***Implementación de Algoritmo de Deutsch y Deutsch-Jozsa***

**Santiago Diaz Rojas**

**Escuela Colombiana de Ingeniería**

[**santiago.diaz-r@mail.escuelaing.edu.co**](mailto:santiago.diaz-r@mail.escuelaing.edu.co)

**21 de noviembre de 2024**

***Este reporte se entrega para cumplir con los requisitos parciales del curso CNYT:***

***Computación Cuántica - 2024-1***

[Tabla de contenidos](#_Toc183121307)

[1 Introducción 3](#_Toc183121308)

[2 Algoritmo de Deutsch 4](#_Toc183121309)

[2.1 Problema 4](#_Toc183121310)

[2.2 Implementando las funciones en el computador cuántico 5](#_Toc183121311)

[2.3 Implementando el algoritmo de Deutsch en un computador cuántico 12](#_Toc183121312)

[3 Algoritmo de Deutsch-Jozsa 13](#_Toc183121313)

[3.1 Problema 13](#_Toc183121314)

[3.2 Implementando las funciones en el computador cuántico 14](#_Toc183121315)

[3.3 Implementando el algoritmo de Deutsch-Josza en un computador cuántico 14](#_Toc183121316)

[4 Conclusiones 14](#_Toc183121317)

[5 Bibliografía 14](#_Toc183121318)

# Introducción

La computación cuántica es una disciplina emergente en el campo de la informática que busca superar las limitaciones de los computadores tradicionales al aprovechar los principios únicos de la mecánica cuántica, como la superposición, el entrelazamiento y la interferencia. Estas propiedades permiten resolver problemas complejos con mayor eficiencia, lo que tiene el potencial de transformar áreas como la criptografía, la inteligencia artificial y la optimización. IBM se destaca como uno de los líderes en este ámbito, ofreciendo acceso a su plataforma IBM Quantum Experience, que facilita a investigadores y entusiastas trabajar con circuitos cuánticos reales o simulados. Aunque la computación cuántica está en sus primeras etapas de desarrollo, su crecimiento constante promete cambiar radicalmente cómo abordamos problemas computacionales. Este documento utilizará la librería Qiskit de Python como base para explorar y evaluar diversos circuitos cuánticos.

Este informe tiene como objetivo presentar un análisis accesible de las capacidades y características del computador cuántico de IBM, explicando cómo funcionan los qubits y las puertas cuánticas para realizar cálculos que superan las posibilidades de los sistemas tradicionales. También se hablará sobre el impacto actual de la computación cuántica en diferentes industrias, los desafíos que enfrenta su implementación a gran escala y se incluirá un ejemplo práctico para demostrar cómo estas máquinas pueden resolver problemas reales.

A lo largo del documento, se explicarán dos circuitos fundamentales en la computación cuántica: el algoritmo de Deutsch y el algoritmo de Deutsch-Jozsa. Se presentará una interpretación sencilla de los problemas que resuelven estos algoritmos, además de sus implementaciones prácticas. En primer lugar, se mostrará cómo evaluar cada entrada de las funciones utilizadas en estos algoritmos y, posteriormente, se presentará una implementación más directa de cada uno, acompañada de un análisis de sus resultados. El propósito es facilitar al lector la comprensión de estas herramientas cuánticas y su utilidad.

# Algoritmo de Deutsch

El algoritmo de Deutsch fue el primero en aprovechar la capacidad del paralelismo cuántico para resolver problemas. Originalmente, utiliza dos qubits, aunque existe una versión optimizada que funciona con solo uno. Si bien su beneficio práctico es modesto, este algoritmo abrió el camino para el desarrollo de otros algoritmos cuánticos que han demostrado ser más eficientes que los métodos clásicos.

El problema en el que se basa el algoritmo fue planteado por primera vez en 1985, antes de que existiera el modelo actual de circuitos cuánticos. Los conceptos de puertas lógicas universales y circuitos cuánticos surgieron unos años más tarde, en 1989. Posteriormente, el algoritmo de Deutsch fue ampliado y generalizado, dando lugar al conocido algoritmo de Deutsch-Jozsa.

## Problema

Supongamos que tenemos una función booleana de 1 bit sin conocer los detalles de su implementación. Queremos determinar si esta función es equilibrada o constante. Una función booleana de 1 bit es equilibrada si de lo contrario, la función es constante, en cuyo caso . Existen cuatro funciones booleanas posibles :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *x* |  |  |  |  |
| 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |

Así las formas de las funciones son las siguientes:

Sin embargo, como determinar que la función es balanceada o constate, esto se evalúa generando la matriz correspondiente de la función la cual describe si la combinación la combinación de qubits de entrada y de salida es la misma, y evaluándola en el algoritmo de Deutcsh.

**Generador de matriz**



**Algoritmo de Deutsch**

Imagen que contiene Diagrama

Descripción generada automáticamente

## Implementando las funciones en el computador cuántico

Imagen que contiene Diagrama

Descripción generada automáticamente

Se puede observar la implementación de la función de manera gráfica, a continuación, se mostrará cada uno de los circuitos implementados y el resultado para cada una de los posibles qubits de salida.

Qubit de entrada 00:

Gráfico

Descripción generada automáticamente

Qubit de entrada 01:

Diagrama

Descripción generada automáticamente con confianza baja

Qubit de entrada 10:

Diagrama

Descripción generada automáticamente con confianza baja

Qubit de entrada 11:

Diagrama

Descripción generada automáticamente

Como se puede apreciar cada uno de los resultados de los circuitos ejecutados anteriormente siguen el invariante de que sus qubits de salida son iguales a los deentrada esto se debe a que la función es constante, y describe por su matriz:

De donde se puede inferir que si la matriz es la identidad, la función es constante.

Imagen que contiene reloj, competencia de atletismo

Descripción generada automáticamente

En este caso la función tiene la particularidad de que sus qubits va a ser igual al de salida, definiendo el resultado de sus circuitos de la siguiente manera:

Qubit de entrada 00:

Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

Qubit de entrada 01:

Gráfico

Descripción generada automáticamente con confianza baja

Qubit de entrada 10:

Diagrama

Descripción generada automáticamente con confianza baja

Qubit de entrada 11:

Diagrama

Descripción generada automáticamente con confianza baja

En este caso la matriz de la función presenta una alteración y es que cuando el valor de entrada no coincide con el de salida, se genera la matriz negación, como se puede ver a continuación:

Imagen que contiene Diagrama

Descripción generada automáticamente

En este caso la función tiene la particularidad de que sus qubits de entrada son la negación en la salida, definiendo el resultado de sus circuitos de la siguiente manera:

Qubit de entrada 00:

Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

Qubit de entrada 01:

Imagen de la pantalla de un celular con letras

Descripción generada automáticamente con confianza baja

Qubit de entrada 10:

Imagen que contiene Diagrama

Descripción generada automáticamente

Qubit de entrada 11:

Imagen que contiene Diagrama

Descripción generada automáticamente

En este caso la matriz de la función presenta una alteración y es que cuando el valor de entrada no coincide con el de salida, se genera la matriz negación, como se puede ver a continuación:

Gráfico de burbujas

Descripción generada automáticamente con confianza media

En este caso la función tiene la particularidad de que sus qubits siempre van al qubit 1 en la salida:

Qubit de entrada 00:

Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

Qubit de entrada 01:

Imagen de la pantalla de un celular con letras

Descripción generada automáticamente con confianza baja

Qubit de entrada 10:

Gráfico

Descripción generada automáticamente con confianza baja

Qubit de entrada 11:

Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

En este caso la matriz de la función presenta una alteración y es que cuando el valor de entrada no coincide con el de salida, se genera la matriz negación, como se puede ver a continuación:

## Implementando el algoritmo de Deutsch en un computador cuántico

El algoritmo de deutsch, nos indica cuando una función es constante o balanceada, a continuación, se va a ver la implementación de cada una de las anteriores funciones, y sus resultados:

Gráfico

Descripción generada automáticamente

Gráfico

Descripción generada automáticamente

Gráfico

Descripción generada automáticamente

Gráfico

Descripción generada automáticamente

Como podemos ver si el algoritmo nos arroja 1 en el qubit de salida esto implica que la función es balanceada, de lo contrario no arroja 0, indicando que es constante.

# Algoritmo de Deutsch-Jozsa

El algoritmo de Deutsch-Jozsa es un algoritmo cuántico determinista que generaliza el algoritmo de Deutsch y se destaca por ser el primer ejemplo de un algoritmo cuántico que supera exponencialmente a su equivalente clásico determinista en términos de velocidad.

Este algoritmo es un ejemplo clave de cómo el paralelismo cuántico puede aprovecharse para resolver problemas de decisión con una eficiencia incomparable, y ha servido como base para la creación y análisis de otros algoritmos cuánticos avanzados. Su importancia radica no solo en su velocidad exponencialmente superior, sino también en su simplicidad conceptual, lo que lo convierte en un punto de partida ideal para comprender el potencial de los computadores cuánticos en problemas prácticos y teóricos.

## Problema

El algoritmo de Deutsch-Jozsa resuelve el problema de determinar si una función booleana f(x), que toma una entrada de n-bits y devuelve un valor de 1-bit, es constante (tiene el mismo valor para todas las entradas) o balanceada (devuelve 0 para la mitad de las entradas y 1 para la otra mitad). En otras palabras, lo que permite es que el qubit de entrada sea de mas de un bit, dando, así como resultado la posibilidad de evaluar el tipo de función del que estamos tratando.

Diagrama, Esquemático

Descripción generada automáticamente

## Implementando las funciones en el computador cuántico

Se realiza la implementación de 4 funciones una constante y 3 balanceadas, esto con el fin de ver como es el comportamiento de la función con respecto a los valores de entrada, en este caso nos piden que el n = 4, por lo tanto requerimos 4 qubits de entrada y 1 qubit de control, sin embargo solo necesitamos medir los 4 primeros que son los que nos indican la salida del circuito, por complejidad espacian no se muestra el procedimiento completo, por esto se recomienda visualizar el .ipynb adjunto, a continuación se presentan algunos casos:

Imagen de la pantalla de una computadora

Descripción generada automáticamente con confianza baja

Como se puede apreciar se confirma la descripción del circuito y su representación gráfica donde la función es la matriz generada con ayuda del algoritmo de deutsch inicial:



## Implementando el algoritmo de Deutsch-Josza en un computador cuántico

Gráfico

Descripción generada automáticamente

Imagen que contiene texto, marcador

Descripción generada automáticamente

Como podemos ver esta función representa una matriz constante por lo tante el qubit de salida se dirige a 0.

Diagrama

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente con confianza media

Diagrama

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Gráfico, Diagrama

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Como podemos ver estas 3 ultimas funciones no representan una matriz constante por lo tante sus qubits de salida se dirige nunca se dirigen a 0.

# Conclusiones

Se presentó un enfoque para visualizar matrices binarias mediante grafos dirigidos, donde las conexiones reflejan la relación entre los estados binarios de entrada y los qubits de salida. Este trabajo facilitó la comprensión de cómo se representan matemáticamente las operaciones en sistemas cuánticos y su interpretación visual. La estructura y las conexiones permiten analizar patrones y propiedades de funciones lógicas complejas.

Durante el análisis y creación de grafos, se profundizó en cómo las matrices definen transformaciones entre qubits de entrada y salida en un sistema cuántico. Además, se reforzó el entendimiento de la dualidad entre las representaciones matriciales y gráficas, clave en aplicaciones como algoritmos cuánticos balanceados o constantes (por ejemplo, Deutsch-Jozsa).

Lo aprendido tiene aplicaciones en áreas como la optimización de algoritmos cuánticos, diseño de circuitos lógicos y simulaciones de sistemas complejos. La capacidad de transformar representaciones matemáticas abstractas en modelos visuales ayuda a identificar patrones y posibles errores.

# Bibliografía

Portugal, R. (2022). Basic Quantum Algorithms. En *arXiv [quant-ph]*. <http://arxiv.org/abs/2201.10574>

*Qiskit*. (s/f). Ibm.com. Recuperado el 22 de noviembre de 2024, de <https://www.ibm.com/quantum/qiskit>