UNIVERSITAT DE BARCELONA
Facultat de Matemàtiques i Informàtica

GRAU DE MATEMÀTIQUES

Treball final de grau

Computació quàntica: L'algoritme de Shor

Autora: Ariadna Celma Miralles

Director: Dr. Luis Dieulefait
Realitzat a: Departament de Matemàtiques i Informàtica

Barcelona, 24 de gener de 2022

Abstract

This work introduces the bases of quantic computation to describe Shor's algorithm. This algorithm allows us to factor a number N with polynomial speed. This implies a qualitative increase in calculation speed, since a quantic computer could calculate in hours or days what a classic computer may calculate in milion years. For that reason, criptographies like RSA may become obsolete, because factoring its public key and deciphering the message would be too quick.

Resum

En aquest treball ens introduirem en les bases de la computació quàntica per poder descriure l'Algoritme de Shor. Aquest és un algoritme que permet factoritzar un nombre N en un temps polinòmic. Això comportaria un enorme increment en la velocitat de càlcul, de manera que un ordinador quàntic podria realitzar-lo en hores o dies quan els ordinadors actuals amb computació clàssica requeririen milions d'anys. A causa d'això, criptografies com la RSA quedarien obsoletes, ja que seria molt ràpid factoritzar-ne la clau pública i desxifrar el missatge.

2020 Mathematics Subject Classification. 11Y05, 11Y11, 81P65, 81P68, 81P94

Agraïments

Per començar voldria agrair moltíssim l'ajuda rebuda per part del meu tutor, en Luis, en aquest treball de final de grau. Amb el seu interès i suport ha fet d'aquesta experiència un camí més amè.

A les meves companyes de la universitat, Hortènsia, Sara i Mònica, per aquests anys compartits plens de bones vivències i, sobretot, Iris, gràcies per tots els moments que m'has regalat de riures, anècdotes i companyonia durant tots aquests anys.

A totes les meves amigues i amics, tan de Reus com de Santiago, que m'han acompanyat en aquesta etapa i m'han fet costat en els moments més dificultosos.

I, sobretot, un agraïment etern a tota la meva família. Mercè, Ferran, Alexandre i Nàroa, sense el vostre suport i amor incondicional no hauria arribat fins aquí. Gràcies.

Índex

1 Introducció

En el primer curs de la carrera se'ns va introduir a l'aritmètica modular i a l'encriptació de missatges, en l'assignatura d'Aritmètica. Personalment, em va semblar una matèria molt interessant i que em va cridar molt l'atenció. Una cosa que ha estat tant present en la història com és la desencriptació de missatges, tant en gran escala, guerres mundials, com en tant petita com pot ser una nota secreta entre amics; veure les matemàtiques que porten darrere i aconseguir comprendre-les em va semblar un privilegi. Més endavant vaig descobrir també la història d'Alan Turing, el matemàtic que durant la segona guerra mundial va aconseguir desenvolupar una màquina capaç de desxifrar el codi de la màquina alemanya *Enigma*. Tot plegat em picava la curiositat i m'incrementava les ganes d'aprendre sobre el tema. Així que, quan buscant temàtiques sobre els treballs de final de grau vaig topar-me amb l'opció de criptografia, no vaig trigar gens en decantar-m'hi. Quan se'm va proposar enfocar-lo en la computació quàntica, no negaré que em vaig espantar una mica. Però més tard vaig pensar que què millor que tractar el tema en l'actualitat, per conèixer una mica més el món on vivim.

Aquest treball comença amb una petita introducció a la computació clàssica que ens permet entrar just després en la seva evolució quàntica. Introduïm la notació de Dirac per visualitzar els estats en forma de vectors i fer més comprensibles i "palpables" les operacions amb que els manipulem.

En l'apartat 3, introduïm els conceptes de circuits quàntics i les portes bàsiques que els formen. Amb ells, serem capaços de descriure les transformacions dels qubits imprescindibles per l'Algoritme de Shor.

Posteriorment tenim descrit el mètode d'encriptació RSA, el principal afectat per l'algoritme de Shor, ja que un cop factoritzat el número N el missatge és immediat de desxifrar.

En el penúltim apartat ens trobarem amb les transformacions quàntiques de Fourier i amb l'estimació quàntica de fase, ambdós imprescindibles i precursors de l'últim, i més important, dels apartats: l'Algoritme de Shor. En aquest expliquem com funciona aquest algoritme quàntic per factoritzar nombres enters grans i observem un exemple pràctic amb el seu circuit particular.

2 INTRODUCCIÓ A LA COMPUTACIÓ QUÀNTICA

2.1 Què és un ordinador quàntic?

Algú podria dir que un ordinador quàntic és aquell que funciona seguint les lleis de la mecànica quàntica. No és així. Qualsevol ordinador actual funciona sota aquestes lleis, però no són ordinadors quàntics.

Un ordinador quàntic és aquell que utilitza certes transformacions específiques de la mecànica quàntica, com per exemple la superposició, per tal de realitzar operacions sobre les dades. Les lleis de la mecànica quàntica permeten que aquestes transformacions succeeixin en unes condicions molt específiques i controlades.

La computació quàntica pot ser enormement més eficaç del que s'havia arribat a imaginar en la teoria computacional clàssica. El temps que pot trigar un ordinador quàntic a resoldre aquestes tasques augmenta molt més lentament amb la mida de l'*input* que el que augmentaria en les "equivalents" en un ordinador clàssic.

2.2 *Cbits* i els seus estats

Un ordinador clàssic opera amb cadenes (seqüències) de zeros i uns, convertint-les en altres de diferents. Cada posició en aquestes cadenes és el que s'anomena *bit* i pot contenir tant un 0 com un 1.

Per representar aquestes col·leccions de *bits* l'ordinador ha de contenir el corresponent grup de sistemes físics, cadascun dels quals pot existir en dos estats físics completament oposats, associats als valors 0 o 1 del corresponent *bit* que representa el sistema. (Per exemple, un sistema físic podria ser 0 si la direcció és a la dreta o 1 si és a l'esquerra.)

És comú en la ciència de la computació quàntica usar el terme "*bit*" per descriure els dos estats en el sistema clàssic. Per evitar confusions, els anomenarem *Cbit* en els sistemes físics clàssics i *Qbit* (qubit) en els quàntics.

Representem l'estat de cada *Cbit* en caixes de la següent forma: $|\ \rangle$, on posem a l'interior el corresponent valor de l'estat, 0 o 1.

En principi la representació dels estats de quatre *Cbits*, per exemple 1010, s'escriuria $|1\rangle|0\rangle|1\rangle|0\rangle$, però per facilitar la lectura s'acaba posant tots en la mateixa "caixa": $|1010\rangle$. Algunes vegades també veurem les caixes amb subíndex, que ens estarà indicant el número de *Cbits*, ja que per exemple no podríem distingir si $|6\rangle$ (escrit en base 10) és un estat de 3-*Cbit*, $|110\rangle$, o de 4-*Cbit*, $|0110\rangle$, etc.

Dirac és qui introdueix aquesta notació en els inicis de la computació quàntica, per facilitar l'escriptura i manipulació de vectors (*kets*). Ens serà útil començar a pensar en els estats com vectors, ja que així és com els representarem.

Per començar, podem veure què podem fer quan prenem els estats $|0\rangle$ i $|1\rangle$ d'un sol *Cbit* representant-los com dos vectors unitaris ortogonals en un espai bidimensional:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{i} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

En el cas de 2 *Cbits* l'espai vectorial seria de dimensió 4, amb bases ortonormals $\{00\}$, $\{01\}$, $\{10\}$ i $\{11\}$.

2.3 *Qbits* i els seus estats

En els ordinadors quàntics, la variable bàsica és el qubit: una variant quàntica del *bit* clàssic. Té les mateixes restriccions que aquest últim: només pot emmagatzemar una única peça binària d'informació, i només ens pot donar com a *output* el 0 o el 1. Tot i així, també pot ser manipulat de diverses maneres que només es poden descriure amb la mecànica quàntica.

Amb els vectors podem descriure més estats complexos a part del $|0\rangle$ i el $|1\rangle$. Com aquests dos estats formen una base ortonormal, podem representar qualsevol vector de dimensió 2 com una combinació d'ells dos. Això és el que anomenarem Superposició.

L'estat $|\psi\rangle$ associat a un qubit pot ser qualsevol vector unitari en l'espai vectorial de dimensió 2, abastat per $|0\rangle$ i $|1\rangle$ sobre els nombres complexos. L'estat general d'un qubit és:

$$a|0\rangle + b|1\rangle = \begin{pmatrix} a \\ b \end{pmatrix},$$

on a i b són dos nombres complexos amb l'única condició que $|\psi\rangle$ sigui un vector unitari en l'espai vectorial complex, és a dir, que $|a|^2 + |b|^2$ sigui 1.

L'estat $|\psi\rangle$ és una superposició dels estats $|0\rangle$ i $|1\rangle$ amb amplituds a i b. Si a o b és 0, i l'altra és 1, aleshores tenim els estats clàssics. Generelitzem amb n qubits.

2.3.1 Operacions reversibles

Els ordinadors quàntics realitzen una part important de la seva màgia a través d'operacions reversibles, les quals transformen l'estat inicial del qubit a la seva forma final utilitzant només processos que poden ser invertits. Només hi ha un únic procés que no pot ser invertit, aquest és la mesura, que al mateix temps és l'única manera d'extreure informació útil dels qubits un cop han adquirit la seva forma final.

En una operació reversible cada estat final sorgeix d'un únic estat inicial. L'única operació reversible no trivial que podem aplicar a un *Cbit* és l'operació NOT, que intercanvia l'estat 0 i el 1. (NOT és reversible perquè existeix l'inversa, ja que si apliquem dues vegades aquestes operacions és com aplicar l'identitat).

$$\text{NOT} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Si, per exemple, mirem per un parell de *Cbits*, l'operació reversible més genèrica és qualsevol permutació dels seus 4 possibles estats, per exemple "S_{ij}", que intercanvia els estats dels *Cbits* i i j.
L'extensió en els qubits és la Controlled NOT o cNOT. Aquesta actua de la següent manera: si l'estat i és 0, no canvia l'estat del *Cbit* j. Però si és 1, sí que el canvia.

2.4 Notació de Dirac

La notació de Dirac s'usa per descriure estats quàntics. Siguin $a, b \in \mathbb{C}^2$:

Ket: $|a\rangle = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$ (vector columna)

Bra: $\langle b| = |b\rangle^{T*} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}^{T*} = \begin{pmatrix} b_1^* & b_2^* \end{pmatrix}$ (vector fila conjugat)

(on, si $b = c + di$, aleshores $b^* = c - di$)

Bra-ket: $\langle b|a\rangle = \begin{pmatrix} b_1^* & b_2^* \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = a_1 b_1^* + a_2 b_2^* = \langle a|b\rangle^*$

Ket-bra: $|a\rangle\langle b| = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \begin{pmatrix} b_1^* & b_2^* \end{pmatrix} = \begin{pmatrix} a_1 b_1^* & a_1 b_2^* \\ a_2 b_1^* & a_2 b_2^* \end{pmatrix}$

Definim els estats $|0\rangle := \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ i $|1\rangle := \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, que són ortogonals. (Fàcil de comprovar si fem el producte escalar $\langle 0|1\rangle = 1 \cdot 0 + 0 \cdot 1 = 0$)

2.5 Mesures

Si tu tens n *Cbits*, cadascun representat per 0 o 1, pots saber l'estat de cadascun a simple vista. No hi ha problema en descobrir l'estat d'aquests *Cbits*, ni, per tant, saber el resultat de qualsevol operació realitzada en ells. És a dir, l'estat dels *Cbits* no es veu alterat en el procés de llegir-los.

En canvi, si tu tens n qubits en superposició dels estats base computacionals, no hi ha res que es pugui fer per extreure l'informació continguda en les amplituds. No es poden llegir aquestes amplituds i, per tant, no es pot descobrir en quin estat està.

Només hi ha una manera d'extreure aquesta informació de n qubits donat un estat: s'anomena la Mesura. Realitzar la mesura consisteix en realitzar un cert test en cada qubit, el resultat del qual será 0 o 1. L'estat determina la probabilitat dels possibles resultats, seguint la següent norma: la probabilitat de generar cert resultat ve donada pel quadrat de la norma de l'amplitud de l'estat que volem calcular, en l'expansió de l'estat $|\psi\rangle$ dels qubits en els 2^n estats base.

Aquesta norma és coneguda com la norma de Born. Ens mostra el lligam entre les amplituds i els números que pots llegir quan mesures els qubits. Les condicions de normalització són els requeriments per tal que les probabilitats dels 2^n resultats sumin 1.

A diferència de les portes unitàries, que tenen un únic *output* (estat després de la porta) per cada *input* (estat abans de la porta), l'estat dels qubits que resulta d'una porta de mesura només és estadísticament determinat per l'estat dels qubits entrats. Per tant, l'acció de la mesura no pot ser desfeta. Donat l'estat final $|x\rangle$ no hi ha cap manera de recuperar l'estat inicial $|\psi\rangle$. És irreversible.

Si n qubits, inicialment descrits per un estat $|\psi\rangle$, s'envien a través d'una porta de mesura de n-qubits i el resultat mostra x, aleshores un associa els qubits emergents de la mesura a l'estat clàssic base $|\psi\rangle_n$. Això vol dir que totes les amplituds que caracteritzaven els estats entrants han desaparegut en l'estat resultant. L'únic que s'ha aconseguit de la mesura és determinar la probabilitat d'un output particular. Així doncs, quan envies n qubits a través d'una porta de mesura, elimines la possibilitat d'extreure cap més

informació de l'estat original $|\psi\rangle$. D'això en diem que l'estat col·lapsa.

Triem bases ortogonals per descriure (mesurar) estats quàntics. Si fem una mesura en la base $\{|0\rangle, |1\rangle\}$, l'estat col·lapsa en algun dels estats $|0\rangle$ o $|1\rangle$, ja que els dos són estats propis de σ_z (operador Pauli-z), i anomenem a això z-mesura.

Hi ha infinites bases diferents. Les més comunes són:

$$\{|+\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle \ , \ |-\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\}$$
$$\{|+i\rangle = \tfrac{1}{\sqrt{2}}\ (|0\rangle + i|1\rangle \ , \ |-i\rangle = \tfrac{1}{\sqrt{2}}\ (|0\rangle - i|1\rangle)\}$$

Que corresponen als estats propis de σ_x i σ_y, respectivament.

2.6 La regla de Born

Per trobar la probabilitat de mesurar $|x\rangle$, fem el producte escalar de $|x\rangle$ i de l'estat que estem mesurant (en aquest cas $|\psi\rangle$), i llavors calculem el seu quadrat.

Aquesta norma ens mostra que les amplituds estan relacionades amb les probabilitats que obtenim. Si volem que aquestes sumin 1, necessitem que el vector estigui normalitzat, i.e., necessitem que $\langle\psi|\psi\rangle = 1$.

Aleshores, si teníem que: $|\psi\rangle = a|0\rangle + b|1\rangle$, llavors $|a|^2 + |b|^2 = 1$.

La probabilitat que un estat $|\psi\rangle$ col·lapsi durant una mesura en la base $\{|x\rangle, |x^T\rangle\}$ a l'estat $|x\rangle$ ve donada per

$$P(x) = |\langle x|\psi\rangle|^2$$

i, com ja hem dit, si està normalitzat compleix

$$\sum_i P(x_i) = 1.$$

Exemples:

1. $|\psi\rangle = \frac{1}{\sqrt{3}}\ (|0\rangle + \sqrt{2}\cdot|1\rangle)$ el mesurem en la base $\{|0\rangle,|1\rangle\}$

   $\mathrm{P}(0) = |\ \langle 0|\ \frac{1}{\sqrt{3}}\ (|0\rangle + \sqrt{2}\cdot|1\rangle)\rangle|^2 = |\frac{1}{\sqrt{3}}\langle 0|0\rangle + \sqrt{\frac{2}{3}}\langle 0|1\rangle|^2 = \frac{1}{3}$

   $\mathrm{P}(1) = |\langle 1|\frac{1}{\sqrt{3}}(|0\rangle + \sqrt{2}\cdot|1\rangle)\rangle|^2 = |\frac{1}{\sqrt{3}}\langle 1|0\rangle + \sqrt{\frac{2}{3}}\langle 1|1\rangle|^2 = \frac{2}{3}$

   Ja que $\langle 0|1\rangle = 0 = \langle 1|0\rangle$ i $\langle 0|0\rangle = 1 = \langle 1|1\rangle$.

2. $|\psi\rangle = \frac{1}{\sqrt{2}}\ (|0\rangle$ - $|1\rangle)$ el mesurem en la base $\{|+\rangle, |-\rangle\}$:

   $P(+) = |\langle +|\psi\rangle|^2 = |\frac{1}{\sqrt{2}}(\langle 0| + \langle 1|)\cdot\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)|^2 = \frac{1}{4}|\langle 0|0\rangle - \langle 0|1\rangle + \langle 1|0\rangle - \langle 1|1\rangle|^2 = \frac{1}{4}\cdot 0 = 0$

   $P(-) = |\langle -|\psi\rangle|^2 = |\frac{1}{\sqrt{2}}(\langle 0| - \langle 1|)\cdot\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)|^2 = \frac{1}{4}|\langle 0|0\rangle - \langle 0|1\rangle - \langle 1|0\rangle + \langle 1|1\rangle|^2 = \frac{1}{4}\cdot 4 = 1$

El resultat és l'esperat, ja que $|\psi\rangle = |-\rangle \Longrightarrow \langle +|\psi\rangle = \langle +|-\rangle = 0$ (perquè són ortogonals).

2.7 Esfera de Bloch

Ja sabem que l'estat general d'un qubit és $|\psi\rangle = a|0\rangle + b|1\rangle$, amb $a, b \in \mathbb{C}$. Podem ser més específics en aquesta descripció. Com que no podem mesurar la fase global, només podem mesurar la diferència de fase entre els estats $|0\rangle$ i $|1\rangle$. Treballarem amb una classe d'equivalència de la parella incial de nombres complexos multiplicant-los per una constant no nula. D'aquesta manera, afegint el terme que ens diu la fase relativa entre tots dos, ens queda: $|\psi\rangle = a|0\rangle + e^{i\varphi}b|1\rangle$, amb $a, b, \varphi \in \mathbb{R}$. Això ho hem fet multiplicant per un $e^{-i\phi_1}$ amb norma 1, on ϕ_1 és l'argument de a, per obtenir llavors l'expressió descrita anteriorment amb a real.

Finalment, com que teníem que l'estat del qubit estava normalitzat ($\sqrt{a^2 + b^2} = 1$), podem utilitzar l'identitat trigonomètrica i escriure-ho com: $\sqrt{\sin^2 x + \cos^2 x} = 1$, que expressa els termes reals a i b en termes de la variable θ:

$$a = \cos\tfrac{\theta}{2} \qquad b = \sin\tfrac{\theta}{2}$$

Podem escriure qualsevol estat normalitzat com $|\psi\rangle = \cos(\frac{\theta}{2})|0\rangle + e^{i\varphi}\sin(\frac{\theta}{2})|1\rangle$, on $\varphi \in [0, 2\pi]$ descriu la fase relativa i l'angle $\theta \in [0, \pi]$ que determina la probabilitat de mesurar $|0\rangle$ o $|1\rangle$:

$$\mathrm{P}(0) = \cos^2(\tfrac{\theta}{2}) \qquad \text{i} \qquad \mathrm{P}(1) = \sin^2(\tfrac{\theta}{2})$$

Tots els estats normalitzats purs poden ser il·lustrats en la superfície de l'esfera de radi 1 ($|r| = 1$), que anomenem l'esfera de Bloch.

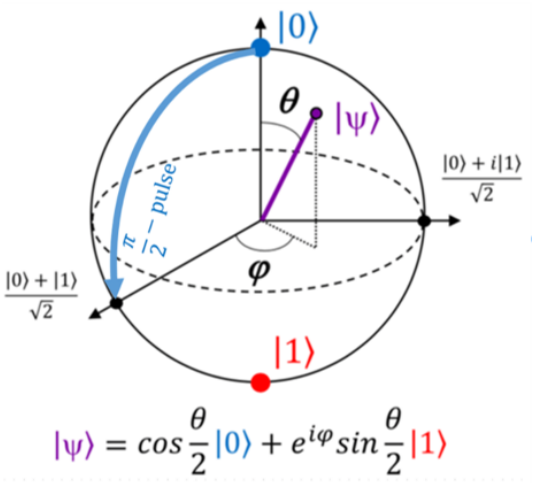
Les coordenades de cada estat són donades pel vector de Bloch: $r = \begin{pmatrix} \sin\theta\cos\varphi \\ \sin\theta\sin\varphi \\ \cos\theta \end{pmatrix}$.

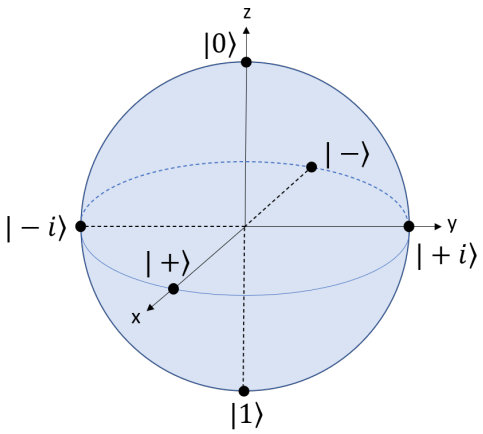
Exemples i dibuix de l'esfera:

- $|0\rangle$: $\theta = 0$, φ arbitrari $\Longrightarrow \vec{r} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$
- $|1\rangle$: $\theta = \pi$, φ arbitrari $\Longrightarrow \vec{r} = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}$
- $|+\rangle$: $\theta = \frac{\pi}{2}$, $\varphi = 0 \Longrightarrow \vec{r} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$
- $|-\rangle$: $\theta = \frac{\pi}{2}$, $\varphi = \pi \Longrightarrow \vec{r} = \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix}$

- $|+i\rangle : \theta = \frac{\pi}{2}, \varphi = \frac{\pi}{2} \Longrightarrow \vec{r} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$
- $|-i\rangle : \theta = \frac{\pi}{2}, \varphi = \frac{3\pi}{2} \Longrightarrow \vec{r} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$

Compte: En l'esfera de Bloch, els angles són el doble de grans que en l'espai de Hilbert. I.e., en un estat general, $|\psi\rangle = \cos(\frac{\theta}{2})|0\rangle + e^{i\varphi}\sin(\frac{\theta}{2})|1\rangle$, θ és l'angle en l'esfera de Bloch, mentre que $\frac{\theta}{2}$ és l'angle real que tenim en l'espai de Hilbert. Per exemple, $|0\rangle$ i $|1\rangle$ són ortogonals, però en l'esfera de Bloch el seu angle és de 180°.

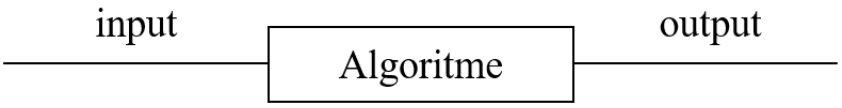




3 CIRCUITS I PORTES

3.1 Model de circuit

En computació quàntica, un circuit és la pràctica de representar mitjançant un diagrama l'acció d'una seqüència de portes (*gates*) que duen a terme computacions elementals a una sèrie de qubits. Els estats inicials els trobem a l'esquerra, els finals a la dreta, i al mig totes les portes. La línia que entra i surt de la caixa representa la transformació unitària.



3.2 Portes de qubits singulars

- Exemple clàssic: NOT 0 a 1 i 1 a 0
- Exemple quàntic:

  Les portes quàntiques sempre estan representades per matrius unitàries ($U^T U = 1$):

$$U = \begin{pmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{pmatrix} = u_{00}|0\rangle\langle 0| + u_{01}|0\rangle\langle 1| + u_{10}|1\rangle\langle 0| + u_{11}|1\rangle\langle 1|$$

  – X-porta: (porta NOT) bit-flip, rotació al voltant de l'eix x per π. Intercanvia les amplituds dels estats $|0\rangle$ i $|1\rangle$.

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = |0\rangle\langle 1| + |1\rangle\langle 0|$$

  Exemples:

  * $\sigma_x\ |0\rangle = |0\rangle\ \langle 1|0\rangle + |1\rangle\ \langle 0|0\rangle = 0 + |1\rangle = |1\rangle$

    $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$

  * $\sigma_x\ |1\rangle = |0\rangle\ \langle 1|1\rangle + |1\rangle\ \langle 0|1\rangle = |0\rangle + 0 = |0\rangle$

    $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$

  * $\sigma_x\ |+\rangle = |+\rangle$ (no l'afecta, igual que a la resta, ja que és la rotació en l'eix de les x)

– Z-porta: phase-flip, rotació al voltant de l'eix z per π.

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = |0\rangle\langle 0| - |1\rangle\langle 1|$$

Exemples:

* $\sigma_z\ |+\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = |-\rangle$
* $\sigma_z\ |-\rangle = (|0\rangle\langle 0|$ - $|1\rangle\langle 1|) \cdot \frac{1}{\sqrt{2}}\ (|0\rangle$ - $|1\rangle) = |+\rangle$

– Y-porta: bit i phase-flip, rotació al voltant de l'eix y per π.

$$\sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = -i|0\rangle\langle 1| + i|1\rangle\langle 0| = i \cdot \sigma_x \cdot \sigma_z$$

σ_x , σ_y i σ_z són les matrius de Pauli i $\sigma_i^2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, que és obvi ja que si apliquem dues vegades qualsevol de les portes fem una rotació de 360°.

Observem que la porta Z no té cap efecte sobre $|0\rangle$ i $|1\rangle$, això és perquè són estats propis d'aquesta porta. Per això, les bases computacionals també s'anomenen Z-bases. D'igual forma tenim les X-bases : $|+\rangle$ i $|-\rangle$.

Usant només les portes Pauli és impossible canviar el qubit inicial a qualsevol estat diferent a $|0\rangle$ o $|1\rangle$, i.e., no podem aconseguir superposició. Per això tenim la següent porta: la porta Hadamard.

– Matriu Hadamard (H-porta): Aquesta porta ens permet sortir dels pols de l'esfera de Bloch i crear superposició de $|0\rangle$ i $|1\rangle$. Es pot pensar com la rotació sobre el vector de Bloch (1 0 1) (la línia entre els eixos x i z), o com la transformació de l'estat del qubit entre les bases z i x.

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{\sqrt{2}}\ (\ |0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| - |1\rangle\langle 1|)$$

Exemples:

* $\mathrm{H}|0\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = |+\rangle$
* $\mathrm{H}|1\rangle = \frac{1}{\sqrt{2}}\ (\ |0\rangle\langle 0|1\rangle + |0\rangle\langle 1|1\rangle + |1\rangle\langle 0|1\rangle - |1\rangle\langle 1|1\rangle) = \frac{1}{\sqrt{2}}\ (\ |0\rangle - |1\rangle) = |-\rangle$

Crea superposició! A més, $\mathrm{H}|+\rangle = |0\rangle$ i $\mathrm{H}|-\rangle = |1\rangle$.

– De forma similar, $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$ afegeix 90° a la fase φ.

* $\mathrm{S}|+\rangle = |+i\rangle$
* $\mathrm{S}|-\rangle = |-i\rangle$

$S \cdot H$ s'aplica per canviar de la base Z a la Y.

Mesura en altres bases. Per exemple, podem mesurar amb les X-bases. Calculem les probabilitats de mesurar $|+\rangle$ o $|-\rangle$: $\mathrm{p}(|+\rangle) = |\langle +|\psi\rangle|^2$ i $\mathrm{p}(|-\rangle) = |\langle -|\psi\rangle|^2$. Després de la mesura, la superposició s'ha destruït.

Veiem també que podem crear una X-porta entreposant una Z-porta entre dues H-portes: X = HZH. Això el que fa és que: començant amb la Z-base, la H-porta ens canvia el qubit a la X-base, la Z-porta realitza NOT en la X-base, i al final la segona H-porta retorna el qubit a la Z-base. Ho podem comprovar multiplicant les respectives matrius:

$$HZH = \tfrac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}\tfrac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} = \tfrac{1}{2}\begin{pmatrix}1 & -1\\ 1 & 1\end{pmatrix}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix} = X$$

3.3 Representació d'estats amb multi-qubits

Ja hem vist que, quan tenim un únic qubit, aquest té dues amplituds complexes. Aleshores, quan tenim per exemple dos qubits, tenim quatre amplituds complexes ja que la base és {00, 01, 10, 11}:

$$|x\rangle = x_{00}|00\rangle + x_{01}|01\rangle + x_{10}|10\rangle + x_{11}|11\rangle = \begin{pmatrix}x_{00}\\ x_{01}\\ x_{10}\\ x_{11}\end{pmatrix}$$

En general, si estem treballant amb n-qubits, tindrem 2^n estats base i, per tant, 2^n amplituds complexes.

Es mesuren de la mateixa manera :

$$\mathrm{P}(|00\rangle) = |\langle 00|x\rangle|^2 = |x_{00}|^2 \quad \text{i} \quad |x_{00}|^2 + |x_{01}|^2 + |x_{10}|^2 + |x_{11}|^2 = 1.$$

Si tenim dos qubits per separat, podem escriure el seu estat col·lectiu utilitzant el producte de kronecker:

$$|a\rangle = \begin{pmatrix}a_0\\ a_1\end{pmatrix} \qquad |b\rangle = \begin{pmatrix}b_0\\ b_1\end{pmatrix}$$

$$|ba\rangle = \begin{pmatrix}b_0 \times \begin{pmatrix}a_0\\ a_1\end{pmatrix}\\ b_1 \times \begin{pmatrix}a_0\\ a_1\end{pmatrix}\end{pmatrix} = \begin{pmatrix}b_0a_0\\ b_0a_1\\ b_1a_0\\ b_1a_1\end{pmatrix}$$

Com hem dit abans, tindrem 2^n i, en conseqüència, el vector tindrà 2^n components. Per tant, significa que creix exponencial al nombre de qubits. Aquesta és la raó per la qual els ordinadors quàntics amb grans nombres de qubits siguin tant difícils de simular.

3.4 Portes singulars en multi-qubits

De forma similar, per calcular les matrius que actuen sobre aquests vectors d'estat usem el producte tensorial:

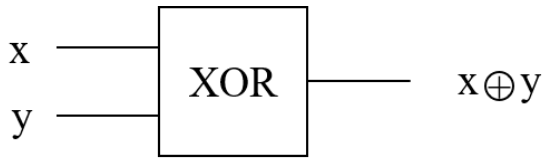
$$M_1|q_1\rangle \otimes M_2|q_0\rangle = (M_1 \otimes M_2)|q_1q_0\rangle$$

Mirem un exemple. Agafem $M_1 = I$ i $M_2 = X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. Aleshores:

$$M_1 \otimes M_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 \times \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & 0 \times \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ 0 \times \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & 1 \times \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

3.5 Portes de dos qubits

- Exemple clàssic: XOR, x i $y \mapsto x + y$



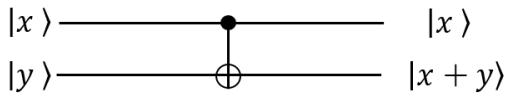
  Però, com que la teoria quàntica és unitària, només considera matrius unitàries i, per tant, portes reversibles. ($U^T U = 1 \mapsto U^{-1} = U^T$)

- Exemple quàntic:
  La porta CNOT (Controlled NOT) és una porta de condició que realitza una X-porta al qubit objectiu si l'estat del de control és $|1\rangle$. Per tant, en el cas de 2 qubits tenim dues opcions: que el primer qubit sigui el de control i el segon l'objectiu, o a l'inrevés.

$$CNOT_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10|$$

$$CNOT_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} = |00\rangle\langle 00| + |01\rangle\langle 11| + |10\rangle\langle 10| + |11\rangle\langle 01|$$

  En el cas de la $CNOT_1$:



  On la y del *output* és la suma de x i y del *input*. Si tenim $|0\rangle$ en $|x\rangle$, $|y\rangle$ no canvia. Si tenim $|1\rangle$ en $|x\rangle$, aleshores $|y\rangle$ sí que canvia.

  Mirem com afecten les dues CNOT en els estats base:

$$CNOT_1|00\rangle_{xy} = CNOT \begin{pmatrix}1\\0\\0\\0\end{pmatrix} = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} = |00\rangle_{xy}$$

$$CNOT_1|01\rangle_{xy} = CNOT \begin{pmatrix}0\\1\\0\\0\end{pmatrix} = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} = |01\rangle_{xy}$$

$$CNOT_1|10\rangle_{xy} = CNOT \begin{pmatrix}0\\0\\1\\0\end{pmatrix} = \begin{pmatrix}0\\0\\0\\1\end{pmatrix} = |11\rangle_{xy}$$

$$CNOT_1|11\rangle_{xy} = CNOT \begin{pmatrix}0\\0\\0\\1\end{pmatrix} = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} = |10\rangle_{xy}$$

$$CNOT_2|00\rangle_{xy} = CNOT \begin{pmatrix}1\\0\\0\\0\end{pmatrix} = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} = |00\rangle_{xy}$$

$$CNOT_2|01\rangle_{xy} = CNOT \begin{pmatrix}0\\1\\0\\0\end{pmatrix} = \begin{pmatrix}0\\0\\0\\1\end{pmatrix} = |11\rangle_{xy}$$

$$CNOT_2|10\rangle_{xy} = CNOT \begin{pmatrix}0\\0\\1\\0\end{pmatrix} = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} = |10\rangle_{xy}$$

$$CNOT_2|11\rangle_{xy} = CNOT \begin{pmatrix}0\\0\\0\\1\end{pmatrix} = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} = |01\rangle_{xy}$$

En les següents taules resumim com les dues CNOT alteren els qubits d'entrada (*inputs*), i els qubits de sortida (*outputs*) que obtenim:

ENTRADA		SORTIDA	
x	y	x	y
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

ENTRADA		SORTIDA	
x	y	x	y
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	1

A continuació podem veure com actuen en qubits amb superposició. Mirarem un exemple: $(\frac{1}{\sqrt{2}} \quad \frac{1}{\sqrt{2}} \quad 0 \quad 0)$, el produeix l'estat $|0\rangle \otimes |+\rangle = |0+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$.

Si li apliquem la $CNOT_2$ ens queda $(\frac{1}{\sqrt{2}} \quad 0 \quad 0 \quad \frac{1}{\sqrt{2}})$. Per tant, aconseguim l'estat: $CNOT|0+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

Observem que, aplicant $CNOT_1$ a un vector $(\frac{1}{\sqrt{2}} \quad 0 \quad \frac{1}{\sqrt{2}} \quad 0)$ obtenim també $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.
Comprovació:

$CNOT_2|0+\rangle = (|00\rangle\langle 00| + |01\rangle\langle 11| + |10\rangle\langle 10| + |11\rangle\langle 01|)\ (\frac{1}{\sqrt{2}}(|00\rangle + |01\rangle) =$

$= \frac{1}{\sqrt{2}}(|00\rangle\langle 00|00\rangle + |01\rangle\langle 11|00\rangle + |10\rangle\langle 10|00\rangle + |11\rangle\langle 01|00\rangle + |00\rangle\langle 00|01\rangle + |01\rangle\langle 11|01\rangle +$

$+ |10\rangle\langle 10|01\rangle + |11\rangle\langle 01|01\rangle) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

(perquè sabem que $\langle 00|00\rangle$ i $\langle 01|01\rangle$ són 1 i els demés 0)

$$CNOT_1 \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

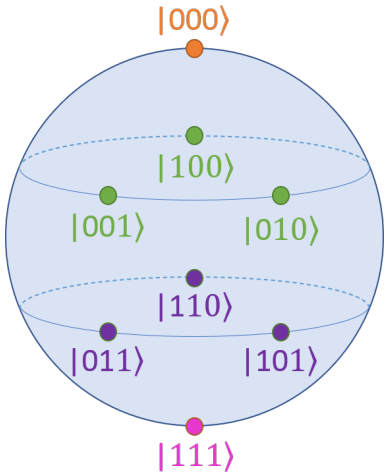
Podem veure que cada funció f pot ser descrita per un circuit reversible. A un irreversible podem afegir-hi *inputs* per tal de tornar-lo així reversible.

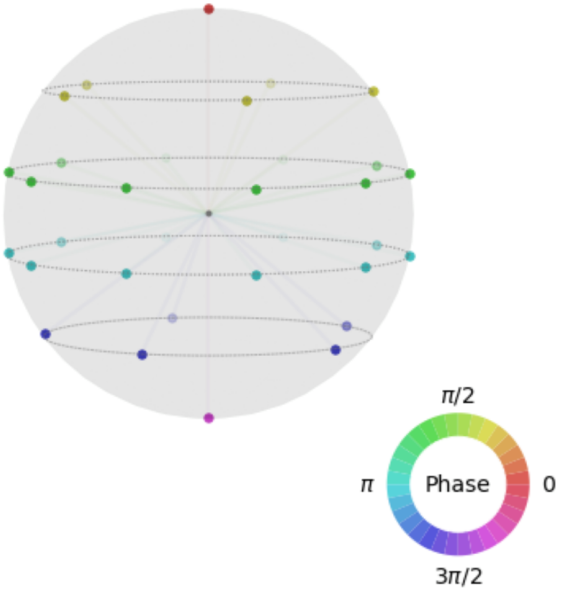
Els circuits quàntics poden descriure totes les funcions que es poden realitzar en un ordinador clàssic, i.e., tot el que fem en un ordinador clàssic ho podem fer també en un de quàntic.

4 La Q-esfera

L'esfera de Bloch només pot il·lustrar l'estat d'un qubit. Per il·lustrar múltiples qubits usem la Q-esfera.

- Per 1-quibit:
  - El "pol nord" representa l'estat $|0\rangle$, el "pol sud" representa l'estat $|1\rangle$.
  - La mida de les boles és proporcional a la probabilitat de la mesura de l'estat respectiu.
  - El color indica la fase relativa comparat de l'estat $|0\rangle$
- Per n-quibits: hi ha 2^n estats de base, e.g., per n = 3 tenim {000, 001, 010, 100, 011, 101, 110, 111}
  Dibuixem aquests estats base com punts distribuïts equitativament en l'esfera amb $0^{\times n}$ al "pol nord" i $1^{\times n}$ al "pol sud", amb tots els altres estats alineats en paral·lels, de manera que el nombre de 1's en cada latitud és constant i en augment de nord a sud.





5 Encriptació RSA

Objectiu: trobar el període r d'una funció f que és periòdica en la suma ordinària, satisfent que $f(x) = f(y)$ per x diferent a y sii x i y difereixen per un múltiple de r. Trobar el període d'una funció periòdica pot ser la clau per factoritzar productes de grans nombres primers, un problema matemàtic amb aplicacions bastant pràctiques.

Podem pensar que trobar el període d'una funció és fàcil, ja que ens imaginem, per exemple, funcions com la del sinus, contínues i que varien lentament. Però les que hauríem de tenir en ment són aquelles que no són contínues i que n'és molt complicat extreure'n el període sense l'informació adequada.

Els algoritmes clàssics més coneguts per trobar el període r d'una funció triguen un temps relacionat amb l'exponencial de $n^{\frac{1}{3}}$, on n és el nombre de *bits* de r. Però l'any 1994, Peter Shor va descobrir que es podia explotar el poder de la computació quàntica per conèixer el període r, en un temps només una mica major a n^3.

L'habilitat de trobar períodes eficaçment, combinat amb alguns trucs de la teoria de nombres, permet factoritzar el producte de dos nombres primers grans. El gran esforç computacional requerit per a totes les tècniques clàssiques de factorització sustenen la seguretat de l'encriptació pel mètode RSA.

Qualsevol ordinador capaç de trobar eficaçment períodes serà una amenaça enorme a la seguretat de comunicació comercial i militar.

5.1 Preliminars

Les entitats bàsiques darrere l'encriptació RSA són els grups finits, on l'operació és la mútiplicació per un cert mòdul N. En l'aritmètica mòdul-N, tots els números que difereixen per un múltiple de N són considerats iguals (en aquest mòdul), per tant només hi ha N quantitats diferents $\{0, ..., N-1\}$.

Sigui G_N el grup de nombres positius menors de N que no tenen factors en comú amb N. Com que la factorització en nombres primers és única, el producte de dos nombres de G_N tampoc té factors en comú amb N, per tant, també pertany a G_N (G_N és tancat per la multiplicació). Si a, b i c de G_N, amb $ab \equiv ac \pmod N$, aleshores $a(b-c)$ és un múltiple de N i, com que a no té factors comuns amb N, $b-c$ és un múltiple de N, per tant, $b \equiv c \pmod N$. Com ja vam veure en aritmètica de primer any, si $a \in G_N$, aleshores a té invers en G_N i l'anomenarem d.

Cada nombre a del grup finit G té un ordre k, que és el menor nombre tal que $a^k = 1 \pmod N$.

L'ordre de cada membre de G és un divisor del nombre de membres de G (ordre de G). Si p és un nombre primer, aleshores el grup G_p conté $p-1$ nombres, ja que cap nombre menor que p comparteix factors amb p. Aleshores, com que $p-1$ és un múltiple d'ordre k de qualsevol a en G_p, segueix que qualsevol a menor que p satisfà:

$$a^{p-1} = 1 \pmod p \qquad \text{Teorema petit de Fermat.}$$

Tot a no divisible per p la podem escriure de la forma $a = mp + a'$ amb m enter i a' menor a p.

L'encriptació de RSA utilitza una extensió del Teorema Petit de Fermat en el cas caracteritzat per dos nombres primers diferents, p i q. Si un enter a no és divisible ni per p ni per q, aleshores cap potència de a és divisible per p ni per q. Per tant, en particular, a^{q-1} no és divisible per p, concluim que $(a^{q-1})^{p-1} = 1 \pmod{p}$ i per la mateixa raó $(a^{p-1})^{q-1} = 1 \pmod{q}$. Amb aquestes dues condicions obtenim que $a^{(q-1)(p-1)} - 1$ és múltiple tan de p com de q. Com que p i q són primers diferents, ha de ser múltiple de pq i, per tant,

$$a^{(q-1)(p-1)} = 1 \pmod{pq}.$$

Notem que, com que a no és divisible ni per p ni per q, no té factors comuns amb pq i per tant està en G_{pq}. El nombre d'elements de G_{pq} és $pq-1-(p-1)-(q-1) = (p-1)(q-1)$, ja que hi ha $pq-1$ enters positius menors que pq, entre els quals hi ha $p-1$ múltiples de q i uns altres $q-1$ múltiples de p. El que acabem de provar és un cas particular del Teorema d'Euler.

Obtenim la versió que és la base de l'encriptació RSA agafant qualsevol potència s i multiplicant a les dues bandes per a.

$$a^{1+s(q-1)(p-1)} = a \pmod{pq}$$

Notem, finalment, que, si c és un enter que no té cap factor en comú amb $(p-1)(q-1)$, aleshores c està en $G_{p-1,q-1}$ i, per tant, té una inversa en aquest grup. I.e., hi ha una d que satisfà que :

$$cd = 1 \pmod{(p-1)(q-1)}$$

Per tant, per algun enter, $cd = 1 + s(p-1)(q-1)$.

I per tant, qualsevol a satisfà que: $a^{cd} = a \pmod{pq}$. Per tant, si $b = a^c \pmod{pq}$, aleshores $b^d = a \pmod{pq}$.

Això constitueix la base de l'encriptació RSA.

5.2 RSA

En Bob vol rebre un missatge de l'Alice codificat per tal que només ell pugui llegir-lo. Per fer-ho, escull dos nombres primers molt grans (d'uns 200 dígits), p i q. Li dóna a l'Alice a través d'un canal públic el producte d'aquests dos nombres, $N = pq$, i un nombre codificador gran, c, que ha escollit de manera que no tingui factors en comú amb $(p-1)(q-1)$. No ha revelat els valors de p i q per separat; sabent que és impossible factoritzar un nombre d'uns 400 dígits amb els ordinadors actuals està tranquil que tan l'Alice com qualsevol intermediari no serà capaç de descobrir-ne els valors coneixent-ne tan sols el producte. En Bob, coneixent p i q, i per tant, $(p-1)(q-1)$, pot trobar l'invers de c mod $((p-1)(q-1))$, d, que satisfà que $cd \equiv 1 \pmod{(p-1)(q-1)}$. Ell es guarda el d per ell sol per utilitzar-lo en la descodificació.

L'Alice codifica un missatge representant-lo com una cadena de menys de 400 dígits usant, per exemple, alguna versió de la codificació ASCII (American Standard Code for Information Interchange). Si el missatge té més de 400 caràcters haurà de dividir-lo en

parts. Ella interpreta cada cadena com un nombre a menor que N. Utilitzant el número codificador c i el valor de $N = pq$ que ha rebut d'en Bob, calcula $b = a^c \pmod{pq}$, i li envia a en Bob a través d'un canal públic. Quan ell rep b, utilitza la coneixença de d per calcular $b^d \pmod{pq}$, que és el missatge a de l'Alice.

Si algun interceptor sigués capaç de factoritzar N trobant p i q, seria capaç de descodificar el missatge de l'Alice igual que ho ha fet en Bob. Però factoritzar un nombre tan gran com N està molt lluny de les capacitats de la computació clàssica.

Descobrir el període de forma eficaç és un interès en la ciència de la criptografia no només per la factorització eficient, sinó perquè podria permetre que els interceptors descodifiquessin el missatge de l'Alice sense que coneguessin ni haguessin de computar els factors p i q. Així funcionaria:

L'intermediari utilitza el seu "buscador de període" per calcular l'ordre r del missatge codificat $b = a^c$ de l'Alice que ella ha enviat públicament. Aquest missatge sabem que està en G_{pq}. Ara, l'ordre r d'aquest missatge és el mateix que l'ordre de a. Això és degut a que el subgrup G_{pq} generat per a conté $a^c = b$, i, per tant, conté el subgrup generat per b, però el subgrup generat per b conté $b^d = a$, i, per tant, el subgrup generat per a. Com que cada subgrup conté a l'altre, aquests han de ser el mateix. I, com que els ordres de a i b són el nombre d'elements del subgrup que generen, tenen el mateix ordre. Per tant, si l'interceptor pot trobar aquest ordre r del missatge codificat b, també coneix l'ordre del missatge original a.

Com que en Bob ha escollit el c de tal manera que no tingui factors en comú amb $(p-1)(q-1)$, i com que r divideix $(p-1)(q-1)$ de G_{pq}, c i r no poden tenir factors en comú. Per tant, c és congruent mòdul r a un cert c' de G_r, que té un invers d' en G_r, i d' és també un invers mòdul r de c:

$$cd' = 1 \pmod{r}.$$

Per tant, donat un c (que en Bob ha cedit públicament) i un r (que l'interceptor ha descobert del missatge codificat b de l'Alice i del N anunciat per en Bob), és fàcil per l'interceptor calcular d' amb un ordinador clàssic, utilitzant en mòdul r el mateix algoritme que utilitza en Bob per trobar d en mòdul $(p-1)(q-1)$. Tenim que, per algun m enter:

$b^{d'} = a^{cd'} = a^{1+mr} = a(a^r)^m = a \pmod{pq}$ ja que com r és l'ordre de a mòdul pq tenim que $a^r \equiv 1 \pmod{pq}$.

6 Preliminars a l'Algoritme de Shor

Recordem que: $|\psi\rangle = a|0\rangle + b|1\rangle$. Aleshores:

$$\langle 0|\psi\rangle = a \longrightarrow Prob(0) = \|a\|^2$$
$$\langle 1|\psi\rangle = b \longrightarrow Prob(1) = \|b\|^2$$
$$\|a\|^2 + \|b\|^2 = 1$$

Implicació 1: Les operacions d'estats quàntics han de preservar norma 1.

- Matrius que preserven la norma $\equiv$ unitàries ($U^T U = 1 \mapsto U^{-1} = U^T$)
- $|det(U)| = 1$

Implicació 2: Per un qubit sol, les portes unitàries són rotacions en la superfície de l'esfera de Bloch.

Implicació 3: Els valors i els vectors propis d'una matriu unitària són especials:

1. $U\ |x\rangle = \lambda_x |x\rangle$, on λ_x és el valor propi (que ha de ser de la forma $e^{i\theta}$) i |x⟩ el vector propi.

   (Quan tu apliques l'operador unitari, el que estàs fent és aplicar una fase en els seus vectors propis)

2. λ_x i λ_y, si $\lambda_y \neq \lambda_x$ aleshores els vectors propis corresponents |x⟩ i |y⟩ són ortogonals, i.e., ⟨x|y⟩ = 0.

6.1 Preliminars

El problema que estem intentant resoldre: donada una funció $f(x)$ que sigui periòdica, trobar el període.

Definició estricta: $f(x) = f(y)$ per $x \neq y$ si i només si $|x - y| = k \cdot p$, on p representa el període.

En l'exemple de l'Algoritme de Shor, amb $n = \log N$, tenim les següents velocitats:

Clàssic: $O\ (exp\ (c \cdot n^{\frac{1}{3}} \cdot (log\ n)^{\frac{2}{3}}))$

Quàntic: $O(n^2\ log(n)\ log(\ log(n)))$, on $log(n)\ log(\ log(n))$ és una mica més ràpid que $O(n^3)$.

Ens fixem que en el quàntic no hi ha exponencial, sinó que és polinomial.

Raons per les quals funciona:

1. Transformacions quàntiques de Fourier
2. Exponenciació modular

El problema de factoritzar un nombre que és producte de dos primers és la base de la seguretat en els nostres ordinadors. Per tant, té sèries implicacions per a la seguretat, però encara no és fàcilment accessible avui en dia.

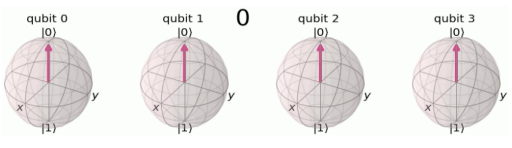
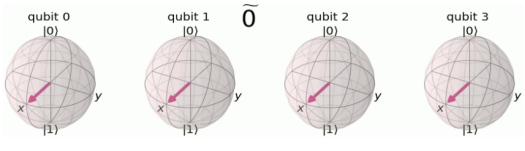
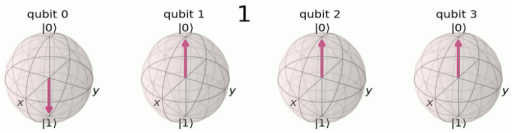
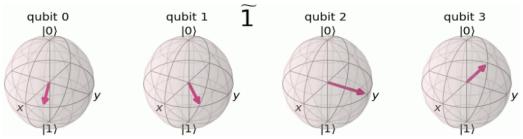
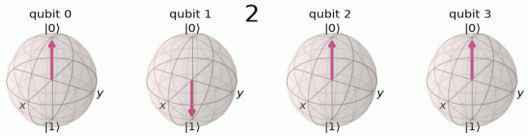
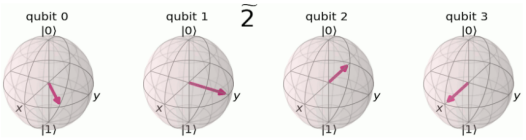
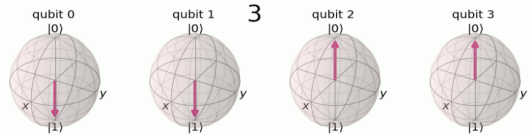
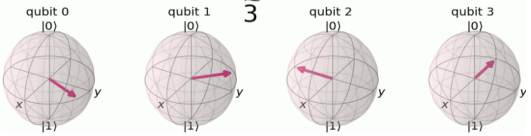
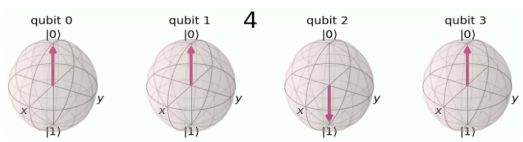
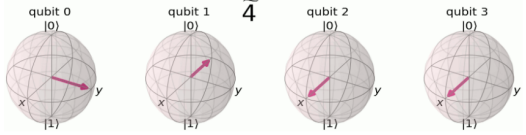
6.2 Transformacions quàntiques de Fourier

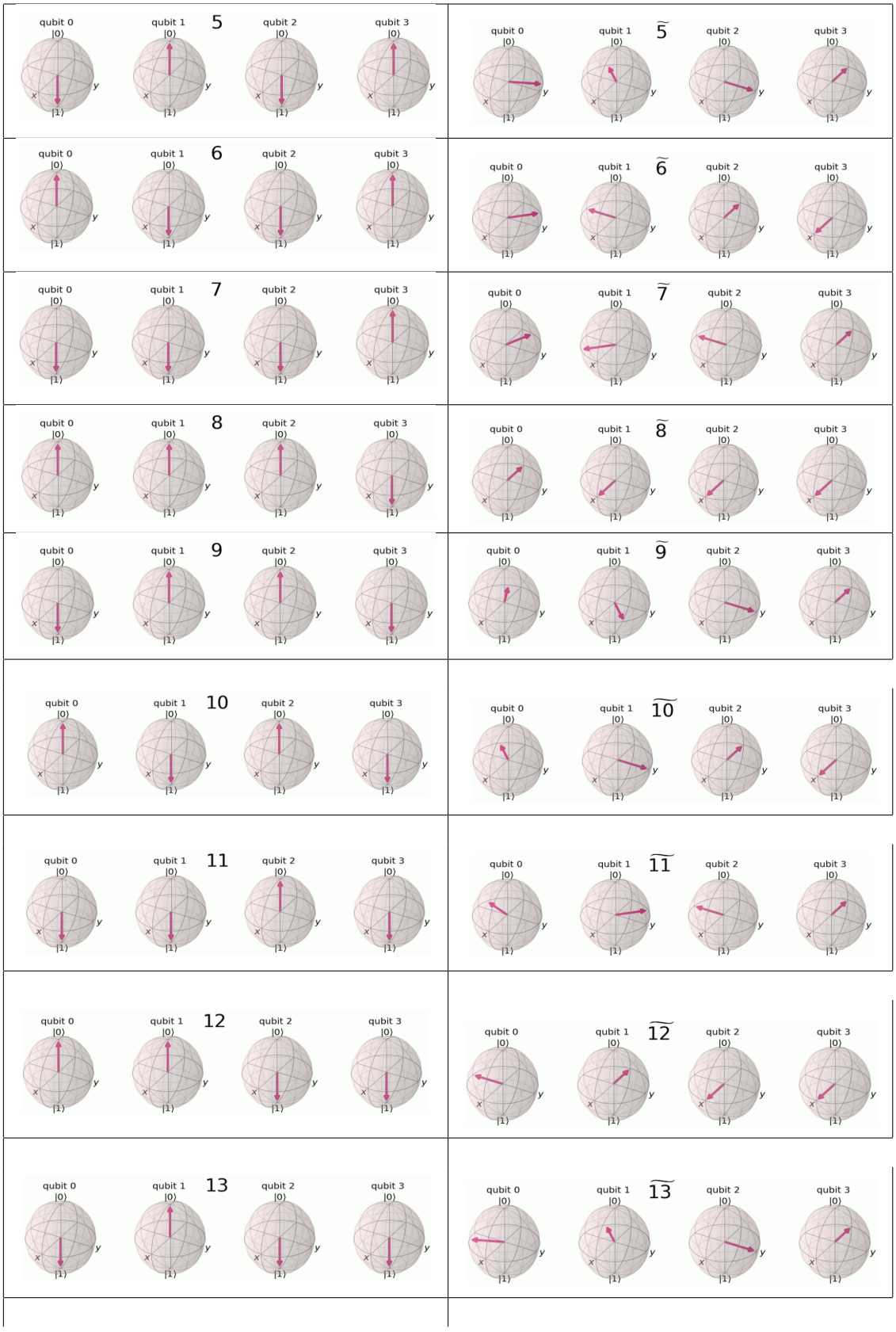
Les transformacions quàntiques de Fourier (QFT) són transformacions entre dues bases, la computacional i les de Fourier. Les portes H són les transformacions quàntiques de Fourier per un sol qubit, que passa de les z-bases $|0\rangle$ i $|1\rangle$ a les x-bases $|+\rangle$ i $|-\rangle$. De la mateixa manera, tots els estats de qubits múltiples en bases computacionals tenen els seus estats corresponents en les bases de Fourier. Les QFT simplement són la funció que transforma d'una base a l'altre.
Per exemple, per 1 qubit:

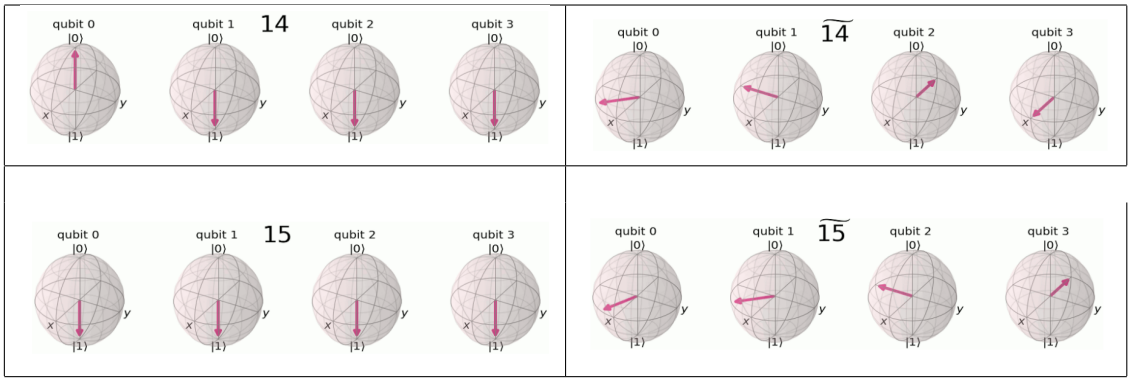
1. Bases computacionals: $\{|0\rangle, |1\rangle\}$
2. Bases de Fourier: $\{|+\rangle, |-\rangle\}$

En la següent taula observarem com canvien les posicions en les bases computacionals i en les de Fourier, en 4 qubits:

En la computacional, emmagatzem números en binari usant els estats $|0\rangle$ i $|1\rangle$. Veiem que cada qubit té una freqüència diferent. El de més a l'esquerra canvia en cada increment del número, el segon cada dos, el tercer cada quatre i l'últim cada vuit.
En canvi, en les bases de Fourier emmagatzem els nombres utilitzant diverses rotacions sobre l'eix z. Veiem que el primer qubit avança a $\frac{2\pi}{16}$, el segon a $\frac{2\pi}{8}$, el tercer a $\frac{2\pi}{4}$ i l'últim a $\frac{2\pi}{2} = \pi$.

Computacional	Fourier
	
	
	
	
	


qubit 0
qubit 1
qubit 2
qubit 3
|0⟩
|1⟩
x
y
5
$\widetilde{5}$
6
$\widetilde{6}$
7
$\widetilde{7}$
8
$\widetilde{8}$
9
$\widetilde{9}$
10
$\widetilde{10}$
11
$\widetilde{11}$
12
$\widetilde{12}$
13
$\widetilde{13}$



Construcció del circuit quàntic que aplica les QFT:

- **1 qubit**: $\{|0\rangle, |1\rangle\}$, 2 estats base
- **2 qubits**: $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, 4 estats base
- ...
- **n qubits**: N estats base, on $N = 2^n$

$$|\tilde{x}\rangle \equiv QFT\ |x\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{\frac{2\pi i x y}{N}} |y\rangle$$

On $|\tilde{x}\rangle$ és la base de Fourier i $|x\rangle$ és la computacional.

Exemple: 1-qubit, per tant, $N = 2^1 = 2$

$$QFT\ |x\rangle = \frac{1}{\sqrt{2}} \sum_{y=0}^{2-1} e^{\frac{2\pi i x y}{2}} |y\rangle = \frac{1}{\sqrt{2}} (e^{\frac{2\pi i x \cdot 0}{2}} |0\rangle + e^{\frac{2\pi i x \cdot 1}{2}} |1\rangle) = \frac{1}{\sqrt{2}} (1 \cdot |0\rangle + e^{\pi i x} \cdot |1\rangle)$$

- $QFT\ |0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + e^{\pi i \cdot 0} \cdot |1\rangle) = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) = |+\rangle$
- $QFT\ |1\rangle = \frac{1}{\sqrt{2}} (|0\rangle + e^{\pi i \cdot 1} \cdot |1\rangle) = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) = |-\rangle$, ja que $e^{i\pi} = -1$.

Fem un petit incís sobre la notació en qubits múltiples. Per exemple, amb n = 3 tindríem:

$$|\tilde{x}\rangle = \frac{1}{\sqrt{8}} \sum_{y=0}^{7} e^{\frac{2\pi i x y}{2^3}} |y\rangle$$

Com és molt difícil escriure $\sum_{y_1=0}^{1} \sum_{y_2=0}^{1} \sum_{y_3=0}^{1} ... \sum_{y_n=0}^{1}$, millor escrivim $\sum_{y=0}^{N-1}$, i tenim que $|y\rangle = |4\rangle$ l'escrivim com $|100\rangle$, on $|y\rangle = |y_1\ y_2\ y_3\ ...\ y_n\rangle$.

Aleshores: $|\tilde{x}\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{\frac{2\pi i x y}{N}} |y\rangle$

$y = [y_1\ y_2\ y_3\ ...\ y_n] = 2^{n-1}y_1 + ... + 2^0 y_n \quad$ (e.g. $y = 3 = [11] = 2^1 \cdot 1 + 2^0 \cdot 1$)

Escrivim, doncs:

$$y = \sum_{k=1}^{n} y_k 2^{n-k}$$

Per tant:

$$|\tilde{x}\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{\frac{2\pi i x \sum_{k=1}^{n} y_k 2^{n-k}}{N}} |y\rangle$$

Com que N = 2^n, ens queda:

$$|\tilde{x}\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{2\pi i x \sum_{k=1}^{n} \frac{y_k}{2^k}} |y_1\ y_2\ y_3\ ...\ y_n\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} \prod_{k=1}^{n} e^{2\pi i x \frac{y_k}{2^k}} |y_1\ y_2\ y_3\ ...\ y_n\rangle =$$
$$= \frac{1}{\sqrt{N}} (|0\rangle + e^{\frac{2\pi i x}{2^1}} |1\rangle) \otimes (|0\rangle + e^{\frac{2\pi i x}{2^2}} |1\rangle) \otimes (|0\rangle + e^{\frac{2\pi i x}{2^3}} |1\rangle) \otimes ... \otimes (|0\rangle + e^{\frac{2\pi i x}{2^n}} |1\rangle\ )$$

Veiem, doncs, que el que fem és:

Com que $|x\rangle = |x_1\ x_2\ x_3\ ...\ x_n\rangle = |x_1\rangle \otimes |x_2\rangle \otimes |x_3\rangle \otimes ... \otimes |x_n\rangle$, observem que cada $|x_j\rangle$ se'ns trasnforma en $|0\rangle + e^{\frac{2\pi i x}{2^j}} |1\rangle$.

$$|x_1\rangle \longrightarrow |0\rangle + e^{\frac{2\pi i x}{2^1}} |1\rangle$$
$$|x_2\rangle \longrightarrow |0\rangle + e^{\frac{2\pi i x}{2^2}} |1\rangle$$
$$|x_3\rangle \longrightarrow |0\rangle + e^{\frac{2\pi i x}{2^3}} |1\rangle$$
$$...$$

Exemple: n = 3 qubits, $N = 2^3 = 8$

$|x\rangle = |5\rangle = |101\rangle$

$QFT\ |x\rangle = |\tilde{x}\rangle = |\tilde{5}\rangle = \frac{1}{\sqrt{8}} (|0\rangle + e^{\frac{2\pi i 5}{2}} |1\rangle) \otimes (|0\rangle + e^{\frac{2\pi i 5}{4}} |1\rangle) \otimes (|0\rangle + e^{\frac{2\pi i 5}{8}} |1\rangle\ )$

6.3 El circuit quàntic que implementa QFT

Hem vist que cada qubit va de $|x_k\rangle$ a $|0\rangle + e^{\frac{2\pi i x}{2^k}}|1\rangle$. El circuit que implementa la QFT utilitza dues portes: la primera és la porta Hadamard per a un sol qubit i la segona és la $CROT_k$ per a dos qubits (rotació controlada).

- $H|x_k\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{\frac{2\pi i x_k}{2}}|1\rangle)$
- $CROT_k = \begin{pmatrix} I & 0 \\ 0 & UROT_k \end{pmatrix}$ on $UROT_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^k}} \end{pmatrix}$

$CROT_k$ actua sobre 2-qubits, on el primer és el de control i el segon l'objectiu.

Observem que:

1. $H\ |x_k\rangle = |0\rangle + e^{\frac{2\pi i x_k}{2}}|1\rangle = \begin{cases} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) & si \quad x_k = 0 \\ \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) & si \quad x_k = 1 \end{cases}$

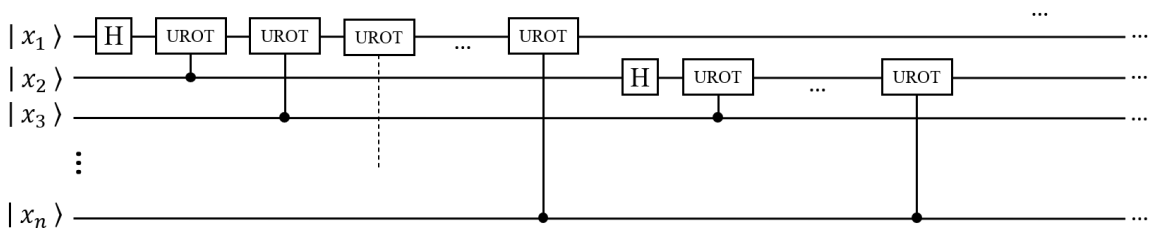
   ja que $\begin{cases} e^{\frac{2\pi i x_k}{2}} = 1 & si \quad x_k = 0 \\ e^{\frac{2\pi i x_k}{2}} = -1 & si \quad x_k = 1 \end{cases}$

2. $UROT_k\ |x_j\rangle = e^{\frac{2\pi i x_j}{2^k}}|x_j\rangle$

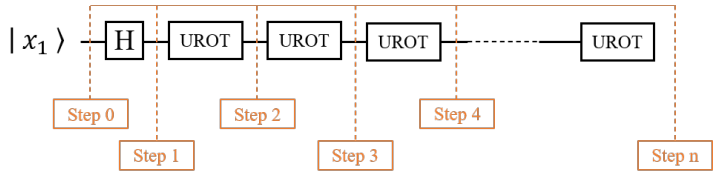
   $\begin{cases} e^{\frac{2\pi i x_j}{2^k}}\ |x_j\rangle = |0\rangle & si \quad x_j = 0 \\ e^{\frac{2\pi i x_j}{2^k}}\ |x_j\rangle = e^{\frac{2\pi i}{2^k}}|1\rangle & si \quad x_j = 1 \end{cases}$

   $UROT_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^k}} \end{pmatrix}$, suma $\frac{2\pi}{2^k}$ a la fase per l'estat $|1\rangle$

Circuit quàntic que implementa QFT



Veiem què passa en el primer qubit:



Step 0: $|x_1\ x_2\ x_3\ ...\ x_n\rangle$

Step 1: $\frac{1}{\sqrt{2}}[|0\rangle + e^{\frac{2\pi i x_1}{2}}|1\rangle] \otimes\ |x_2\ x_3\ ...\ x_n\rangle$

Step 2: $\frac{1}{\sqrt{2}}[|0\rangle + e^{\frac{2\pi i x_2}{2^2}} e^{\frac{2\pi i x_1}{2}}|1\rangle] \otimes\ |x_2\ x_3\ ...\ x_n\rangle$

Step 3: $\frac{1}{\sqrt{2}}[|0\rangle + e^{\frac{2\pi i x_3}{2^3}} e^{\frac{2\pi i x_2}{2^2}} e^{\frac{2\pi i x_1}{2}}|1\rangle] \otimes\ |x_2\ x_3\ ...\ x_n\rangle$

...

Step n: $\frac{1}{\sqrt{2}}[|0\rangle + e^{\frac{2\pi i x_n}{2^n}} ... e^{\frac{2\pi i x_3}{2^3}} e^{\frac{2\pi i x_2}{2^2}} e^{\frac{2\pi i x_1}{2}}|1\rangle] \otimes\ |x_2\ x_3\ ...\ x_n\rangle$

Com $x = 2^{n-1}x_1 + 2^{n-2}x_2 + ... + 2^1 x_{n-1} + 2^0 x_n$ podem escriure

$$\frac{1}{\sqrt{2}}[|0\rangle + e^{\frac{2\pi i x}{2^n}}|1\rangle] \otimes\ |x_2\ x_3\ ...\ x_n\rangle$$

El primer qubit del circuit és el primer de l'expressió. Aplicant-ho ara als qubits 2 fins al n ens queda:

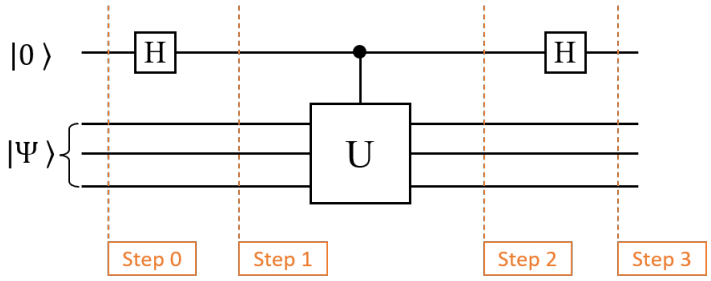
$$\frac{1}{\sqrt{2}}[|0\rangle + e^{\frac{2\pi i x}{2^n}}|1\rangle] \otimes\ \frac{1}{\sqrt{2}}[|0\rangle + e^{\frac{2\pi i x}{2^{n-1}}}|1\rangle] \otimes\ ... \otimes\ \frac{1}{\sqrt{2}}[|0\rangle + e^{\frac{2\pi i x}{2^1}}|1\rangle]$$

Aquest circuit implementa les QFT, excepte per l'ordre invers dels qubits a l'*output*.

6.4 Estimació de fase quàntica:

Objectiu: donat un operador unitari U, l'algoritme estima θ en $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$. Ψ és el vep i $e^{2\pi i\theta}$ el vap corresponent. Com U és unitària tots els vaps tenen norma 1.

Sabem que quan mesurem $\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)$ serà $|0\rangle$ amb probabilitat $\frac{1}{2}$ i $|1\rangle$ amb probabilitat $\frac{1}{2}$. Si tenim $e^{\frac{i\pi}{2}}\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)$ serà $|0\rangle$ amb probabilitat $|\frac{e^{\frac{i\pi}{2}}}{\sqrt{2}}|^2 = \frac{1}{2}$ i $|1\rangle$ amb probabilitat $|\frac{e^{\frac{i\pi}{2}}}{\sqrt{2}}|^2 = \frac{1}{2}$.



Step 0: $|0\rangle|\psi\rangle$

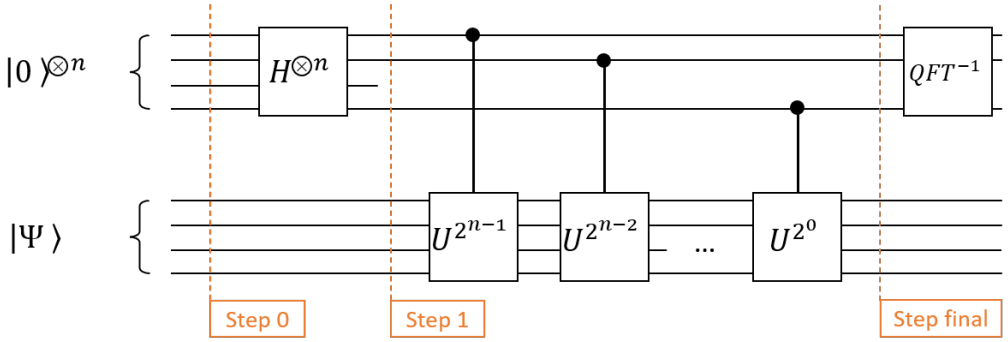
Step 1: $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|\psi\rangle$

Step 2: $\frac{1}{\sqrt{2}}(|0\rangle|\psi\rangle + |1\rangle e^{i\theta_\psi}|\psi\rangle)$ (L'operador unitari aplica el canvi de fase només a $|1\rangle$)

Step 3: $\frac{1}{\sqrt{2}}((\frac{|0\rangle+|1\rangle}{\sqrt{2}})|\psi\rangle + (\frac{|0\rangle-|1\rangle}{\sqrt{2}})e^{i\theta_\psi}|\psi\rangle) = \frac{1}{2}[(1+e^{i\theta_\psi})|0\rangle + (1-e^{i\theta_\psi})|1\rangle]|\psi\rangle$

Probabilitat de mesurar $|0\rangle$: $|\frac{1}{2}(1+e^{i\theta_\psi})|^2$
Probabilitat de mesurar $|1\rangle$: $|\frac{1}{2}(1-e^{i\theta_\psi})|^2$

Obtindrem més precisió si utilitzem més qubits:



Step 0: $|0\rangle^{\otimes n}|\psi\rangle$

Step 1: $\frac{1}{\sqrt{2}}^n(|0\rangle + |1\rangle)^{\otimes n}|\psi\rangle$

(...)

I com $U^{2^x}|\psi\rangle = U^{2^x-1}U|\psi\rangle = U^{2^x-1}e^{i\theta_\psi}|\psi\rangle = e^{i\theta_\psi}e^{i\theta_\psi}U^{2^x-2}|\psi\rangle$, tenim que:

Step final: $(\frac{1}{\sqrt{2}})^n(|0\rangle + e^{i\theta_\psi 2^{n-1}}|1\rangle) \otimes (|0\rangle + e^{i\theta_\psi 2^{n-2}}|1\rangle) \otimes ... \otimes (|0\rangle + e^{i\theta_\psi 2^0}|1\rangle)|\psi\rangle$.

Si comparem amb les transformacions quàntiques de Fourier:

$$|\tilde{x}\rangle = \frac{1}{\sqrt{N}}(|0\rangle + e^{\frac{2\pi ix}{2^1}}|1\rangle) \otimes (|0\rangle + e^{\frac{2\pi ix}{2^2}}|1\rangle) \otimes (|0\rangle + e^{\frac{2\pi ix}{2^3}}|1\rangle) \otimes ... \otimes (|0\rangle + e^{\frac{2\pi ix}{2^n}}|1\rangle\ )$$

Observem que la estimació quàntica de fase és el mateix que les QFT, excepte que $\theta_\psi \longrightarrow \frac{\theta_\psi}{2^n} 2\pi$.

L'algoritme d'estimació de fase quàntica utilitza "*phase kickback*" per escriure la fase de U (en les bases de Fourier) als t qubits en el registre de recompte. Aleshores utilitzem l'invers de les QFT per traduir-ho de les bases de Fourier a les computacionals, les quals sabem mesurar.

En les bases de Fourier el qubit de més amunt completa una rotació quan el comptador va de 0 a 2^t. Per comptar fins un nombre x entre 0 i 2^t, rotem aquest qubit per $\frac{x}{2^t}$ al voltant de l'eix de les z. Pel següent qubit rotem $\frac{2x}{2^t}$, després, $\frac{4x}{2^t}$.

Quan usem un qubit per controlar la porta U, el qubit fa un gir proporcional a la fase $e^{2i\pi\theta}$. Podem utilitzar portes CU successives per repetir aquesta rotació un apropiat nombre de vegades fins que haguem codificat la fase θ com un nombre entre 0 i 2^t en la base de Fourier.

Després apliquem l'invers de QFT per convertir-ho a la base computacional.

En resum: donada una U unitària, $U|\psi\rangle = e^{2\pi i\theta|\psi\rangle}$, l'estimació de fase ens dóna $2^n\theta$, on aquí n és el nombre de qubits usats per estimar θ.

7 Algoritme de Shor

Problema: factoritzar un nombre $N = p \cdot q$, on p i q són primers molt grans.

Petit tast de l'aritmètica modular:

$5 \div 3$: quocient = 1 i residu = 2, aleshores podem escriure $5 \equiv 2 \pmod 3$

$x =$	1	2	3	4	5	6	7	8	9	
$x \equiv$	1	2	0	1	2	0	1	2	0	(mod 3)

Notem que:

$x \equiv 0 \longrightarrow$ és múltiple de 3

$x \equiv 1 \longrightarrow$ és múltiple de 3 més 1

$x \equiv 2 \longrightarrow$ és múltiple de 3 més 2

En general, $x \equiv y \pmod 3 \longrightarrow x = 3k + y$ per algun k pertanyent als enters.

També notem que la periodicitat de l'aritmètica modular $x \equiv y \pmod N$ significa que $y \in \{0, \ldots, N-1\}$, on y denota el menor enter positiu que és congruent amb x mòdul N (e.g. $x \equiv y \pmod 3$ significa que $y \in \{0, 1, 2\}$)

Ara es veurà que, l'algortime de Shor, que serveix per trobar el període, es considera en realitat un algoritme per factoritzar. Observem només el cas rellevant de l'encriptació RSA, on volem factoritzar el producte de dos grans primers, $N = pq$, encara que la relació entre el descobriment del període i la factorització sigui més general.

Si tenim una manera de determinar un període (per exemple l'algoritme de Shor), i volem trobar els primers grans de N, p i q, escollim un nombre a aleatori coprimer a N. La possibilitat que a sigui múltiple de p o q són minúscules quan són tan grans, però si no te'n fies ho pots comprovar mitjançant l'algoritme d'Euclides.

Utilitzant el nostre cercador de període, trobem l'ordre de a en G_{pq}: el menor r que compleix que $a^r \equiv 1 \pmod{pq}$.

Aleshores podem utilitzar aquesta informació per factoritzar N de dues maneres, si l'elecció de a ha estat encertada.

Suposem primer que hem tingut la sort que r és parell. Aleshores podem calcular $x \equiv a^{\frac{r}{2}} \pmod{pq}$ i notem que $0 = x^2 - 1 = (x-1)(x+1) \pmod{pq}$.

Ara, $x - 1 = a^{\frac{r}{2}} - 1$ no és congruent a 0 mòdul pq ja que r és la menor potència de a congruent a 1 en aquest mòdul.

Suposem que, a més, encara hem tingut més sort, i $x + 1 = a^{\frac{r}{2}} + 1$ no és congruent 0 $\pmod{pq}$.

En aquest cas, ni $x - 1$ ni $x + 1$ és divisible per N, però sabem que el seu producte sí que ho és. Com que p i q són primers, això és possible només si un dels dos divideix $x - 1$ (diguem p) i l'altre divideix $x + 1$ (diguem q).

Com que els únics divisors de N són p i q, tenim que p és el màxim comú divisor de N i $x - 1$, i q el mcd de N i $x + 1$. Aleshores podem trobar p i q amb l'algoritme d'Euclides.

Veiem que per aconseguir factoritzar necessitem una mica de sort en l'elecció de la a. Està demostrat en l'apèndix M del llibre "Quantum computer science: An Introduction" que la probabilitat d'aconseguir una a amb aquestes condicions és del 0.5, per tant, no ens hauria de costar molts intents aconseguir-ho.

Protocol per l'algoritme de Shor esquemàtic:

Tenim $N = p \cdot q$

1. Triar un número a que sigui coprimer amb N
2. Trobar l'ordre de r de la funció $a^r \equiv 1 \pmod{N}$.
   ($\equiv$ l'r més petit tal que $a^r \equiv 1 \pmod{N}$)
3. Si r és parell (és necessari):
   $x \equiv a^{\frac{r}{2}} \pmod{N}$
   si $x + 1 \neq 0 \pmod{N}$ aleshores $\{p, q\}$ (almenys un dels dos) estan continguts en $\{\text{mcd}(\ x + 1\ ,\ N),\ \text{mcd}(\ x - 1\ ,\ N)\}$
   Si no, hem de trobar una altra a.

Exemple:

Prenem el número 21 i extreurem els seus factors amb l'algoritme de Shor.

1. Triar un número a que sigui coprimer amb 21: per exemple $a = 17$
2. Trobar l'ordre de r de la funció $17^r \equiv 1 \pmod{21}$.

$x =$	0	1	2	3	4	5	6	...
$17^x \pmod{21} =$	1	17	16	20	4	5	1	...

3. Com que $r = 6$ és parell:
   $x \equiv 17^{\frac{6}{2}} \pmod{21}$, i.e., $x \equiv 17^3 \pmod{21} = 20 \pmod{21}$
   Com que $x + 1 \equiv 0 \pmod{21}$ hem de triar una altra a.

1. Agafem ara $a = 13$
2. Trobar l'ordre de r de la funció $13^r \equiv 1 \pmod{21}$.

$x =$	0	1	2	3	4	5	6	...
$13^x \pmod{21} =$	1	13	1	13	1	13	1	...

3. Com que $r = 2$ és parell:
   $x \equiv 13^{\frac{2}{2}} \pmod{21}$, i.e., $x \equiv 13^1 \pmod{21} = 13 \pmod{21}$
   Com que $x + 1 \equiv 14 \pmod{21} \neq 0 \pmod{21}$, podem calcular:

$$\text{mcd}(x + 1,\ N) = \text{mcd}\ (14,\ 21) = 7;$$
$$\text{mcd}(x - 1,\ N) = \text{mcd}\ (12,\ 21) = 3;$$

   I tenim que p i/o q es troben en $\{7,\ 3\}$. En aquest cas, tots dos, així que ja hem factoritzat el nombre.

7.1 Funcionament de l'algortime de Shor a partir d'un exemple amb circuit

En aquest apartat veurem com actua l'algoritme de Shor a través d'un exemple extret del curs virtual de Qiskit.

Prenem $N = 15$. Sabem que $15 = [1111]$ (quatre qubits). Primer fem el protocol com anteriorment.

1. Agafem ara $a = 13$
2. Trobar l'ordre de r de la funció $13^r \equiv 1 \pmod{15}$.

$$\begin{array}{rcccccc} x = & 0 & 1 & 2 & 3 & 4 & 5 & \dots \\ 13^x \pmod{15} = & 1 & 13 & 4 & 7 & 1 & 13 & \dots \end{array}$$

3. Com que $r = 4$ és parell:

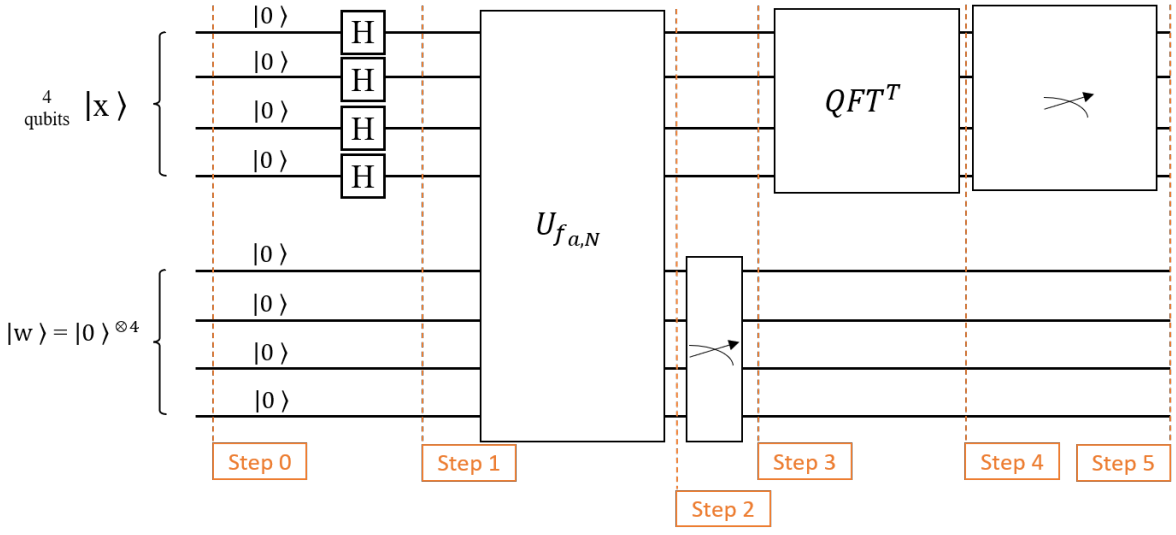
   $x \equiv 13^{\frac{4}{2}} \pmod{15}$, i.e., $x \equiv 13^2 \pmod{15} = 4 \pmod{15}$

   Com que $x + 1 \equiv 5 \pmod{15} \neq 0 \pmod{15}$, podem calcular:

$$\begin{aligned} \text{mcd}(x + 1,\, N) &= \text{mcd}\,(5,\, 15) = 5; \\ \text{mcd}(x - 1,\, N) &= \text{mcd}\,(3,\, 15) = 3; \end{aligned}$$

   I tenim que p i/o q es troben en $\{3, 5\}$. En aquest cas, tots dos, així que ja hem factoritzat el nombre.

Ara mirem el circuit per factoritzar $N = 15$:



On $f_{a,N}(x) \equiv a^x \pmod{N}$.

$|x\rangle|w\rangle \longrightarrow |x\rangle|w \oplus f_{a,N}(x)\rangle$

Step 0: $|0\rangle^{\otimes 4}|0\rangle^{\otimes 4}$

Setp 1: $\frac{1}{4}[H^{\otimes 4}|0\rangle]|0\rangle^{\otimes 4} = \frac{1}{4}[|0\rangle_4 + |1\rangle_4 + |2\rangle_4 + ... + |15\rangle_4]|0\rangle_4$

Step 2: $\frac{1}{4}$ [$|0\rangle_4|0 \oplus 13^0 (\text{mod } 15)\rangle_4 + |1\rangle_4|0 \oplus 13^1 (\text{mod } 15)\rangle_4 + |2\rangle_4|0 \oplus 13^2 (\text{mod } 15)\rangle_4$

$+ ... + |15\rangle_4|0 \oplus 13^{15} (\text{mod } 15)\rangle_4] = \frac{1}{4}$ [$|0\rangle_4|13^0 (\text{mod } 15)\ \rangle_4 + |1\rangle_4|13^1 (\text{mod } 15)\ \rangle_4 +$

$+ |2\rangle_4|13^2 (\text{mod } 15)\rangle_4 + ... + |15\rangle_4|13^{15} (\text{mod } 15)\rangle_4] = \frac{1}{4}$ [$|0\rangle_4|1 (\text{mod } 15)\rangle_4 +$

$+ |1\rangle_4|13 (\text{mod } 15)\rangle_4 + |2\rangle_4|4 (\text{mod } 15)\rangle_4 + ... + |15\rangle_4|7 (\text{ mod } 15)\rangle_4] =$

$= \frac{1}{4}$ [$|0\rangle_4|1\rangle_4 + |1\rangle_4|13\rangle_4 + |2\rangle_4|4\rangle_4 + |3\rangle_4|7\rangle_4 + |4\rangle_4|1\rangle_4 + |5\rangle_4|13\rangle_4 + |6\rangle_4|4\rangle_4 + |7\rangle_4|7\rangle_4$

$+ |8\rangle_4|1\rangle_4 + |9\rangle_4|13\rangle_4 + |10\rangle_4|4\rangle_4 + |11\rangle_4|7\rangle_4 + |12\rangle_4|1\rangle_4 + |13\rangle_4|13\rangle_4 + |14\rangle_4|4\rangle_4 + |15\rangle_4|7\rangle_4]$

Step 3: Mesurar la w. Per exemple: "7". Si $|w\rangle = |7\rangle_4$, $|x\rangle$ es transforma en:

$|x\rangle|w\rangle = \frac{1}{2}[|3\rangle_4 + |7\rangle_4 + |11\rangle_4 + |15\rangle_4] \otimes |7\rangle_4$ (el $\frac{1}{2}$ ve de 4 combinacions $\frac{1}{\sqrt{4}} = \frac{1}{2}$)

Step 4: Aplicar QFT^T a $|x\rangle$, on ^{T} és la trasposada complexa conjugada

$QFT|x\rangle = |\tilde{x}\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{\frac{2\pi i x y}{N}} |y\rangle$

$QFT^T|\tilde{x}\rangle = |x\rangle \equiv QFT\ |x\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{\frac{-2\pi i x y}{N}} |y\rangle$

$QFT^T|3\rangle_4 = \frac{1}{\sqrt{16}} \sum_{y=0}^{15} e^{\frac{-2\pi i 3 y}{16}} |y\rangle$

$QFT^T|7\rangle_4 = \frac{1}{\sqrt{16}} \sum_{y=0}^{15} e^{\frac{-2\pi i 7 y}{16}} |y\rangle$

$QFT^T|11\rangle_4 = \frac{1}{\sqrt{16}} \sum_{y=0}^{15} e^{\frac{-2\pi i 11 y}{16}} |y\rangle$

$QFT^T|15\rangle_4 = \frac{1}{\sqrt{16}} \sum_{y=0}^{15} e^{\frac{-2\pi i 15 y}{16}} |y\rangle$

$QFT^T|x\rangle = \frac{1}{8} \sum_{y=0}^{15} [e^{\frac{-i\pi 3y}{8}} + e^{\frac{-\pi i 7y}{8}} + e^{\frac{-\pi i 11 y}{8}} + e^{\frac{-\pi i 15 y}{8}}]|y\rangle = \frac{1}{8}[4|0\rangle_4 + 4i|4\rangle_4 - 4|8\rangle_4 - 4i|12\rangle_4]$

$(e^{\frac{-i\pi 3y}{8}} = \cos(\frac{3\pi}{8}) - i \sin(\frac{3\pi}{8}))$

(dels 15 només 4 són diferents a 0, es veu amb un programa)

Step 5: mesurar $|x\rangle$ 0, 4, 8, 12 amb la mateixa probabilitat.

(Els passos que resten és realitzen de forma clàssica.)

Els resultats de la mesura arriben a un pic prop de $j\frac{N}{r}$ per a certa j enter i on el r és el període que estem buscant. Per exemple: $j\frac{16}{r} = 4$ si $j = 1$ i $r = 4$.

r és parell? Sí.

Agafem $x \equiv 13^{\frac{4}{2}} \pmod{15}$, i.e., $x \equiv 13^2 \pmod{15} = 4 \pmod{15}$
Com que $x + 1 \equiv 5 \pmod{15} \neq 0 \pmod{15}$, podem calcular:

$$\text{mcd}(x + 1,\, N) = \text{mcd}\ (5,\, 15) = 5;$$
$$\text{mcd}(x - 1,\, N) = \text{mcd}\ (3,\, 15) = 3;$$

Tenim, doncs, $\{p,\, q\} = \{3,\, 5\}$.

Si calculem $j\frac{16}{r} = 8$, tenim dues opcions: $j = 1$ amb $r = 2$ o $j = 2$ amb $r = 4$ (aquesta última acabem de veure que funciona).

Agafem $x \equiv 13^{\frac{2}{2}} \pmod{15}$, i.e., $x \equiv 13^1 \pmod{15} = 13 \pmod{15}$
Com que $x + 1 \equiv 14 \pmod{15} \neq 0 \pmod{15}$, podem calcular:

$$\text{mcd}(x + 1,\, N) = \text{mcd}\ (14,\, 15) = 1;$$
$$\text{mcd}(x - 1,\, N) = \text{mcd}\ (12,\, 15) = 3;$$

Obtenim una solució parcial $\{1,\, 3\}$

Si calculem $j\frac{16}{r} = 12$, $j = 3$ amb $r = 4$, que ja hem vist que funciona.

Resultat final: L'ordinador quàntic ens diu

- $|0\rangle_4$ hem de tornar-ho a provar, ja que no ens dona informació.
- $|4\rangle_4$ ens dona 3 i 5.
- $|8\rangle_4$ ens dona 3 i 5 o 3, i després podem extreure el 5.
- $|12\rangle_4$ ens dona 3 i 5.

Recordem que $f_{a,N}(x) \equiv a^x \pmod{\text{N}}$

$x = [x_1\ x_2\ x_3\ ...\ x_n] = 2^{n-1}x_1 + 2^{n-2}x_2 + ... + 2^0 x_n$

$f_{a,N}(x) \equiv a^x \pmod{\text{N}} = a^{2^{n-1}x_1 + 2^{n-2}x_2 + ... + 2^0 x_n} = a^{2^{n-1}x_1} \cdot a^{2^{n-2}x_2} \cdot ... \cdot a^{2^0 x_n}$

I això es calcularia amb l'ordinador clàssic. En el circuit fiquem la versió quàntica d'aquesta funció, donada per una matriu unitària $U_{f_{a,N}}$.

8 Conclusions

La computació quàntica sorgeix de la necessitat d'obtenir una tecnologia capaç de processar dades en una velocitat òptima. Gràcies a aquesta es realitzaran càlculs complexos en minuts o hores, quan en la computació clàssica requeririen anys.

En aquest treball hem vist que la computació quàntica parteix de la superposició: representar qualsevol estat com una combinació dels estats base. Amb aquest canvi s'amplia infinitament el ventall de possibilitats que ens brindava la computació clàssica, que només utilitzava els estats purs o estats base.

Hem fet un breu repàs de l'encriptació RSA, recordant que es basa en la "impossibilitat" de factoritzar amb els ordinadors actuals un nombre producte de dos primers molt grans. I hem vist que l'algoritme de Shor, amb la seva capacitat de trobar el període i, per tant, factoritzar, aconseguiria deixar-la obsoleta. Amb aquest fet, cauria gran part de la seguretat que tenim avui en dia, ja que està basada en aquest mètode d'encriptació i amb d'altres de similars, també de clau pública, que es veurien igualment afectats per l'algoritme.

L'algoritme de Peter Shor pot factoritzar N en temps polinòmic, més concretament $\mathrm{O}(\log(N)^3)$. Això significa un enorme increment de velocitat, ja que fins ara es tractava d'un temps exponencial.

Encara no sabem quan existirà un ordinador quàntic amb prou capacitat com per fer caure el sistema de seguretat basat en l'encriptació; alguns preveuen que serà en vàries dècades, mentre que d'altres opinen que serà en pocs anys. Davant d'aquesta previsió, ha sorgit el que se'n diu la "criptografia post-quàntica", que pretén trobar algoritmes que no es vegin compromesos per la computació quàntica o, almenys, a molt més llarg termini.

Per fer-nos una petita idea, per trencar la RSA d'un parell de primers p i q d'uns 100 dígits (en base 10) amb l'algoritme de Shor necessitaríem que l'ordinador quàntic pogués treballar amb aproximadament uns 1000 qubits de forma estable. Actualment, el que disposa de més qubits en condicions òptimes pel funcionament en té 127.

Referències

https://qiskit.org/textbook/

https://qiskit.org/learn/intro-qc-qh/

"Quantum computer science: An Introduction.", N.David Mermin

"Quantum computing from a mathematical perspective: a description of the quantum circuit model", J. Ossorio-Castillo i José M. Tornero

"Lecture Notes on Cryptography" ,Shafi Goldwasser i Mihir Bellare

"Quantum Computing for the Quantum Curious", Ciaran Hughes, Joshua Isaacson, Anastasia Perry, Ranbel F. Sun i Jessica Turner