

Algoritmo cuántico de Deutsch y Jozsa en GAMA

M. Paredes López

Departamento de Matemáticas, Cinvestav,
Av. Instituto Politécnico Nacional 2508, CDMX

A. Meneses Viveros

Departamento de Computación, Cinvestav,
Av. Instituto Politécnico Nacional 2508, CDMX

G. Morales-Luna

Departamento de Computación, Cinvestav,
Av. Instituto Politécnico Nacional 2508, CDMX

Received 13 September 2017; accepted 1 March 2018

Una propiedad importante de la computación cuántica es su paralelismo implícito, que permite procesar un número exponencial de transformaciones básicas mediante un número lineal de qubits en un sistema cuántico. El algoritmo de Deutsch y Jozsa ilustra la reducción, proporcionada por el cómputo cuántico, en la complejidad del procesamiento. Presentamos aquí la implementación y la ejecución del algoritmo cuántico de Deutsch y Jozsa en GAMA, herramienta que hemos desarrollado para la simulación y monitorización de algoritmos cuánticos. La simulación permite explorar los detalles de sus componentes mediante el módulo de depuración de GAMA que permite mostrar las diferentes configuraciones que cada componente puede asumir.

Descriptores: Computación cuántica; paralelismo; qubits; complejidad computacional; simulación cuántica

An important feature of quantum computing is its inherent parallelism, allowing to process an exponential number of basic transforms with just a linear number of qubits. The Deutsch-Jozsa algorithm exemplifies the computational complexity reduction. This work reports the implementation and execution of the Deutsch-Josza quantum algorithm in GAMA, a programming language for quantum computing simulation developed by ourselves. Through this simulation, it is possible to explore all the components involved by tracing all the different configurations that each component may take.

Keywords: Quantum computing; parallelism; qubits; computational complexity; quantum simulation.

PACS: 03.67.Ac; 03.67.Lx

1. Introducción

La computación cuántica aprovecha fenómenos de la mecánica cuántica para realizar computaciones, tales como la superposición, el entrelazamiento o el paralelismo implícito. El cómputo cuántico permite desarrollar algoritmos para acelerar procesos, o transmisiones de información o, incluso, encontrar nuevas propiedades de física cuántica [1].

Un *algoritmo* es una sucesión finita de pasos para resolver una tarea, y uno *cuántico* consiste en aplicar una serie finita de transformaciones (o *compuertas cuánticas*) a un estado inicial para producir un estado final. En la actualidad se han desarrollado diversos algoritmos cuánticos en diferentes áreas tales como criptografía, búsqueda, optimización, resolución de sistemas lineales y simulación de sistemas cuánticos, entre otros [2]. El algoritmo de Deutsch y Jozsa es uno de los primeros algoritmos cuánticos que surgieron [3]. Y aunque se ha implementado y simulado usando diferentes técnicas [4–9], y ha demostrado obtener resultados en un solo paso de tipo cuántico en comparación con el algoritmo secuencial clásico que requiere un número exponencial, sus aplicaciones han sido limitadas. Sin embargo, recientemente, este algoritmo ha sido utilizado para resolver distintos problemas, entre ellos, en lenguajes formales para determinar diferentes partes en una estructura de datos de tipo arreglo y aplicar funciones específicas a cada una de las partes [10]. También la utili-

zación del algoritmo de Deutsch y Jozsa ha sido propuesto dentro del contexto del análisis de datos masivos [11].

Dada la importancia del algoritmo de Deutsch y Jozsa en el entendimiento de algoritmos cuánticos, presentamos su implementación en el lenguaje de programación cuántica GAMA (lenguaje de propósito específico diseñado para la simulación de algoritmos cuánticos [12]). En particular, tal implementación está orientada a la enseñanza de algoritmos cuánticos. Durante su desarrollo encontramos que la parte modular de este algoritmo consiste en la construcción de la compuerta cuántica U_f , introducida en la Sec. 2.5, con la cual es posible la evaluación de una sola función para múltiples valores de entrada.

Este artículo tiene la composición siguiente: en la Sec. 2 introducimos brevemente los conceptos fundamentales de la computación cuántica, así como la definición de un algoritmo cuántico, en las Secs. 3 y 4 describimos el algoritmo cuántico de Deutsch y Jozsa y su implementación, en la Sec. 5 presentamos los resultados y una breve discusión de los resultados, y, finalmente, en la Sec. 6 presentamos las conclusiones del trabajo.

2. Fundamentos de algoritmos cuánticos

En la computación cuántica se trabaja con sistemas cuánticos finitos y se trata con espacios vectoriales complejos de dimensión finita.

2.1. Bits cuánticos

En el modelo de computación clásica la unidad básica de información es el *bit*, el cual asume uno de dos posibles valores, ya sea 0 o 1. Análogamente, el modelo de computación cuántica tiene como unidad básica el *qubit*, que puede asumir una superposición de dos valores, o estados, distintos, digamos $|0\rangle$ o $|1\rangle$, los cuales representan los dos valores de un bit clásico.

Los vectores $|0\rangle = [1 \ 0]^T$ y $|1\rangle = [0 \ 1]^T$ forman la base canónica en \mathbb{C}^2 . Los *covectores*, o *funcionales lineales* $\langle x| = |x\rangle^H$ (el superíndice H denota transposición conjugada), con $x \in \{0, 1\}$, forman la base del espacio dual $(\mathbb{C}^2)^*$, que al ser de Hilbert, es isomorfo a \mathbb{C}^2 mismo.

Esta notación tiene la ventaja de poder especificar las transformaciones cuánticas (ver Sec. 2.4) sobre estados cuánticos en términos de la base canónica. Por ejemplo, la transformación que intercambia los estados $|0\rangle$ y $|1\rangle$ está dada por la matriz $X = |0\rangle\langle 1| + |1\rangle\langle 0|$.

Un *qubit* es una superposición de bits clásicos, es decir, es una combinación lineal $a|0\rangle + b|1\rangle$, donde a y b son números complejos tales que $|a| + |b| = 1$. El espacio de Hilbert complejo de dimensión 2, con base ortonormal $E = [|0\rangle, |1\rangle]$, contiene a los qubits como sus vectores unitarios.

El mecanismo de *medición* (ver Sec. 2.3) de qubits se hace con respecto a una base ortonormal, en este caso E , la probabilidad de que el valor medido sea $|0\rangle$ es $|a|^2$ y la probabilidad de que el valor medido sea $|1\rangle$ es $|b|^2$.

Aunque un qubit puede variar de manera continua entre un conjunto de estados cuánticos, después de su medición es posible que asuma un único estado *determinista*, es decir, extrae un simple bit clásico. La medición de qubits es un proceso que solamente puede ser aplicada una vez, ya que un qubit al ser medido pierde su estado de superposición.

2.2. Registros cuánticos

Dado un entero $k \geq 1$, sea $n = 2^k$. Un *sistema cuántico* de k -qubits varía en un espacio de estados de n dimensiones. Cualquier vector de la forma

$$\psi = \sum_{i=0}^{n-1} \mu_i |i\rangle \quad (1)$$

donde $\mu_i \in \mathbb{C}$ y $\sum_{i=0}^{n-1} |\mu_i|^2 = 1$, es un vector unitario en \mathbb{C}^n . De hecho, al representar cada índice en base 2, $i = (i_{k-1} \cdots i_1 i_0)_2$ se podría escribir

$$\psi = \sum_{i_{k-1}, \dots, i_1, i_0=0}^1 \mu_{i_{k-1} \cdots i_1 i_0} |i_{k-1} \cdots i_1 i_0\rangle \quad (2)$$

Tal vector unitario ψ es un *quregistro*.

La relación entre k y n es lo que determina el poder real de la computación cuántica: un *quregistro* puede asumir, al aplicar un operador de medición, un número exponencial, en términos de k , de estados deterministas. Así, la concatenación de k *qubits*, produce vectores de dimensión 2^k . Desde el

punto de vista algebraico, los estados cuánticos se combinan mediante el *producto tensorial* (\otimes): el producto tensorial de k qubits es un k -*quregistro*.

Recordamos brevemente: sean V y W dos espacios complejos de Hilbert, de dimensión finita, con bases ortonormales respectivas $\{v_i\}_{i=0}^{n-1}$ y $\{w_j\}_{j=0}^{m-1}$. El *producto tensorial* $V \otimes W$ tiene como base $\{v_i \otimes w_j\}_{i=0, \dots, n-1}^{j=0, \dots, m-1}$. Por tanto, $\dim(V \otimes W) = \dim(V) \cdot \dim(W)$.

En la potencia tensorial $(\mathbb{C}^2)^{\otimes k} \approx \mathbb{C}^{2^k}$, se suele escribir

$$|i_{k-1} \cdots i_1 i_0\rangle = |i_{k-1}\rangle \otimes \cdots \otimes |i_1\rangle \otimes |i_0\rangle.$$

También se tiene que en la esfera unitaria del espacio \mathbb{C}^{2^k} existen vectores que no son productos tensoriales de vectores de dimensión menor. Estos son llamados *entrelazados*.

El origen del poder del cómputo cuántico se deriva de la posibilidad de codificar procesos como la evolución de estados cuánticos.

2.3. Toma de mediciones

La *medición de qubits* constituye otra gran diferencia entre el modelo clásico y el modelo cuántico, puesto que se puede medir (leer) el valor que tiene un bit. Sin embargo, no se puede leer el valor de un qubit sin alterarlo, ya que éste se convierte en un estado determinista. Dado un qubit

$$\psi = \mu_0 |0\rangle + \mu_1 |1\rangle \quad (3)$$

donde $\mu_0, \mu_1 \in \mathbb{C}$, $|\mu_0|^2 + |\mu_1|^2 = 1$, su medición proporcionará $|0\rangle$ con probabilidad $|\mu_0|^2$ o $|1\rangle$ con probabilidad $|\mu_1|^2$.

De manera similar, al tener un k -*quregistro* ψ como en la ec. (1), para cada $i = 0, \dots, 2^k - 1$, la probabilidad de que tras la medición el estado ψ asuma el estado determinista $|i_2\rangle = |i_{k-1} \cdots i_1 i_0\rangle$ es $|\mu_i|^2 = |\mu_{i_{k-1} \cdots i_1 i_0}|^2$ (véase la Ec. (2)).

Esto propicia comportamientos particulares entre los quregistros entrelazados y los que no lo son. Por ejemplo, en un k -*quregistro* de la forma $\psi = (a_0|0\rangle + a_1|1\rangle) \otimes \phi$, donde ϕ es un $(k-1)$ -*quregistro* se tendría que, al tomar una medición en su primer qubit, éste transitaría a uno $|\varepsilon\rangle$, con $\varepsilon \in \{0, 1\}$, con probabilidad $|a_\varepsilon|^2$. Luego ψ transitaría a $|\varepsilon\rangle \otimes \phi$, es decir, la medición del primer qubit no influye sobre las de los demás. En cambio, para un estado maximalmente entrelazado, digamos $a_0|0 \cdots 0\rangle + a_1|1 \cdots 1\rangle$, con $|a_0|^2 + |a_1|^2 = 1$, al hacer una medición en su primer qubit, éste transitaría a uno $|\varepsilon\rangle$, con $\varepsilon \in \{0, 1\}$, con probabilidad $|a_\varepsilon|^2$, y el quregistro completo transitaría a $|\varepsilon \cdots \varepsilon\rangle$, es decir ¡al determinar el primer qubit, los demás quedarán determinados también y, en este caso, con el mismo valor! Esta es la célebre “acción escalofriante” (*spooky action*) que originó la llamada *paradoja ERP* (por Einstein, Rosen y Podolsky), de los estados entrelazados.

2.4. Compuertas cuánticas

Así como una computadora clásica utiliza circuitos lógicos que permiten la manipulación de información mediante compuertas, en el cómputo cuántico existen mecanismos para manipular *qregistros*. El análogo a los circuitos lógicos son las *compuertas cuánticas* o bien *quocompuertas*. Las compuertas cuánticas y las composiciones de ellas transforman linealmente un estado inicial ψ_1 en un estado final ψ_2 , conservando siempre sus normas y ángulos, es decir, son transformaciones *unitarias* [13]. Cuando se fija una base del espacio, cualquier transformación unitaria queda descrita por una matriz. Sea A^\dagger la conjugada transpuesta de la matriz A . Una matriz A es unitaria si se cumple que $AA^\dagger = I$. Las compuertas son transformaciones unitarias en espacios de Hilbert [13]. Dado este hecho, una de las consecuencias más importantes de las compuertas cuánticas es que son *reversibles*, véase [1].

2.4.1. Transformaciones lineales sobre qubits

Para entender mejor el concepto de compuerta cuántica, veremos algunos ejemplos muy sencillos que operan sobre qubits. A continuación, mostramos las matrices de Pauli, las cuales pueden generar cualquier transformación lineal de dimensión 2, a partir de una combinación lineal de ellas con coeficientes complejos. En la primera columna aparece cada matriz y en la segunda la transformación lineal correspondiente, especificada por su acción en la base canónica:

$$\begin{aligned} I &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & I : |0\rangle &\mapsto |0\rangle \\ X &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & X : |0\rangle &\mapsto |1\rangle \\ Z &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} & Z : |0\rangle &\mapsto |0\rangle \\ && |1\rangle &\mapsto -|1\rangle \end{aligned}$$

Estas transformaciones tienen un nombre convencional. I es la transformación identidad, X es la negación y Z es un corrimiento de fase. Cualquiera de éstas cumple la propiedad de ser transformaciones unitarias.

Otra compuerta muy común es *Cnot* (*Controlled-NOT gate*), la cual “niega al segundo qubit siempre que el primero esté afirmado”. Actuando en la base de los 2-qregistros es

$$\text{Cnot} : \begin{aligned} |0x\rangle &\mapsto |0\rangle \otimes |x\rangle \\ |1x\rangle &\mapsto |1\rangle \otimes X|x\rangle, \end{aligned}$$

y tiene como matriz

$$\text{Cnot} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & X \end{pmatrix}$$

donde X es la compuerta de negación. El primer qubit es *de control* y el segundo *de salida* (se dice también “*target*”). *Cnot* es unitaria y no puede ser descompuesta como el producto tensorial de dos transformaciones de qubits.

Otra transformación importante que opera sobre un qubit es la de *Hadamard*:

$$H : \begin{aligned} |0\rangle &\mapsto \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ |1\rangle &\mapsto \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \end{aligned}$$

la cual tiene importantes aplicaciones. Cuando ésta es aplicada a $|0\rangle$, H crea una superposición del estado $(1/\sqrt{2})(|0\rangle + |1\rangle)$. Aplicada a n bits genera una superposición de 2^n posibles estados:

$$(H \otimes H \otimes \cdots \otimes H)|00\cdots 0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle.$$

$H^{\otimes n} = (H \otimes H \otimes \cdots \otimes H)$ es llamada de *Walsh*, o de *Walsh-Hadamard*, y es práctica común denotarla como W_n . De manera recursiva:

$$W_1 = H, \quad W_{n+1} = H \otimes W_n.$$

A diferencia de la computación clásica, en la que se puede realizar la copia de información sin ningún problema, en la computación cuántica la accesibilidad a la información tiene una limitante importante. La mecánica cuántica no permite la copia exacta de estados cuánticos.

Teorema de no-clonación. *No es posible hacer una copia o clonación de un estado cuántico* [14].

Como un mero bosquejo, supongamos que U fuera una transformación unitaria que clone estados, es decir $U(|a0\rangle) = |aa\rangle$ cualquiera que sea $|a\rangle$. Si se considera un par de estados cuánticos ortogonales, digamos $|a\rangle$ y $|b\rangle$, para el estado $|c\rangle = (1/\sqrt{2})(|a\rangle + |b\rangle)$ resulta, por un lado

$$\begin{aligned} U(|c0\rangle) &= U\left(\frac{1}{\sqrt{2}}(|a0\rangle + |b0\rangle)\right) \\ &= \frac{1}{\sqrt{2}}(U(|a0\rangle) + U(|b0\rangle)) \\ &= \frac{1}{\sqrt{2}}(|aa\rangle + |bb\rangle) \end{aligned}$$

y por otro lado, siendo U de clonación

$$\begin{aligned} U(|c0\rangle) &= |cc\rangle \\ &= \frac{1}{\sqrt{2}}(|a\rangle + |b\rangle) \otimes \frac{1}{\sqrt{2}}(|a\rangle + |b\rangle) \\ &= \frac{1}{\sqrt{2}}(|aa\rangle + |bb\rangle) + \frac{1}{\sqrt{2}}(|ab\rangle + |ba\rangle) \end{aligned}$$

lo que implica $(1/\sqrt{2})(|ab\rangle + |ba\rangle) = 0$, lo cual no es posible.

También es claro que clonar es imposible mediante la sola toma de mediciones, pues esta última destruye estados.

2.5. Algoritmos cuánticos

Un *algoritmo cuántico* consiste de la ejecución de una serie de compuertas cuánticas sobre entidades, que pueden ser *qubits* o *queregistros*, seguida de una toma de medición.

Para simular un algoritmo cuántico, luego de una inicialización de la máquina de estados y de sus *qubits* y *queregistros*, se aplican las transformaciones unitarias indicadas. Dado que la *toma de mediciones* es probabilista, se ha de disponer de un generador de números aleatorios. Dependiendo de la *toma de mediciones* se decide si acaso se concluye el proceso, o bien, se lo reinicia.

El poder de la computación cuántica lo ocasiona su *parallelismo inherente* y el entrelazamiento.

Es posible, por ejemplo, dada una función f , la evaluación simultánea de valores $f(x)$ para muchos valores de x en la simple aplicación de una compuerta cuántica U_f .

En efecto, una forma cuántica de llevar a cabo esto es considerando una computadora de 2-queregistros que iniciando en el estado $|x, y\rangle$, aplica U_f para llegar al estado $|x, y \oplus f(x)\rangle$, donde \oplus es la adición módulo 2; el primer registro x denota al registro de datos de entrada, el segundo registro y es de salida. Nótese que si $y = 0$ la operación \oplus deja el valor de $f(x)$. Si el registro de entrada es inicializado en la superposición $|h\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$, la cual es creada aplicando la compuerta de Hadamard al estado $|0\rangle$, al aplicar U_f resulta

$$U_f|h0\rangle = \frac{|0, f(0)\rangle + |1, f(1)\rangle}{\sqrt{2}}.$$

Este estado contiene información de ambos valores $f(0)$ y $f(1)$, pero no da uno determinado, y, al tomar una medición, se arribará a cualquiera de los estados deterministas $|0, f(0)\rangle$, $|1, f(1)\rangle$ con igual probabilidad.

3. Algoritmo cuántico de Deutsch y Jozsa

La idea de los algoritmos cuánticos es almacenar información en una superposición de estados cuánticos, manipularlos mediante transformaciones unitarias y extraer información útil del estado resultante.

Veamos el algoritmo de *Deutsch y Jozsa*, el cual fue de los primeros algoritmos de tipo cuántico [15]. Éste puede resolver más rápido, usando efectos cuánticos [3], un problema que una computadora clásica lo haría en un número exponencial de exponencialmente.

Una función booleana $f : \mathbb{Z}_2^n \rightarrow \mathbb{Z}_2$ es *constante* si es idénticamente 0 o 1, y es *equilibrada* si el número de veces que asume el valor 0 es el mismo que el que asume el valor 1, es decir

$$\begin{aligned} \text{card}\{x \in \mathbb{Z}_2^n \mid f(x) = 0\} &= 2^{n-1} \\ &= \text{card}\{x \in \mathbb{Z}_2^n \mid f(x) = 1\}. \end{aligned}$$

Se ve fácilmente que hay $\binom{2^n}{2^{n-1}} = O(n^{2^{n-1}})$ funciones booleanas equilibradas, lo cual es un crecimiento doblemente exponencial. Por ejemplo, los primeros valores son los siguientes:

n	1	2	3	4	5
$\binom{2^n}{2^{n-1}}$	2	6	70	12870	601080390

TABLA I. Funciones constantes o equilibradas de dos entradas.

x_1	x_2	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
0	0	0	0	0	0	1	1	1	1
0	1	0	0	1	1	0	0	1	1
1	0	0	1	0	1	0	1	0	1
1	1	0	1	1	0	1	0	0	1

Sea D_n el conjunto de todas las funciones booleanas constantes o equilibradas. En el cuadro 3 se muestra el conjunto D_2 de las $8 = 2 + 6$ funciones constantes (f_0 y f_7) o equilibradas (f_1-f_6).

El propósito del algoritmo de Deutsch y Jozsa es decidir, para una función dada $f : \mathbb{Z}_2^n \rightarrow \mathbb{Z}_2$ que esté en D_n , $f \in D_n$, si acaso es constante o equilibrada “utilizando un solo paso de cómputo”.

Clásicamente, habría que evaluar los valores $f(x)$ de f para todas las entradas x . En el peor caso se tendría que recorrer las 2^n posibilidades. Veamos el procedimiento del algoritmo cuántico, donde con una sola evaluación de f se determina si es constante o equilibrada.

Se considera el operador unitario U_f definido en el espacio de los $(n+1)$ -queregistros, es decir, actúa en el espacio complejo de dimensión 2^{n+1} . En cada entrada $|xy\rangle = |x\rangle \otimes |y\rangle$, donde $|x\rangle$ es un n -quregistro e $|y\rangle$ es un qubit, produce el $(n+1)$ -quregistro

$$|\Psi\rangle = U_f|xy\rangle = |x\rangle \otimes |y \oplus f(x)\rangle.$$

U_f aparece bosquejado en la Fig. 1.

Así pues

$$U_f : |xy\rangle = |x\rangle \otimes |y\rangle \mapsto |x\rangle \otimes |y \oplus f(x)\rangle \quad (4)$$

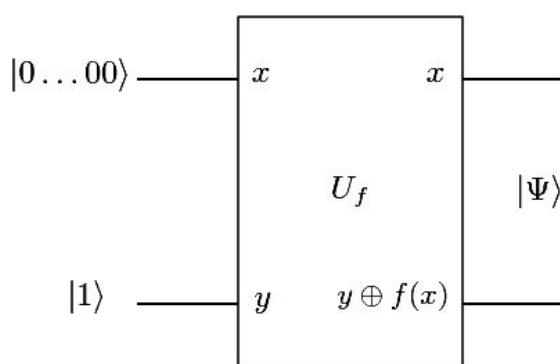


FIGURA 1. Ilustración de la compuerta cuántica U_f utilizada en el algoritmo de Deutsch y Jozsa.

Inicialmente se considera el estado de entrada

$$\underbrace{|0\rangle \otimes \dots \otimes |0\rangle}_{n \text{ veces}} \otimes |1\rangle = \underbrace{|0\dots0}_n |1\rangle = |0^n 1\rangle. \quad (5)$$

Luego, a cada qubit se le aplica la compuerta de Hadamard:

$$\underbrace{H \otimes \dots \otimes H}_{n+1 \text{ veces}} |0^n 1\rangle = \frac{1}{2^{n/2}} \sum_{x \in \mathbb{Z}_2^n} |x\rangle \otimes H|1\rangle. \quad (6)$$

Observamos que para cada $x \in \mathbb{Z}_2^n$, U_f actúa como sigue

$$U_f : |x0\rangle \mapsto |xf(x)\rangle, \quad |x1\rangle \mapsto |x(1 \oplus f(x))\rangle$$

y también, puesto que $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$,

$$\begin{aligned} U_f(|x\rangle \otimes H|1\rangle) &= \frac{1}{\sqrt{2}} \begin{cases} |x0\rangle - |x1\rangle & \text{si } f(x) = 0 \\ |x1\rangle - |x0\rangle & \text{si } f(x) = 1 \end{cases} \\ &= \frac{1}{\sqrt{2}}(-1)^{f(x)}(|x0\rangle - |x1\rangle). \end{aligned}$$

De (6) se ha de tener

$$U_f \circ H^{\otimes(n+1)} |0^n 1\rangle = \frac{1}{2^{(n+1)/2}} \sum_{x \in \mathbb{Z}_2^n} (-1)^{f(x)} (|x0\rangle - |x1\rangle).$$

Al aplicar $H^{\otimes(n+1)}$ a cada $(n+1)$ -quregistro $|xb\rangle$, $b \in \{0, 1\}$:

$$\begin{aligned} H^{\otimes(n+1)}|xb\rangle &= \frac{1}{2^{(n+1)/2}} \left(\bigotimes_{i=0}^{n-1} (|0\rangle + (-1)^{x_i} |1\rangle) \right) \otimes \\ &\quad (|0\rangle + (-1)^b |1\rangle) \\ &= \frac{1}{2^{(n+1)/2}} \left(\sum_{z \in \mathbb{Z}_2^n} (-1)^{\langle x|z \rangle} |z\rangle \right) \otimes \\ &\quad (|0\rangle + (-1)^b |1\rangle) \end{aligned}$$

donde $\langle x|z \rangle$ es el producto interno en \mathbb{Z}_2^n . Así,

$$H^{\otimes(n+1)}(|x0\rangle - |x1\rangle) = \frac{2}{2^{(n+1)/2}} \left(\sum_{z \in \mathbb{Z}_2^n} (-1)^{\langle x|z \rangle} |z\rangle \right) \otimes |1\rangle$$

de donde se podrá expresar

$$H^{\otimes(n+1)} \circ U_f \circ H^{\otimes(n+1)} |0^n 1\rangle = \sum_{z \in \mathbb{Z}_2^n} c_z |z1\rangle$$

siendo, para cada $z \in \mathbb{Z}_2^n$,

$$c_z = \frac{1}{2^n} \sum_{x \in \mathbb{Z}_2^n} (-1)^{f(x) + \langle x|z \rangle}.$$

La probabilidad de que al tomar una medición, respecto a la base canónica, quede el estado determinista $|0^n 1\rangle$ es entonces $|c_0|^2$ y éste tiene como valor 1 si f es constante y valor 0 si f es equilibrada. Por tanto, al tomar una medición a $H^{\otimes(n+1)} \circ U_f \circ H^{\otimes(n+1)} |0^n 1\rangle$, si se obtiene $|0^n 1\rangle$, se decide con toda certeza que f es constante, en otro caso, f es equilibrada. Es decir, con la medición de los primeros n -qubits se puede saber si f se trata de una función equilibrada o constante [15-17].

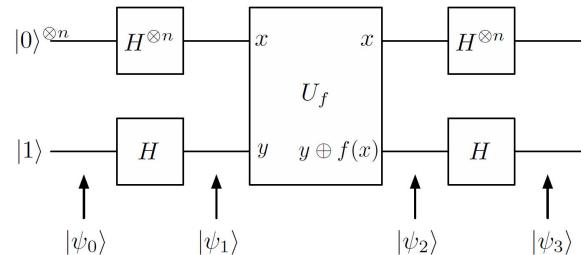


FIGURA 2. Algoritmo de Deutsch y Jozsa para un quregistro de control $|0\rangle^{\otimes n}$ y un estado $|1\rangle$.

4. Implementación del algoritmo de Deutsch y Jozsa

Dada la definición del algoritmo de Deutsch y Jozsa en la Sec. 3, a continuación introducimos el procedimiento para llevar a cabo su implementación en el simulador de algoritmos cuánticos.

La Fig. 2 muestra esquemáticamente la serie de transformaciones de tipo cuántica ($|\psi_1\rangle$, $|\psi_2\rangle$ y $|\psi_3\rangle$) que se le aplican al estado inicial $|\psi_0\rangle$ para resolver el problema de Deutsch y Jozsa. Particularmente, estas tres transformaciones son las que generan los estados intermedios $|\psi_1\rangle$, $|\psi_2\rangle$ y $|\psi_3\rangle$ que constituyen el algoritmo cuántico de Deutsch y Jozsa.

A continuación describimos la serie de transformaciones necesarias en el algoritmo cuántico de Deutsch y Jozsa para posteriormente introducir el pseudocódigo del algoritmo en la Sec. 5.

El estado inicial, $|\psi_0\rangle$, es generado a partir del producto tensorial entre un quregistro de control $|0^{\otimes n}\rangle$ (usado para almacenar los argumentos de la función) y un qubit de función $|1\rangle$ (utilizado para evaluar la función) [6],

$$|\psi_0\rangle = |0\rangle^{\otimes n} \otimes |1\rangle.$$

La primera transformación, $|\psi_1\rangle$, consiste en aplicar la compuerta cuántica de Hadamard al estado inicial, $|\psi_0\rangle$,

$$\begin{aligned} |\psi_1\rangle &= H^{\otimes n+1}|\psi_0\rangle \\ &= H^{\otimes n}|0\rangle^{\otimes n} \otimes H|1\rangle \\ &= \sum_{x \in \mathbb{Z}_2^n} \frac{1}{\sqrt{2^n}} |x\rangle \otimes \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right], \end{aligned} \quad (7)$$

donde x , es la representación binaria de un número entero entre $[0, 2^{n-1}]$, utilizando n qubits.

La segunda transformación consiste en aplicar la compuerta U_f a $|\psi_1\rangle$. El operador U_f se encarga de realizar el mapeo de un quregistro de la siguiente manera:

$$U_f : |xa\rangle = |x\rangle \otimes |a\rangle \rightarrow |x\rangle \otimes |a \oplus f(x)\rangle$$

donde \oplus denota la suma entera módulo 2, es decir:

$$0 \oplus 0 = 0, \quad 0 \oplus 1 = 1, \quad 1 \oplus 0 = 1, \quad 1 \oplus 1 = 0.$$

El operador U_f tiene el efecto de introducir la fase $(-1)^{f(x)}$ al estado $|x\rangle$. Así al aplicar U_f al estado $|\psi_1\rangle$, de la Ec. (7) tenemos

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \mathbb{Z}_2^n} (-1)^{f(x)} |x\rangle \otimes \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right], \quad (8)$$

Finalmente, la tercera transformación consiste en aplicar el operador de Hadamard a $|\psi_2\rangle$, de lo que obtenemos

$$|\psi_3\rangle = \frac{1}{\sqrt{2^n}} \sum_{x,z \in \mathbb{Z}_2^n} (-1)^{f(x)+\langle x|z\rangle} |z\rangle \otimes |1\rangle, \quad (9)$$

El estado resultante es $|\psi_3\rangle$. Para conocer si la función es equilibrada o constante, es necesario interpretar la medición del estado $|\psi_3\rangle$. Si el resultado de la medición es el estado determinista $|0^{\otimes n}1\rangle$ entonces sabemos que se trata de una función constante, en cualquier otro caso la función es equilibrada.

4.1. El operador U_f

Como se aprecia en la Ec. (8), toda la información de la función a evaluar f se incluye en el quregistro $|\psi_2\rangle$ al aplicar el operador U_f . En esta sección veremos la manera de expresar a U_f en forma matricial, con el propósito de tener una representación que se pueda utilizar en la simulación.

Retomemos brevemente el comportamiento del algoritmo. Notemos que el qubit de función debe ser $|1\rangle$, debido a que al aplicarle una compuerta de Hadamard a tal qubit aparece un signo negativo, como se aprecia en la Ec. (7). Este signo se distribuye hacia la mitad de los elementos del quregistro $|\psi_0\rangle$ durante la primera transformación. Este signo

negativo es el que dispersa hacia los estados $|x\rangle$ de la Ec.(8) al aplicarle el operador U_f al estado $|\psi_1\rangle$ y que en la operación de medición ayuda a determinar si la función $f(x)$ es constante o equilibrada.

Cuando el operador U_f se aplica al estado

$$|x\rangle \otimes H|1\rangle = |x\rangle \otimes \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle),$$

se tiene

$$\begin{aligned} U_f (|x\rangle \otimes H|1\rangle) &= U_f \left(|x\rangle \otimes \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \right) \\ &= \frac{1}{\sqrt{2}} \begin{cases} |x0\rangle - |x1\rangle & \text{si } f(x) = 0 \\ |x1\rangle - |x0\rangle & \text{si } f(x) = 1 \end{cases} \\ &= (-1)^{f(x)} |x\rangle \otimes H|1\rangle. \end{aligned}$$

Y podemos ver que el signo negativo, para cada estado $|x\rangle$, depende del valor de $f(x)$.

Para expresar matricialmente el operador U_f , es necesario definirlo en términos de submatrices $U_{f(x)}$. Podemos escribir a la matriz $U_{f(x)}$ de tamaño 2×2 como:

$$U_{f(x)} = \begin{pmatrix} 1 \oplus f(x) & f(x) \\ f(x) & 1 \oplus f(x) \end{pmatrix}. \quad (10)$$

La forma general al operador U_f como una matriz $2^{n+1} \times 2^{n+1}$ es definida como

$$U_f = \begin{pmatrix} U_{f(0)} & 0 & \cdots & 0 \\ 0 & U_{f(1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & U_{f(2^n-1)} \end{pmatrix}. \quad (11)$$

Entonces otra forma de escribir la Ec. (8) es: $U_f(2^n - 1)$

$$|\psi_3\rangle = \begin{pmatrix} U_{f(0)} & 0 & \cdots & 0 \\ 0 & U_{f(1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & U_{f(2^n-1)} \end{pmatrix} (H^{\otimes n} |0\rangle^{\otimes n} \otimes H|1\rangle). \quad (12)$$

Para el caso de $n = 2$, la matriz U_f se escribe como

$$U_f = \begin{pmatrix} 1 \oplus f_{00} & f_{00} & 0 & 0 & 0 & 0 & 0 & 0 \\ f_{00} & 1 \oplus f_{00} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 \oplus f_{01} & f_{01} & 0 & 0 & 0 & 0 \\ 0 & 0 & f_{01} & 1 \oplus f_{01} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \oplus f_{10} & f_{10} & 0 & 0 \\ 0 & 0 & 0 & 0 & f_{10} & 1 \oplus f_{10} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \oplus f_{11} & f_{11} \\ 0 & 0 & 0 & 0 & 0 & 0 & f_{11} & 1 \oplus f_{11} \end{pmatrix}, \quad (13)$$

donde

$$f_{00} = f(0, 0) \quad (14)$$

$$f_{01} = f(0, 1) \quad (15)$$

$$f_{10} = f(1, 0) \quad (16)$$

$$f_{11} = f(1, 1). \quad (17)$$

Se puede ver en las matrices (11) y (13) que si la función f es constante, entonces se tiene a la matriz unitaria $2^{n+1} \times 2^{n+1}$ o a una matriz con submatrices $U_{f(x)}$ antidiagonales en su diagonal principal. Por lo que al aplicar esta matriz a $|\psi_1\rangle$ se obtendrá a $\pm|\psi_1\rangle$, y al aplicar la compuerta de Hadamard, del último paso del algoritmo de Deutsch y Jozsa se tiene $\pm|0\rangle^{\otimes n} \otimes |1\rangle$. Entonces al realizar la medición sobre los primeros n -qubits obtenemos que la amplitud del estado $|0\rangle^{\otimes n}$ es uno.

Si la función es equilibrada, entonces U_f se expresa como una combinación de matrices $U_{f(x)}$ diagonales y antidiagonales. De esta forma, el estado resultante $|\psi_3\rangle = |z\rangle \otimes H|1\rangle$ contiene al elemento $|z\rangle$, el cual es cualquier estado distinto de $|0\rangle^{\otimes n}$. Así, la medición final dará como resultado que la amplitud de $|0\rangle^{\otimes n}$ es 0.

4.2. Método experimental

Como plataforma experimental utilizamos GAMA, el cual es un lenguaje de programación para simular la ejecución de algoritmos cuánticos en una computadora clásica [12]. Este lenguaje es de tipo imperativo, estructurado y de propósito específico. GAMA cuenta con un módulo de depuración que permite llevar la secuencia de los diferentes cambios de los estados de las variables en las diferentes entidades de un algoritmo cuántico, incluyendo qubits, qregisters, qgates, así como las operaciones entre ellas: mediciones, producto interior (\cdot) y productor exterior (\otimes).

5. Resultados y discusión

5.1. Pseudocódigo

Ahora explicaremos el pseudocódigo del algoritmo cuando $n = 2$ con el fin de ilustrarlo, implementarlo y probarlo en nuestro simulador de algoritmos cuánticos. El estado inicial $|\psi_0\rangle$ se expresa como: $|\psi_0\rangle = |00\rangle \otimes |1\rangle$ y el estado final $|\psi_3\rangle$ se obtiene, como hemos mencionado, después de la aplicación de una serie de transformaciones de la compuerta de Hadamard y U_f a dicho estado inicial como $|\psi_3\rangle = H^{\otimes 3}U_fH^{\otimes 3}|\psi_0\rangle$ donde $H^{\otimes 3} = H \otimes H \otimes H$ y U_f está definido como en la Ec.(13).

El algoritmo 1 muestra el pseudocódigo de los pasos necesarios para resolver este ejemplo. Primero, se lleva a cabo la inicialización del operador U_f con la función `Initialize()` descrita en el algoritmo 2. Como hemos mencionado el operador U_f está conformado de evaluaciones de la función $f(x_1, x_2)$. Para demostrar el funcionamiento

ALGORITMO 1. Algoritmo cuántico de Deutsch y Jozsa para la función $f(x_1, x_2)$.

```

 $U_f = Initialize();$ 
 $\psi_0 = |00\rangle \otimes |1\rangle;$ 
 $H3 = H^{\otimes 3};$ 
 $\psi_1 = H3 * \psi_0;$ 
 $\psi_2 = U_f * \psi_1;$ 
 $\psi_3 = H3 * \psi_2;$ 
 $\psi = measure(\psi_3, 0);$ 
return  $\psi;$ 
```

ALGORITMO 2. Función `Initialize()`.

Ensure: Una quocompuerta uf que representa el operador U_f
Ogate $uf = \{$

```

 $\{(1 + f(0, 0)) \% 2, f(0, 0), 0, 0, 0, 0, 0, 0\},$ 
 $\{f(0, 0), (1 + f(0, 0)) \% 2, 0, 0, 0, 0, 0, 0\},$ 
 $\{0, 0, (1 + f(0, 1)) \% 2, f(0, 1), 0, 0, 0, 0\},$ 
 $\{0, 0, f(0, 1), (1 + f(0, 1)) \% 2, 0, 0, 0, 0\},$ 
 $\{0, 0, 0, 0, (1 + f(1, 0)) \% 2, f(1, 0), 0, 0\},$ 
 $\{0, 0, 0, 0, f(1, 0), (1 + f(1, 0)) \% 2, 0, 0\},$ 
 $\{0, 0, 0, 0, 0, 0, (1 + f(1, 1)) \% 2, f(1, 1)\},$ 
 $\{0, 0, 0, 0, 0, 0, f(1, 1), (1 + f(1, 1)) \% 2\}$ 
```

$\}$;

return uf ;

(aquí, $z \% 2$ denota “ z módulo 2”, es decir el residuo de z al ser dividido entre 2.)

ALGORITMO 3. Función $f(x_1, x_2)$.

Require: Dos valores de entrada x_1 y x_2 .

Ensure: El resultado de la función evaluada en los valores de entrada.

```

Int  $f(\text{Int } x_1, \text{ Int } x_2) \{$ 
  if  $x_1 == x_2$  then
    return 0;
  end if
  return 1; }
```

del algoritmo, seleccionamos una de las funciones presentadas anteriormente en el Cuadro 3. En este caso, la función equilibrada f_3 del Cuadro 3 fue implementada y su pseudocódigo se muestra en el algoritmo 3. Los siguientes pasos corresponden a las transformaciones hechas al estado inicial ψ_0 . La primera equivale a aplicarle la compuerta $H^{\otimes 3}$. La segunda consiste en la aplicación del operador U_f y finalmente la tercera en aplicarle la compuerta $H^{\otimes 3}$ nuevamente.

Finalmente, el estado ψ es el estado resultante, al cual se le realizará una toma de medición, haciéndolo por cualquier qubit que corresponda al qregister de control, que en este caso tomamos el qubit 0 para medirlo.

5.2. Simulación en GAMA

Presentamos el código GAMA para llevar a cabo la simulación del algoritmo. Por ahora, estamos interesados únicamente en obtener el resultado de la simulación y no en el de estados intermedios durante una depuración. De tal manera que el estado resultante de cada una de las variables involucradas se puede observar en la pestaña inferior *Configurations*, en donde el quregistro `psi` contiene la información de si acaso la función se trata de una constante o bien de una equilibrada. En la Sec. 5.3 veremos la interpretación al resultado del algoritmo dado un tal quregistro.

5.3. Resultados

Como resultado de la simulación anterior, para el caso $n = 2$ y la función equilibrada f_3 del Cuadro 3, obtuvimos que la medición del quregistro ψ_3 generó el quregistro $\psi = (1.0 + 0.0i)|111\rangle$. Considerando que el valor del quregistro ψ_0 de entrada es $|001\rangle$, diremos por un lado que si el quregistro resultante arroja el mismo valor del quregistro inicial se trata de una función constante y por otro lado, si se trata de cualquier otro estado sabemos que es una función equilibrada. Bajo esta premisa observamos que el quregistro $\psi = |111\rangle$ indica que la función $f(x_1, x_2)$ es equilibrada.

En este trabajo hemos realizado la simulación del algoritmo cuántico de Deutsch y Jozsa en el lenguaje GAMA. Para ello, hemos establecido un mecanismo para construir la compuerta cuántica U_f en forma matricial, que ayuda a obtener la simulación de su comportamiento. Se puede apreciar que el qubit de función $|1\rangle$ de entrada se utiliza para inducción la secuencia de signos negativos, necesarios para que el algoritmo funcione, y para que el registro de control alcance la dimensión apropiada para operarse con la compuerta cuántica U_f . Por otro lado, y tal vez más delicado, es el hecho de que la compuerta U_f debe inicializarse para cada una de las funciones que el algoritmo de Deutsch y Jozsa tenga de entrada, y si la inicialización no es tomada en cuenta en la complejidad el algoritmo cuántico resulta ser más eficiente que el clásico. Sin embargo, si este mecanismo de inicialización se toma como parte de la ejecución del algoritmo, la complejidad podría ser igual o mayor a la del algoritmo clásico ya que GAMA lleva a cabo una simulación de cómputo cuántico dentro de una computadora clásica.

Además, también implementamos el algoritmo de Deutsch y Jozsa para los siguientes valores $n = 3, 4, 5$. El objetivo de este experimento es conocer su comportamiento y sus implicaciones. Al incrementar el número de qubits, la dimensión de la compuerta U_f aumenta de acuerdo a los detalles mostrados en el Cuadro 5.3. En general, la dimensión de la compuerta U_f para resolver un problema de n qubits es $2^{n-1} \times 2^{n-1}$. Para cualquier tamaño de quregistro, el algoritmo de Deutsch y Jozsa aplica una vez la compuerta Hadamard al registro inicial, una vez la compuerta U_f y finalmente una compuerta Hadamard. Aplicar una compuerta cuántica a un quregistro equivale a la multiplicación de una matriz por

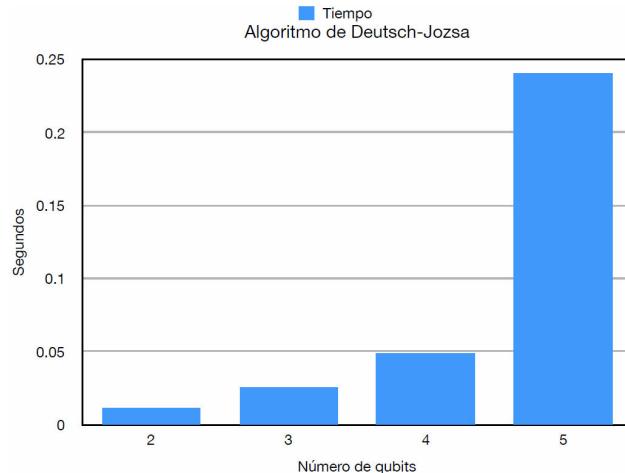


FIGURA 3. Gráfica de tiempos del algoritmo Deutsch y Jozsa con respecto al número de qubits.

TABLE II. Tamaño de las compuertas cuánticas en el algoritmo de Deutsch y Jozsa.

Número de qubits	Dimensión de U_f y Hadamard
2	8X8
3	16X16
4	32X32
5	64X64

un vector. Lo que implica que la complejidad del algoritmo en una computadora clásica es 3 veces $O(m^2)$, donde m es la dimensión de la matriz y del vector. En el algoritmo cuántico, la dimensión de las compuertas es de $m = 2^{n-1}$, y su complejidad después de simularlo en una computadora clásica es de $O((2^{n-1})^2)$. Esto ocasiona que la inicialización de la compuerta cuántica U_f consuma mayor tiempo debido a que el número de funciones a evaluar crece también. Sin embargo, no solo el tiempo de procesamiento tiene un impacto sino también la capacidad de almacenamiento de la máquina puede llegar a ser crucial si se trata de una simulación con un número de qubits muy alto.

La Fig. 3 muestra la gráfica de tiempos de las corridas del algoritmo de Deustch-Jozsa, utilizando una computadora de 32-bits para funciones de dos a cinco qubits de entrada, de lo que podemos apreciar un crecimiento exponencial. Como mencionamos anteriormente esto se debe al tiempo que consume la inicialización de la compuerta U_f . Una de las limitantes del simulador es que el número máximo de qubits que se pueden utilizar en un algoritmo está limitado al número de matrices que se puedan crear dentro del tamaño de la memoria de la computadora.

6. Conclusiones

En este trabajo hemos realizado la simulación del algoritmo cuántico de Deutsch y Jozsa en el lenguaje GAMA con el fin

de explorar sus diferentes componentes. Para ello, hemos establecido un mecanismo para construir la compuerta cuántica U_f en forma matricial, que ayuda a obtener la simulación de su comportamiento. Se puede apreciar que el qubit de función $|1\rangle$ se utiliza para inducción de la secuencia de signos negativos, necesarios para que el algoritmo funcione, y para que el registro de control alcance la dimensión apropiada para operarse con la compuerta cuántica U_f . Por otro lado, la compuerta U_f debe inicializarse para cada una de las funciones que el algoritmo de Deutsch y Jozsa tenga de entrada, y si la inicialización no es tomada en cuenta en la complejidad el algoritmo cuántico resulta ser más eficiente que el clásico. Sin embargo, si este mecanismo de inicialización se toma co-

mo parte de la ejecución del algoritmo, la complejidad podría ser igual o mayor a la del algoritmo clásico.

Agradecimientos

Los autores agradecen al apoyo financiero proporcionado por el Consejo Nacional de Ciencia y Tecnología (CONACyT), así como también a ABACUS, laboratorio de Matemáticas Aplicadas y Supercómputo de alto rendimiento. Evidentemente, nuestra institución de adscripción, el Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, es de especial reconocimiento.

1. R. Feynman, R. Leighton, M. Sands, and E. Hafner, *The Feynman Lectures on Physics*; Vol. I (AAPT, 1965).
2. A. Montanaro, *Quantum algorithms*: an overview, (2015). arXiv preprint arXiv:1511.04206.
3. D. Deutsch and R. Jozsa, *Proc Roy Soc Lond A* **439** (1992) 553-558.
4. P. Bianucci, A. Muller, C. Shih, Q. Wang, Q. Xue, and C. Piermarocchi, *Physical Review B* **69** (2004) 161303.
5. I. L. Chuang, L. M. Vandersypen, X. Zhou, D. W. Leung, and S. Lloyd, *Experimental realization of a quantum algorithm*, (1998) arXiv preprint quant-ph/9801037.
6. D. Collins, K. W. Kim, and W. C. Holton, *Phys. Rev. A* **58** (1998) R1633-R1636.
7. N. Linden, H. Barjat, and R. Freeman, *Chemical Physics Letters*, **296** (1998) 61-67.
8. M. Scholz, T. Aichele, S. Ramelow, and O. Benson, *Physical review letters*, **96** (2006) 180501.
9. J. Zhao, X. Tan, D. Lan, H. Wu, G. Xue, H. Yu, and Y. Yu, *physica status solidi (b)*, **254** (2017).
10. A. Maitra, S. Maitra, and A. K. Pal, *Data Analytics in Quantum Paradigm: An Introduction*, (Springer International Publishing, Cham, 2017). pp. 313-330.
11. M. Batty, A. Casaccino, A. J. Duncan, S. Rees, and S. Severini, *An Application of the Deutsch-Jozsa Algorithm to Formal Languages and the Word Problem in Groups*. In Y. Kawano and M. Mosca, editors, *Theory of Quantum Computation, Communication, and Cryptography*, (Springer Berlin Heidelberg, 2008). pp. 57-69.
12. M. Paredes, *Simulación de Cómputo Cuántico*, tesis de maestría. (Departamento de Computación, Cinvestav, 2007).
13. S. J. Lomonaco, *A Rosetta Stone for Quantum Mechanics with an Introduction to Quantum Computation*, (Proceedings in Symposia in Applied Mathematics, 2000).
14. W. K. Wootters and W. H. Zurek, *Nature*, **299** (1982) 802-803.
15. D. Bouwmeester, A. K. Ekert, and A. Zeilinger, *The Physics of Quantum Information: Quantum Cryptography, Quantum Teleportation, Quantum Computation*. (Springer Publishing Company, Incorporated, 1st edition, 2010).
16. M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*: 10th Anniversary Edition. Cambridge University Press, New York, NY, USA, 10th edition, (2011).
17. B. Thaller, *Visual Quantum Mechanics: Selected Topics with Computer Generated Animations of Quantum Mechanical Phenomena with Cdrom*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1st edition, (2000). Rev. Mex.