Aplicaciones del Algoritmo de Grover

Quantum Counting, Problemas NP-Completos, Optimización

Federico Fuidio

Facultad de Ingeniería
Universidad de Montevideo
Montevideo

Qiskit Fall Fest FIUBA - 10/11/2025

Índice

Quantum Counting

Introducción

- Tenemos una función $f : \{0,1\}^n \to \{0,1\}$ con M entradas tales que $f(x) = 1$.
- **No conocemos el número M.**
- Tenemos un oráculo O_f que implementa la operación $O_f |x\rangle = (-1)^{f(x)} |x\rangle$. Por tanto, podemos implementar el operador de Grover $G = AO_f$.
- **Objetivo:** Estimar el valor de M.

Quantum Counting
Quantum Phase Estimation - QPE

Qué hace QPE?

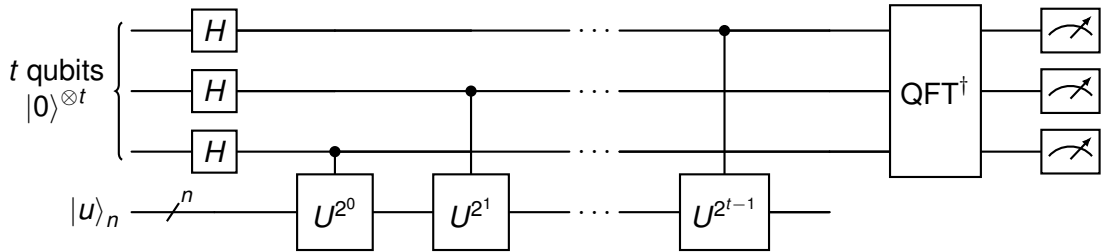
- QPE permite estimar los valores propios de un operador unitario U.
- El algoritmo de QPE comienza con una combinación de vectores propios $|u_j\rangle$ con valores propios asociados $\lambda_j = \exp(i2\pi\phi_j)$:

$$|u\rangle = \sum_j \alpha_j |u_j\rangle .$$

- Usando $t = m + \lceil \log_2 \left(2 + \frac{1}{2\varepsilon}\right) \rceil$ qubits, nos aseguramos de obtener una estimación de ϕ_j con precisión de m bits y con una probabilidad de al menos $|\alpha_j|^2(1-\varepsilon)$.

Quantum Counting

Quantum Phase Estimation - QPE



Al medir los t qubits del primer registro, la estimación de la fase es:

$$\hat{\phi} = 0.b_1 b_2 \dots b_t = \sum_{i=1}^{t} b_i 2^{-i}$$

Quantum Counting

QPE aplicado al operador de Grover

Veamos qué pasa si aplicamos QPE con el operador de Grover $G = (AO_f)$ y estado inicial $|u\rangle_n = |+\rangle^{\otimes n}$:

$$|+\rangle^{\otimes n} = \frac{\sqrt{N-M}}{\sqrt{N}} |\alpha\rangle + \frac{\sqrt{M}}{\sqrt{N}} |\beta\rangle .$$

Vimos que, en el espacio generado por $|\alpha\rangle$ y $|\beta\rangle$, los valores propios de G son:

$$\lambda = e^{\pm i\theta}.$$

Con $\theta = 2 \arcsin\left(\sqrt{M}/\sqrt{N}\right) \Rightarrow$ QPE nos dará una estimación de θ, que a su vez nos permite estimar M.

Quantum Counting

QPE aplicado al operador de Grover

Así, mediremos:

- $\hat{\phi}_{\text{QPE}} \approx \phi_{\text{QPE}} = \theta/(2\pi)$, correspondiente al valor propio $e^{+i\theta}$. Podemos estimar M como:

$$M = \sin^2(\theta/2)N \approx \sin^2\left(\pi\hat{\phi}_{\text{QPE}}\right)N = \hat{M}.$$

- $\hat{\phi}_{\text{QPE}} \approx \phi_{\text{QPE}} = (2\pi - \theta)/(2\pi) = 1 - \theta/(2\pi)$, correspondiente al valor propio $e^{-i\theta}$. Podemos estimar M como:

$$M = \sin^2(\theta/2)N \approx \sin^2\left(\pi - \pi\hat{\phi}_{\text{QPE}}\right)N = \hat{M}.$$

Quantum Counting

QPE aplicado al operador de Grover

Nosotros implementamos $-G$ en lugar de G. Por tanto, las fases de los valores propios tienen un *shift* de π: $\lambda = e^{i(\pm\theta+\pi)}$.

- $\hat{\phi}_{\text{QPE}} \approx \phi_{\text{QPE}} = (\pi+\theta)/(2\pi) = 1/2 + \theta/(2\pi)$, correspondiente al valor propio $e^{+i(\theta+\pi)}$. Podemos estimar M como:

$$M = \sin^2(\theta/2)N \approx \sin^2\left(\pi\hat{\phi}_{\text{QPE}} - \pi/2\right)N = \hat{M}.$$

- $\hat{\phi}_{\text{QPE}} \approx \phi_{\text{QPE}} = (\pi-\theta)/(2\pi) = 1/2 - \theta/(2\pi)$, correspondiente al valor propio $e^{-i\theta}$. Podemos estimar M como:

$$M = \sin^2(\theta/2)N \approx \sin^2\left(\pi/2 - \pi\hat{\phi}_{\text{QPE}}\right)N = \hat{M}.$$

Quantum Counting

Análisis de error

Cuál es el error cometido en la estimación de $\hat{M}$ que podemos asegurar con probabilidad $1 - \varepsilon$?

$$\left| \hat{\phi}_{QPE} - \phi_{QPE} \right| < 2^{-m}$$

Quantum Counting

Análisis de error

Cuál es el error cometido en la estimación de $\hat{M}$ que podemos asegurar con probabilidad $1 - \varepsilon$?

$$\left|\hat{\phi}_{QPE} - \phi_{QPE}\right| < 2^{-m}$$

$$\Rightarrow \left|\pi\hat{\phi}_{QPE} - \pi\phi_{QPE}\right| < \pi 2^{-m}$$

Quantum Counting

Análisis de error

Cuál es el error cometido en la estimación de $\hat{M}$ que podemos asegurar con probabilidad $1 - \varepsilon$?

$$\left|\hat{\phi}_{QPE} - \phi_{QPE}\right| < 2^{-m}$$

$$\Rightarrow \left|\pi\hat{\phi}_{QPE} - \pi\phi_{QPE}\right| < \pi 2^{-m}$$

$$\Rightarrow \left|\left(\pi\hat{\phi}_{QPE} - \pi/2\right) - \left(\pi\phi_{QPE} - 2\pi\right)\right| < \pi 2^{-m}$$

Quantum Counting

Análisis de error

Cuál es el error cometido en la estimación de $\hat{M}$ que podemos asegurar con probabilidad $1 - \varepsilon$?

$$\left|\hat{\phi}_{QPE} - \phi_{QPE}\right| < 2^{-m}$$

$$\Rightarrow \left|\pi\hat{\phi}_{QPE} - \pi\phi_{QPE}\right| < \pi 2^{-m}$$

$$\Rightarrow \left|\left(\pi\hat{\phi}_{QPE} - \pi/2\right) - \left(\pi\phi_{QPE} - 2\pi\right)\right| < \pi 2^{-m}$$

$$\Rightarrow \left|\hat{\theta}/2 - \theta/2\right| < \pi 2^{-m}$$

Quantum Counting

Análisis de error

Cuál es el error cometido en la estimación de $\hat{M}$ que podemos asegurar con probabilidad $1 - \varepsilon$?

$$\left|\hat{\phi}_{QPE} - \phi_{QPE}\right| < 2^{-m}$$

$$\Rightarrow \left|\pi\hat{\phi}_{QPE} - \pi\phi_{QPE}\right| < \pi 2^{-m}$$

$$\Rightarrow \left|\left(\pi\hat{\phi}_{QPE} - \pi/2\right) - \left(\pi\phi_{QPE} - 2\pi\right)\right| < \pi 2^{-m}$$

$$\Rightarrow \left|\hat{\theta}/2 - \theta/2\right| < \pi 2^{-m}$$

$$\Rightarrow \left|\hat{M} - M\right| = N\left|\sin^2\left(\hat{\theta}/2\right) - \sin^2\left(\theta/2\right)\right| \leq N\left|\hat{\theta}/2 - \theta/2\right| < \pi N 2^{-t} = \pi 2^{-m+n}$$

Quantum Counting
Comentarios

Comentarios finales

- $|\hat{M} - M| \leq \pi 2^{-m+n}$ con una probabilidad de al menos $1 - \varepsilon$.
- Como $\pi 2^{-3} \approx 0.3927$, podemos asegurar (con probabilidad $1 - \varepsilon$) medir correctamente M luego de redondear $\hat{M}$ al entero más cercano, tomando $m = n + 3$.
- Se puede demostrar que se necesitan $\mathcal{O}(\sqrt{N})$ aplicaciones del operador de Grover controlado $C(-G)$, tanto para asegurar que $M \neq 0$ con alta probabilidad como para obtener una buena estimación de R.

3-SAT

Problema de satisfabilidad booleana (SAT)

- El problema de **satisfacibilidad booleana (SAT)** consiste en determinar si existe una asignación de valores `TRUE/FALSE` a las variables de una fórmula booleana que haga que toda la expresión sea `TRUE`.
- Si existe, se dice que la fórmula es **satisfacible**.

Ejemplos:

- $(x_1 \vee \neg x_2) \wedge (x_2 \vee x_4 \vee \neg x_5)$
- $(x_1 \wedge \neg x_2) \wedge (\neg x_1 \vee x_2)$

3-SAT

Conceptos importantes

- Un **literal** es una variable o su negación.
- Una **cláusula** es una disyunción (OR) de literales.
- Una fórmula está en **forma normal conjuntiva (CNF)** si es una conjunción (AND) de cláusulas.

$$\overbrace{(x_1 \vee \neg x_2)}^{\text{Cláusula 1}} \wedge \overbrace{(x_2 \vee x_4 \vee \neg x_5)}^{\text{Cláusula 2}} \Rightarrow \text{Está en CNF}$$

$$(x_1 \vee \neg x_2) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge \underbrace{(x_1 \wedge \neg x_4)}_{\text{No es una cláusula}} \Rightarrow \text{No está en CNF}$$

3-SAT

Descripción del problema

El **3-SAT** consiste en determinar si una fórmula en forma normal conjuntiva (CNF) puede satisfacerse, donde cada cláusula contiene como máximo tres literales.

Ejemplo satisfacible

$$F(x_1, x_2, x_3) = (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_2)$$

$$F(\texttt{TRUE}, \texttt{FALSE}, \texttt{FALSE}) = \texttt{TRUE}$$

Ejemplo insatisfacible

$$F(x_1, x_2) = (x_1 \vee x_2) \wedge (\neg x_1 \vee x_2) \wedge (x_1 \vee \neg x_2) \wedge (\neg x_1 \vee \neg x_2)$$

3-SAT

Solución con Grover

Para poder resolver este problema con Grover, dada una fórmula $F(x_1, x_2, \ldots, x_n) = C_1 \wedge C_2 \wedge \cdots \wedge C_m$, debemos implementar un operador O de forma que:

$$O\,|x_1 x_2 \ldots x_n\rangle\,|y\rangle = |x_1 x_2 \ldots x_n\rangle\,|y \oplus F(x)\rangle$$

Agregamos m qubits auxiliares para guardar los valores de $C_1, C_2, \ldots, C_m$. Así, buscamos implementar las operaciones:

$$|x_1 x_2 \ldots x_n\rangle\,|0\rangle^{\otimes m}\,|y\rangle \to |x_1 x_2 \ldots x_n\rangle\,|C_1 C_2 \ldots C_m\rangle\,|y\rangle$$

$$|x_1 x_2 \ldots x_n\rangle\,|C_1 C_2 \ldots C_m\rangle\,|y\rangle \to |x_1 x_2 \ldots x_n\rangle\,|C_1 C_2 \ldots C_m\rangle\,|y \oplus C_1 \wedge C_2 \wedge \cdots \wedge C_m\rangle$$

$$|x_1 x_2 \ldots x_n\rangle\,|C_1 C_2 \ldots C_m\rangle\,|y \oplus C_1 \wedge C_2 \wedge \cdots \wedge C_m\rangle \to |x_1 x_2 \ldots x_n\rangle\,|0\rangle^{\otimes m}\,|y \oplus F(x)\rangle$$

3-SAT

Solución con Grover

Veamos cómo implementar $|x_1 x_2 \dots x_n\rangle |0\rangle^{\otimes m} |y\rangle \to |x_1 x_2 \dots x_n\rangle |C_1 C_2 \dots C_m\rangle |y\rangle$. Para lograr implementar esta operación debemos implementar un OR, de forma que $|a\rangle |b\rangle |c\rangle |0\rangle \to |a\rangle |b\rangle |c\rangle |a \vee b \vee c\rangle$. Notamos que con una compuerta X multicontrolada podemos implementar un AND:

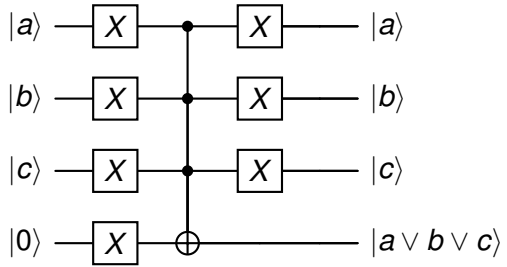
$$C^3(X) |abc\rangle |0\rangle = |abc\rangle |a \wedge b \wedge c\rangle$$

Por tanto, podemos implementar el OR como:

$$\begin{aligned} C^3(X) |a \oplus 1\rangle |b \oplus 1\rangle |c \oplus 1\rangle |1\rangle &= |a \oplus 1\rangle |b \oplus 1\rangle |c \oplus 1\rangle \overbrace{|1 \oplus (\neg a \wedge \neg b \wedge \neg c)\rangle}^{=\neg(\neg a \wedge \neg b \wedge \neg c) = a \vee b \vee c} \quad (1) \\ &= |a \oplus 1\rangle |b \oplus 1\rangle |c \oplus 1\rangle |a \vee b \vee c\rangle \quad (2) \end{aligned}$$

3-SAT

Solución con Grover



- El estado inicial es $|a\rangle\,|b\rangle\,|c\rangle\,|0\rangle$
- Luego de aplicar X en todos los qubits obtenemos: $|a \oplus 1\rangle\,|b \oplus 1\rangle\,|c \oplus 1\rangle\,|1\rangle$
- Después de la X multicontrolada obtenemos: $|a \oplus 1\rangle\,|b \oplus 1\rangle\,|c \oplus 1\rangle\,|a \vee b \vee c\rangle$
- Finalmente, se aplica una X los tres primeros qubits: $|a\rangle\,|b\rangle\,|c\rangle\,|a \vee b \vee c\rangle$

3-SAT

Solución con Grover

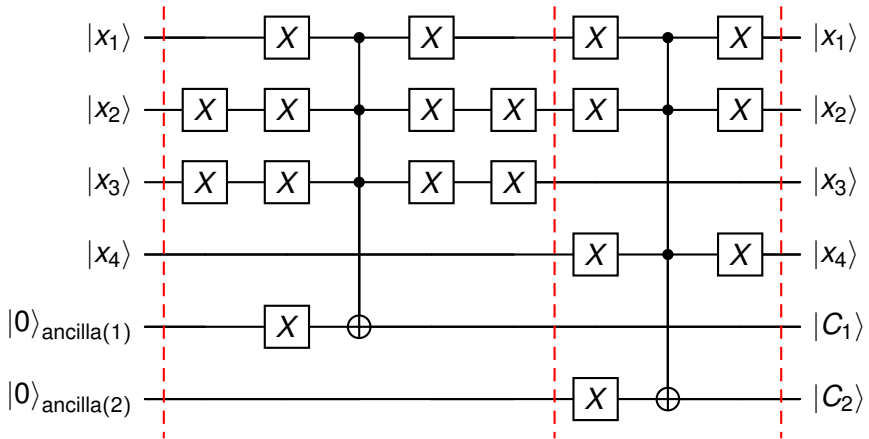
Para implementar $|x_1 x_2 \dots x_n\rangle |0\rangle^{\otimes m} |y\rangle \to |x_1 x_2 \dots x_n\rangle |C_1 C_2 \dots C_m\rangle |y\rangle$:

- Implementamos OR $\left|x_j\right\rangle |x_k\rangle |x_l\rangle |0\rangle_{\text{ancilla}(i)}$ siempre que $C_i = (x_j \vee x_k \vee x_l)$.
- Si una cláusula tiene algún literal negado, basta aplicar una X antes y después de la operación OR.

3-SAT

Solución con Grover

Ejemplo: $C_1 \wedge C_2 = (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_1 \vee x_2 \vee x_4)$



3-SAT

Solución con Grover

Ahora debemos implementar la segunda operación:

$$|x_1 x_2 \dots x_n\rangle \, |C_1 C_2 \dots C_m\rangle \, |y\rangle \to |x_1 x_2 \dots x_n\rangle \, |C_1 C_2 \dots C_m\rangle \, |y \oplus C_1 \wedge C_2 \wedge \dots \wedge C_m\rangle$$

Esto se puede implementar con una compuerta X en el último qubit $|y\rangle$ controlada por los m qubits auxiliares:

$$C^m(X) \, |C_1 C_2 \dots C_m\rangle \, |y\rangle = |C_1 C_2 \dots C_m\rangle \, |y \oplus C_1 \wedge C_2 \wedge \dots \wedge C_m\rangle \, .$$

3-SAT

Solución con Grover

Finalmente debemos implementar:

$$|x_1 x_2 \dots x_n\rangle \, |C_1 C_2 \dots C_m\rangle \overbrace{|y \oplus C_1 \wedge C_2 \wedge \dots \wedge C_m\rangle}^{|y \oplus F(x)\rangle} \to |x_1 x_2 \dots x_n\rangle \, |0\rangle^{\otimes m} \, |y \oplus F(x)\rangle$$

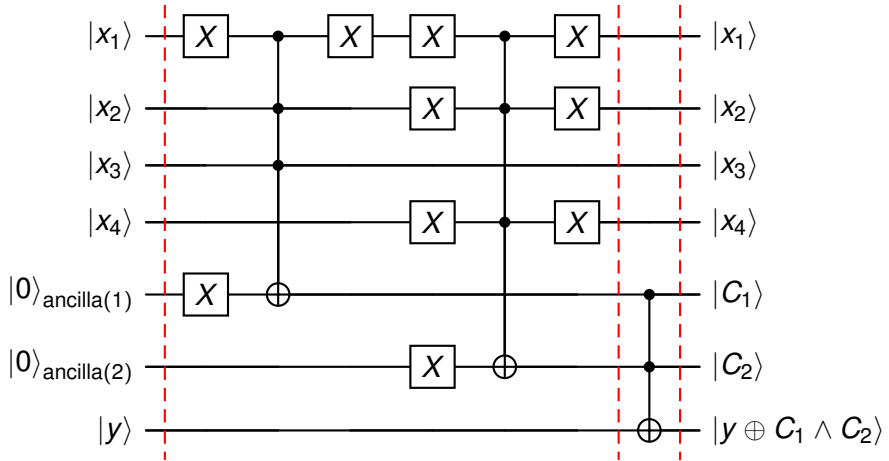
Para esto nos basta con implementar la operación inversa al primer operador en los dos primeros registros:

$$|x_1 x_2 \dots x_n\rangle \, |C_1 C_2 \dots C_m\rangle \to |x_1 x_2 \dots x_n\rangle \, |0\rangle^{\otimes m} .$$

Como todos los operadores usados en la primera operación, X y $C^3(X)$ son su propio inverso, sólo hace falta repetir las mismas compuertas en el orden inverso.

3-SAT

Solución con Grover



3-SAT
Solución con Grover

3-SAT con Grover

- El operador de Grover G se implementa en $\mathcal{O}(m)$.
- Encontramos una solución con probabilidad alta en $\mathcal{O}(m2^{n/2}) \approx \mathcal{O}(m(1.41)^n)$.
- Clásicamente, con fuerza bruta, se necesita $\mathcal{O}(m2^n)$.
- Existen algoritmos clásicos que logran $\mathcal{O}(mc^n)$ para $c < \sqrt{2}$.

Grover Adaptive Search - GAS

Introducción

Veamos cómo usar Grover para resolver problemas de optimización.

- Formulaciones QUBO (Quadratic Unconstrained Binary Optimization): $\min_{x \in \{0,1\}^n} \{f(x)\}$ donde:

$$f(x) = \sum_{i,j} Q_{i,j} x_i x_j$$

- Formulaciones CPBO (Constrained Polynomial Binary Optimization): más general que QUBO. $f(x)$ puede ser un polinomio de cualquier orden.
- Vamos a centrarnos en el caso donde todos los coeficientes del polinomio son enteros. Para el caso de QUBO, implica que $Q_{i,j} \in \mathbb{Z}$.

Grover Adaptive Search - GAS

Introducción

La idea es generar un operador de Grover que amplifique los elementos que cumplen $f(x) < y$ para cualquier $y \in \mathbb{Z}$. Esto nos permitirá iterativamente mejorar el valor del mínimo.
Necesitamos dos operaciones:

$$U_1 \left|x_1 x_2 \dots x_n\right\rangle \left|0\right\rangle^{\otimes m} \left|z\right\rangle = \left|x_1 x_2 \dots x_n\right\rangle \left|f(x) - y\right\rangle_m \left|z\right\rangle$$

$$U_2 \left|x_1 x_2 \dots x_n\right\rangle \left|f(x) - y\right\rangle_m \left|z\right\rangle = \left|x_1 x_2 \dots x_n\right\rangle \left|f(x) - y\right\rangle_m \left|z \oplus (f(x) - y < 0)\right\rangle$$

Grover Adaptive Search - GAS

Implementación de U_1

Definimos el operador $U_G(\theta)$, que actúa en m qubits como:

$$U_G(\theta)\,|y\rangle_m = e^{iy\theta}\,|y\rangle_m$$

Notemos que $U_G(\theta)U_G(\phi) = U_G(\theta+\phi)$, ya que:

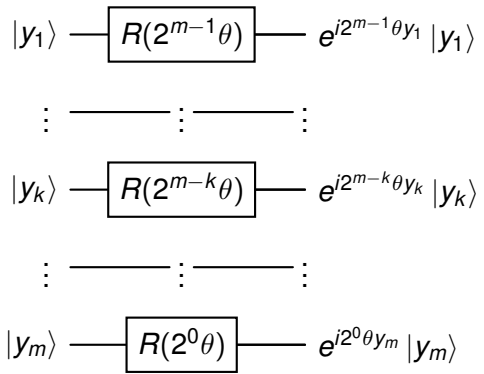
$$U_G(\theta)U_G(\phi)\,|y\rangle = U_G(\theta)e^{iy\phi}\,|y\rangle = e^{iy\theta}e^{iy\phi}\,|y\rangle = e^{iy(\theta+\phi)}\,|y\rangle$$

Podemos implementar esta operación con una compuerta $R(2^{m-k}\theta)$ en el qubit k, siendo:

$$R(\theta) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}$$

Grover Adaptive Search - GAS

Implementación de U_1



$$U_G(\theta)\,|y\rangle = U_G(\theta)\,|y_1 y_2 \dots y_m\rangle = e^{i\left(\sum_{k=1}^{m} 2^{m-k} y_k\right)\theta}\,|y_1 y_2 \dots y_m\rangle = e^{iy\theta}\,|y\rangle$$

Grover Adaptive Search - GAS

Implementación de U_1

Notemos que:

$$U_G\left(\frac{2\pi}{2^m}k\right) H^{\otimes n} |0\rangle^{\otimes m} = \sum_{j=0}^{2^m-1} e^{2\pi ijk/2^m} |j\rangle_m$$

Por otro lado:

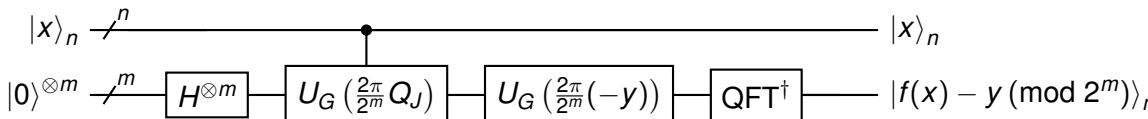
$$\mathrm{QFT}\,|k \pmod{2^m}\rangle = \sum_{j=0}^{2^m-1} e^{2\pi ijk/2^m} |j\rangle_m$$

$$\Rightarrow H^{\otimes m} |0\rangle^{\otimes m} \xrightarrow{m} \boxed{U_G\left(\tfrac{2\pi}{2^m}k\right)} \boxed{\mathrm{QFT}^\dagger} |k \pmod{2^m}\rangle$$

Grover Adaptive Search - GAS

Implementación de U_1

Podemos usar la representación en complemento a 2 en k, para $2^{-m+1} \leq k < 2^{m-1}$.
Elegimos m de forma de poder representar los valores de $f(x)$ en ese rango.
U_1 se puede implementar como:

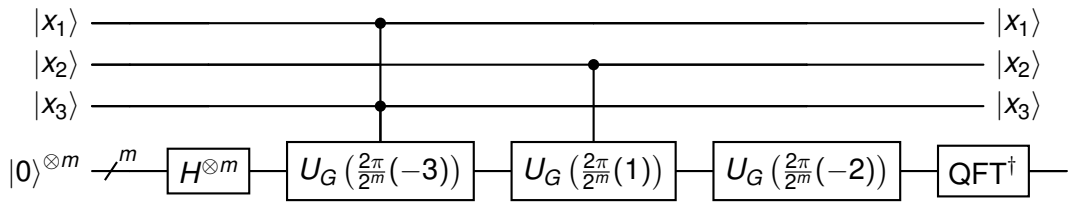


Donde $Q_J \in \{Q_{i,j}\}$. Los controles en el primer registro son los qubits i, j si $Q_J = Q_{i,j}$.

Grover Adaptive Search - GAS

Implementación de U_1

Para $f(x) = -3x_1x_3 + x_2$:



El estado al final del circuito será:

$$|x_1x_2x_3\rangle\,|-3x_1x_3 + x_2 - 2 \,(\text{mod}\, 2^m)\rangle$$

Grover Adaptive Search - GAS

Implementación de U_2

Como en el registro de qubits auxiliares estamos usando complemento a dos, el estado del primer qubit nos indica el signo de $f(x) - y$:

$$\text{CNOT} \left|y_1\right\rangle \left|z\right\rangle = \left|y_1\right\rangle \left|z \oplus (f(x) - y < 0)\right\rangle$$

Por lo que U_2 consiste simplemente en una $C(X)$ en el qubit $|z\rangle$, controlada por $|y_1\rangle$. Finalmente, para implementar el oráculo O, aplicamos el inverso de U_1 para volver a dejar las ancillas en $|0\rangle^{\otimes m}$. Así,

$$O = U_1^\dagger U_2 U_1$$

Grover Adaptive Search - GAS

Algoritmo completo

GAS

- Inicializar $i = 0$, $y_0 = 0$, y $x^{(0)} = (x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)}) = (0, 0, \ldots, 0)$.
- Aplicar el algoritmo de Grover para obtener un estado tal que $f(x) < y_i$. Repetir la cantidad necesaria de veces para asegurar una alta probabilidad de encontrar un x válido, en caso de existir.
- Si se encuentra un x tal que $f(x) < y_i$, actualizar: $y_{i+1} = f(x)$, $x^{(i+1)} = x$, $i \leftarrow i + 1$, y volver al paso anterior.
- En caso contrario, retornar $x^{(i)}$ como el estado que minimiza la función objetivo.

Gracias!

Federico Fuidio

Facultad de Ingeniería
Universidad de Montevideo
Montevideo

Qiskit Fall Fest FIUBA - 10/11/2025