



Universidad Católica  
**San Pablo**

## Título de la Tesis

José Carlos Delgado Ramos

Orientador: Prof Dr. Yván Jesús Túpac Valdivia

*Tesis profesional presentada al Programa Profesional de Ciencia de la Computación como parte de los requisitos para obtener el Título Profesional de Lic. en Ciencia de la Computación.*

UCSP- Universidad Católica San Pablo  
Noviembre de 2014

*Esto lo pondré cuando lo termine.*

# Abreviaturas

**QIEA** Quantum-Inspired Evolutionary Algorithm

**PDD** Densidad Determinada Probabilística

# Resumen

---

En la presente tesis se propone una modificación multipoblacional de evolución paralelizada del Algoritmo Evolutivo de Inspiración Cuántica (QIEA) para  $\mathbb{R}$ , el cual introduce un operador de reproducción (*crossover*) basado en el concepto de entrelazamiento (*entanglement*) a nivel intergeneracional entre las diversas poblaciones de individuos cuánticos, con el fin de aplicar algunas mejoras obtenidas por un individuo a otro ya evolucionado tras un determinado número de iteraciones, y de esta manera minimizar el riesgo de converger únicamente hacia un óptimo local.

# Abstract

---

In this thesis, a multi-population parallelized evolution modification of the Quantum-Inspired Evolutionary Algorithm (QIEA) for  $\mathbb{R}$  is proposed. This modification introduces a crossover operator based on the concept of *entanglement* at an intergenerational level among different populations of quantum individuals, with the purpose of applying the improvements obtained by a certain individual to another yet evolutioned individual after a certain number of iterations, so in this way we become able to minimize the risk of convergence towards a local optimal solution.

# Índice general

<b>1. Introducción</b>	<b>2</b>
1.1. Motivación y Contexto . . . . .	2
1.2. Planteamiento del Problema . . . . .	2
1.3. Objetivos . . . . .	3
1.3.1. Objetivos Específicos . . . . .	3
1.4. Organización de la tesis . . . . .	3
1.5. Cronograma . . . . .	3
<b>2. Estado del Arte</b>	<b>4</b>
2.1. Variantes y modificaciones del Algoritmo Evolutivo de Inspiración Cuántica	4
2.2. Aplicaciones de los Algoritmos Evolutivos de Inspiración Cuántica . . . . .	5
2.3. Consideraciones Finales . . . . .	5
<b>3. Marco Teórico</b>	<b>6</b>
3.1. Computación Cuántica . . . . .	6
3.2. Computación Evolutiva . . . . .	7
3.3. Algoritmo Evolutivo de Inspiración Cuántica . . . . .	10
3.4. Algoritmo Evolutivo de Inspiración Cuántica - $\mathbb{R}$ . . . . .	11
3.5. Consideraciones Finales . . . . .	13
<b>4. Propuesta</b>	<b>14</b>
<b>Bibliografía</b>	<b>18</b>

# Índice de tablas

# Índice de figuras

3.1. Proceso de un algoritmo evolutivo. [Eiben and Smith, 2003] . . . . .	9
3.2. Rotación del ángulo al momento de la aplicación del operador Q-gate . . .	12



# Capítulo 1

## Introducción

Este es el primer capítulo de la tesis. Se inicia con el desarrollo de la introducción de la tesis. Es importante que el texto utilice la tabla de abreviaturas correctamente. En el archivo `abreviaturas.tex` contiene la tabla de abreviaturas. Para citar alguna de ellas debes usar los comandos `\ac{tu-sigla-aqui}`. Si es la primera vez que utilizas la sigla ella se expandirá por completo. Por ejemplo, el comando `\ac{CMM}` va a producir: **CMM!** (**CMM!**). Si más adelante repites el mismo comando sólo aparecerá la sigla **CMM!**. Para explorar mucho más este comando es necesario leer su manual disponible en: <http://www.ctan.org/tex-archive/macros/latex/contrib/acronym/>

### 1.1. Motivación y Contexto

En esta sección se va desde aspectos generales a aspectos específicos (como un embudo). No se olvide que es la primera parte que tiene contacto con el lector y que hará que este se interese en el tema a investigar.

El objetivo de esta sección es llevar al lector hacia el tema que se va a tratar en forma específica y dejar la puerta abierta a otras investigaciones

### 1.2. Planteamiento del Problema

En esta sección se realiza el planteamiento del problema que queremos resolver con la tesis. Sea muy puntual y no ocupe más de un párrafo en especificar cual es el problema que desea atacar.

---

## 1.3. Objetivos

En esta sección se colocan los objetivos generales de la tesis. Máximo dos. Si necesita ampliar estos objetivos utilice la sección de objetivos específicos.

### 1.3.1. Objetivos Específicos

En esta sección se coloca el los objetivos específicos de la tesis, que serán aquellos que contesten a las interrogantes de investigación.

## 1.4. Organización de la tesis

En esta sección se coloca cuantos capítulos contendrá la tesis y que se tratará en cada uno de ellos en forma resumida. Dediquele un parrafo de dos o tres lineas a explicar cada capítulo.

## 1.5. Cronograma

Esta sección sólo es para aquellos alumnos que estén presentando su plan de tesis. Esta sección no va en la tesis final.

## Capítulo 2

# Estado del Arte

Mientras que los primeros estudios sobre computación cuántica fueron propuestos en la década de los ochenta [Benioff, 1980] [Benioff, 1982], el primer trabajo sobre técnicas de inteligencia computacional de inspiración cuántica recién fue presentado hacia finales de la década de los noventa [Narayanan and Moore, 1996]. A diferencia de los “auténticos” algoritmos cuánticos tales como el algoritmo de Grover [Grover, 1996] o el algoritmo de Shor [Shor, 1994], los algoritmos de inspiración cuántica no requieren de una computadora cuántica para ser implementados, sino que implementan niveles adicionales de aleatoriedad inspirados en conceptos y principios de la computación cuántica, tales como los *qubits*, la *interferencia* o la *superposición de estados* [Nowotniak, 2010].

El primer trabajo que realmente se puede considerar como un algoritmo evolutivo de inspiración cuántica fue presentado en el año 2001 [Han and Kim, 2000], el mismo que presenta el algoritmo evolutivo de inspiración cuántica base. Es acá cuando se introducen los conceptos de qubits y superposición de estados, además de la representación de individuos basados en el concepto de qubits. Se tratarán estos conceptos con mayor profundidad en el siguiente capítulo.

### 2.1. Variantes y modificaciones del Algoritmo Evolutivo de Inspiración Cuántica

Debido al poco tiempo de vida de este campo -menos de tres lustros, en realidad-, el área se encuentra aún en crecimiento y sólo existen unos pocos cientos de artículos sobre el tema [Nowotniak and Kucharski, 2014]. Desde el año 2007 se ha dado un incremento notable en la investigación relacionada, concentrando esfuerzos mayoritariamente en aplicaciones, seguido por innovaciones tanto en los operadores genéticos como en las representaciones [Nowotniak, 2010].

En el ámbito de operadores genéticos, ya sean modificados o generalizados, se ha realizado avances tales como la implementación de estrategias de búsqueda de escala mutativa y un método de optimización de caos basado en mapas de Hénon [Teng et al., 2010], una búsqueda en la disminución de operaciones al momento de evaluar la aptitud de los indi-

viduos [Malossini et al., 2008], aplicación de rotaciones en los ángulos de las puertas lógicas cuánticas [Zhang et al., 2006], combinación con Cuadrados Latinos [Tsai et al., 2012], generalización del qubit para una cantidad  $n$  de variables además del operador  $Q$ -gate [Alegría and Túpac, 2014].

Cabe señalar también que en muchos casos se utiliza codificación cuántica binaria de la misma manera que el algoritmo original, mientras que en algunos casos se ha hecho la extensión correspondiente para utilizar valores reales [Qin et al., 2008] [Fan et al., 2007] [Zhao et al., 2009]. Esto, a partir del trabajo desarrollado por [da Cruz, 2007] que propone un algoritmo que genera representaciones en números reales en vez de únicamente representaciones binarias como el algoritmo original.

Si bien la mayoría de algoritmos evolutivos de inspiración cuántica están relacionados estrechamente a los algoritmos genéticos, hay casos en los que se ha llevado dicha inspiración a otros algoritmos evolutivos, tales como a la evolución diferencial [Hota and Pat, 2010] y a la programación genética [Dias et al., 2011] [Dias and Pacheco, 2013].

## 2.2. Aplicaciones de los Algoritmos Evolutivos de Inspiración Cuántica

Al ser derivados de los algoritmos evolutivos, los algoritmos cuántica de inspiración cuántica han sido aplicados en diversos problemas combinatorios y de optimización numérica [Nowotniak, 2010]. Entre algunas de las áreas en las que se han aplicado satisfactoriamente este tipo de problemas, tenemos: procesamiento de imágenes [Talbi et al., 2007], problemas de diseño de redes [Xing et al., 2009], problemas de calendarización de flujo de tienda [Li and Wang, 2007], optimización de sistemas de energía [Miao et al., 2009], generación de instrumentos a escala nanométrica [Dias et al., 2011], entre otros.

Además, los algoritmos representados con números reales han sido exitosamente aplicados a problemas de optimización en ingeniería [Mani and Patvardhan, 2010], modelamiento de opciones de facturación [Fan et al., 2007], optimización de sistemas de energía [Jeong et al., 2010], análisis de datos financieros [Fan et al., 2008], entrenamiento de redes neuronales [Zhao et al., 2009] [Venayagamoorthy and Singhal, 2005], entre otras aplicaciones adicionales.

## 2.3. Consideraciones Finales

El desarrollo de algoritmos evolutivos de inspiración cuántica, al ser un área novedosa y aún poco explorada, tiene todavía mucho que ofrecer. A pesar de ello, ya se ha demostrado su competencia y adaptabilidad a problemas de diversa índole, con garantía de obtención de resultados positivos. Es por esto que se muestran en el presente -y a futuro- como un área de gran potencial y como una alternativa factible para el desarrollo de nuevas soluciones a problemas de distinta índole.

## Capítulo 3

### Marco Teórico

En este capítulo se examinará de forma más profunda las dos áreas de inspiración para la presente tesis: la **computación cuántica** y la **computación evolutiva**.

#### 3.1. Computación Cuántica

La computación cuántica es un paradigma de computación que surge en contraposición a la computación clásica, la cual basa su existencia en transistores. En la computación cuántica, se aplican a la computación los principios presentes en la física cuántica, tales como la superposición de estados, entrelazamiento, entre otras [Hey, 1999].

La potencia disponible en la computación clásica está ligada directamente al tamaño de los transistores que componen sus chips. Actualmente, el tamaño de dichos transistores ya está aproximándose a la escala de nanómetros, lo cual limita el aumento de ciclos por reloj para los procesadores, motivo por el cual el foco en la computación clásica ha cambiado para concentrarse en campos como la paralelización [Censor, 1997].

Es aquí donde la computación cuántica muestra sus posibilidades. A diferencia del paradigma tradicional donde la unidad básica es el bit, en la computación cuántica la unidad básica es el *qubit*. La diferencia entre el bit tradicional y el qubit es que mientras el primero sólo puede almacenar un único valor (ya sea este 0 o 1), el segundo puede contener 0, 1 o una superposición de ambos valores, fenómeno al que se llama *superposición de estados*. Es esta superposición de estados la que permite asumir una cantidad simultánea de valores a la vez, y por lo tanto desarrollar una cantidad fantástica de operaciones simultáneas en vez de la única que permite la computación clásica [Nielsen and Chuang, 2010].

Esta diferencia (que se evidencia en la representación de algoritmos cuánticos mediante máquinas de Turing cuánticas en vez de las tradicionales) presenta posibilidades poco imaginables hasta hace poco. Si se lograra construir un computador cuántico y ejecutar en él algoritmos que aprovechen su poder, se podría llegar al extremo de ejecutar tareas de complejidad no-polinomial para un ordenador tradicional (problemas NP-completos). Sin embargo, los avances científicos en el campo todavía no han llegado al punto de poder generar un prototipo práctico de computador cuántico, por lo que sus propiedades

continúan siendo más una posibilidad que un hecho [Han et al., 2001].

Además de ello, cabe resaltar que para aprovechar toda la potencia de procesamiento que un ordenador cuántico podría ofrecer, se deberá desarrollar algoritmos cuánticos que empleen los conceptos relacionados a la computación cuántica. Por ahora, se tienen algunos ejemplos como el algoritmo de Shor (para descomponer un número en factores polinomiales) [Shor, 1994] o el de Grover (para buscar una secuencia no ordenada de datos con  $N$  componentes) [Grover, 1996], entre otros. Basta ver, por ejemplo, el algoritmo de Shor, que demuestra que el poder de la computación cuántica es tal que podría acabar con el campo de la criptografía tradicional al ser capaz de romper algoritmos de encriptación como tales como RSA.

Dado que los algoritmos cuánticos “auténticos” son por ahora inaplicables en la vida real, se ha buscado adaptar distintos conceptos de la computación cuántica a otras áreas para poder aprovechar algunas de sus ventajas en la computación tradicional. De esos intentos surge la computación evolutiva de inspiración cuántica, la cual se explica posteriormente en este capítulo.

La unidad básica de un computador cuántico de dos estados es el llamado bit cuántico o *qubit*, el cual puede encontrarse en un estado ‘0’, un estado ‘1’ o en una superposición de ambos. Se puede representar este estado como:

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (3.1)$$

donde  $\alpha$  y  $\beta$  son valores complejos que representan las amplitudes de probabilidad de ambos estados. Por lo tanto,  $|\alpha|^2$  es la probabilidad que el valor del qubit sea de ‘0’ al observarlo, y  $|\beta|^2$  la probabilidad correspondiente para el valor ‘1’. La normalización del estado garantiza que se cumpla:

$$|\alpha|^2 + |\beta|^2 = 1 \quad (3.2)$$

De existir un sistema de  $m$ -qubits, éste puede representar tanta información como  $2^m$  estados a la vez. Sin embargo, al observar el qubit, la superposición del mismo colapsa, asumiendo así un estado único.

## 3.2. Computación Evolutiva

La computación evolutiva es una rama dependiente de la inteligencia artificial, y a su vez, una metodología de optimización inspirada en la evolución biológica y el comportamiento de organismos vivientes [Zhang et al., 2011], y comprende en sí a los algoritmos evolutivos. La computación evolutiva incluye metodologías y variantes tales como: algoritmos genéticos, estrategias evolutivas, programación genética, sistemas de clasificación mediante aprendizaje, evolución diferencial, algoritmo para la estimación de la distribución, inteligencia de enjambre, optimización por colonia de hormigas, entre otros.

Un algoritmo evolutivo es una abstracción de procesos y principios establecidos por el Darwinismo y el Neo-Darwinismo. Por lo general comprenden las siguientes ideas: dada una población de individuos, distintas circunstancias en el ambiente ejercen presión sobre los mismos provocando el fenómeno conocido como selección natural (supervivencia del más apto), causando un incremento en las aptitudes de dichos individuos. Estas circunstancias ambientales son las encargadas de realizar la medición de la aptitud de los individuos.

Dada una función que mide esta calidad y la cual se busca maximizar, se genera aleatoriamente un conjunto de individuos candidatos (que podrían ser elementos en el dominio de la función) a los cuales se les aplica la función ya mencionada para medir y comparar la aptitud de cada individuo por separado. Basándonos en esta medición de aptitud, algunos de los individuos candidatos son escogidos para propiciar la siguiente generación mediante la recombinación y/o mutación. La recombinación es un operador que se aplica a dos o más candidatos-progenitor seleccionados y que permite intermezclar características de ambos produciendo como resultado a un individuo-hijo. La mutación, por otro lado, es la aplicación de un cambio puntual sobre un candidato resultando en uno distinto. Esta generación de nuevos individuos candidatos compete en aptitud -y en algunos casos, edad- con los individuos de la generación anterior por un lugar en la siguiente. Este proceso continua iterando hasta que se evalúa a algún individuo de calidad lo suficientemente alta (hallazgo de una solución) o que se llegue a un límite en el tiempo de computación [Eiben and Smith, 2003].

Vale la pena indicar también que muchos componentes de los algoritmos evolutivos son estocásticos debido a que, a pesar de las mayores posibilidades de supervivencia o reproducción que poseen los mejores individuos, los más débiles mantienen incluso alguna chance de hacerlo también.

---

**Algorithm 1** Pseudocódigo para un algoritmo evolutivo estándar

---

INITIALIZE *population* with random candidate solutions;

EVALUATE each candidate;

**while** NOT termination-condition **do**

    SELECT parents;

    RECOMBINE pairs of parents;

    MUTATE the resulting offspring;

    EVALUATE new candidates;

    SELECT individuals for next generation;

**end while**

---

Debido a que todas las variedades de algoritmos evolutivos siguen los lineamientos formulados tanto en el algoritmo estándar como en la Figura 3.1, las diferencias entre ellas se reducen a detalles técnicos, como la forma de representar las soluciones. Lo ideal es utilizar la representación de datos más adecuada según la naturaleza del problema a resolver. Si bien son varios los paradigmas desarrollados a partir de la computación evolutiva, son tres los más importantes: Estrategias de Evolución, Programación Evolutiva y Algoritmos Genéticos, las cuales evolucionan soluciones para problemas parametrizados. A ellas se les sumaría un cuarto paradigma: Programación Genética, el cual evoluciona los programas computacionales en sí con el fin de solucionar problemas computacionales

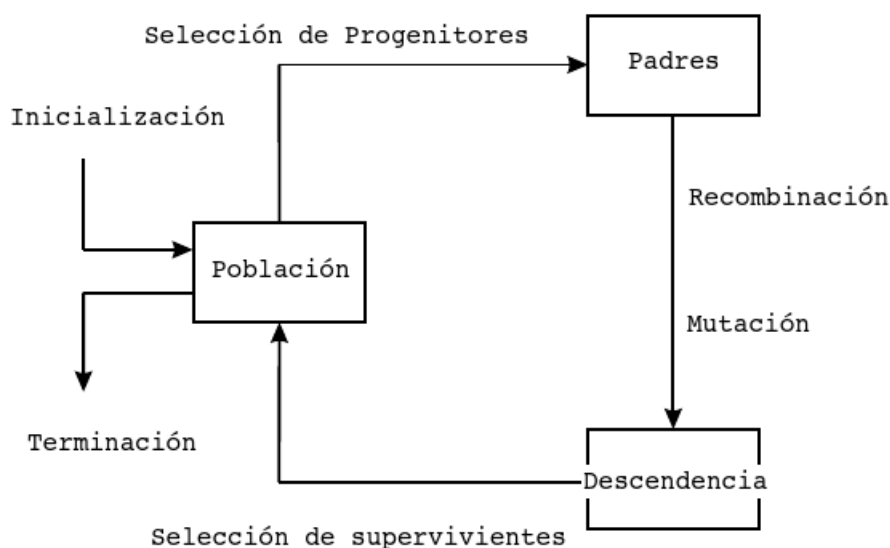


Figura 3.1: Proceso de un algoritmo evolutivo. [Eiben and Smith, 2003]

[Kicinger et al., 2005].

Es necesario mencionar algunos conceptos propios de los algoritmos evolutivos, entre los cuales se encuentra al **individuo** que es una solución propuesta al problema; la **población** que es el conjunto de individuos a evaluar y evolucionar; la **generación** que es una iteración del algoritmo en el que se evalúa la aptitud de los individuos de la población para posteriormente obtener una población nueva tras realizar cambios aplicando operadores tales como *mutación* o *recombinación*; el **fenotipo** que son los rasgos observables de cada individuo; y el **genotipo** que es la codificación genética factible de convertirse en un individuo.

Los algoritmos evolutivos emplean determinadas herramientas comunes en sus distintas variables:

- Una forma de codificar las soluciones. Esta forma varía de tal forma que podemos encontrar el uso de cadenas de alfabetos finitos como en los algoritmos genéticos, el de árboles en la programación evolutiva, o de vectores con valores reales en las estrategias evolutivas, entre otros [Kicinger et al., 2005]
- Una función de aptitud que depende tanto de los individuos como de la forma de evaluarlos.
- Un mecanismo de selección, el cual se basa en la aptitud.
- Un conjunto de operadores para reproducir y alterar a los individuos codificados.

Los algoritmos evolutivos poseen gran cantidad de aplicaciones, entre las que podemos contar problemas de optimización [Coello, 1999] [Zhou et al., 2011], exploración (arte evolutivo) [Romero and Machado, 2008], optimización de procesos químicos [Singulani et al., 2008], entre muchos otros. Presentan gran cantidad de ventajas, entre las cuales podemos mencionar:



- Aplicabilidad en problemas donde no hay otros métodos disponibles, ya sea por presencia de restricciones no lineales, discontinuidad, multi-modalidad, problemas de ruido, etc.
- Adecuados para problemas que requieren múltiples soluciones, debido a la existencia de una población de las mismas.
- Altamente paralelizables.

Por supuesto, los algoritmos evolutivos también presentan algunos inconvenientes, tales como:

- Los efectos que los errores del usuario pueden producir al momento de ajustar parámetros, los cuales pueden resultar en errores o en un desempeño menor que óptimo [Hinterding et al., 1997].
- El ajuste de los parámetros puede tomar tiempo.
- El valor óptimo de los parámetros pueden variar durante la evolución.
- No existe una garantía para hallar soluciones óptimas en un periodo de tiempo determinado, aunque para evitar ello se pueden aplicar pruebas de convergencia asintótica.

### 3.3. Algoritmo Evolutivo de Inspiración Cuántica

Los algoritmos evolutivos de inspiración cuántica buscan aprovechar los paradigmas de la física cuántica para mejorar su rendimiento al resolver problemas. Para la formulación de un algoritmo de inspiración cuántica se debe cumplir con los siguientes requerimientos:

- Tener una representación numérica o un método para convertir en representación numérica.
- Determinar una configuración inicial.
- Definir una condición de finalización.
- Dividir el problema en sub-problemas más simples.
- Identificar el número de universos (estados de superposición).
- Cada sub-problema debe asociarse a un universo.
- Los cálculos deben ser independientes en cada universo.
- Debe haber alguna interacción entre universos, y ésta debe, al menos permitir hallar la solución, o ayudar a que cada sub-problema en cada universo sea capaz de encontrarla.

El algoritmo evolutivo de inspiración cuántica original fue descrito originalmente en [Han and Kim, 2000] y posteriormente analizado [Han et al., 2001]. Como se puede ver en el capítulo anterior, han surgido buen número de variaciones y aplicaciones, todos basados en el original descrito a continuación.

QIEA emplea una representación denominada Q-bit -la cual equivale al qubit que puede representar ya sea a uno de los valores  $\{0, 1\}$  o a la superposición de ambos-, el cual está definida por dos valores  $[\alpha, \beta]$

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (3.3)$$

donde  $|\alpha|^2 + |\beta|^2 = 1$ , y que hacen referencia a la probabilidad que el Q-bit colapse -ya sea a uno o a otro- de los dos estados posibles al momento de ser observado. De la misma forma se puede definir un vector de  $n$  Q-bits de esta forma:

$$\begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \beta_1 & \beta_2 & \dots & \beta_n \end{bmatrix} \quad (3.4)$$

donde también se cumple  $|\alpha_j|^2 + |\beta_j|^2 = 1 \forall j = 0, 1, \dots, n$ .

Para modificar los Q-bits se emplea un operador denominado Q-gate, el cual se encarga de rotar el ángulo y hacerlo inclinarse hacia un valor  $\{0, 1\}$ , de tal manera que asegure el cumplimiento de la ecuación 3.2:

$$U(\Delta\theta_i) = \begin{bmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{bmatrix} \quad (3.5)$$

donde  $\Delta\theta_i, i = 0, 1, \dots, m$  es el ángulo de rotación de cada Q-bit. La magnitud de este valor tiene efecto en la convergencia, pero de ser muy alto se corre el riesgo de diverger o de converger prematuramente hacia un óptimo local.

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = U(\Delta\theta_i) \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \quad (3.6)$$

La estructura del algoritmo QIEA es la siguiente:

### 3.4. Algoritmo Evolutivo de Inspiración Cuántica - $\mathbb{R}$

La propuesta original de un algoritmo evolutivo de inspiración cuántica para  $\mathbb{R}$  busca generar un conjunto de estados observables continuos y no discretos como el algoritmo base [da Cruz, 2007]. La inspiración de este modelo está en el uso de funciones de onda.

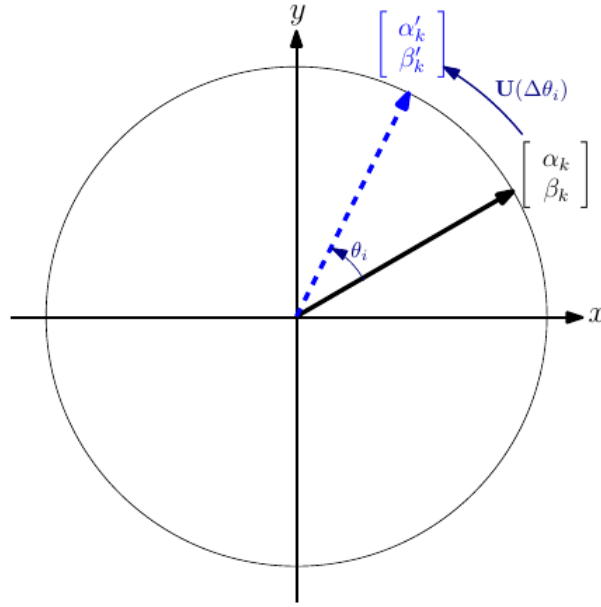


Figura 3.2: Rotación del ángulo al momento de la aplicación del operador Q-gate

---

**Algorithm 2** Pseudocódigo para el algoritmo evolutivo de inspiración cuántica

---

```

INITIALIZE  $t \leftarrow 0$ ;
INITIALIZE  $Q(t)$ ;
MAKE  $P(t)$  by observing the states of  $Q(t)$ ;
EVALUATE  $P(t)$ ;
STORE best solutions among  $P(t)$  into  $\mathbf{b}$ ;
while NOT termination-condition do
     $t \leftarrow t + 1$ ;
    MAKE  $P(t)$  by observing states of  $Q(t - 1)$ ;
    EVALUATE  $P(t)$  ;
    UPDATE  $Q(t)$  using Q-gates;
    STORE best solutions among  $P(t)$  into  $\mathbf{b}$ ;
end while

```

---

Se calcula el momento de un fotón a partir de su masa al moverse a la velocidad de la luz  $m = hv/c^2$  derivada de la ecuación fundamental de Einstein  $E = mc^2$ .

$$p = mc = hv/c = h/\lambda \quad (3.7)$$

donde  $\lambda$  es la longitud de onda de la luz.

Al reemplazar  $\lambda = h/p$  en la ecuación utilizada en electromagnetismo donde una onda estacionaria con longitud de onda  $\lambda$  se propaga hacia el lado positivo del eje  $x$  definida como  $\psi(x) = e^{i2\pi x/\lambda}$  donde  $i = \sqrt{-1}$  se obtiene:

$$\psi(x) = e^{ipx2\pi/h} \quad (3.8)$$

Según [Gillespie, 1974], la Densidad Determinada Probabilística (PDD) para la ubicación de una partícula con función de onda  $\psi$  está definida por  $|\psi|^2$ , por lo que al integrar todo el espacio en el que la partícula podría ser encontrada, se obtiene la posibilidad de encontrarla en cualquier lugar del espacio:

$$\int_{-\infty}^{\infty} |\psi|^2 d\tau = 1 \quad (3.9)$$

Debido a que a cada observación de una partícula, ésta asumirá diferentes valores de posición según la probabilidad de estar localizada en determinada región del espacio, se usa este concepto en el modelo AEIQ- $\mathbb{R}$  para representar los valores de los individuos cuánticos.

De tal forma, se obtiene una población de individuos cuánticos  $Q_t = q_1, \dots, q_m$ , en la generación  $t$ , donde cada individuo cuántico  $q_i$  está formado por  $n$  genes,  $q_{ij} = q_{i1}, \dots, q_{in}$ , donde a su vez cada gen  $q_{ij}$  está formado por funciones densidades de probabilidad (en vez de variables como en el algoritmo original).

### 3.5. Consideraciones Finales

Según se ha visto, el algoritmo evolutivo de inspiración cuántica original es sencillo y de implementación relativamente rápida, mientras que el algoritmo para  $\mathbb{R}$  sube ligeramente el listón en términos de inspiración pero añadiendo nuevas capacidades a las posibilidades de la inspiración cuántica. Se tomará este algoritmo como punto de partida para la propuesta que se desarrolla en el siguiente capítulo, el cual busca introducir un nuevo mecanismo de recombinación (*crossover*) en un entorno concurrente de ejecución con más de una población cuántica generadora. Según se ha visto, el algoritmo evolutivo de inspiración cuántica original es sencillo y de implementación relativamente rápida, mientras que el algoritmo para  $\mathbb{R}$  sube ligeramente el listón en términos de inspiración pero añadiendo nuevas capacidades a las posibilidades de la inspiración cuántica. Se tomará este algoritmo como punto de partida para la propuesta que se desarrolla en el siguiente capítulo, el cual busca introducir un nuevo mecanismo de recombinación (*crossover*) en un entorno concurrente de ejecución con más de una población cuántica generadora.

## Capítulo 4

# Propuesta

La propuesta que presenta esta tesis busca extender el algoritmo evolutivo de inspiración cuántica original, e introducir el concepto de *entrelazamiento intergeneracional* como justificación para la implementación de un operador de recombinación. Se busca introducir una versión concurrente del algoritmo base, el cual a partir de una determinada iteración  $u$  combinará un Q-bit que sea parte del individuo generador de mejores soluciones con otro Q-bit de otro individuo caracterizado por entregar soluciones peores que las del resto. De esta manera, se intenta que la búsqueda de valores óptimos no se vea desviada por la tendencia a un máximo local en vez de al máximo global que se busca obtener.

# Bibliografía

- [Alegria and Túpac, 2014] Alegria, J. M. and Túpac, Y. J. (2014). A generalized quantum-inspired evolutionary algorithm for combinatorial optimization problems.
- [Benioff, 1980] Benioff, P. (1980). The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of Statistical Physics*, 22(5):563–591.
- [Benioff, 1982] Benioff, P. (1982). Quantum mechanical models of turing machines that dissipate no energy. *Physical Review Letters*, 48(23):1581.
- [Censor, 1997] Censor, Y. (1997). *Parallel optimization: Theory, algorithms, and applications*. Oxford University Press.
- [Coello, 1999] Coello, C. A. C. (1999). A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information systems*, 1(3):269–308.
- [da Cruz, 2007] da Cruz, A. (2007). *Algoritmos evolutivos com inspiração quântica para problemas com representação numérica*. PhD thesis, PhD thesis, Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brasil.(In Portuguese).
- [Dias and Pacheco, 2013] Dias, D. M. and Pacheco, M. A. C. (2013). Quantum-inspired linear genetic programming as a knowledge management system. *The Computer Journal*, 56(9):1043–1062.
- [Dias et al., 2011] Dias, D. M., Singulani, A. P., Pacheco, M. A. C., de Souza, P. L., Pires, M. P., and Neto, O. P. V. (2011). Self-assembly quantum dots growth prediction by quantum-inspired linear genetic programming. In *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pages 2075–2082. IEEE.
- [Eiben and Smith, 2003] Eiben, A. E. and Smith, J. E. (2003). *Introduction to evolutionary computing*. springer.
- [Fan et al., 2007] Fan, K., Brabazon, A., O’Sullivan, C., and O’Neill, M. (2007). Option pricing model calibration using a real-valued quantum-inspired evolutionary algorithm. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 1983–1990. ACM.
- [Fan et al., 2008] Fan, K., Brabazon, A., O’Sullivan, C., and O’Neill, M. (2008). Quantum-inspired evolutionary algorithms for financial data analysis. In *Applications of Evolutionary Computing*, pages 133–143. Springer.

- 
- [Gillespie, 1974] Gillespie, D. T. (1974). *A quantum mechanics primer: An Elementary Introduction to the Formal Theory of Non-relativistic Quantum Mechanics*. John Wiley and Sons.
- [Grover, 1996] Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219. ACM.
- [Han and Kim, 2000] Han, K.-H. and Kim, J.-H. (2000). Genetic quantum algorithm and its application to combinatorial optimization problem. In *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, volume 2, pages 1354–1360. IEEE.
- [Han et al., 2001] Han, K.-H., Kim, J.-H., et al. (2001). Analysis of quantum-inspired evolutionary algorithm. In *Proceedings of the 2001 International Conference on Artificial Intelligence*, pages 727–730. Citeseer.
- [Hey, 1999] Hey, T. (1999). Quantum computing: an introduction. *Computing & Control Engineering Journal*, 10(3):105–112.
- [Hinterding et al., 1997] Hinterding, R., Michalewicz, Z., and Eiben, A. E. (1997). Adaptation in evolutionary computation: A survey. In *Evolutionary Computation, 1997., IEEE International Conference on*, pages 65–69. IEEE.
- [Hota and Pat, 2010] Hota, A. R. and Pat, A. (2010). An adaptive quantum-inspired differential evolution algorithm for 0–1 knapsack problem. In *Nature and Biologically Inspired Computing (NaBIC), 2010 Second World Congress on*, pages 703–708. IEEE.
- [Jeong et al., 2010] Jeong, Y.-W., Park, J.-B., Jang, S.-H., and Lee, K. Y. (2010). A new quantum-inspired binary pso: application to unit commitment problems for power systems. *Power Systems, IEEE Transactions on*, 25(3):1486–1495.
- [Kicinger et al., 2005] Kicinger, R., Arciszewski, T., and Jong, K. D. (2005). Evolutionary computation and structural design: A survey of the state-of-the-art. *Computers & Structures*, 83(23):1943–1978.
- [Li and Wang, 2007] Li, B.-B. and Wang, L. (2007). A hybrid quantum-inspired genetic algorithm for multiobjective flow shop scheduling. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 37(3):576–591.
- [Malossini et al., 2008] Malossini, A., Blanzieri, E., and Calarco, T. (2008). Quantum genetic optimization. *Trans. Evol. Comp*, 12(2):231–241.
- [Mani and Patvardhan, 2010] Mani, A. and Patvardhan, C. (2010). Solving ceramic grinding optimization problem by adaptive quantum evolutionary algorithm. In *Intelligent Systems, Modelling and Simulation (ISMS), 2010 International Conference on*, pages 43–48. IEEE.
- [Miao et al., 2009] Miao, H., Wang, H., and Deng, Z. (2009). Quantum genetic algorithm and its application in power system reactive power optimization. In *Computational Intelligence and Security, 2009. CIS'09. International Conference on*, volume 1, pages 107–111. IEEE.
-

- 
- [Narayanan and Moore, 1996] Narayanan, A. and Moore, M. (1996). Quantum-inspired genetic algorithms. In *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*, pages 61–66. IEEE.
- [Nielsen and Chuang, 2010] Nielsen, M. A. and Chuang, I. L. (2010). *Quantum computation and quantum information*. Cambridge university press.
- [Nowotniak, 2010] Nowotniak, R. (2010). Survey of quantum-inspired evolutionary algorithms.
- [Nowotniak and Kucharski, 2014] Nowotniak, R. and Kucharski, J. (2014). Higher-order quantum-inspired genetic algorithms. *arXiv preprint arXiv:1407.0977*.
- [Qin et al., 2008] Qin, C., Liu, Y., and Zheng, J. (2008). A real-coded quantum-inspired evolutionary algorithm for global numerical optimization. In *Cybernetics and Intelligent Systems, 2008 IEEE Conference on*, pages 1160–1164. IEEE.
- [Romero and Machado, 2008] Romero, J. J. and Machado, P. (2008). *The art of artificial evolution: a handbook on evolutionary art and music*. Springer.
- [Shor, 1994] Shor, P. W. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*, pages 124–134. IEEE.
- [Singulani et al., 2008] Singulani, A. P., Vilela Neto, O. P., Aurélio Pacheco, M. C., Vellasco, M. B., Pires, M. P., and Souza, P. L. (2008). Computational intelligence applied to the growth of quantum dots. *Journal of Crystal Growth*, 310(23):5063–5065.
- [Talbi et al., 2007] Talbi, H., Batouche, M., and Draa, A. (2007). A quantum-inspired evolutionary algorithm for multiobjective image segmentation. *International Journal of Mathematical, Physical and Engineering Sciences*, 1(2):109–114.
- [Teng et al., 2010] Teng, H., Zhao, B., and Cao, A. (2010). Chaos quantum genetic algorithm based on henon map. *Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on*, 1:922–925.
- [Tsai et al., 2012] Tsai, J.-T., Chou, J.-H., and Ho, W.-H. (2012). Improved quantum-inspired evolutionary algorithm for engineering design optimization. *Mathematical Problems in Engineering*, 2012.
- [Venayagamoorthy and Singhal, 2005] Venayagamoorthy, G. K. and Singhal, G. (2005). Quantum-inspired evolutionary algorithms and binary particle swarm optimization for training mlp and srn neural networks. *Journal of Computational and Theoretical Nanoscience*, 2(4):561–568.
- [Xing et al., 2009] Xing, H., Liu, X., Jin, X., Bai, L., and Ji, Y. (2009). A multi-granularity evolution based quantum genetic algorithm for qos multicast routing problem in wdm networks. *Computer Communications*, 32(2):386–393.
- [Zhang et al., 2006] Zhang, G.-x., Li, N., Jin, W.-d., and Hu, L.-z. (2006). Novel quantum genetic algorithm and its applications. *Frontiers of Electrical and Electronic Engineering in China*, 1(1):31–36.



- [Zhang et al., 2011] Zhang, J., Zhan, Z.-h., Lin, Y., Chen, N., Gong, Y.-j., Zhong, J.-h., Chung, H. S., Li, Y., and Shi, Y.-h. (2011). Evolutionary computation meets machine learning: A survey. *Computational Intelligence Magazine, IEEE*, 6(4):68–75.
- [Zhao et al., 2009] Zhao, S., Xu, G., Tao, T., and Liang, L. (2009). Real-coded chaotic quantum-inspired genetic algorithm for training of fuzzy neural networks. *Computers & Mathematics with Applications*, 57(11).
- [Zhou et al., 2011] Zhou, A., Qu, B.-Y., Li, H., Zhao, S.-Z., Suganthan, P. N., and Zhang, Q. (2011). Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation*, 1(1):32–49.