ionq.com/>. 58
- [31] D Wave. D wave, 2023. URL <https://www.dwavesys.com/>. 58
- [32] Xanadu. Xanadu quantum hardware, 2023. URL <https://www.xanadu.ai/>. 58
- [33] D. Bacon. Decoherence, control, and symmetry in quantum computers, 2003. 58
- [34] John Clarke and Frank Wilhelm. Superconducting quantum bits. *Nature*, 453: 1031–42, 07 2008. doi: 10.1038/nature07128. 58
- [35] J. I. Cirac and P. Zoller. Quantum computations with cold trapped ions. *Phys. Rev. Lett.*, 74:4091–4094, May 1995. doi: 10.1103/PhysRevLett.74.4091. URL <https://link.aps.org/doi/10.1103/PhysRevLett.74.4091>. 58
- [36] Niall F. Robertson, Albert Akhriev, Jiri Vala, and Sergiy Zhuk. Approximate Quantum Compiling for Quantum Simulation: A Tensor Network based approach. 1 2023. 58
- [37] Roman Schmied. Quantum state tomography of a single qubit: comparison of methods. *Journal of Modern Optics*, 63(18):1744–1758, 2016. doi: 10.1080/09500340.2016.1142018. 59
- [38] Isaac L. Chuang and M. A. Nielsen. Prescription for experimental determination of the dynamics of a quantum black box. *Journal of Modern Optics*, 44(11-12): 2455–2467, 1997. doi: 10.1080/09500349708231894. 61
- [39] Richard Jozsa. Fidelity for mixed quantum states. *Journal of Modern Optics*, 41 (12):2315–2323, 1994. doi: 10.1080/09500349414552171. 62, 63
- [40] Andrzej Jamiołkowski. Linear transformations which preserve trace and positive semidefiniteness of operators. *Reports on Mathematical Physics*, 3:275–278, 1972. doi: 10.1016/0034-4877(72)90011-0. 62
- [41] IBMQ. Computer resources, 2023. URL <https://quantum-computing.ibm.com/services/resources?tab=systems>. 63
- [42] Qiskit Textbook. Measurement error mitigation, 2023. URL <https://learn.qiskit.org/course/quantum-hardware/measurement-error-mitigation>. 65

- [43] Simon J Devitt, William J Munro, and Kae Nemoto. Quantum error correction for beginners. *Reports on Progress in Physics*, 76(7):076001, jun 2013. doi: 10.1088/0034-4885/76/7/076001. URL <https://dx.doi.org/10.1088/0034-4885/76/7/076001>. 65
- [44] Avimita Chatterjee, Koustubh Phalak, and Swaroop Ghosh. Quantum error correction for dummies, 2023. 65
- [45] Swamit S. Tannu and Moinuddin K. Qureshi. Mitigating measurement errors in quantum computers by exploiting state-dependent bias. In *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, MICRO '52, page 279–290, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450369381. doi: 10.1145/3352460.3358265. URL <https://doi.org/10.1145/3352460.3358265>. 65, 66