Algoritmos Cuánticos Introductorios

A. Algoritmo de Bernstein-Vazirani

En el problema de Bernstein-Vazirani se tiene una función de n bits $f : \{0,1\}^n \to \{0,1\}$ que presenta un bit de salida. Se sabe que $f_s(\mathbf{x}) = \mathbf{x}.\mathbf{s}$, donde $\mathbf{s}$ es una cadena de n bits desconocida y $\mathbf{x}.\mathbf{s} = x_1 s_1 + x_2 s_2 + \cdots + x_n s_n$. El objetivo es encontrar la cadena de n bits $\mathbf{s}$. Cuál es la complejidad de este problema considerado clásicamente? Si buscamos el resultado exacto, es decir desde un punto de vista determinista, cada consulta a la función nos puede dar 1 solo bit de información sobre $\mathbf{s}$ (porque la salida de f es de 1 bit). Por lo tanto, el número de consultas debe ser como mínimo n.
En el algoritmo cuántico se trabaja con la compuerta unitaria

$$U_s = \sum_{\mathbf{x}\in\{0,1\}^n} \sum_{y\in\{0,1\}} |\mathbf{x}\rangle \langle \mathbf{x}| \, |\mathbf{y} \oplus (\mathbf{s}.\mathbf{x})\rangle \langle \mathbf{y}| \,.$$

La idea del algoritmo de Bernstein-Vazirani es similar a la de Deutsch-Josza: usar el truco de kick-back de fase y una superposición de todos las cadenas de bits posibles como entrada para crear el estado

$$|\psi_{\mathbf{s}}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}} (-1)^{f_s(\mathbf{x})} |\mathbf{x}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}} (-1)^{\mathbf{x}.\mathbf{s}} |\mathbf{x}\rangle \,.$$

Supongamos ahora que generamos otro estado como $|\psi_s\rangle$ pero con otra cadena en lugar de $\mathbf{s}$ que llamaremos $\mathbf{t}$. Vamos a demostrar que $|\psi_{\mathbf{s}}\rangle$ y $|\psi_{\mathbf{t}}\rangle$ son ortogonales.

$$\begin{aligned}
\langle\psi_{\mathbf{s}}|\psi_{\mathbf{t}}\rangle &= \tfrac{1}{2^n} \textstyle\sum_{\mathbf{x}\in\{0,1\}} (-1)^{\mathbf{x}.\mathbf{s}} \langle \mathbf{x}| \sum_{\mathbf{y}\in\{0,1\}} (-1)^{\mathbf{y}.\mathbf{t}} |\mathbf{y}\rangle \\
&= \tfrac{1}{2^n} \textstyle\sum_{\mathbf{x},\mathbf{y}\in\{0,1\}} (-1)^{\mathbf{x}.\mathbf{s}+\mathbf{y}.\mathbf{t}} \langle x|y\rangle \\
&= \tfrac{1}{2^n} \textstyle\sum_{\mathbf{x},\mathbf{y}\in\{0,1\}} (-1)^{\mathbf{x}.\mathbf{s}+\mathbf{y}.\mathbf{t}} \delta_{\mathbf{x},\mathbf{y}} \\
&= \tfrac{1}{2^n} \textstyle\sum_{\mathbf{x}\in\{0,1\}} (-1)^{\mathbf{x}.\mathbf{s}+\mathbf{x}.\mathbf{t}} \\
&= \tfrac{1}{2^n} \textstyle\sum_{\mathbf{x}\in\{0,1\}} (-1)^{\mathbf{x}.(\mathbf{s}+\mathbf{t})}.
\end{aligned}$$

El producto y la adición son entre bits y por lo tanto, módulo 2.
Consideremos la suma para $\mathbf{k}$ fijo

$$\begin{aligned}\sum_{\mathbf{x}\in\{0,1\}}(-1)^{\mathbf{x}.\mathbf{k}} &= \left(\sum_{x_1\in\{0,1\}}(-1)^{x_1k_1}\right)\left(\sum_{x_2\in\{0,1\}}(-1)^{x_2k_2}\right)\cdots\left(\sum_{x_n\in\{0,1\}}(-1)^{x_nk_n}\right)\\ &= 2\delta_{k_1,0}2\delta_{k_2,0}\ldots 2\delta_{k_n,0}.\\ &= 2^n\delta_{\mathbf{k},0}.\end{aligned}$$

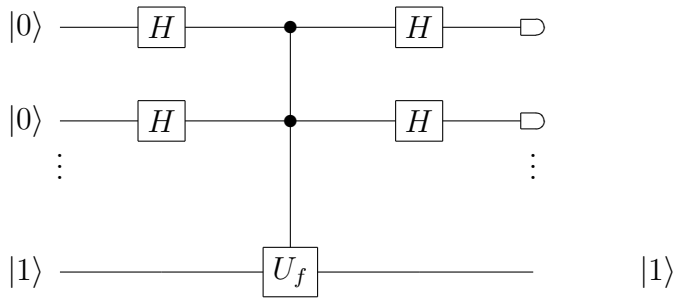
Ésto implica que $\langle\psi_{\mathbf{s}}|\psi_{\mathbf{t}}\rangle = \delta_{(\mathbf{s}+\mathbf{t},0)}$, o como esta suma es modulo 2,

$$\langle\psi_{\mathbf{s}}|\psi_{\mathbf{t}}\rangle = \delta_{(\mathbf{s},\mathbf{t})}.$$

En otras palabras, estos estados son ortogonales.
Qué significa ésto para un algoritmo cuántico? Un conjunto ortonormal de vectores forma una base tal que se puede *medir en esa base*. Se puede hacer un cambio de base a la computacional y de ahí sacar la cadena $\mathbf{s}$.

$$H^{\otimes n}\left|\psi_{\mathbf{s}}\right\rangle = \left|\mathbf{s}\right\rangle.$$

De esta forma encontramos un algoritmo para identificar $\mathbf{s}$ usando *una sola* consulta cuántica. El circuito es el mismo que el del problema de Deutsch-Josza.



El algoritmo clásico, si bien requiere n consultas es polinomial y por lo tanto, clásicamente tratable. Una complicación adicional al problema lo convierte en el *problema recursivo de Bernstein-Varizani* que es superpolinomial clásicamente y se resuelve en n pasos cuánticamente aplicando recursivamente el algoritmo visto antes.También se conoce como el *Muestreo Recursivo de Fourier*.

En el primer paso del problema recursivo de Bernstein-Vazirani se requiere encontrar alguna función $g : \{0,1\}^n \to \{0,1\}$ sobre $\mathbf{s}$, $g(\mathbf{s})$. Esta función es fácil de calcular teniendo $\mathbf{s}$ pero difícil no teniéndolo, se conoce como función 'hard core'. Por ejemplo, se puede tener $g(\mathbf{s}) = \mathbf{s}.\mathbf{x}(mod2)$, para todo x. Para convertir este problema en recursivo (de nivel 2), se compone consigo mismo. Ahora ya no se tiene acceso a la función (oráculo) como antes y en cambio el oráculo son dos cadenas de n bits, digamos $\mathbf{x} \in \{0,1\}^n$ y $\mathbf{y} \in \{0,1\}^n$. La función que buscamos es de la forma: $\{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ y está dada por $f(\mathbf{x},\mathbf{y}) = \mathbf{s}_x.\mathbf{y}$, donde $\mathbf{s}_x$ representa 2^n cadenas diferentes de bits que satisfacen $g(\mathbf{s}_x) = \mathbf{s}.\mathbf{x}$ para algún $\mathbf{s}$. El problema de Bernstein-Vazirani de nivel 2 consiste en encontrar $g(\mathbf{s})$.

Por qué este problema es más difícil que el no-recursivo? Supongamos que obtenemos $\mathbf{s}_x.\mathbf{y}$. Sabemos que nos tomará algún trabajo encontrar algún $\mathbf{s}_x$ para un x fijo (porque este es el problema no-recursivo original). Pero aún despues de encontrarlo y calcular $g(\mathbf{s}_x)$, tenemos que repetir este trabajo para el resto de los x. Así vemos que este problema da mucho más trabajo que el original.

Se puede aumentar el nivel de recurrencia de esta forma k veces. En ese nivel k la función toma como entrada k cadenas de n bits, $\mathbf{x}_i$ y calcula $f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k) = \mathbf{x}_k.\mathbf{s}_{x_1,x_2,\ldots,x_{k-1}}$. La cadena secreta $\mathbf{s}_{x_1,x_2,\ldots,x_{k-1}}$ es la solución de otro problema de nivel menor: $g(\mathbf{s}_{x_1,x_2,\ldots,x_{k-1}}) = \mathbf{x}_{k-1}.\mathbf{s}_{x_1,x_2,\ldots,x_{k-2}}$. Ésto sigue hasta que en el nivel final se tiene $g(\mathbf{s}_{x_1}) = \mathbf{x}_1.\mathbf{s}$.

Históricamente éste fue el primer problema en el que se encontró una separación super-polinomial entre los algoritmos clásicos (BPP) y cuánticos(BQP).

B. Estimación Cuántica de Fase

El problema de la *estimación cuántica de la fase* es importante porque conduce naturalmente a la *transformada cuántica de Fourier* que es usada por todos los algoritmos que veremos más adelante.

Notemos, para empezar, que la segunda compuerta de Hadamard, tanto en el algoritmo de Deutsch como en el de Deutsch-Jozsa era usada para obtener información codificada en las fases de los estados. Recordemos que

$$\begin{aligned} H|x\rangle &= \tfrac{1}{\sqrt{2}}(|0\rangle + (-1)^x|1\rangle) \\ &= \tfrac{1}{\sqrt{2}}\sum_{z\in\{0,1\}}(-1)^{xz}|z\rangle\,. \end{aligned}$$

Se puede pensar que la compuerta de Hadamard codifica información sobre el valor de x en las fases relativas entre los estado $|0\rangle$ y $|1\rangle$. Como la compuerta de Hadamard es auto-inversa, aplicando otra vez H se recupera el estado $|x\rangle$ de nuevo:

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

Claro que estas fases $(-1)^x$ tienen una forma muy particular. En general la fase es un número complejo de la forma $e^{2\pi i\omega}$, con $\omega \in (0,1)$ número real.
Supongamos que tenemos el estado

$$\frac{1}{\sqrt{2^n}}\sum_{y=0}^{2n-1} e^{2\pi i\omega y}|y\rangle\,,$$

donde $\omega \in (0,1)$. Hasta ahora habíamos considerado $\mathbf{y}$ como n-uplas de valores binarios, pero ahora son cadenas de n bits que representan enteros de 0 a $2n-1$. $|y\rangle$ se entiende como el estado de la base $|\mathbf{y}\rangle$, con $\mathbf{y}$ la codificación binaria del entero y.

Problema de Estimación de Fase

Se tiene el estado $\frac{1}{\sqrt{2^n}}\sum_{y=0}^{2^n-1} e^{2\pi i\omega y}|y\rangle\,.$
Problema Estimar la fase ω.

Escribamos primero ω como

$$\omega = 0.x_1x_2x_3\ldots$$

lo que significa $x_1.2^{-1} + x_2.2^{-2} + x_3.2^{-3} + \ldots$.
Similarmente

$$2^k\omega = x_1x_2x_3\ldots x_k.x_{k+1}x_{k+2}\ldots$$

y como $e^{2\pi ik} = 1$ para k entero, se tiene

$$e^{2\pi i2^k(\omega)} = e^{2\pi i(0.x_{k+1}x_{k+2}\cdots)}.$$

Empecemos con 1-qubit de entrada y $\omega = 0.x_1$, el estado será

$$\begin{aligned}
\tfrac{1}{\sqrt{2}} \textstyle\sum_{y=0}^{1} e^{2\pi i(0.x_1)y} |y\rangle &= \tfrac{1}{\sqrt{2}} \textstyle\sum_{y=0}^{1} e^{2\pi i(\frac{x_1}{2})y} |y\rangle \\
&= \tfrac{1}{\sqrt{2}} \textstyle\sum_{y=0}^{1} e^{\pi i(x_1 y)} |y\rangle \\
&= \tfrac{1}{\sqrt{2}} \textstyle\sum_{y=0}^{1} (-1)^{x_1 y} |y\rangle \\
&= \tfrac{1}{\sqrt{2}} (|0\rangle + (-1)^{x_1} |1\rangle)
\end{aligned}$$

Recordando lo que vimos antes podemos usar la compuerta de Hadamard de 1-qubit para determinar x_1 y por lo tanto ω.

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

Ahora vamos a probar una identidad muy útil:

$$\begin{aligned}
\tfrac{1}{\sqrt{2^n}} \textstyle\sum_{y=0}^{2^n-1} e^{2\pi i \omega y} |y\rangle &= \tfrac{1}{2^{n/2}} (|0\rangle + e^{2\pi i(2^{n-1}\omega)} |1\rangle) \otimes (|0\rangle + e^{2\pi i(2^{n-2}\omega)} |1\rangle) \otimes \cdots \otimes (|0\rangle + e^{2\pi i(\omega)} |1\rangle) \\
&= \tfrac{1}{2^{n/2}} (|0\rangle + e^{2\pi i(0.x_n)} |1\rangle) \otimes (|0\rangle + e^{2\pi i(0.x_{n-1}x_n)} |1\rangle) \otimes \dots \\
&\quad \cdots \otimes (|0\rangle + e^{2\pi i(0.x_1 x_2 \dots x_n)} |1\rangle)
\end{aligned}$$

Para demostrar esta igualdad se siguen los pasos:

$$\begin{aligned}
\frac{1}{2^{n/2}} \textstyle\sum_{y=0}^{2^n-1} e^{2\pi i \omega y} |y\rangle &= \frac{1}{2^{n/2}} \textstyle\sum_{y_1=0}^{1} \cdots \sum_{y_n=0}^{1} e^{2\pi i \omega (\sum_{l=1}^{n} y_l 2^{-l})} |y_1 \dots y_n\rangle \\
&= \frac{1}{2^{n/2}} \textstyle\sum_{y_1=0}^{1} \cdots \sum_{y_n=0}^{1} \bigotimes_{l=1}^{n} e^{2\pi i \omega y_l 2^{-l}} |y_l\rangle \\
&= \frac{1}{2^{n/2}} \bigotimes_{l=1}^{n} \big[\textstyle\sum_{y_l=0}^{1} e^{2\pi i \omega y_l 2^{-l}} |y_l\rangle \big] \\
&= \frac{1}{2^{n/2}} \bigotimes_{l=1}^{n} \big[|0\rangle + e^{2\pi i \omega 2^{-l}} |1\rangle \big] \\
&= \frac{1}{2^{n/2}} (|0\rangle + e^{2\pi i(0.x_n)} |1\rangle) \otimes (|0\rangle + e^{2\pi i(0.x_{n-1}x_n)} |1\rangle) \otimes \dots \\
&\quad \cdots \otimes (|0\rangle + e^{2\pi i(0.x_1 x_2 \dots x_n)} |1\rangle)
\end{aligned}$$

Veamos qué pasa ahora con el estado de 2-qubits $\frac{1}{\sqrt{2^2}}\sum_{y=0}^{2^2-1} e^{2\pi i\omega y}\,|y\rangle$ y $\omega = 0.x_1x_2$. Usando la identidad anterior, el estado se puede escribir como

$$\frac{1}{\sqrt{2^2}}\sum_{y=0}^{2^2-1} e^{2\pi i(0.x_1x_2)y}\,|y\rangle = \frac{1}{2}(|0\rangle + e^{2\pi i(0.x_2)}\,|1\rangle) \otimes (|0\rangle + e^{2\pi i(0.x_1x_2)}\,|1\rangle).$$

Como vimos antes, x_2 se puede determinar del primer qubit usando una compuerta de Hadamard. Para obtener x_1 se tiene que hacer algo con el segundo qubit. Si $x_2 = 0$ el segundo qubit estará en el estado $\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i(0.x_1)}\,|1\rangle)$ y se puede usar una compuerta de Hadamard para obtener x_1. Pero si $x_2 = 1$ no es tan fácil. Definimos un *operador rotación de fase* de 1-qubit R_2 por la matriz (en base computacional)

$$R_2 = \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^2}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i(0.01)} \end{bmatrix},$$

donde 0.01 en el exponente está escrito en base 2 o sea que es 2^{-2}. La inversa de R_2 es

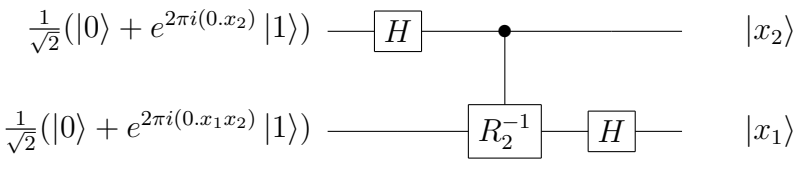
$$R_2^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & e^{-2\pi i(0.01)} \end{bmatrix}.$$

Si $x_2 = 1$ y aplicamos R_2^{-1} al segundo bit:

$$\begin{aligned} R_2^{-1}\Big(\frac{|0\rangle + e^{2\pi i(0.x_1 1)}\,|1\rangle}{\sqrt{2}}\Big) &= \frac{|0\rangle + e^{2\pi i(0.x_1 1 - 0.01)}\,|1\rangle}{\sqrt{2}} \\ &= \frac{|0\rangle + e^{2\pi i(0.x_1)}\,|1\rangle}{\sqrt{2}}. \end{aligned}$$

Despues de aplicar R_2^{-1} se puede obtener x_1 con una compuerta de Hadamard. Como sólo es necesario aplicar R_2^{-1} si $x_2 = 1$ y el estado del primer qubit es $|x_2\rangle$ despues de pasar por Hadamard, se puede aplicar una *compuerta controlada* R_2^{-1} (controlada por el primer qubit) al segundo qubit.

El circuito de la Figura resuelve el problema de Estimación de Fase (exactamente) para un estado de 2-qubits con $\omega = 0.x_1x_2$.



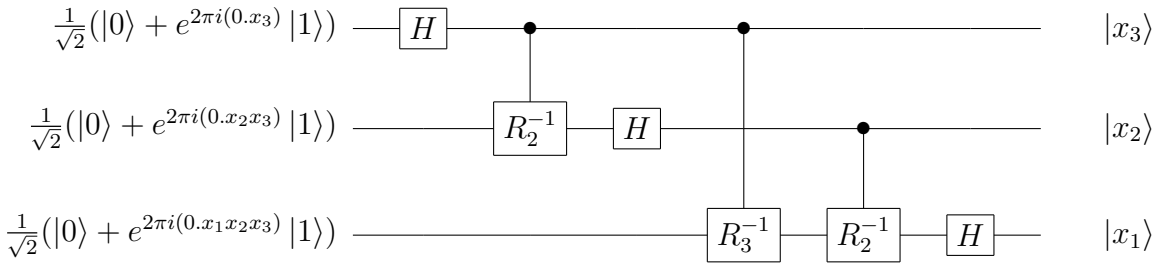
La solución propuesta a la estimación de fase puede ser generalizada a estados de más qubits. Por ejemplo para determinar $\omega = 0.x_1x_2x_3$ en el estado de 3-qubits $\frac{1}{2^{3/2}} \sum_{y=0}^{2^3-1} e^{2\pi i(0.x_1x_2x_3)y} |y\rangle$. De acuerdo al desarrollo demostrado antes este estado se puede expresar como:

$$\Big(\frac{(|0\rangle + e^{2\pi i(0.x_3)} |1\rangle)}{\sqrt{2}}\Big) \otimes \Big(\frac{(|0\rangle + e^{2\pi i(0.x_2x_3)} |1\rangle)}{\sqrt{2}}\Big) \otimes \Big(\frac{(|0\rangle + e^{2\pi i(0.x_1x_2x_3)} |1\rangle)}{\sqrt{2}}\Big).$$

Definimos una compuerta general de rotación de fase de 1-qubit R_k

$$R_k = \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^k}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i(0.0\ldots1)} \end{bmatrix},$$

donde el 1 en el exponente está en la posición k.
En este caso, tenemos que aplicar R_3^{-1} y R_2^{-1} para obtener x_1. El circuito se muestra en la figura



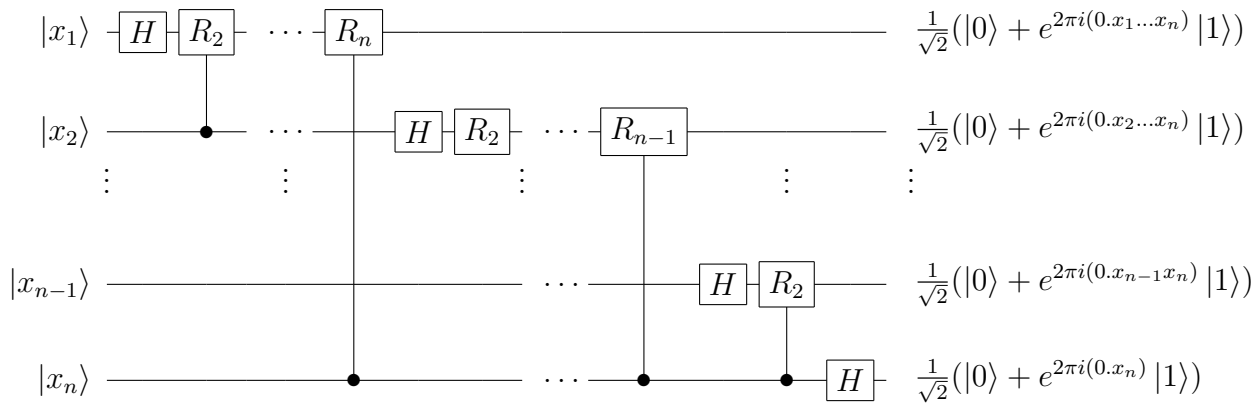
Es elemental la generalización a estados de n qubits.
Hemos visto la estimación de fase cuando ésta es de la forma $\omega = 0.x_1x_2\ldots x_n$ o, dicho de otra forma, cuando ω es de la forma $\frac{x}{2^n}$ para algún entero n. Para ω arbitrario se usa el mismo circuito de estimación de fase que resultará en una aproximación x tal que $\frac{x}{2^n}$ es cercano a ω con alta probabilidad dada por n.
El circuito de estimación de fase sobre estados de n qubits que estima una fase de la forma $0.x_1x_2\ldots x_n$ dando como salida el estado $|x_n\ldots x_2x_1\rangle$ realiza la transformación

$$\frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} e^{2\pi i \frac{x}{2^n} y} |y\rangle \rightarrow |x\rangle .$$

Consideremos la inversa de esta transformación

$$|x\rangle \rightarrow \frac{1}{2^{n/2}} \sum_{y=0}^{2^n-1} e^{2\pi i \frac{x}{2^n} y} |y\rangle .$$

Esta transformación se asemeja a la *transformada discreta de Fourier*. Se llama la *Transformada Cuántica de Fourier* (QFT_{2^n}) o directamente QFT. Es importante señalar que se realiza simplemente aplicando el circuito de estimación de fase a la inversa (orden y compuertas inversas). El circuito correspondiente se muestra en la Figura:



C. Estados Periódicos

Vamos a estudiar el comportamiento de la QFT en los llamados *estados periódicos*. Una superposición de estados periódicos es de la forma

$$|\phi_{r,b}\rangle = \frac{1}{\sqrt{m}} \sum_{z=0}^{m-1} |zr + b\rangle .$$

Decimos que este estado es periódico con *período* r, *desplazamiento* b y m repeticiones del período.

Problema: Hallar el Período r de un Estado Periódico, Dado mr

Entrada:

- Entero mr
- Oráculo generador de estados cuánticos

$$|\phi_{r,b}\rangle = \frac{1}{\sqrt{m}} \sum_{z=0}^{m-1} |zr + b\rangle \,,$$

donde $b \in \{0, 1, \ldots, r-1\}$ es elegido aleatoriamente con distribución uniforme.

Problema: Encontrar r.

Si se mide directamente $|\phi_{r,b}\rangle$ en la base computacional, se obtiene $zr + b$ para algún valor $z \in \{0, 1, \ldots, m-1\}$ elegido uniforme y aleatoriamente. Como $b \in \{0, 1, \ldots, r-1\}$ también es elegido al azar y con distribución uniforme, la probabilidad de que la medición resulte en algún entero $x \in \{0, 1, \ldots, mr-1\}$ es $\frac{1}{mr}$ para todos los valores posibles y no se logra entonces información útil para deducir r.
Sin embargo, si aplicamos QFT_{mr}^{-1} a $|\phi_{r,b}\rangle$ se obtiene

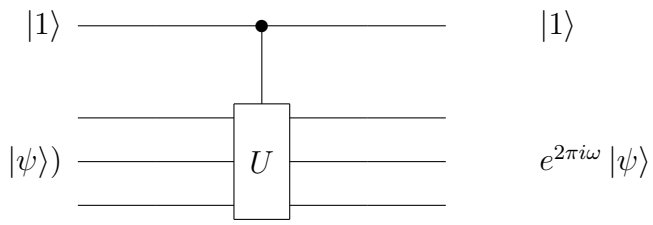
$$QFT_{mr}^{-1} |\phi_{r,b}\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{-2\pi i \frac{b}{r} k} |mk\rangle$$

Si medimos este estado obtendremos un valor $x = mk$ con k entre 0 y $r-1$. Como sabemos mr, podemos calcular $\frac{x}{mr} = \frac{k}{r}$ y expresarlo como fracción reducida. Hay un problema, sin embargo, en el caso que k y r tengan algún factor común, porque el denominador no será r sino algún divisor de r. Supongamos por ejemplo que $m = 3, r = 20, x = 24$. Inicialmente conocemos $mr = 60$ y midiendo $QFT_{60}^{-1} |\phi_{r,b}\rangle$ nos dió el número $x = 24$. En este caso entonces, $\frac{24}{60} = \frac{8}{20}$ y $k = 8$. Pero al reducir la fracción obtendremos $\frac{24}{60} = \frac{2}{5}$. Se perdió el factor 4 porque es también factor de 24.
Para resolver este problema se repite el procedimiento para obtener dos resultados x_1 y x_2, que cumplen $\frac{x_1}{mr} = \frac{k_1}{r}$ y $\frac{x_2}{mr} = \frac{k_2}{r}$, con k_1 y k_2 enteros entre 0 y $r-1$. Por medio del *Algoritmo Extendido de Euclides* se puede encontrar los enteros c_1, c_2, r_1, r_2 tal que $MCD(c_1, r_1) = MCD(c_2, r_2) = 1$ y $\frac{k_1}{r} = \frac{c_1}{r_1}$ y $\frac{k_2}{r} = \frac{c_2}{r_2}$. Ésto significa que r_1 y r_2 son divisores de r, o sea que es múltiplo común de r_1 y r_2. De hecho, es el mínimo común múltiplo de ambos con probabilidad $\frac{6}{\pi^2}$.

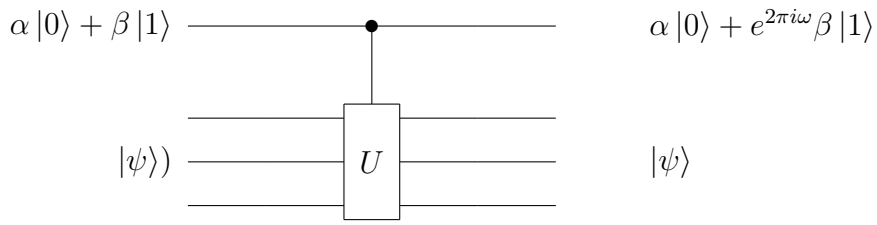
D. Estimación de Autovalores

Cuando estudiamos el algoritmo de Deutsch señalamos que podíamos pensar el operador U_f como un operador controlado $c - U_{f(x)}$. También vimos que el estado $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$ del qubit target era una autovector de $U_{f(x)}$ con autovalor $(-1)^{f(x)}$ y mostramos que se puede asociar este autovalor al bit de control. Vamos a extender esta idea para estimar autovalores de operadores multi-qubits.

Consideremos un operador unitario de n-qubits U que tiene un autovector $|\psi\rangle$ con autovalor $e^{2\pi i\omega}$. Si armamos una compuerta U-controlada, ya hemos visto que U sólo se aplica si el qubit control está en $|1\rangle$. Si el qubit target está en $|\psi\rangle$ se obtiene lo mostrado en la Figura



Si el qubit control está en una superposición $\alpha\,|0\rangle + \beta\,|1\rangle$ se obtiene



donde se aplicó el kick-back de la fase.

Problema: Estimación de Autovalores

Entrada: Operador U que tiene un autovector $|\psi\rangle$ con autovalor $e^{2\pi i\omega}$.

Problema: Obtener una buena estimación de ω

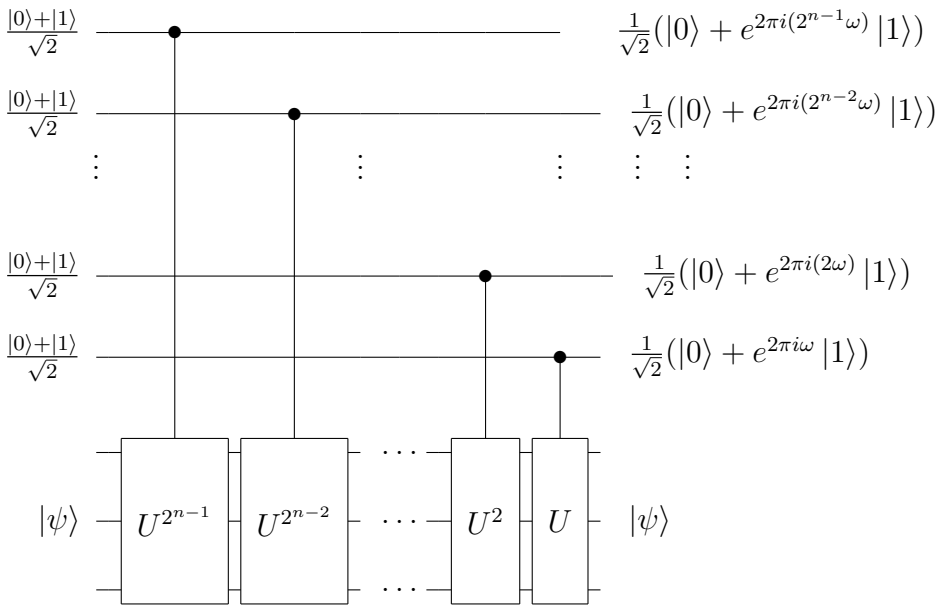
Recordemos que la QFT inversa nos permite obtener ω dado

$$\begin{aligned}\frac{1}{\sqrt{2^n}}\sum_{y=0}^{2^n-1} e^{2\pi i\omega y}\,|y\rangle \;&=\; \frac{1}{2^{n/2}}(|0\rangle + e^{2\pi i(2^{n-1}\omega)}\,|1\rangle)\otimes(|0\rangle + e^{2\pi i(2^{n-2}\omega)}\,|1\rangle)\otimes\ldots\\ &\quad\cdots\otimes(|0\rangle + e^{2\pi i(\omega)}\,|1\rangle)\end{aligned}$$

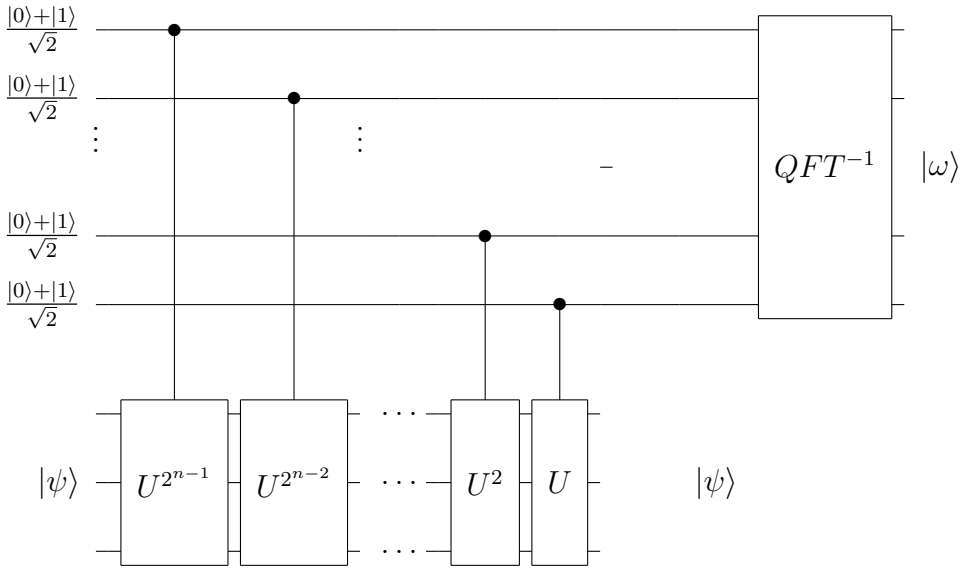
Entonces si podemos imaginar un circuito que arme este estado, podremos aplicar QFT^{-1} para estimar el autovalor. Con ese objeto veamos que $|\psi\rangle$ es también un autovector de U^2 con autovalor $e^{2.2\pi i\omega}$ así como de U^l, con l entero y autovalor $e^{l.2\pi i\omega}$. Por lo tanto tenemos que trabajar con una compuerta U^{2^j}-controlada, el qubit de control $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$ y el qubit target en $|\psi\rangle$ para obtener:

$$c-U^{2^j}\Big(\Big(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\Big)\,|\psi\rangle\Big) = \Big(\frac{|0\rangle+e^{2\pi i(2^j\omega)}\,|1\rangle}{\sqrt{2}}\Big)\,|\psi\rangle\,.$$

Por lo tanto el estado buscado se obtiene con el circuito:



Si ahora se aplica QFT^{-1} a la salida se obtendrá un estado $|\omega\rangle$ que da una buena estimación del parámetro ω del autovalor.
El circuito se muestra en la Figura



Los qubits de control en el estado $\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)\ldots\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)$ se pueden obtener partiendo de $|0\rangle^{\otimes n}$ y aplicando la Transformación de Hadamard $H^{\otimes n}$. Se puede verificar fácilmente que

$$QFT\,|0\rangle^{\otimes n} = H^{\otimes n}\,|0\rangle^{\otimes n}$$

Además como tenemos una secuencia de operaciones U^{2^k} controladas sobre el k-ésimo bit x_k de $x = 2^{n-1}x_{n-1} + \cdots + 2x_1 + x_0$, es fácil de ver que es equivalente a aplicar U x veces y entonces se puede escribir como un solo operador U^x.

El circuito general para estimar el autovalor $e^{2\pi i\omega}$ del operador U se muestra en la figura.