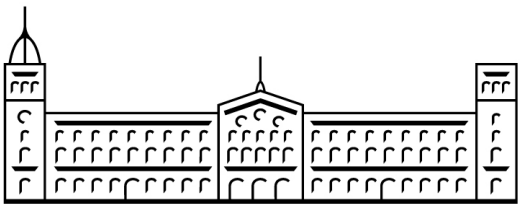


Treball final de grau

GRAU DE MATEMÀTIQUES

Facultat de Matemàtiques i Informàtica
Universitat de Barcelona

Algoritmes quàntics i criptografia

Autor: Cristian Moreno Pulido

Director: Dr. Artur Travesa
Realitzat a: Departament de Matemàtiques i Informàtica

Barcelona, gener de 2017

Abstract

From Shor's algorithm follows an algorithm that would allow us, in a proper quantum computer, to factorize integers effectively. This could break the RSA encryption algorithm. This paper tries to study this algorithm as well as some further variations that improve Shor's algorithm in certain conditions or are able to attack other cryptographic systems.

Resum

De l'algoritme de Shor es desprèn un algoritme que permetria, en un ordinador quàntic adequat, factoritzar nombres enters de manera efectiva. D'aquesta manera es podria trencar l'algoritme criptogràfic RSA. En el treball es tracta d'estudiar aquest algoritme així com també algunes variacions posteriors que milloren el de Shor en certes condicions particulars o que són capaços d'atacar altres sistemes criptogràfics.

Agraïments

M'agradaria agrair en primer lloc al Dr. Travesa per haver acceptat ser el meu tutor. També vull agrair-li totes les hores que m'ha dedicat i la paciència que ha tingut per guiar-me durant tot el treball.
En segon lloc, vull expressar la meva gratitud amb l'Alberto Càmara pels seus consells i la seva ajuda. Per últim, voldria agrair el suport que m'han donat els meus companys i la meva família.

Índex

1 Introducció

Hilbert space is a big place.
-Carlton Caves, físic nord-americà.

El *bit* és una unitat de mesura de la d'informació comunament utilitzada en ciències de la computació i que indica un símbol que pot prendre dos valors diferents. Acostumem a representar aquests valors amb 0 i 1. D'altra banda, amb dos bits, podem representar 4 missatges diferents: $00, 01, 10, 11$. A mesura que anem augmentant el nombre de bits, creix de manera exponencial el nombre de missatges diferents que podem escriure. El bits tambe poden representar les configuracions o estats d'un sistema físic també poden representar bits, per exemple, un circuit elèctric que pot estar tancat o obert representa un bit.
A principis del segle XX va aparèixer un nou paradigma en el camp de la física: la mecànica quàntica. Aquesta nova branca parla de la informació accessible que tenim d'un sistema físic. Sota certes condicions, alguns d'aquests sistemes presenten diverses configuracions, però no es troben en cap d'elles en concret, sinó que es troben en un estat que representa una probabilitat de trobar-se en cadascuna de les configuracions possibles. Per exemple, per descriure l'estat d'un electró en un àtom podríem dir que el 50% de les vegades es trobarà en el nivell fonamental i el 50% de les vegades es trobarà en un nivell excitat. Tot i que el sistema presenti dues configuracions, ja no serà cert que representi un bit. És a dir, algunes entitats físiques tenen una natura intrínsecament probabilística, i diem que són sistemes quàntics.
La computació quàntica és l'estudi de les tasques de processament de dades que es poden aconseguir utilitzant sistemes quàntics. Per tal de realitzar aquest estudi, utilizem el concepte de qubit, que generalitza el concepte de bit mitjançant les lleis de la mecànica quàntica: amb un qubit representem al mateix temps els dos valors 0 i 1. Si bé es cert que existeixen objectes físics que permeten implementar qubits, hi ha tota una sèrie d'inconvenients teòrics i enginyerils que dificulten en gran mesura construir computadors basats en aquest concepte. Actualment, el màxim nombre de qubits amb el qual s'està treballant és 20 (cf. [2]).
En aquest treball expliquem un model de computació quàntica basada en qubits i estudiem diferents algoritmes que poden ser implementats en un ordinador quàntic. En particular, l'algoritme de Shor, un algoritme de factorització amb corbes el·líptiques i un algoritme de càlcul del logaritme discret. Aquests algoritmes són de gran importància en teoria de nombres i també per les seves implicacions en criptografia.

2 Conceptes previs

Notació. Si $z \in \mathbb{C}$, aleshores z^* denotarà el seu nombre complex conjugat.

2.1 Espais de Hilbert

Definició 2.1. Direm que un espai vectorial E sobre $\mathbb{C}$ és *prehilbertià* si està dotat d'un producte hermitià; és a dir, d'una aplicació

$$\begin{array}{rccc} \langle , \rangle : & E \times E & \longrightarrow & \mathbb{C} \\ & (u, v) & \mapsto & \langle u, v \rangle, \end{array}$$

per a la qual se satisfà que

1. $\langle v, w \rangle = \langle w, v \rangle^*, \forall v, w \in E$ i, en particular, $\langle v, v \rangle \in \mathbb{R}$,
2. $\langle v, u + w \rangle = \langle v, u \rangle + \langle v, w \rangle, \forall v, w, u \in E$,
3. $\langle v, \lambda w \rangle = \lambda \langle v, w \rangle, \forall v, w \in E, \forall \lambda \in \mathbb{C}$,
4. $\langle v, v \rangle > 0, \forall v \in E - \{0\}$ i $\langle v, v \rangle = 0$ si, i només si, $v = 0$.

Definició 2.2. Un *espai de Hilbert* H és un espai prehilbertià sobre $\mathbb{C}$ tal que la norma induïda pel producte hermitià defineix una mètrica completa.

Exemple 2.3. L'exemple bàsic d'espai de Hilbert és $\mathbb{C}^n$ amb el producte hermitià

$$\langle (a_1, \ldots, a_n), (b_1, \ldots, b_n) \rangle = \sum_{k=1}^{n} a_k^* b_k.$$

Lema 2.3.1. *Tot subespai vectorial tancat d'un espai de Hilbert té estructura d'espai de Hilbert.*

Ens interessen els vectors de mòdul 1. Si H és un espai de Hilbert de dimensió arbitrària, hi ha una manera senzilla d'associar a un vector $v \in H - \{0\}$ un altre vector $\widehat{v}$ tal que $\|\widehat{v}\| = 1$. Simplement hem de considerar $\widehat{v} = \frac{v}{\|v\|}$. Als vectors de norma 1 els anomenem *normalitzats* o *unitaris*. D'altra banda, és natural considerar el concepte d'ortogonalitat de dos vectors. Si $u, v \in H$, direm que són *ortogonals* si $\langle u, v \rangle = 0$. També podem parlar d'un *conjunt ortogonal de vectors* com un conjunt de vectors de H, $\{v_i\}_{i \in I}$, tal que tots els vectors que hi pertanyen són ortogonals dos a dos. Per últim, diem *conjunt ortonormal de vectors* a un conjunt ortogonal de vectors unitaris.

Sigui H un espai de Hilbert i $\langle , \rangle$ el seu producte hermitià. Si H' denota l'espai dual (lineal) de H, aleshores, per a qualsevol $v \in H$, $f_v := \langle v, \rangle \in H'$.

Definició 2.4. Un operador lineal

$$A : H \longrightarrow H$$

és diu *acotat* si, per a qualsevol $R \in \mathbb{R}^+$, existeix el suprem

$$\sup \{\|A(x)\|_H : x \in H \text{ i } \|x\|_H \leq R\}.$$

Observació 1. Si H és de dimensió finita, aleshores tot operador $A : H \longrightarrow H$ és acotat.

Teorema 2.5. *Sigui*

$$A : H \longrightarrow H$$

un operador lineal acotat. Per a cada $v \in H$, podem considerar una forma lineal donada per

$$\begin{array}{rccc} f_v \circ A : & H & \longrightarrow & \mathbb{C} \\ & x & \mapsto & \langle v, A(x)\rangle, \end{array}$$

que és contínua i li correspon un únic $w_v \in H$ per al qual se satisfà que

$$f_v \circ A(x) = \langle v, A(x)\rangle = \langle w_v, x\rangle$$

per a tot $x \in H$. Aleshores, existeix un únic operador lineal, $A^\dagger$, tal que

$$\begin{array}{rccc} A^\dagger : & H & \longrightarrow & H \\ & v & \mapsto & w_v. \end{array}$$

Direm que $A^\dagger$ és l'operador adjunt de A. Si, a més, $A = A^\dagger$ direm que A és un operador autoadjunt.

Observació 2. De les propietats del producte hermitià s'extreu que $A^\dagger$ és un operador acotat.

Exemple 2.6. Si $H = \mathbb{C}^n$ i, en una certa base, l'operador A en forma matricial ve donat per la matriu $M \in \mathscr{M}_{n\times n}(\mathbb{C})$, aleshores la forma matricial de $A^\dagger$ en la mateixa base ve donada per $M^\dagger := (M^*)^t \in \mathscr{M}_{n\times n}(\mathbb{C})$, la matriu transposada de la seva complexa conjugada.

Definició 2.7. Direm que un operador lineal A és unitari si $AA^\dagger = A^\dagger A = \mathbb{I}$. És fàcil veure que, en aquest cas, els valors propis de A són de mòdul 1.

A continuació, anem a introduir la notació que utilitzarem al llarg del treball. Donat un espai de Hilbert H de dimensió finita, considerem l'aplicació lineal

$$\begin{array}{rccc} f : & H & \longrightarrow & Hom(H, \mathbb{C}) \\ & v & \mapsto & f(v) := f_v, \end{array}$$

on

$$\begin{array}{rccc} f_v : & H & \longrightarrow & \mathbb{C} \\ & w & \mapsto & \langle v, w\rangle. \end{array}$$

Com que $\langle , \rangle$ és no degenerada aleshores f és un antiisomorfisme d'espais vectorials. Aquest fet motiva utilitzar una nova notació proposada originalment per Paul Dirac i que és emprada comunament en física.

Notació. A partir d'ara, donat un vector $\phi \in H$, el denotarem $|\phi\rangle$. Si considerem la imatge de ϕ per f, trobem una forma lineal que denotarem $\langle\phi|$. Fent ús de la notació anterior, si tenim dos vectors $|u\rangle, |v\rangle \in H$ el producte hermitià del primer pel segon es denotarà per $\langle u|v\rangle$, indicant que el producte hermitià és el resultat d'aplicar la forma lineal associada a $|u\rangle$ a $|v\rangle$.

2.2 Producte tensorial

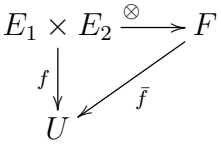
Teorema 2.8. *Siguin E_1, E_2 espais vectorials de dimensió finita sobre un cos $\mathbb{K}$. Aleshores existeix un espai vectorial F de dimensió finita sobre $\mathbb{K}$ i una aplicació bilineal*

$$\otimes : E_1 \times E_2 \longrightarrow F$$

que satisfà la propietat següent: Per a tot espai vectorial U sobre $\mathbb{K}$ i per a tota aplicació bilineal $f : E_1 \times E_2 \longrightarrow U$, aleshores existeix una única aplicació lineal

$$\bar{f} : F \longrightarrow U$$

tal que el diagrama



commuta.

El parell $(F, \otimes)$ és únic, llevat d'un únic isomorfisme, i s'anomena el *producte tensorial* d'espais vectorials E_1 i E_2. Denotem $E_1 \otimes E_2 := F$, i si $(v, w) \in E_1 \times E_2$, aleshores $v \otimes w := \otimes(v, w)$ s'anomena producte tensorial de v i w.

Lema 2.8.1. *Si $\{v_1, \ldots, v_n\}$ és una base de E_1 i $\{w_1, \ldots, w_m\}$ és una base de E_2, aleshores*

$$\{v_i \otimes w_j : 1 \leq i \leq n, 1 \leq j \leq m\}$$

és una base de $E_1 \otimes E_2$.

Corol·lari 2.9. *Si $dim(E_1) = n$ i $dim(E_2) = m$, aleshores $dim(E_1 \otimes E_2) = nm$.*

Exemple 2.10. Siguin $E_1 = E_2 = \mathbb{C}^2$ sobre $\mathbb{C}$, aleshores $dim(\mathbb{C}^2 \otimes \mathbb{C}^2) = 4$. Si $\{e_1, e_2\}$ és una base de E_1 i $\{v_1, v_2\}$ és una base de E_2 aleshores

$$\{e_1 \otimes v_1, e_1 \otimes v_2, e_2 \otimes v_1, e_2 \otimes v_2\}$$

és una base de $\mathbb{C}^2 \otimes \mathbb{C}^2$. I com que és un espai vectorial de dimensió 4 sobre $\mathbb{C}$ se satisfà que $\mathbb{C}^2 \otimes \mathbb{C}^2 \cong \mathbb{C}^4$.

Teorema 2.11. *Siguin E_1, E_2, E_3 espais vectorials de dimensió finita sobre $\mathbb{K}$. Aleshores existeix un únic isomorfisme*

$$f : E_1 \otimes (E_2 \otimes E_3) \longrightarrow (E_1 \otimes E_2) \otimes E_3.$$

tal que $f(x \otimes (y \otimes z)) = (x \otimes y) \otimes x$ per a qualssevol $x \in E_1$, $y \in E_2$ i $z \in E_3$. Per tant, el producte tensorial és associatiu i podem ometre el parèntesis en el producte de diversos espais vectorials de dimensió finita

$$E_1 \otimes \cdots \otimes E_n.$$

Exemple 2.12. Siguin

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix} \in \mathscr{M}_{n\times m}(\mathbb{C})$$

i

$$B = \begin{pmatrix} b_{11} & \cdots & b_{1k} \\ \vdots & \ddots & \vdots \\ b_{s1} & \cdots & b_{sk} \end{pmatrix} \in \mathscr{M}_{s\times k}(\mathbb{C}).$$

El producte tensorial (o producte de Kronecker) de A i B és

$$A \otimes B := \begin{pmatrix} a_{11}B & \cdots & a_{1m}B \\ \vdots & \ddots & \vdots \\ a_{n1}B & \cdots & a_{nm}B \end{pmatrix} \in \mathscr{M}_{ns\times mk}(\mathbb{C}).$$

2.3 Corbes el·líptiques

En aquesta secció, $\mathbb{K}$ denotarà un cos de característica diferent de 2 i de 3.

Notació. Denotarem per $\mathbb{P}^2_{\mathbb{K}}$ l'espai projectiu de dimensió dos sobre $\mathbb{K}$. Escriurem les coordenades projectives d'un punt en una determinada referència com $(x : y : z)$.

Recordem que els *punts afins* d'un espai projectiu sobre un cos són aquells amb tercera coordenada diferent de 0, i que corresponen a la inclusió dels punts del pla afí $\mathbb{A}^2_{\mathbb{K}}$ en $\mathbb{P}^2_{\mathbb{K}}$ amb tercera coordenada igual a 1. Els denotarem donant només les dues primeres coordenades, $(x : y)$. D'altra banda els *punts de l'infinit* són els que tenen tercera coordenada igual a 0.

Definició 2.13. Una *equació de Weierstrass* generalitzada és una equació cúbica sobre $\mathbb{P}^2_{\mathbb{K}}$ de la forma

$$y^2z + a_1xyz + a_3yz^2 = x^3 + a_2x^2z + a_4xz^2 + a_6z^3,$$

on $a_1, a_2, a_3, a_4, a_6 \in \mathbb{K}$ són constants.

Observació 3. Com que $\mathbb{K}$ és un cos de característica diferent de 2 i de 3, aleshores podem fer transformacions lineals invertibles sobre les indeterminades de tal manera que l'equació resultant és

$$Y^2Z = X^3 + AXZ^2 + BZ^3,$$

on $A, B \in \mathbb{K}$ són constants.

Definició 2.14. Una *corba el·líptica* E sobre $\mathbb{K}$ és una corba projectiva de $\mathbb{P}^2_{\mathbb{K}}$ que té per equació projectiva una equació de Weierstrass

$$y^2z = x^3 + Axz^2 + Bz^3,$$

de coeficients $A, B \in \mathbb{K}$, i tal que $-4A^3 - 27B^2 \neq 0$. Notem que l'únic punt de l'infinit de $\mathbb{P}^2_{\mathbb{K}}$ que pertany a la corba és $O := (0 : 1 : 0)$. Denotarem per $E(\mathbb{K})$ al conjunt de punts de la corba el·líptica, és a dir,

$$E(\mathbb{K}) := \{O\} \cup \{(x : y) \in \mathbb{A}^2_{\mathbb{K}} : y^2 = x^3 + Ax + B\}.$$

Observació 4. Imposar que $-(4A^3 + 27B^2) \neq 0$ equival al fet que la corba no tingui punts singulars.

Definim ara una operació en $E(\mathbb{K})$.

Definició 2.15. Suposem que l'equació de la corba afí és $y^2 = x^3 + Ax + B$. Siguin $P_1 = (x_1 : y_1)$, $P_2 = (x_2 : y_2)$ dos punts afins de $E(\mathbb{K})$. Definim la suma dels dos punts $P_1 + P_2 := P_3$ de la següent manera

(1) Si $x_1 \neq x_2$, aleshores definim

$$m := \frac{y_2 - y_1}{x_2 - x_1},$$

$$x_3 := m^2 - x_1 - x_2,$$

$$y_3 := m(x_1 - x_3) - y_1,$$

i posem $P_3 := (x_3 : y_3)$.

(2) Si $x_1 = x_2$ però $y_1 \neq y_2$, aleshores $P_1 + P_2 := O$. Observem que en aquest cas, l'equació de la corba implica que $y_1 = -y_2 \neq 0$.

(3) Si $P_1 = P_2$ i $y_1 \neq 0$, aleshores definim

$$m := \frac{3x_1^2 + A}{2y_1},$$

$$x_3 := m^2 - 2x_1,$$

$$y_3 := m(x_1 - x_3) - y_1,$$

i posem $P_3 := (x_3 : y_3)$.

(4) Si $P_1 = P_2$ i $y_1 = 0$, aleshores $P_1 + P_2 := O$.

(5) El neutre de la suma és O, és a dir, $P_1 + O := P_1$, $O + P_2 := P_2$ i $O + O := O$.

Teorema 2.16. *$(E(\mathbb{K}), +)$ és un grup abelià.*

Exemple 2.17. Sigui $\mathbb{K} = \mathbb{Z}/7\mathbb{Z}$ i la corba de Weierstrass definida per l'equació $y^2 = x^3 - 4x + 2$. Observem que $-(4 \cdot (-4)^3 + 27 \cdot 2^2) = 1 \neq 0$, per tant l'equació defineix una corba el·líptica. Aleshores

$$E(\mathbb{K}) = \{O, (0 : 3), (0 : 4), (2 : 3), (2 : 4),$$

$$(4 : 1), (4 : 6), (5 : 3), (5 : 4)\}.$$

Notació. Si p és un nombre primer, $\mathbb{F}_p := \mathbb{Z}/p\mathbb{Z}$.

Teorema 2.18 (Hasse)**.** *Siguin p un nombre primer diferent de 2 i de 3, i E una corba el·líptica sobre $\mathbb{F}_p$. Aleshores es compleix que*

$$|E(\mathbb{F}_p)| = p + 1 - a_p,$$

per a cert $a_p \in \mathbb{Z}$ tal que $|a_p| < 2\sqrt{p}$.

Demostració. La demostració es pot trobar a la referència [11]. □

Fins ara, hem parlat de corbes el·líptiques sobre un cos. Cal parlar-ne també sobre l'anell $\mathbb{Z}/n\mathbb{Z}$, on n és un nombre enter no necessàriament primer. En la definició 2.15 necessitem multiplicar per inversos, cosa que no és possible, en general, a $\mathbb{Z}/n\mathbb{Z}$ si n és compost degut a que tindrem divisors de 0. Per tant, necessitem adaptar la definició de la suma al nostre cas.

Definició 2.19. D'ara endavant $G(n)$ denotarà el grup multiplicatiu de $\mathbb{Z}/n\mathbb{Z}$.

Definició 2.20. Sigui

$$B_n^k := \{(x_0, \ldots, x_k\} \in (\mathbb{Z}/n\mathbb{Z})^{k+1} : mcd(x_0, x_1, \ldots, x_k, n) \equiv 1 \pmod{n}\}.$$

Definim una relació d'equivalència, $\sim$, sobre B_n^k per $(x_0 : \cdots : x_k) \sim (y_0 : \cdots : y_k)$ si, i només si, existeix $\lambda \in G(n)$ tal que $(x_0, \ldots, x_k) = \lambda(y_0, \ldots, y_k)$. Definirem l'espai projectiu de dimensió k sobre $\mathbb{Z}/n\mathbb{Z}$ com

$$\mathbb{P}_{\mathbb{K}}^k(\mathbb{Z}/n\mathbb{Z}) := B_n^k/\sim,$$

i denotarem la classe d'equivalència de $(x_0, \ldots, x_k)$ per $(x_0 : \cdots : x_k)$.

Classificarem els punts de $\mathbb{P}_{\mathbb{K}}^k(\mathbb{Z}/n\mathbb{Z})$ en tres conjunts disjunts. En primer lloc, als punts $(x_0, \ldots, x_k)$ on x_k és un element invertible de $\mathbb{Z}/n\mathbb{Z}$ els anomenem *punts afins*, i formen un subconjunt que denotem $\mathbb{P}_{\mathbb{K}}^{Af}(\mathbb{Z}/n\mathbb{Z})$. En segon lloc, als punts $(x_0, \ldots, x_k)$ on $x_k \equiv 0 \pmod{n}$ els anomenem *punts de l'infinit*, i formen un subconjunt que és bijectiu amb $\mathbb{P}_{\mathbb{K}}^{k-1}(\mathbb{Z}/n\mathbb{Z})$ per la inclusió

$$\mathbb{P}_{\mathbb{K}}^{k-1}(\mathbb{Z}/n\mathbb{Z}) \hookrightarrow \{(x_0, \ldots, x_{k-1}, 0) \in B_n^k\}.$$

Per últim, els punts $(x_0, \ldots, x_k)$ on x_k és un divisor de 0 els anomenem *punts especials*. En aquest cas, $mcd(x_k, n)$ és un divisor no trivial de n. Denotarem el conjunt de punts especials per $\mathbb{P}_{\mathbb{K}}^{Esp}(\mathbb{Z}/n\mathbb{Z})$. Així,

$$\mathbb{P}_{\mathbb{K}}(\mathbb{Z}/n\mathbb{Z}) = \mathbb{P}_{\mathbb{K}}^{Af}(\mathbb{Z}/n\mathbb{Z}) \cup \mathbb{P}_{\mathbb{K}}^{k-1}(\mathbb{Z}/n\mathbb{Z}) \cup \mathbb{P}_{\mathbb{K}}^{Esp}(\mathbb{Z}/n\mathbb{Z}).$$

Definició 2.21. Sigui n un natural compost tal que $mcd(6, n) = 1$. Una *corba el·líptica* sobre $\mathbb{Z}/n\mathbb{Z}$ és la corba, en coordenades projectives,

$$yz^2 = x^3 + axz^2 + bz^3,$$

on a, b són elements de $\mathbb{Z}/n\mathbb{Z}$ tals que $4a^3 + 27b^2$ és invertible. Denotarem per $E(\mathbb{Z}/n\mathbb{Z})$ al conjunt de punts projectius de la corba el·líptica. És a dir,

$$E(\mathbb{Z}/n\mathbb{Z}) := \{(x : y : z) \in \mathbb{P}_2(\mathbb{Z}/n\mathbb{Z}), yz^2 = x^3 + axz^2 + bz^3\}.$$

A diferència de les corbes el·líptiques sobre cossos, no necessàriament hi haurà un únic punt de l'infinit. En efecte, si imposem que $z = 0$ en l'equació, obtenim que $x^3 \equiv 0 \pmod{n}$ pot tenir una solució diferent de 0. No obstant, en cas que n sigui lliure de quadrats, l'única solució admissible serà $x \equiv 0 \pmod{n}$ i només hi haurà un únic punt de l'infinit, $O := (0 : 1 : 0)$.
En cas que n es tracti d'un nombre primer, $\mathbb{Z}/n\mathbb{Z}$ és un cos i recuperem la definició que hem donat abans de corba el·líptica sobre un cos. D'altra banda, si n és un nombre compost, i p és un divisor primer de n, sempre podem trobar la reducció mòdul p de la corba de Weierstrass que defineix la corba el·líptica.

Definició 2.22. Sigui p un divisor primer de n i E la corba el·líptica donada per l'equació

$$y^2z = x^3 + axz^2 + bz^3,$$

amb $4a^3 + 27b^2$ invertible a $\mathbb{Z}/n\mathbb{Z}$. Aleshores, si a, b són representants de $a', b' \in \mathbb{F}_p$, definim la *reducció mòdul* p de E com la corba el·líptica E_p sobre $\mathbb{F}_p$ donada per l'equació de Weierstrass

$$y^2z = x^3 + a'xz^2 + b'z^3.$$

El fet que $4a^3+27b^2$ sigui invertible a $\mathbb{Z}/n\mathbb{Z}$ garanteix que $4a'^3+27b'^2$ sigui invertible a $\mathbb{F}_p$, de manera que E_p, efectivament, és una corba el·líptica sobre $\mathbb{F}_p$.

Teorema 2.23. *Siguin n un nombre natural lliure de quadrats tal que $mcd(6, n) = 1$ i E una corba el·líptica sobre $\mathbb{Z}/n\mathbb{Z}$. La reducció mòdul p, on p és un divisor primer de n, indueix una aplicació bijectiva*

$$E(\mathbb{Z}/n\mathbb{Z}) \to \prod_{p|n} E_p(\mathbb{F}_p).$$

Demostració. La demostració d'aquest teorema es pot trobar a la referència [1] □

Prèviament hem donat les lleis d'addició per al cas d'un cos a 2.15. Ara procedim a definir unes lleis d'addició sobre la corba el·líptica $E(\mathbb{Z}/n\mathbb{Z})$, on n és un enter lliure de quadrats tal que $mcd(6, n) = 1$, que té per equació

$$y^2z = x^3 + axz^3 + bz^3.$$

Siguin $P_1 = (x_1, y_1, z_1), P_2 = (x_2, y_2, z_2) \in E(\mathbb{Z}/n\mathbb{Z})$. Definim els elements següents de $\mathbb{Z}/n\mathbb{Z}$:

$$L := x_2z_1 - x_1z_2,$$

$$M := 2y_1z_1,$$

$$A := y_2z_1 - y_1z_2,$$

$$B := 3x_1^2 + az_1^2,$$

$$C := 2x_1z_2 + x_2z_1,$$

$$x_3 := \begin{cases} A^2Lz_1z_2 - L^3(x_1z_2 + x_2z_1) & \text{si L } \neq 0. \\ B^2Mz_1z_2 - M^3(x_2z_1 + x_1z_2), & \text{si L=0 i } y_2z_1 + y_1z_2 \neq 0. \\ 0, & \text{si L=0 i } y_2z_1 + y_1z_2 = 0. \end{cases}$$

$$y_3 := \begin{cases} -L^3y_1z_2 - A^3z_1z_2 + AL^2C, & \text{si L } \neq 0. \\ -M^3y_1z_2 - B^3z_1z_2 + BM^2C, & \text{si L=0 i } y_2z_1 + y_1z_2 \neq 0. \\ 1, & \text{si L=0 i } y_2z_1 + y_1z_2 = 0. \end{cases}$$

$$z_3 := \begin{cases} L^3z_1z_2, & \text{si L} \neq 0. \\ M^3z_1z_2, & \text{si L=0 i } y_2z_1 + y_1z_2 \neq 0. \\ 0, & \text{si L=0 i } y_2z_1 + y_1z_2 = 0. \end{cases}$$

Proposició 2.24. *Amb les notacions anteriors,* $(x_3 : y_3 : z_3) \in E(\mathbb{Z}/n\mathbb{Z})$.

Demostració. Les fórmules anteriors provenen d'utilitzar la definició de suma de punts en una reducció $E_p(\mathbb{F}_p)$, on p és un primer que divideix n. En aquest cas, utilitzem les coordenades de $P_1 = (x_1, y_1, z_1), P_2 = (x_2, y_2, z_2)$ com a representants mòdul n de coordenades en $\mathbb{F}_p$, de manera que (x_3, y_3, z_3) mòdul p és un punt de $E_p(\mathbb{F}_p)$, per a tot p divisor primer de n. Pel teorema 2.23, $(x_3 : y_3 : z_3)$ és un punt de $E(\mathbb{Z}/n\mathbb{Z})$. □

Definició 2.25. Definim una operació anomenada suma en $E(\mathbb{Z}/n\mathbb{Z})$ que denotem per $+$. Siguin dos punts $P_1 = (x_1 : y_1 : z_1), P_2 = (x_2 : y_2 : z_2) \in E(\mathbb{Z}/n\mathbb{Z})$. Per definició, la suma dels punts P_1, P_2 és $P_1 + P_2 := (x_3 : y_3 : z_3)$, seguint la notació anterior.

Teorema 2.26. *Amb l'operació que hem definit,* $E(\mathbb{Z}/n\mathbb{Z})$ *és un grup abelià. A més, la reducció mòdul* p,

$$E(\mathbb{Z}/n\mathbb{Z}) \to E_p(\mathbb{Z}/n\mathbb{Z}),$$

és un morfisme de grups abelians.

3 Elements de la computació quàntica

Per tal de poder elaborar un model matemàtic de la computació quàntica, donarem unes quantes definicions i notacions prèvies.

3.1 Qubits i axiomes de la física quàntica

Definició 3.1. Un *qubit* és un espai de Hilbert isomorf a $\mathbb{C}^2$ amb el producte hermitià

$$\langle (a_0, a_1), (b_0, b_1) \rangle = a_0^* b_0 + a_1^* b_1.$$

La base canònica és ortonormal; denotarem els seus vectors com

$$|0\rangle := \begin{pmatrix} 1 \\ 0 \end{pmatrix} , |1\rangle := \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Observació 5. No s'ha de confondre el vector $|0\rangle$, que forma part de la base, amb el vector 0, el neutre de la suma de l'espai de Hilbert.

Definició 3.2. Un *sistema de n qubits*, on $n \in \mathbb{N}$, és un espai de Hilbert isomorf al producte tensorial de n qubits

$$\mathbb{C}^2 \otimes \overset{n)}{\cdots} \otimes \mathbb{C}^2 \cong \mathbb{C}^{2^n},$$

per tant, és un espai de dimensió 2^n sobre $\mathbb{C}$. El seu producte hermitià està definit per

$$\langle (a_0, \ldots, a_{2^n-1}), (b_0, \ldots, b_{2^n-1}) \rangle = \sum_{k=0}^{2^n-1} a_k^* b_k.$$

Notació. Denotarem els elements de la base canònica ortonormal d'aquest espai de Hilbert per l'expressió binària dels nombres de 0 a $2^n - 1$, o pels propis nombres. Si posem els vectors com l'expressió binària, $a_{n-1} \cdots a_0$, on $a_i \in \{0, 1\}$, de nombres entre 0 i $2^n - 1$ queda patent que els vectors de la base canònica ortonormal d'un sistema de n qubits són el producte tensorial dels vectors de la base canònica ortonormal de n còpies d'un sistema d'un únic qubit. A més, amb aquesta notació, si tenim dos vectors $|a_{n-1} \cdots a_0\rangle, |b_{n-1} \cdots b_0\rangle$, amb $a_i, b_i \in \{0, 1\}$ per a $i \in \{0, \ldots, n-1\}$, el seu producte hermitià es pot calcular a partir del producte hermitià d'estats d'un qubit,

$$\langle a_{n-1} \cdots a_0 | b_{n-1} \cdots b_0 \rangle = \langle a_{n-1} | b_{n-1} \rangle \cdots \langle a_0 | b_0 \rangle.$$

Exemple 3.3. Un sistema de dos qubits és un espai de Hilbert de dimensió 4 sobre $\mathbb{C}$. Els elements de la seva base ortonormal són

$$|00\rangle := \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} , |01\rangle := \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} , |10\rangle := \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} , |11\rangle := \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

A continuació, introduirem la nomenclatura necessària per fer un model matemàtic (simple) de les lleis de la física quàntica. Sigui H un sistema de n qubits.

(1) Anomenarem *estat* a qualsevol vector unitari de H.

(2) Als elements de la base canònica ortonormal $B_H = \{|0\rangle, \cdots, |2^n - 1\rangle\}$, els anomenem *estats ben definits*. Qualsevol altre estat $|u\rangle \in H$ es pot escriure com a combinació lineal d'ells, és a dir, si $|u\rangle \in H$, aleshores

$$|u\rangle = \alpha_0|0\rangle + \cdots + \alpha_{2^n-1}|2^n - 1\rangle,$$

amb $\alpha_k \in \mathbb{C}, \forall k \in \{0, \cdots, 2^n - 1\}$. A més a més, com que $|u\rangle$ és unitari se satisfà que

$$|\alpha_0|^2 + \cdots + |\alpha_{2^n-1}|^2 = 1.$$

En general, a un estat que sigui combinació de més d'un estat fonamental se l'anomena *estat de superposició*.

(3) Donat un estat de superposició $|u\rangle = \alpha_0|0\rangle + \cdots + \alpha_{2^n-1}|2^n - 1\rangle$, definim una variable aleatòria discreta X_{B_H}, associada a la base i a l'estat. Aquesta variable té per espai mostral $\Omega = B_H$ (o sigui, els estats ben definits) i segueix una distribució multinomial d'una sola prova, $M(m = 1, |\alpha_0|^2, \cdots, |\alpha_{2^n-1}|^2)$. Per tant, la funció de massa de probabilitat, P, serà

$$P : \Omega \longrightarrow [0, 1]$$

$$|j\rangle \mapsto P(X_{B_H} = |j\rangle) = |\alpha_j|^2.$$

(4) Direm que hem fet una *observació* si realitzem un experiment aleatori i obtenim un valor experimental de la variable aleatòria X_{B_H}. En aquest cas, si la variable aleatòria pren un valor $|k\rangle \in \Omega$, aleshores el sistema de n qubits passarà a estar en l'estat ben definit $|k\rangle$.

(5) Les transformacions unitàries són operadors lineals que actuen en un sistema de n qubits. Als vectors propis d'una transformació unitària els anomenarem *estats propis*.

Exemple 3.4. Si tenim l'estat $|u\rangle = \frac{\sqrt{3}|0\rangle + i|1\rangle}{2}$ d'un sistema d'un qubit i fem una observació respecte de la base canònica, obtindrem l'estat $|0\rangle$ amb probabilitat $|\frac{\sqrt{3}}{2}|^2 = \frac{3}{4}$ o l'estat $|1\rangle$ amb probabilitat $|\frac{i}{2}|^2 = \frac{1}{4}$.

En general, en cas que puguem escriure $|u\rangle = \beta_0|\psi_0\rangle + \cdots \beta_{2^n-1}|\psi_{2^n-1}\rangle$, on $B_\psi = \{|\psi_0\rangle, \cdots, |\psi_{2^n-1}\rangle\}$ és una base ortonormal d'estats, podem definir una variable aleatòria X_ψ, associada a la B_ψ i a l'estat. L'espai mostral serà, en aquest cas, $\Omega = B_\psi$, i la variable aleatòria segueix una distribució multinomial d'una sola prova, $M(m = 1, |\beta_0|^2, \cdots, |\beta_{2^n-1}|^2)$. Per tant, la funció de massa de probabilitat, P, serà

$$P : \Omega \longrightarrow [0, 1]$$

$$|\psi_j\rangle \mapsto P(X_\psi = |\psi_j\rangle) = |\beta_j|^2.$$

Podrem realitzar observacions respecte de la base B_ψ que donin un valor experimental de la variable aleatòria X_ψ. En aquest cas, si la variable aleatòria pren un valor $|k\rangle \in B_\psi$, el sistema passarà a estar en l'estat $|\psi_k\rangle$. No obstant, en cas de no especificar res en contra, suposarem que sempre fem observacions respecte la base canònica, és a dir, treballarem amb la variable aleatòria X_{B_H}.

Exemple 3.5. Si definim els estats $|\pm\rangle := \frac{|0\rangle \pm |1\rangle}{\sqrt{2}}$, que són ortonormals, i tenim l'estat $|u\rangle = \frac{\sqrt{3}|0\rangle + i|1\rangle}{2} = \left(\frac{\sqrt{3}+i}{2\sqrt{2}}\right)|+\rangle + \left(\frac{\sqrt{3}-i}{2\sqrt{2}}\right)|-\rangle$, podem fer una observació respecte de $\{|+\rangle, |-\rangle\}$, de manera que obtindrem l'estat $|+\rangle$ amb probabilitat $|\frac{\sqrt{3}+i}{2\sqrt{2}}|^2 = \frac{1}{2}$ i l'estat $|-\rangle$ amb probabilitat $|\frac{\sqrt{3}-i}{2\sqrt{2}}|^2 = \frac{1}{2}$.

3.2 Computació i portes quàntiques

La computació clàssica es fonamenta en circuits formats per *bits* i *portes lògiques*. Els bits representen símbols que poden ser 0 o 1. Les portes realitzen tasques senzilles sobre els bits. Es poden modelitzar com aplicacions

$$f : (\mathbb{Z}/2\mathbb{Z})^k \longrightarrow (\mathbb{Z}/2\mathbb{Z})^l,$$

que actuen sobre k bits d'entrada i donen com a sortida l bits. Observem que una composició finita de diferents portes lògiques, en un determinat ordre, és una altra porta lògica que denominem circuit. A més a més, es pot demostrar que existeixen conjunts de portes lògiques que fan de generadors de totes les portes lògiques per mitjà de la composició. Aquests conjunts s'anomenen *conjunts complets*.

Exemple 3.6. Podem construir un conjunt complet de portes lògiques a partir de les portes $\{AND, NOT, XOR\}$, on:

1. La porta NOT consisteix en

$$NOT : \mathbb{Z}/2\mathbb{Z} \longrightarrow \mathbb{Z}/2\mathbb{Z}$$

$$x \mapsto x + 1.$$

2. La porta AND consisteix en

$$AND : (\mathbb{Z}/2\mathbb{Z})^2 \longrightarrow \mathbb{Z}/2\mathbb{Z}$$

$$(x, y) \mapsto xy.$$

3. La porta XOR consisteix en

$$XOR : (\mathbb{Z}/2\mathbb{Z})^2 \longrightarrow \mathbb{Z}/2\mathbb{Z}$$

$$(x, y) \mapsto x + y.$$

Per tal de poder construir totes les altres portes lògiques, es poden utilitzar portes que fan còpies d'un bit,

$$COPY : (\mathbb{Z}/2\mathbb{Z}) \longrightarrow (\mathbb{Z}/2\mathbb{Z})^2$$

$$x \mapsto (x, x),$$

portes que intercanvien el valor de dos bits,

$$SWAP : (\mathbb{Z}/2\mathbb{Z})^2 \longrightarrow (\mathbb{Z}/2\mathbb{Z})^2$$

$$(x, y) \mapsto (y, x),$$

i portes que poden crear bits addicionals en la memòria, els qual s'acostumen a anomenar *bits auxiliars*,

$$AUX_i : (\mathbb{Z}/2\mathbb{Z})^0 \longrightarrow (\mathbb{Z}/2\mathbb{Z})$$

$$0 \mapsto i,$$

on $i \in \{0, 1\}$.

La computació quàntica parteix dels mateixos conceptes, però amb diferències importants. Les lleis de la física quàntica imposen que les operacions que podem realitzar sobre un sistema de n qubits, són les transformacions unitàries; per tant, són bijectives. Com que les portes AND, XOR, $COPY$ i AUX_i no són invertibles, no poden ser implementades en un sistema de n qubits. Això demostra que no podem adaptar de manera directa totes les portes lògiques que es fan servir clàssicament.

Definició 3.7. Anomenarem *porta quàntica de n qubits* a una transformació unitària que actua sobre estats de n qubits,

$$U : \mathbb{C}^{2^n} \longrightarrow \mathbb{C}^{2^n}.$$

Les portes quàntiques actuen sobre estats que serveixen d'entrada i que anomenem *registres*. Un *circuit quàntic* és una composició finita d'un seguit de portes quàntiques que efectuen una transformació unitària convenient sobre els registres i donen un estat final que serà la *sortida* de la computació.

Observació 6. L'observació d'un estat de superposició no és una transformació unitària i, per tant, no la pot realitzar cap circuit quàntic. L'experiment aleatori es fa sobre l'estat de sortida del circuit, i consisteix en la lectura d'aquest estat. En conseqüència, la lectura de l'estat de sortida sempre donarà un dels estats ben definits.

L'observació anterior ens diu que per tal de poder rebre informació de la sortida, hem de fer una observació. Un *ordinador quàntic* serà el conjunt format per un circuit quàntic juntament amb els registres, i l'observació necessària per a la lectura de l'estat de sortida. D'ara endavant, també farem servir les paraules *ordinador clàssic* i *computació clàssica* per referir-nos als ordinadors i a la computació derivats de la màquines de Turing.

3.3 Exemples de portes quàntiques

3.3.1 Portes quàntiques simples o d'un qubit

Les *portes quàntiques simples* són transformacions unitàries

$$U : \mathbb{C}^2 \longrightarrow \mathbb{C}^2.$$

Donem alguns exemples de portes simples que són especialment rellevants. Utilitzarem la base $\{|0\rangle, |1\rangle\}$ per indicar les seves formes matricials.

Exemple 3.8. La porta NOT
És l'equivalent quàntic de la porta NOT. La seva forma matricial ve donada per

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

L'acció d'aquesta porta intercanvia els estats ben definits, és a dir $NOT(|0\rangle) = |1\rangle$ i $NOT(|1\rangle) = |0\rangle$. És fàcil comprovar que $NOT = NOT^{-1} = NOT^{\dagger}$.

Per representar portes, s'utilitzen pictogrames simples, on els estats inicials són a la part esquerra del dibuix i els finals a la dreta.

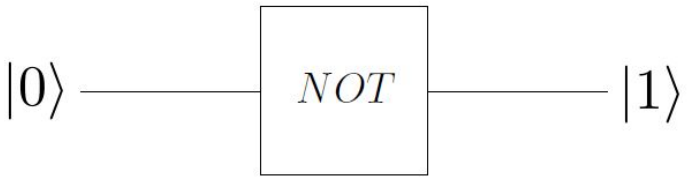


Figura 1: Pictograma de la porta NOT aplicada a un qubit en l'estat $|0\rangle$. Com que només interseca amb una única línia horitzontal, només actua sobre un qubit.

Exemple 3.9. La porta Hadamard

La forma matricial de la porta Hadamard és

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

Observem que $H^2 = \mathbb{I}$, de manera que $H = H^{\dagger} = H^{-1}$. La utilitat d'aquesta porta és la de crear superposició. En aplicar-se sobre els estats ben definits obtenim

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

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

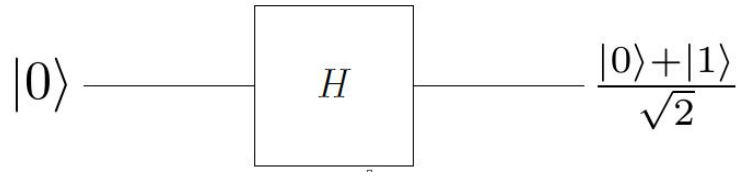


Figura 2: Pictograma de la porta Hadamard aplicada al qubit en l'estat $|0\rangle$.

Exemple 3.10. Porta de desplaçament de fase
La porta de desplaçament de fase es tracta en realitat d'una família uniparamètrica de matrius unitàries. Donat $\delta \in [0, 2\pi)$, la forma matricial de la porta de desplaçament de fase d'angle δ és

$$R_\delta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\delta} \end{pmatrix}.$$

Observem que $R_\delta|0\rangle = |0\rangle$ i $R_\delta|1\rangle = e^{i\delta}|1\rangle$, de manera que no creen superposició en l'aplicar-se sobre els estats ben definits, sinó que els multipliquen per un nombre complex de mòdul 1.

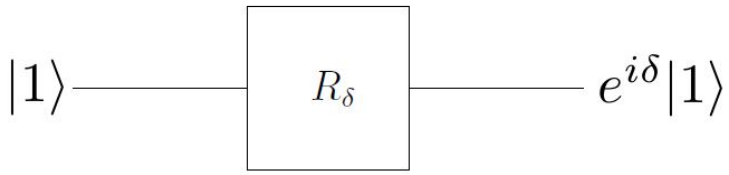


Figura 3: Pictograma de la porta de desplaçament de fase aplicada al qubit $|1\rangle$.

Lema 3.10.1. *Les portes Hadamard i de desplaçament de fase generen* $U(2, \mathbb{C})$.

Demostració. Donada una matriu unitària $M \in U(2, \mathbb{C})$ existeixen quatre paràmetres $\alpha, \beta, \gamma, \delta \in \mathbb{R}$ tals que

$$M_{\alpha,\beta,\gamma,\delta} = exp(i\delta) \begin{pmatrix} cos(\frac{\beta}{2}) & -exp(i\gamma)sin(\frac{\beta}{2}) \\ exp(i\alpha)sin(\frac{\beta}{2}) & exp(i(\alpha+\gamma))cos(\frac{\beta}{2}) \end{pmatrix}.$$

Es comprova fàcilment que

$$R_{\frac{\pi}{2}+\alpha} H R_\beta H R_{-\frac{\pi}{2}+\gamma} = e^{i(-\delta+\frac{\beta}{2})} M,$$

així que només fa falta canviar la fase global, $e^{i(-\delta+\frac{\beta}{2})}$. A partir de la igualtat

$$e^{i\omega}\mathbb{I} = HR_\pi HR_\omega HR_\pi HR_\omega$$

arribem a la igualtat $M = HR_\pi HR_{\delta-\frac{\beta}{2}} HR_\pi HR_{\delta-\frac{\beta}{2}} R_{\frac{\pi}{2}+\alpha} HR_\beta HR_{-\frac{\pi}{2}+\gamma}$. Així qualsevol porta simple és producte de, com a molt, 13 portes Hadamard i de desplaçament de fase. □

3.3.2 Portes quàntiques de dos qubits

Les portes quàntiques de dos qubits són les transformacions unitàries

$$U : \mathbb{C}^4 \longrightarrow \mathbb{C}^4.$$

En aquest cas, hi ha dos conjunts importants: els que es construeixen com a producte tensorial de portes simples i les portes amb un qubit de control. En els exemples utilitzarem la base $|00\rangle, |01\rangle, |10\rangle, |11\rangle$.

Exemple 3.11. Producte tensorial de portes simples Siguin W, V portes simples i $H \cong \mathbb{C}^4$ un sistema de dos qubits. En aquest cas, la imatge del producte tensorial de dos estats $|w\rangle, |v\rangle \in H$ qualssevol pel producte tensorial de les portes és

$$W \otimes V(|w\rangle \otimes |v\rangle) = W(|w\rangle) \otimes V(|v\rangle).$$

Un cas particular seria considerar qualsevol $W \in U(2, \mathbb{C})$ i $V = I$, aleshores $W \otimes V$ és la porta simple que actua sobre el primer qubit d'un sistema de dos qubits i deixa el segon igual.

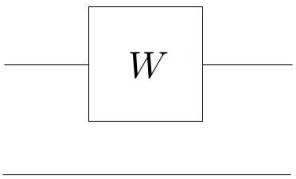


Figura 4: Pictograma de la Porta $W \otimes I$.

Exemple 3.12. Portes controlades

Aquest tipus de porta permet implementar l'operació condicionada "Si es compleix A, aleshores es fa B". Sigui U una porta simple, una porta controlada $C_U^{j,k}$, on $\{j, k\} = \{1, 2\}$, és una porta quàntica de dos qubits associada a U, en la qual el qubit j és de control i l'altre és objectiu. Això vol dir que, en els estats ben definits, la porta simple U actua sobre l'objectiu si, i només si, el qubit de control és $|1\rangle$. Suposem que

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix}.$$

Si el qubit de control és el primer qubit, aleshores la forma matricial de la porta és

$$C_U^{1,2} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{11} & u_{12} \\ 0 & 0 & u_{21} & u_{22} \end{pmatrix}.$$

Si el qubit de control és el segon qubit, aleshores la forma matricial és

$$C_U^{2,1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & u_{11} & 0 & u_{12} \\ 0 & 0 & 1 & 0 \\ 0 & u_{21} & 0 & u_{22} \end{pmatrix}.$$

Un exemple més concret és la porta C_{NOT}. Aquesta porta aplica la porta simple NOT a un dels qubits depenent del qubit de control. Les formes matricials de les portes C_{NOT} són

$$C_{NOT}^{1,2} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, C_{NOT}^{2,1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

Observem que si el qubit de control és $|0\rangle$ l'altre qubit no és modifica, però si és $|1\rangle$ aleshores l'altre canvia de $|0\rangle$ a $|1\rangle$ o de $|1\rangle$ a $|0\rangle$.

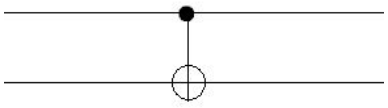


Figura 5: Pictograma de la porta $C_{NOT}^{1,2}$, on el cercle negre indica el qubit de control i el blanc l'objectiu.

Exemple 3.13. Porta Swap
La porta Swap, a la qual anomenarem S, intercanvia el valor dels dos qubits, és a dir, en el estats ben definits ve donada per $S(|jk\rangle) = |kj\rangle$, on $j, k \in \{0, 1\}$. La seva forma matricial, per tant, serà

$$S = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

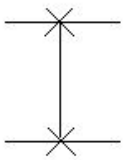


Figura 6: Pictograma de la porta Swap.

Lema 3.13.1. *Algunes propietats de les portes que hem donat com a exemples són*

(1) les portes C_{NOT} i S no són producte tensorial de portes simples,

(2) la porta S es pot posar com a producte de portes C_{NOT}: $S = C_{NOT}^{1,2} C_{NOT}^{2,1} C_{NOT}^{1,2}$,

(3) podem posar la porta C_{NOT}^{21} mitjançant portes Hadamard i la porta C_{NOT}^{12}:

$$C_{NOT}^{2,1} = (H \otimes H) C_{NOT}^{1,2} (H \otimes H).$$

3.3.3 Portes quàntiques de n qubits

En el cas de portes quàntiques de n qubits, es generalitzen els exemples que hem vist a les portes de dos qubits: podem parlar de producte tensorial de portes de k qubits i portes de control associades a matrius de k qubits on hi ha més d'un qubit de control, on $k < n$. En el que segueix, considerem la base $\{|0\rangle, |1\rangle, \ldots, |2^n - 1\rangle\}$.

Exemple 3.14. Porta Toffoli Es tracta d'una porta quàntica de 3 qubits que, sobre els estats ben definits, actua com una porta NOT sobre el tercer qubit si els dos primers qubits es troben en l'estat $|1\rangle$. Per tant, és una porta de control C_{NOT} amb dos qubits de control. La seva forma matricial és

$$Toffoli = \begin{pmatrix} I_{6\times 6} & 0 \\ 0 & NOT \end{pmatrix}$$

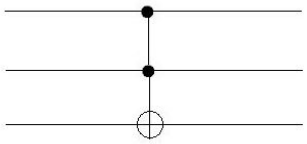


Figura 7: Pictograma de la porta Toffoli.

Observació 7. Hi ha un equivalent clàssic de la porta Toffoli. De fet, es pot demostrar que aquesta porta és un conjunt complet i que, per tant, genera qualsevol circuit de portes lògiques clàssiques.

El que ens proposem ara és mostrar un *conjunt de portes quàntiques universal*. És a dir, un conjunt de portes quàntiques que generen totes les transformacions unitàries per producte tensorial i composició.

Definició 3.15. Una matriu *de dos nivells* és una matriu unitària $d \times d$ (on $d \geq 3$) de coeficients complexos on totes les columnes i files, a excepció de com a molt 2 columnes i 2 files, tenen un únic element de mòdul 1 i la resta 0.

Lema 3.15.1. *Qualsevol matriu unitària $M \in U(d, \mathbb{C})$ es pot posar com a producte de $\frac{d(d-1)}{2}$ matrius de dos nivells.*

Demostració. La demostració es fa per inducció i només en donarem alguns detalls. En cas que $d = 3$, sigui

$$M = \begin{pmatrix} a & d & g \\ b & e & h \\ c & f & j \end{pmatrix}.$$

Si $b = 0$, definim $U_1 := I_{3\times 3}$. En cas contrari, definim

$$U_1 := \begin{pmatrix} \frac{a^*}{\sqrt{|a|^2+|b|^2}} & \frac{b^*}{\sqrt{|a|^2+|b|^2}} & 0 \\ \frac{b}{\sqrt{|a|^2+|b|^2}} & \frac{-a}{\sqrt{|a|^2+|b|^2}} & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

En tots dos casos trobem que la matriu U_1U és de la forma

$$U_1M = \begin{pmatrix} a' & d' & g' \\ 0 & e' & h' \\ c' & f' & j' \end{pmatrix}.$$

Ara, si $c' = 0$, aleshores definim

$$U_2 := \begin{pmatrix} a'^* & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

En cas contrari definim

$$U_2 := \begin{pmatrix} \frac{a'^*}{\sqrt{|a'|^2+|c'|^2}} & 0 & \frac{c'^*}{\sqrt{|a'|^2+|c'|^2}} \\ 0 & 1 & 0 \\ \frac{c'}{\sqrt{|a'|^2+|c'|^2}} & 0 & \frac{-a'}{\sqrt{|a'|^2+|c'|^2}} \end{pmatrix}.$$

En els dos casos, se segueix que

$$U_2U_1M = \begin{pmatrix} 1 & 0 & 0 \\ 0 & e" & h" \\ 0 & f" & j" \end{pmatrix},$$

per a unes altres entrades amb valor que denotem amb ". Finalment, definim $U_3 := (U_2U_1M)^\dagger$. Es verifica que $U_3U_2U_1M = I$, amb U_1, U_2, U_3 de dos nivells i $M = U_1^\dagger U_2^\dagger U_3^\dagger$.
Si suposem que la propietat és certa per a matrius d'ordre més petit que cert d, podem definir $d-1$ matrius de dos nivells $U_1, \ldots, U_{d-1}$, de manera similar al cas amb $d = 3$, tals que

$$U_{d-1} \cdots U_1M = \begin{pmatrix} 1 & 0 \\ 0 & U' \end{pmatrix},$$

on U' és una matriu unitària $(d-1) \times (d-1)$. Utilitzant la hipòtesi d'inducció, U' és el producte de $\frac{(d-1)(d-2)}{2}$ matrius de dos nivells, i per tant M és producte de $(d-1) + \frac{(d-1)(d-2)}{2} = \frac{d(d-1)}{2}$ matrius de dos nivells. □

Corol·lari 3.16. *La construcció anterior indica que en el cas particular en què $d = 2^n$, necessitaríem $2^{n-1}(2^n - 1)$ portes de dos nivells.*

Definició 3.17. Siguin s, t dos nombres naturals. Suposem que tenen com a representacions binàries les successions de n bits

$$s_{n-1}s_{n-2} \cdots s_0,$$

$$t_{n-1}t_{n-2} \cdots t_0,$$

respectivament. Un *codi Gray* entre s i t és una successió finita de successions de n bits que comença en s i acaba en t, on dos termes consecutius de la successió només poden diferir en un únic bit.

Exemple 3.18. Un codi Gray entre 9 i 14 és

$$\begin{array}{cccc} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{array}$$

Proposició 3.19. *Les portes Hadamard, de desplaçament de fase i C_{NOT} permeten construir qualsevol porta de dos nivells en un espai de n qubits qualsevol.*

Demostració. Sigui U una matriu de dos nivells actuant en un sistema de n qubits, tal que les files i columnes s i t, on $0 \leq s < t < n$, són les úniques columnes i files que poden contenir més d'un element no nul. D'aquesta manera, la matriu actua de manera no trivial en, com a molt, dos estats ben definits $|c\rangle$ i $|d\rangle$. Si denotem la entrada de la fila i i la columna j de U per u_{ij}, considerem la matriu unitària

$$U' := \begin{pmatrix} u_{ss} & u_{st} \\ u_{ts} & u_{tt} \end{pmatrix},$$

que actua sobre un sistema d'un únic qubit, i un codi Gray entre c i d que contingui m termes, amb $m \leq n+1$,

$$a_1 := c \to a_2 \to \cdots \to a_{m-1} \to a_m := d.$$

Considerem ara la seqüència de transformacions

$$|a_1\rangle \overset{V_1}{\mapsto} |a_2\rangle \overset{V_2}{\mapsto} \cdots \overset{V_{m-3}}{\mapsto} |a_{m-2}\rangle \overset{V_{m-2}}{\mapsto} |a_{m-1}\rangle,$$

per a certes transformacions unitàries V_i. Els estats implicats en la seqüència es poden escriure com $|a_i\rangle = |a_{i,n-1} \ldots a_{i,j_i} \ldots a_{i,0}\rangle$ i $|a_{i+1}\rangle = |a_{i,n-1} \ldots a'_{i,j_i} \ldots a_{i,0}\rangle$, on $|a_{i,j_i}\rangle, |a'_{i,j_i}\rangle$ són els estats dels qubits en què difereixen. Definim V_i com la transformació unitària que actua sobre un estat ben definit $|b\rangle = |b_{n-1} \ldots b_{j_i} \ldots b_0\rangle$ com

$$V_i(|b\rangle) = |b_{n-1} \ldots NOT^{f(a_i,b)}(b_{j_i}) \ldots b_0\rangle,$$

on

$$f(a_i, b) = \begin{cases} 1, \text{ si } a_{i,k} = b_k, \text{ per a } k \in \{1, \ldots, j_i - 1, j_i + 1, \ldots, n-1\}, \\ 0, \text{ altrament.} \end{cases}$$

En conseqüència, V_i només afecta $|a_i\rangle$ i $|a_{i+1}\rangle$ i deixa la resta invariants. L'efecte de tota la seqüència sobre els estats implicats en la seqüència és

$$|a_1\rangle \mapsto |a_{m-1}\rangle,$$

$$|a_2\rangle \mapsto |a_1\rangle,$$

$$|a_3\rangle \mapsto |a_2\rangle,$$

$$\vdots$$

$$|a_{m-1}\rangle \mapsto |a_{m-2}\rangle.$$

Suposem que els estats $|g_{m-1}\rangle$ i $|g_m\rangle$ difereixen en el qubit que ocupa la posició j_m. Després d'aplicar les transformacions V_i, apliquem la transformació unitària que actua sobre un estat ben definit $|b\rangle = |b_{n-1} \dots b_{j_{m-1}} \dots b_0\rangle$ com

$$|b\rangle \mapsto |b_{n-1} \dots U'^{f(a_{m-1},b)}(b_{i,j_i}) \dots b_0\rangle,$$

on

$$f(a_{m-1}, b) = \begin{cases} 1, \text{ si } a_{m-1,k} = b_k, \text{ per a } k \in \{1, \dots, j_{m-1} - 1, j_{m-1} + 1, \dots, n-1\}, \\ 0, \text{ altrament.} \end{cases}$$

Aquesta transformació aplica la porta U' únicament a l'estat $|a_m\rangle = |d\rangle$ sobre el qubit que ocupa la posició en què $|a_{m-1}\rangle$ i $|a_m\rangle$ difereixen. Finalment desfem les transformacions V_i aplicant-les en ordre invers. El resultat final és la transformació U, per a la qual hem utilitzat com a molt $2(n-1)$ portes quàntiques controlades, les quals es poden realitzar amb $O(n)$ portes simples i C_{NOT}, a més de la porta simple U' que es pot realitzar amb $O(n)$ portes. □

Corol·lari 3.20. *Qualsevol transformació unitària de n qubits es pot construir mitjançant un circuit de portes simples i portes C_{NOT}. Direm que les portes simples juntament amb la porta C_{NOT} formen un sistema universal de portes quàntiques.*

La descripció que hem fet indica que per implementar una transformació unitària qualsevol es necessiten $O(4^n n^2)$ portes simples i C_{NOT}. Aquest mètode no serà de caràcter general si volem obtenir algoritmes òptims, però demostra que la computació quàntica és possible sempre que sapiguem construir físicament un subconjunt de portes quàntiques adient. A diferència d'allò que passa en computació clàssica, tal com plantegem la construcció de circuits quàntics, per a cada problema que volguem resoldre haurem de construir un circuit quàntic diferent amb un nombre finit de portes quàntiques universals. No obstant, això ens allibera del fet d'haver de saber implementar qualsevol porta quàntica universal, ja que només haurem de saber implementar aquelles que siguin necessàries pel problema concret.

3.4 Recursos necessaris i complexitat

Igual que succeeix en els ordinadors clàssics, per poder implementar circuits i emmagatzemar estats necessitarem uns certs recursos i ser capaços de realitzar certes tasques. En concret, suposarem que:

(1) Som capaços de treballar amb qualsevol sistema de n qubits, on n és un nombre natural qualsevol.

(2) Podem implementar totes les portes que necessitem d'un conjunt universal de portes quàntiques.

(3) Podem construir un estat ben definit qualsevol d'un sistema de n qubits, i aconseguir qualsevol altre estat ben definit amb menys de n portes NOT.

(4) Som capaços de fer observacions respecte qualsevol base ortonormal.

D'altra banda, per tal de determinar la dificultat d'un algoritme en un circuit quàntic hem de trobar una manera de quantificar la seva complexitat. En els ordinadors clàssics, aquesta quantificació es fa per mitjà dels recursos que necessita per poder-se implementar: la memòria i el temps que triga l'algoritme a resoldre un determinat problema (o el nombre d'operacions que es realitzen). En el cas d'un ordinador quàntic utilitzarem els mateixos recursos:

(1) la memòria necessària per implementar els registres, mesurada en el nombre de qubits necessaris,

(2) el temps, mesurat en el nombre de portes quàntiques universals utilitzades en l'algoritme.

Definició 3.21. Direm que un algoritme quàntic és *eficient* si el nombre de portes quàntiques universals i la memòria que empra la seva implementació són d'ordre un polinomi en el nombre de qubits del registre.

3.5 Adaptació d'algoritmes clàssics

Una manera de construir algoritmes quàntics és, simplement, considerar el seu anàleg clàssic. El problema és que, en un ordinador quàntic, totes les operacions que es fan sobre els registres són reversibles, és a dir, tenen inversa ja que es construeixen a partir de transformacions unitàries. En canvi, ja hem comentat que molts algoritmes clàssics utilitzen portes lògiques que no són reversibles. No obstant, la computació clàssica reversible està continguda en la computació quàntica, en considerar les matrius unitàries que només tenen com a entrades 0 i 1. En conseqüència, si mostrem una tècnica per tal de poder fer reversibles aquells algoritmes que funcionen amb portes no reversibles demostraríem que tota la computació clàssica pot ser simulada en un ordinador quàntic.
La porta Toffoli clàssica és una porta lògica reversible que constitueix un conjunt complet de portes lògiques clàssiques, de manera que podem construir qualsevol circuit amb aquesta porta actuant sobre els 3 bits que correspongui en cada cas. Aquesta porta actua sobre 3 bits com

$$(A, B, C) \to (A, B, AB \oplus C),$$

on $\oplus$ indica suma mòdul 2.
Per tal de fer una implementació reversible d'un circuit amb l'ajuda de portes Toffoli, podem expressar les operacions $\{AND, NOT, XOR, COPY, SWAP\}$ amb només portes Toffoli i certs bits addicionals.
Per exemple, si partim dels 3 bits $(A, B, 0)$, i apliquem la porta Toffoli, obtenim els tres bits (A, B, AB). És a dir, hem calculat el resultat de fer l'operació AND dels dos primers bits i hem col·locat el resultat en el tercer bit. Per recuperar l'estat $(A, B, 0)$, només hem d'aplicar de nou la porta Toffoli,

$$(A, B, AB) \mapsto (A, B, AB \oplus AB) = (A, B, 0).$$

De manera anàloga, es poden construir les altres portes de manera reversible. El fet de no poder implementar les portes AUX_i és degut al fet que, en el cas de computació reversible, la dimensió de l'espai de sortida i el d'arribada han de ser les mateixes. Per aquest motiu s'ha de suposar que prèviament a fer la computació s'ha preparat un nombre de bits auxiliars suficient com per poder realitzar-la amb portes Toffoli. Per algoritmes de complexitat com a mínim polinòmica, això es pot fer mitjançant un increment de memòria del mateix ordre de complexitat que l'algoritme que s'estigui implementant.
En particular, si l'algoritme és polinòmic, la memòria i el nombre de portes emprades també ho serà.

4 Algoritmes quàntics bàsics

Notació. D'ara endavant farem servir *log* per denotar el logaritme en base 2.

Notació. Denotarem la probabilitat condicionada de A donat B per $P(A;B)$.

4.1 Transformada de Fourier quàntica

La transformada de Fourier discreta, F, és l' endomorfisme lineal de l'espai vectorial $\mathbb{C}^n$ tal que

$$(x_0,\ldots,x_{n-1}) \to F(x_0,\ldots,x_{n-1}) = (y_0,\ldots,y_{n-1}),$$

on $y_k = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} x_j exp(\frac{2\pi ikj}{n})$.

Per a un sistema de n qubits podem definir una transformació unitària molt similar.

Definició 4.1. Siguin H un sistema de n qubits i $q := 2^n$. Definim la *transformada de Fourier quàntica* com la transformació lineal que, sobre els estats ben definits, actua de la manera

$$|j\rangle \overset{F_q}{\mapsto} \frac{1}{\sqrt{q}} \sum_{k=0}^{q-1} exp\left(\frac{2\pi ijk}{q}\right)|k\rangle.$$

Es pot comprovar que, efectivament, aquesta transformació és unitària i per tant pot ser implementada en un circuit quàntic.

Observació 8. La definició de transformada de Fourier quàntica també té sentit per a qualsevol q tal que $0 < q \leq 2^n$. No obstant, la implementació que donarem només serà vàlida per a $q = 2^n$.

Exemple 4.2. La transformada de Fourier quàntica dels estats ben definits de $\mathbb{C}^4$ és

$$F_2(|00\rangle) = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle),$$

$$F_2(|01\rangle) = \frac{1}{2}(|00\rangle + i|01\rangle - |10\rangle - i|11\rangle),$$

$$F_2(|10\rangle) = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle),$$

$$F_2(|11\rangle) = \frac{1}{2}(|00\rangle - i|01\rangle - |10\rangle + i|11\rangle)$$

Teorema 4.3. *És possible implementar la transformada de Fourier quàntica en un sistema de n qubits amb circuits quàntics.*

Demostració. Siguin H un sistema de n qubits i $q := 2^n$, amb $n \geq 3$. Hem de veure que existeix una seqüència finita de transformacions unitàries que poden ser construïdes amb portes de 1 o 2 qubits i que, en aplicar-les a un estat, donen com a resultat la transformada de Fourier quàntica d'aquest estat en H. Per començar,

utilitzarem dos tipus de portes. Una és la porta Hadamard aplicada al qubit $l \in \{0, \ldots, n-1\}$, és a dir,

$$H_l := 1 \otimes \cdots \otimes 1 \otimes \overset{l)}{H} \otimes 1 \otimes \cdots \otimes 1.$$

L'altra porta és la porta de desplaçament de fase controlada, que en un subespai de dos qubits amb el primer qubit de control té la forma

$$C_R^{s,t} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\theta_{s,t}} \end{pmatrix},$$

on $\theta_{s,t} := \frac{\pi}{2^{t-s}}$. Considerem la seqüència de portes lògiques

$$\overline{F}_n = H_0 C_R^{0,1} C_R^{0,2} \cdots C_R^{0,n-2} C_R^{0,n-1} H_1 C_R^{1,2} \cdots C_R^{1,n-2} C_R^{1,n-1} H_2 \cdots$$

$$H_{n-3} C_R^{n-3,n-2} C_R^{n-3,n-1} H_{n-2} C_R^{n-2,n-1} H_{n-1}.$$

Hem utilitzat $n + \frac{n(n-1)}{2} = \frac{n(n+1)}{2} = \frac{n^2+n}{2}$ portes quàntiques en total. La matriu resultant d'aquest producte no es correspon a F_n, sinó que, aplicada a un estat ben definit $|a\rangle \in H$, ens dóna $\frac{1}{\sqrt{q}} \sum_{c=0}^{q-1} exp(\frac{2\pi i a c}{q}) |b_c\rangle$, on b_c denota el nombre que s'obté en llegir les xifres binàries de c en ordre invers. Ho demostrarem per inducció sobre n.

(a) Si $n = 1$, aleshores $\overline{F}_1 = H_0 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$. Si $n = 2$, aleshores

$$\overline{F}_2 = H_0 C_R^{0,1} H_1 = \frac{1}{\sqrt{4}} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & i & -i \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -i & i \end{pmatrix}.$$

Les entrades de les matrius són els productes hermitians d'estats ben definits. Es comprova fàcilment que són les matrius que buscàvem.

(b) Suposem que per a cert $n \geq 2$ se satisfà la tesi.

(c) Ara considerem $\overline{F}_{n+1}$ i els estats ben definits

$$|a\rangle := |a_n \cdots a_0\rangle,$$
$$|b\rangle := |b_n \cdots b_0\rangle,$$
$$|c\rangle := |b_0 \cdots b_n\rangle,$$

on la successió de a_i és la representació binària de a i similarment per a b. Observem que les portes de control de desplaçament de fase únicament multipliquen un estat per un determinat nombre complex de mòdul 1. En canvi, les portes Hadamard modifiquen el estats ben definits, creant superposició. Definim ara

$$|\psi\rangle := \overline{F}_{n+1}(|a\rangle),$$
$$|a'\rangle := |a_{n-1}\cdots a_0\rangle \text{ , amb } a = a_n 2^n + a',$$
$$|b'\rangle := |b_{n-1}\cdots b_0\rangle \text{ , amb } b = b_n 2^n + b',$$
$$|c'\rangle := |b_0\cdots b_{n-1}\rangle \text{ , amb } c = 2c' + b_n.$$

Les portes quàntiques que hem afegit respecte $\overline{F}_n$ per implementar $\overline{F}_{n+1}$ únicament afecten al qubit n de $|a\rangle$, $|a_n\rangle$. D'aquesta manera, podem escriure

$$|\psi\rangle = \overline{F}_{n+1}(|a\rangle) = |\psi_n\rangle \otimes \overline{F}_n(|a'\rangle),$$

on $|\psi_n\rangle$ és un estat d'un sistema d'un qubit. Així,

$$\langle b|\psi\rangle = \langle b_n|\psi_n\rangle\langle b'|\overline{F}_n(|a'\rangle)\rangle = \langle b_n|\psi_n\rangle exp(\frac{2\pi i a'c'}{q}),$$

on en l'última igualtat hem utilitzat la hipòtesi d'inducció. Les portes que afecten al qubit n són, començant a actuar per la dreta, $C_R^{0,n}\cdots C_R^{n-1,n}H_n$. A més, si ens fixem en l'ordre en que es multipliquen les matrius, els qubits de control són els qubits de $|a\rangle$, ja que aquests actuen com a control abans que siguin modificats per una porta Hadamard. D'aquesta manera,

$$|\psi_n\rangle = \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{a_n}\prod_{j=0}^{n-1} exp\left(\frac{i\pi}{2^{n-j}}a_j\right)|1\rangle) =$$

$$\frac{1}{\sqrt{2}}(|0\rangle + exp(i\pi a_n)exp\left(\sum_{j=0}^{n-1} i\frac{\pi}{2^{n-j}}a_j\right)|1\rangle) =$$

$$\frac{1}{\sqrt{2}}(|0\rangle + exp\left(i\pi a_n + \sum_{j=0}^{n-1} i\frac{\pi}{2^{n-j}}a_j\right)|1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + exp\left(\frac{i\pi}{2^n}\sum_{j=0}^{n} a_j 2^j)\right)|1\rangle =$$

$$\frac{1}{\sqrt{2}}(|0\rangle + exp\left(\frac{i\pi}{2^n}a\right)|1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + exp\left(\frac{i\pi}{q}a\right)|1\rangle).$$

En cas que $b_n = 0$, aleshores $a = a'$, $c = 2c'$ i

$$\langle b_n|\psi_n\rangle = \frac{1}{\sqrt{2}}.$$

Per tant,

$$\langle b|\psi\rangle = \frac{1}{\sqrt{2}}\frac{1}{\sqrt{q}}exp\left(\frac{2\pi i a'c'}{q}\right) = \frac{1}{\sqrt{2q}}exp\left(\frac{2\pi i a'c'}{q}\right) = \frac{1}{\sqrt{2^{n+1}}}exp\left(\frac{2\pi i ac}{2^{n+1}}\right),$$

com volíem veure.
Si $b_n = 1$, aleshores $a = a' + qa_n$, $c = 2c' + 1$ i

$$\langle b_n|\psi_n\rangle = \frac{1}{\sqrt{2}}exp\left(\frac{i\pi}{q}a\right).$$

Per tant,

$$\langle b|\psi\rangle = \frac{1}{\sqrt{2}} exp\left(\frac{i\pi}{q}a\right)\frac{1}{\sqrt{q}} exp\left(\frac{2\pi i a'c'}{q}\right) =$$

$$\frac{1}{\sqrt{2q}} exp\left(\frac{i\pi}{q}a + \frac{i\pi(a-a_n q)(c-1)}{q}\right) = \frac{1}{\sqrt{2q}} exp\left(\frac{2\pi iac}{2q} - i\pi a_n(c-1)\right) =$$

$$\frac{1}{\sqrt{2q}} exp\left(\frac{2\pi iac}{2q}\right) = \frac{1}{\sqrt{2^{n+1}}} exp\left(\frac{2\pi iac}{2^{n+1}}\right),$$

on hem utilitzat que $c-1$ és parell. De nou, és el que calia demostrar.

Com hem comentat, aquest seguit de portes quàntiques no es correspon a la transformada de Fourier quàntica. El que resta fer és, o bé llegir els estats en ordre invers, o revertir els estats ben definits amb com a molt $\frac{n}{2}$ portes Swap. □

Finalment, fem un resum de l'algoritme.

Algoritme 1. *Aquest algoritme dóna la transformada de Fourier d'un estat qualsevol. L'input inicial és un estat de superposició d'un sistema de $n \geq 3$ qubits.*

1. *Implementar l'estat inicial al circuit.*
2. *Aplicar la porta Hadamard al qubit $n-1$.*
3. *Aplicar des de $j = n-2$ fins a $j = 0$ el producte de portes*

$$H_j C_R^{j-1,j} \cdots C_R^{j-1,n-1}.$$

4. *Aplicar des de $j = 0$ fins a $j = \frac{n-1}{2} - 1$, si n és senar, i fins $j = \frac{n}{2} - 1$, si j és parell, la porta swap que afecta als qubits j i $n-j-1$.*

En total utilitzem menys de $\frac{n^2+n}{2} + \frac{n}{2}$ portes quàntiques i per tant l'algoritme és $O(n^2)$ en el nombre de portes i $O(n)$ en el nombre de qubits necessaris.

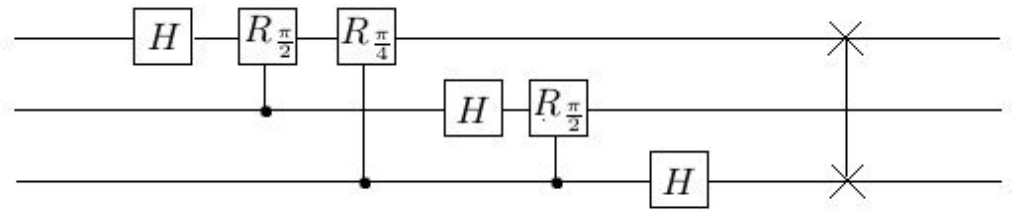


Figura 8: Circuit de la transformada de Fourier quàntica per 3 qubits.

4.2 Algoritme d'estimació de fase

La transformada de Fourier quàntica és a la base de molts altres algoritmes. En concret, mitjançant la transformada de Fourier quàntica trobem un algoritme que ens permet saber, amb certa aproximació, els valors propis d'un operador unitari. Com que aquests valors propis han de tenir mòdul 1 es poden escriure com $e^{2\pi i\phi}$, així, per poder aproximar el valor propi és suficient saber una estimació de $\phi \in [0,1)$, que té representació binària

$$\phi = \sum_{k=1}^{\infty} \phi_k 2^{-k}, \qquad \phi_k \in \{0,1\}.$$

Siguin U una transformació unitària que actua sobre estats d'un sistema de n qubits que anomenem H i $|u\rangle \in H$ un estat propi del qual volem estimar el valor propi. L'algoritme d'estimació de fase utilitza $t+n$ qubits, on els primers t qubits estan inicialitzats a l'estat $|0 \overset{t)}{\cdots} 0\rangle$. El valor de t dependrà de la precisió que volem que tingui la nostra estimació. Els darrers n qubits contenen l'estat propi $|u\rangle$.
Per a construir el circuit quàntic que efectuí l'estimació de fase utilitzarem dos tipus de portes. En primer lloc, les portes Hadamard aplicades al qubit j, que denotem per H_j. Apliquem a cadascun dels primers t qubits una d'aquestes portes. En segon lloc considerarem la transformació U^{2^j}, per a $j \in \mathbb{N}$, aplicada a l'estat $|u\rangle$ i controlada per cadascun dels primers t qubits. Denotem aquesta porta per $C^{k,|u\rangle}_{U^{2^j}}$, on indiquem que apliquem la porta U^{2^j} sobre $|u\rangle$ controlada pel qubit que ocupa la posició k, és a dir, U^{2^j} s'aplica a $|u\rangle$ si, i només si, el qubit k es troba en l'estat $|1\rangle$. Aleshores la seqüència de portes és

$$C^{0,|u\rangle}_{U^{2^{t-1}}} C^{1,|u\rangle}_{U^{2^{t-2}}} \cdots C^{t-2,|u\rangle}_{U^{2^1}} C^{t-1,|u\rangle}_{U^{2^0}} H_{t-1} \cdots H_1 H_0,$$

començant a actuar per la dreta. L'estat dels primers t qubits després d'aquesta transformació és

$$\frac{1}{2^{t/2}}(|0\rangle + exp(2\pi i 2^{t-1}\phi)|1\rangle) \otimes (|0\rangle + exp(2\pi i 2^{t-2}\phi)|1\rangle) \otimes \cdots \otimes (|0\rangle + exp(2\pi i 2^0 \phi)|1\rangle) =$$

$$\frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} exp(2\pi i k\phi)|k\rangle.$$

L'últim pas de l'algoritme és aplicar la transformació inversa de la transformada de Fourier quàntica als primers t qubits.

Lema 4.3.1. *Si ϕ té menys de t xifres binàries, aleshores aquest algoritme ens permet saber exactament la representació binària de ϕ.*

Demostració. Per hipòtesi, $\phi = \phi_1 2^{-1} + \phi_2 2^{-2} + \cdots + \phi_{t-1} 2^{-t}$, on $\phi_j \in \{0,1\}$, $1 \leq j \leq n$. En particular, això vol dir que $2^t\phi \in \{0,1,\ldots,2^t-1\}$.
D'altra banda, és fàcil comprovar que la transformació inversa a la transformada de Fourier quàntica en t qubits és

$$|k\rangle \overset{F_t^{-1}}{\to} \frac{1}{\sqrt{2^t}} \sum_{l=0}^{2^t-1} exp\left(-\frac{2\pi i l k}{2^t}\right)|l\rangle.$$

Si apliquem aquesta transformació a l'estat $\frac{1}{\sqrt{2^t}}\sum_{j=0}^{2^t-1} exp(2\pi ik\phi)|k\rangle$ que ens resulta en fer la primera seqüència de portes quàntiques obtenim

$$F_t^{-1}\left(\frac{1}{\sqrt{2^t}}\sum_{k=0}^{2^t-1} exp(2\pi ik\phi)|k\rangle\right) = \frac{1}{2^t}\sum_{k=0}^{2^t-1} exp(2\pi ik\phi)\sum_{l=0}^{2^t-1} exp\left(-\frac{2\pi ilk}{2^t}\right)|l\rangle =$$

$$\frac{1}{2^t}\sum_{l=0}^{2^t-1}\left(\sum_{k=0}^{2^t-1} exp\left(2\pi ik\frac{2^t\phi - l}{2^t}\right)\right)|l\rangle = \frac{1}{2^t}\sum_{l=0}^{2^t-1} 2^t\delta_{2^t\phi,l}|l\rangle = |2^t\phi\rangle = |\phi_{t-1}\cdots\phi_0\rangle.$$

Com que el resultat és un estat ben definit, l'obtindrem amb probabilitat 1 en fer una observació, i podrem trobar les xifres de la representació binària de ϕ. □

A continuació demostrem que, en cas que ϕ tingui una representació de més de t xifres binàries, l'algoritme ens dóna la millor aproximació de ϕ en t xifres i menor que ϕ. És a dir, el $b \in \{0,\ldots,2^t-1\}$ tal que si definim $\delta := \phi - \frac{b}{2^t}$, aleshores $0 \leq \delta < \frac{1}{2^t}$.
Siguin

$$|\psi\rangle := F_t^{-1}\left(\frac{1}{\sqrt{2^t}}\sum_{k=0}^{2^t-1} exp(2\pi ik\phi)|k\rangle\right) = \frac{1}{2^t}\sum_{l=0}^{2^t-1}\left(\sum_{k=0}^{2^t-1} exp\left(2\pi ik\frac{2^t\phi - l}{2^t}\right)\right)|l\rangle$$

i

$$\alpha_l := \langle b+l \pmod{2^t}|\psi\rangle = \frac{1}{2^t}\sum_{k=0}^{2^t-1} exp\left(2\pi ik\frac{2^t\phi-(b+l)}{2^t}\right) =$$

$$\frac{1}{2^t}\frac{1-exp(2\pi i(2^t\phi-(b+l))}{1-exp\left(2\pi i\left(\frac{2^t\phi-(b+l)}{2^t}\right)\right)} = \frac{1}{2^t}\frac{1-exp(2\pi i(2^t\delta-l))}{1-exp\left(2\pi i\left(\delta-\frac{l}{2^t}\right)\right)},$$

on $l \in \{-(2^{t-1}-1),\ldots,2^{t-1}\}$. En el càlcul de α_l, hem utilitzat que $\frac{2^t\phi-(b+l)}{2^t} \notin \mathbb{Z}$ per calcular la progressió geomètrica.

Si fem una observació sobre l'estat $|\psi\rangle$ obtindrem un estat $|m\rangle$ ben definit. Per tal de trobar una estimació és convenient que la probabilitat d'observar un valor proper a b sigui alta. Per aquest motiu estudiarem el valor de $P(|m-b|>e)$, on e és un enter que caracteritza la tolerància que desitgem. La probabilitat d'observar un estat $|m\rangle$ amb aquestes característiques és

$$P(|m-b|>e) = \sum_{l=-2^{t-1}+1}^{-(e+1)} |\alpha_l|^2 + \sum_{l=e+1}^{2^{t-1}} |\alpha_l|^2.$$

Com que $|1-exp(i\theta)| \leq 2$ per a qualsevol $\theta \in \mathbb{R}$, tenim que

$$|\alpha_l| \leq \frac{2}{2^t|1-exp(2\pi i(\delta-\frac{l}{2^t}))|}.$$

D'altra banda, $|1 - exp(i\theta)| \geq \frac{2|\theta|}{\pi}$, si $-\pi \leq \theta \leq \pi$. Aleshores

$$|\alpha_l| \leq \frac{1}{2^{t+1}(\delta - \frac{l}{2^t})},$$

en ser $-\pi \leq 2\pi(\delta - \frac{l}{2^t}) \leq \pi$, si $l \in \{-(2^{t-1}-1), \ldots, 2^{t-1}\}$. Finalment, podem acotar $P(|m-b| > e)$,

$$P(|m-b| > e) \leq \sum_{-2^{t-1}+1}^{-(e+1)} \frac{1}{4(l-2^t\delta)^2} + \sum_{l=e+1}^{2^t-1} \frac{1}{4(l-2^t\delta)^2},$$

i com que $0 \leq 2^t\delta \leq 1$,

$$p(|m-b| > e) \leq \frac{1}{4}\left[\sum_{l=-2^{t-1}+1}^{-(e+1)} \frac{1}{l^2} + \sum_{l=e+1}^{2^t-1} \frac{1}{(l-1)^2}\right] \leq$$

$$\frac{1}{2}\sum_{l=e}^{2^{t-1}-1} \frac{1}{l^2} \leq \frac{1}{2}\int_{e-1}^{2^{t-1}-1} \frac{dl}{l^2} \leq \frac{1}{2(e-1)}.$$

Si volem estimar ϕ amb un error menor que 2^{-s} i probabilitat més gran que $1-\epsilon$, per a cert ϵ petit, escollim $e = 2^{t-s} - 1$, i $t := s + p$, on $p := \lceil log_2\left(2 + \frac{1}{2\epsilon}\right)\rceil$. Aleshores,

$$P(|m-b| \leq e) \geq 1 - \frac{1}{2(2^p-2)} \geq 1 - \epsilon.$$

A més, els m que satisfan la condició anterior compleixen que

$$|\frac{m}{2^t} - \phi| \leq |\frac{m-b}{2^t}| + |\frac{b}{2^t} - \phi| \leq \frac{e}{2^t} + \delta \leq \frac{e+1}{2^t} = \frac{1}{2^s},$$

de manera que en observar $|m\rangle$, obtindrem una aproximació de ϕ amb un error menor que 2^{-s}.

Ara que hem vist que l'algoritme té probabilitat prou alta d'encert, procedim a fer-ne un resum.

Algoritme 2. *Suposarem que som capaços d'implementar portes U^{2^j} controlades per qualsevol qubit i que sabem preparar un estat propi $|u\rangle$ de U amb valor propi $e^{2\pi i\phi}$. Aquest algoritme ens permet trobar una aproximació de ϕ amb un error menor que 2^{-s}, per a cert $s \in \mathbb{N}$, amb probabilitat més gran que $1-\epsilon$, per a cert ϵ petit escollit prèviament. L'input inicial és l'estat producte tensorial de de $t = s + \lceil log\left(2 + \frac{1}{2\epsilon}\right)\rceil$ qubits inicialitzats a $|0\rangle$ i de n qubits inicialitzats a l'estat $|u\rangle$.*

1. Implementar l'estat inicial al circuit.

2. *Aplicar des de* $j = 0$ *fins a* $j = t - 1$ *una porta Hadamard al qubit* j.

$$|0\rangle \otimes |u\rangle \mapsto \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle \otimes |u\rangle$$

3. *Aplicar des de* $j = 0$ *fins a* $j = t - 1$ *la porta* $C_{U^{2^j}}^{k,|u\rangle}$, *on* $k = t - 1 - j$.

$$\frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle \otimes |u\rangle \mapsto \frac{1}{2^t} \sum_{j=0}^{2^t-1} exp(2\pi i j\phi)|j\rangle \otimes |u\rangle$$

4. *Aplicar la inversa de la transformada de Fourier als primers* t *qubits.*

$$\frac{1}{2^t} \sum_{j=0}^{2^t-1} exp(2\pi i j\phi)|j\rangle \otimes |u\rangle \mapsto \frac{1}{2^t} \sum_{l=0}^{2^t-1} \left(\sum_{k=0}^{2^t-1} exp\left(2\pi i k \frac{2^t\phi - l}{2^t}\right) \right) |l\rangle \otimes |u\rangle$$

5. *Fer una observació del primer registre de l'estat final i obtenir un estat ben definit* $|m\rangle$. *Trobem una aproximació de* ϕ *en calcular* $\frac{m}{2^t}$.

En total, utilitzem n portes Hadamard, n portes U^{2^j} controlades, i la transformada de Fourier, que empra $O(n^2)$ portes universals. La complexitat dependrà de quantes portes universals que necessitem per implementar les portes U^{2^j} controlades.

La complexitat dependrà de quina sigui la transformació unitària U de la qual volem saber el valor propi.

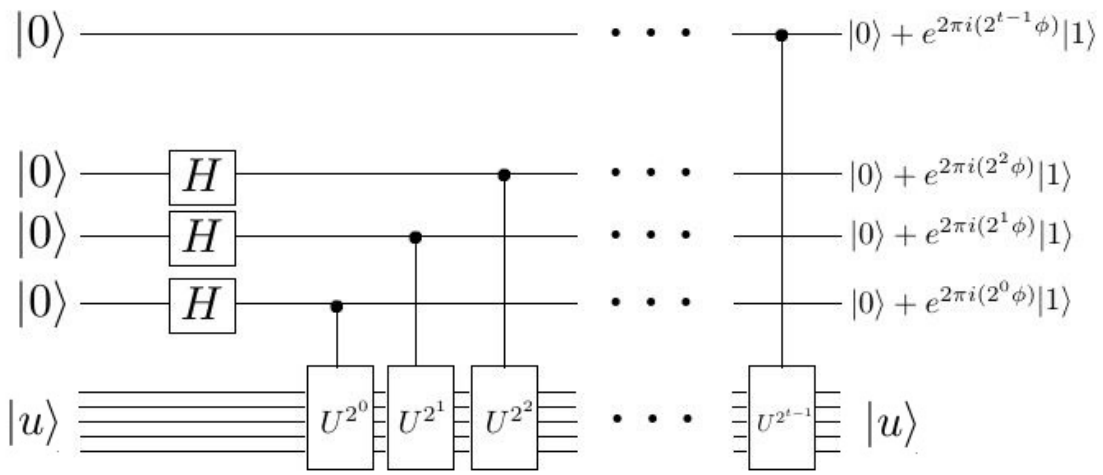


Figura 9: Circuit de l'algoritme d'estimació de fase.

4.3 Algoritme de cerca d'ordre

Siguin G un grup abelià finit amb notació multiplicativa i e el seu neutre. Sempre ens podem plantejar quin és l'ordre d'un element g que no sigui el neutre, és a dir $min\{k \in \mathbb{Z}_+ : g^k = e\}$. Podem utilitzar l'algoritme d'estimació de fase per trobar un algoritme que trobi l'ordre de g. Suposem que cada element del grup pot ser codificat com un estat ben definit d'un sistema de n qubits, que denotem H, on n dependrà del grup G i de la codificació. D'aquesta manera, denotarem la codificació de $h \in G$ per $|h\rangle$. L'operació interna del grup es modelitza per la següent transformació unitària:

$$|h\rangle \overset{U_g}{\to} |hg\rangle,$$

on $h, g \in G$. Aquesta transformació és unitària, degut a l'existència de l'element g^{-1}, i la seva construcció depèn del grup i de la manera que hem codificat; més endavant estudiarem algun exemple concret. Si denotem per r l'ordre de l'element g, podem definir

$$|u_s\rangle := \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} exp\left(\frac{-2\pi isk}{r}\right) |g^k\rangle.$$

Aquest estat és propi de U_g ja que, per a tot $s \in \{0, 1, \ldots, r-1\}$,

$$U_g(|u_s\rangle) = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} exp\left(\frac{-2\pi isk}{r}\right) |g^k\rangle = exp\left(\frac{2\pi is}{r}\right) |u_s\rangle.$$

A més a més els estats $|u_s\rangle, |u_l\rangle$ són ortogonals si $s \neq l$. En efecte,

$$\langle u_s|u_l\rangle = \frac{1}{r} \sum_{k_1=0}^{r-1} \sum_{k_2=0}^{r-1} exp\left(\frac{2\pi isk_1}{r}\right) exp\left(\frac{-2\pi ilk_2}{r}\right) \langle k_1|k_2\rangle =$$

$$\frac{1}{r} \sum_{k_1=0}^{r-1} \sum_{k_2=0}^{r-1} exp\left(\frac{2\pi i(sk_1 - lk_2)}{r}\right) \delta_{k_1,k_2} = \frac{1}{r} \sum_{k_2=0}^{r-1} exp\left(\frac{2\pi i(s-l)k_2)}{r}\right) = \frac{1}{r} r\delta_{s,l} = \delta_{s,l}.$$

Si apliquéssim l'algoritme d'estimació de fase a l'operador U_g i a l'estat propi $|u_s\rangle$, per a cert s, podríem trobar una aproximació de $\frac{s}{r}$, a partir de la qual podríem trobar r. Per poder implementar l'algoritme d'estimació de fase necessitem implementar les portes $U_g^{2^j}$ mitjançant l'algoritme d'exponenciació binària que trobem a l'apèndix. Un problema addicional és que per construir els estats $|u_s\rangle$ sembla que és necessari conèixer r, el qual és precisament el valor que busquem. Podem evitar el problema sabent que

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle = \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} \sum_{k=0}^{r-1} exp\left(\frac{-2\pi isk}{r}\right) |h^k\rangle = \frac{1}{r} \sum_{k=0}^{r-1} r\delta_{0,k}|h^k\rangle = |e\rangle,$$

$$U_g(|e\rangle) = U_g\left(\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle\right) = \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} exp\left(\frac{2\pi is}{r}\right) |u_s\rangle.$$

Com que $|e\rangle$ és un estat ben definit, sabem com construir-lo. A continuació, podem preparar un registre amb $2n+1+\lceil log\left(2+\frac{1}{2\epsilon}\right)\rceil$ qubits en l'estat $|0\rangle$ i un segon registre de n qubits en l'estat $|e\rangle$ i aplicar l'algoritme d'estimació de fase. En aquest cas obtindrem, amb probabilitat $1-\epsilon$,

$$|0\rangle \otimes |e\rangle = |0\rangle \otimes \left(\frac{1}{\sqrt{r}}\sum_{s=0}^{r-1}|u_s\rangle\right) \mapsto \frac{1}{\sqrt{r}}\sum_{s=0}^{r-1}|\phi_s\rangle \otimes |u_s\rangle,$$

on ϕ_s és una estimació de $\frac{s}{r}$ amb error menor que $2^{-(2n+1)}$. Si fem una observació d'aquest estat respecte d'una base que contingui al conjunt $\{|u_0\rangle, \ldots, |u_{r-1}\rangle\}$, obtindrem l'estat $|\phi_s\rangle \otimes |u_s\rangle$ amb probabilitat $\frac{1}{r}$. Així, la probabilitat que, donat un k, $1 \leq k \leq r-1$, l'algoritme ens proporcioni l'estimació de $\frac{k}{r}$ és $\frac{1-\epsilon}{r}$.

Recordem que $r \leq \#G$, on $\#G < 2^n$ és l'ordre del grup. Un cop apliquem l'algoritme d'estimació de fase, el resultat que proporciona l'ordinador quàntic en fer una observació és una aproximació de $\frac{s}{r}$, però encara hem de trobar el valor de r. Per tal d'aconseguir trobar una solució del problema haurem d'afegir una subrutina, que podem implementar en un ordinador clàssic, que utilitzi la informació obtinguda de la sortida de l'ordinador quàntic.
Una possible manera de calcular r és trobar la fracció racional de dos nombres naturals que aproximi suficientment ϕ i identificar r a partir d'aquí. Per tal de trobar aquesta fracció utilitzarem l'algoritme de fraccions continuades (cf. [3] pàg.21), sobre les quals parlem una mica més a l'apèndix. Per tal de donar sentit al càlcul de convergents, hem de demostrar el següent lema.

Lema 4.3.2. *(a) Si suposem que l'algoritme d'estimació de fase ha tingut èxit i obtenim ϕ que aproxima $\frac{s}{r}$, se satisfà que $|\frac{s}{r}-\phi| \leq \frac{1}{2r^2}$.*

(b) $\frac{s}{r}$ és un convergent de la fracció continuada de ϕ.

Demostració. (a) Hem vist que l'error comès per l'algoritme d'estimació de fase està acotat superiorment,

$$\left|\frac{s}{r}-\phi\right| \leq \frac{1}{2^{2n+1}}.$$

Com que $2^{2n+1} = 2\cdot 2^{2n} \geq 2(\#G)^2 \geq 2r^2$, aleshores $|\frac{s}{r}-\phi| \leq \frac{1}{2r^2}$.

(b) Com que representem ϕ amb un nombre finit de qubits es tracta d'un nombre racional. Per tant, té una fracció continuada finita. En cas que $\phi = \frac{s}{r}$ és evident que $\frac{s}{r}$ és un convergent. En cas contrari, siguin $\frac{s}{r} = [q_0; q_1, \ldots q_m]$, la fracció continuada de $\frac{s}{r}$, i b_m, c_m, definits com en la proposició anterior per al cas en que $x = \frac{s}{r}$, amb el que $\frac{b_m}{c_m} = \frac{s}{r}$. Definim ara

$$\delta := 2c_m(c_m\phi - b_m),$$

$$\lambda := 2\left(\frac{c_m b_{m-1} - b_m c_{m-1}}{\delta}\right) - \frac{c_{m-1}}{c_m} = 2\frac{(-1)^m}{\delta} - \frac{c_{m-1}}{c_m}.$$

Com que $mcd(b_m, c_m) = 1$, aleshores se satisfà que $b_m \leq s$ i $c_m \leq r$. En conseqüència, $0 < \delta \leq 1$. De la definició de λ, podem trobar la igualtat següent

$$\phi = \frac{\lambda b_m + b_{m-1}}{\lambda c_m + c_{m-1}},$$

de manera que $\phi = [a_0, \ldots, a_n, \lambda]$. Si escollim que la fracció continuada de $\frac{s}{r}$ tingui un nombre parell de termes (cf. l'observació 12), aleshores $\lambda = \frac{2}{\delta} - \frac{c_{m-1}}{c_m}$. Com que la successió $\{c_n\}_n$ és creixent, aleshores $\lambda = \frac{2}{\delta} - \frac{c_{m-1}}{c_m} \geq 2 - 1 = 1$. Hem demostrat que λ és un racional més gran o igual que 1, per tant té associada una fracció continuada $[\lambda_0; \ldots, \lambda_p]$. Se segueix que

$$\phi = [q_0; \ldots, q_m, \lambda_0, \ldots, \lambda_p],$$

i, per tant, ϕ té a $\frac{s}{r}$ com a convergent.

□

Com que $\frac{s}{r}$ és un convergent de ϕ, podem trobar una fracció equivalent a $\frac{s}{r}$ mitjançant l'algoritme de fraccions continuades. Això ens podria permetre trobar r, sempre que s no sigui igual a 0. Per tant, podrem procedir al càlcul de convergents amb aquest algoritme si, en fer una observació, trobem un estat amb valor propi $\frac{s}{r}$, amb $0 < s < r - 1$ i, a més, l'algoritme d'estimació de fase ens dóna una estimació de $\frac{s}{r}$. La probabilitat que aquests dos fets succeeixin simultàniament és

$$\sum_{s=1}^{r-1} \frac{1-\epsilon}{r} = (1-\epsilon)(1-r^{-1}). \tag{4.1}$$

Si suposem que $r \geq 2$, això vol dir que la probabilitat és més gran que $\frac{1-\epsilon}{2}$. En cas que l'algoritme de cerca d'ordre falli, o trobem que $s = 0$, haurem de tornar a preparar l'estat i aplicar de nou l'estimació d'ordre fins a obtenir una estimació de $\frac{s}{r}$, on $s \neq 0$.

Algoritme 3. *Siguin $(G, \cdot)$ i e el seu neutre. Suposem que tenim una manera de codificar els elements de G amb n qubits. El registre consistirà en*

$$t = 2n + 1 + \lceil log\left(2 + \frac{1}{2\epsilon}\right) \rceil$$

qubits inicialitzats a $|0\rangle$ i n qubits inicialitzats a l'estat $|e\rangle$. Suposem també que som capaços d'implementar l'operació

$$|j\rangle \otimes |h\rangle \overset{V}{\mapsto} |j\rangle \otimes |g^j h\rangle,$$

per a $g \in G$ i $j \in \mathbb{N}$. Aleshores, aquest algoritme retorna l'ordre r de g.

1. *Implementem l'estat inicial $|0\rangle \otimes |e\rangle$.*
2. *Apliquem als primers t qubits des de $j = 0$ fins a $j = t - 1$ la porta Hadamard H_j.*

$$|0\rangle \otimes |e\rangle \mapsto \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle \otimes |e\rangle.$$

3. *Apliquem V,*

$$\frac{1}{\sqrt{2^t}}\sum_{j=0}^{2^t-1}|j\rangle\otimes|e\rangle\mapsto\frac{1}{\sqrt{2^t}}|j\rangle\otimes|g^j\rangle\approx$$

$$\frac{1}{2^t}\sum_{s=0}^{r-1}\sum_{j=0}^{2^t-1}exp\left(2\pi i\frac{sj}{r}\right)|j\rangle\otimes|u_s\rangle.$$

4. *Apliquem la inversa de la transformada de Fourier*

$$\frac{1}{2^t}\sum_{s=0}^{r-1}\sum_{j=0}^{2^t-1}exp\left(2\pi i\frac{sj}{r}\right)|j\rangle\otimes|u_s\rangle\mapsto\frac{1}{r}\sum_{s=0}^{r-1}|\phi_s\rangle\otimes|u_s\rangle,$$

   on ϕ_s és una estimació de $\frac{s}{r}$.

5. *Si mesurem l'estat $|e\rangle$ tornem al pas 1; si no, apliquem el mètode de fracciones continuades.*

En el pas 3, el símbol d'aproximació és degut a que hem reagrupat termes, en ser $V(|r+d\rangle\otimes|g^k\rangle)=|g^{k+d}\rangle$. Només tindrem la igualtat en cas que $r\nmid 2^t$. No obstant, si 2^t és prou gran comparat amb r, l'aproximació és vàlida. Això ocorrerà sovint, ja que $r<\#G<2^n$ i $t>2n$, així 2^t tindrà més del doble de xifres binàries que r. D'altra banda, s'ha de comentar que el càlcul de convergents mitjançant l'algoritme de fraccions continuades té complexitat $O(n^3)$ si s i r tenen n xifres binàries. En aquest cas, la sortida de l'algoritme retorna s' i r' tals que $\frac{s}{r}=\frac{s'}{r'}$, on $\frac{s'}{r'}$ és una fracció irreductible. D'aquesta manera r' és un candidat a l'ordre de g. Si $g^{r'}=e$ ja hem acabat. Hauríem d'afegir al pas 5, que retorni al pas 1 si $g^{r'}\neq e$. Com que la quantitat de nombres primers menors que r és més gran que

$$\frac{r}{2log(r)},$$

per a r suficientment gran, en aplicar tot l'algoritme $2log(r)$ vegades, sense tenir en compte (4.1) trobaríem un s coprimer amb r amb alta probabilitat, de manera que l'algoritme de fraccions continuades ens donaria directament el valor de r.

5 Aplicacions a la Criptografia

5.1 Dos sistemes criptogràfics importants

A continuació comentarem dos algoritmes de xifra-desxifrat d'especial importància: RSA i ElGamal.

5.1.1 RSA

L'algoritme de RSA és una seqüència de passos que permet intercanviar missatges de forma privada entre duess parts, A i B. Aquest algoritme treballa sobre el grup multiplicatiu $G(n)$, per a cert enter n compost lliure de quadrats.

Definició 5.1. La *clau pública* és un parell (n, e). En primer lloc, n és un enter compost positiu que descompon com a producte de r primers diferents com $n = p_1 p_2 \dots p_r$, amb $r \geq 2$. Definim també $\lambda := mcm(p_1 - 1, p_2 - 1, \dots, p_r - 1)$. D'altra banda, e, és un enter positiu tal que $1 \leq e \leq n - 1$, que anomenem *exponent públic* i que satisfà que $mcd(e, \lambda) = 1$. La clau pública es creada per la part que rep el missatge, que anomenem B, i es rebuda per la part que envia el missatge, que anomenem A.

Suposarem que, donat l'enter n, es pot codificar el missatge d'una manera estàndard i coneguda per a totes dues parts amb un enter m tal que $0 \leq m \leq n - 1$.

Definició 5.2. La *clau privada* és el parell (n, d), on n és el mateix nombre definit anteriorment i d és un enter positiu tal que $ed \equiv 1 \pmod{\lambda}$. La clau privada és coneguda de manera exclusiva per B.

Algoritme 4 (RSA). *(1) B construeix la clau pública* (n, e).

(2) B envia la clau pública per un canal de comunicació a A.

(3) A obté la clau pública i codifica el missatge com un enter m *tal que* $0 \leq m \leq n-1$ *seguint un procediment estàndard i conegut.*

(4) A xifra el missatge calculant $c \equiv m^e \pmod{n}$.

(5) A envia el missatge xifrat c *per un canal de comunicació a B.*

(6) B desxifra el missatge mitjançant la clau privada, utilitzant que $m \equiv c^d \pmod{n}$.

La seguretat de l'algoritme, si suposem que ni la implementació ni la codificació son susceptibles a cap tipus d'atac, es basa en el fet que per poder calcular d és necessari el càlcul dels nombres primers que divideixen n. Donat que no es coneixen algoritmes que factoritzin nombres arbitraris en temps polinòmic, una tria adient de l'enter n impossibilita el desxiframent efectiu del missatge. Tot i que el nombre enter n pot tenir més de dos factors primers, si fixem el nombre de bits que ocupa aquest n, és preferible tenir-ne només dos, ja que així la longitud en nombre de bits de cadascun d'aquests primers serà més gran i, en conseqüència, serà més difícil trobar un factor de n.

5.1.2 ElGamal

Aquest algoritme, igual que el RSA, permet intercanviar missatges entre dues parts A i B. Aquest algoritme pot treballa amb qualsevol grup finit; en aquesta explicació donarem notació multiplicativa.

Definició 5.3. La *clau pública* consisteix en (G, p, α, y_B), on G és un grup, un element $\alpha \in G$ amb ordre un nombre primer p i $y_B = \alpha^{x_B}$, on x_B és un enter que forma part de la clau privada. La clau pública és creada per la part que rep el missatge, que anomenem B, i és rebuda per la part que envia el missatge, que anomenem A.

Suposem que, donat el grup G, es pot codificar el missatge d'una manera estàndard i coneguda per les dues parts amb un element $m \in G$.

Definició 5.4. La clau privada és un enter x_B tal que $2 \leq p - 2$, conegut exclusivament per B.

Algoritme 5 (ElGamal)**.** *(1) B construeix la clau pública* (G, p, α, y_B).

(2) B envia la clau pública per un canal de comunicació a A.

(3) A obté la clau pública i codifica el missatge amb un element $m \in G$ *seguint un procediment estàndard i conegut.*

(4) A tria un nombre enter a l'atzar de $\{2, \ldots, p-2\}$ *amb probabilitat uniforme.*

(5) A calcula $C = y_B^k$.

(6) A xifra el missatge calculant el parell (c_1, c_2)*, on*

$$c_1 := \alpha^k,$$

$$c_2 := Cm.$$

(7) A envia el missatge xifrat (c_1, c_2) *per un canal de comunicació a B.*

(8) B desxifra el missatge mitjançant la clau privada, utilitzant que

$$C = c_1^{x_B},$$

$$m = C^{-1} c_2.$$

La seguretat de l'algoritme, si suposem que ni la implementació ni la codificació són susceptibles a cap tipus d'atac, es basa en que si no coneixem la clau privada és necessari saber calcular el logaritme discret de c_1 o de y_B en base α per poder desxifrar el missatge. Com que hi ha grups G per als quals no es coneixen algoritmes que trobin el logaritme discret en temps polinòmic, una tria adient de G i α impossibilita el desxiframent del missatge. Algun exemple de grup que es pot utilitzar per aquest criptosistema és el grup multiplicatiu $G(p)$, on p és un enter primer, o als grups de punts de corbes el·líptiques sobre cossos finits.

5.2 Algoritme de Shor

Sigui un nombre natural $n > 2$ compost i senar. Definim $L := \lceil log(n) \rceil$. La codificació de l'element x de $G(n)$ és simplement $|x\rangle \in \mathbb{C}^{2^L}$. Emprarem la següent transformació unitària.

$$U_x(|y\rangle) = |xy \pmod n\rangle.$$

Per tal de calcular $xy \pmod n$, hi ha un algoritme clàssic que utilitza $O(L^2)$ operacions bàsiques per multiplicar dos nombres de L bits basat en el mètode de multiplicació a mà. D'igual manera, hi ha algoritmes senzills de divisió que empren $= O(L^2)$ operacions bàsiques (cf. [7]). Com que l'algoritme d'exponenciació binària utilitza $O(L)$ productes, vol dir que en total utilitzem $O(L^3)$ operacions bàsiques.

Lema 5.4.1. *Siguin $n > 1$ un nombre natural compost, p un divisor primer de n, un element $x \in G(n)$ qualsevol i $k \in \mathbb{Z}$ tal que k és múltiple de $p - 1$. Aleshores, $mcd(x^k - 1, n) \neq 1$.*

Lema 5.4.2. *Siguin $n > 2$ un nombre natural senar, $k \leq 1$ un nombre natural i x un enter qualssevol. Si $x \equiv -1, 0, 1 \pmod n$, llavors $mcd(x^k - 1, n) = 1$ o bé $mcd(x^k - 1, n) = n$.*

Corol·lari 5.5. *Els $x \in G(n)$ que permeten factoritzar n mitjançant el lema 5.4.1 han de satisfer $2 \leq x \leq n - 2$.*

Proposició 5.6. *Siguin $x \in \{2, \ldots, n-2\}$ i el seu ordre r en $G(n)$. Suposem que r és parell i que $x^{\frac{r}{2}} \not\equiv -1 \pmod n$, aleshores $mcd(x^{\frac{r}{2}} - 1, n)$ és un factor no trivial de n. Si x satisfà aquestes condicions, direm que és vàlid per factoritzar.*

Per implementar un mètode per factoritzar n, utilitzem l'algoritme 3 amb

$$t := 2L + 1 + \lceil \left(2 + \frac{1}{\epsilon}\right) \rceil,$$

qubits en el primer registre per a trobar una aproximació amb $2L+1$ xifres binàries de precisió, amb probabilitat $1 - \epsilon$. L'algoritme d'exponenciació binària, en aquest cas necessita de $O(L)$ productes, i $O(L^2)$ operacions bàsiques per realitzar productes, sumes i divisions enteres. Així, es requereixen $O(L^3)$ operacions bàsiques per a implementar l'exponenciació binària en aquest grup.

Algoritme 6 (de Shor)**.** *Aquest algoritme, donat n un nombre natural senar compost, retorna un factor primer de n.*

1. *Determinem si $n = a^b$, amb $a \geq 3, b \geq 2$. En cas que així sigui, retornem el valor de a.*
2. *Triem, de manera equiprobable $x \in \{2, \ldots, n-2\}$. Calculem $mcd(x, n)$. Si no és igual a 1, retornem $mcd(x, n)$. En cas contrari seguim al següent pas.*
3. *Usem l'algoritme 3 per trobar l'ordre de x a $G(n)$.*
4. *Si r és senar tornem al pas 2. En cas contrari seguim al pas següent.*

5. *Calculem* $y := x^{r/2} \in G(n)$. *Si* $y \equiv -1 \pmod{n}$ *tornem al pas 2. En cas contrari passem al següent pas.*

6. *Retornem* $mcd(y-1, n)$.

A continuació, estudiem la probabilitat de trobar un valor vàlid de x per a factoritzar en el pas 2 de l'algoritme.

Definició 5.7. Donat un nombre natural n i un nombre primer p, definim la valoració p-àdica de n com

$$\upsilon_p(n) = max\{k \in \mathbb{N} : p^k | n\}.$$

Lema 5.7.1. *Siguin p un primer senar i $d := \upsilon_2(\#G(p^\alpha))$, aleshores la probabilitat que l'ordre d'un element $x \in G(p^\alpha)$ triat a l'atzar de manera equiprobable tingui valoració 2-àdica igual a d és $\frac{1}{2}$.*

Demostració. Com que $\#G(p^\alpha) = p^{\alpha-1}(p-1)$ aleshores té ordre parell. Com que es tracta d'un grup cíclic, és fàcil veure que només la meitat dels seus elements són el quadrat d'un altre. Per últim, l'ordre d'un element té valoració p-àdica igual a d si, i només si, no és un quadrat. □

Proposició 5.8. *Sigui $n = p_1^{\alpha_1} \cdots p_k^{\alpha_k}$ la descomposició en factors primers d'un enter positiu senar, amb $k > 1$. Siguin $x \in G(n)$, escollit a l'atzar de manera equiprobable, i r l'ordre de x en $G(n)$. Llavors,*

$$P(x \text{ no sigui vàlid}) = P((r \text{ sigui senar}) \text{ o } (r \text{ sigui parell i } x^{r/2} \equiv -1 \pmod{n})) \leq \frac{2}{2^k}.$$

Demostració. Denotem $S := \{1, \ldots, k\}$. Per a tot $j \in S$ definim r_j com l'ordre de x en $G(p_j^{\alpha_j})$ i $M := mcm(r_1, \ldots, r_k)$. Clarament, $r_j | r$, de manera que $M | r$, i $M = r$ per la definició d'ordre.
El que demostrem ara és que x no serà vàlid per a factoritzar amb el nostre algoritme si, i només si, $\upsilon_2(r_j) = \upsilon_2(r)$ per a tot $j \in S$. En efecte, si r és senar, aleshores, com que $r_j | M$ i $M = r$, r_j és senar, se satisfà que $\upsilon_2(r_j) = 0$ per a tot $j \in S$. Si r és parell i $x^{r/2} \equiv -1 \pmod{n}$, se satisfà que $x^{r/2} \equiv -1 \pmod{p_j^{\alpha_j}}$, per a tot $j \in S$. Per tant, no hi ha cap r_j que divideixi $\frac{r}{2}$ ja que, en aquest cas, $x^{r/2} \equiv x^{r_j} \equiv 1 \pmod{p_j^{\alpha_j}}$ contradient que $x^{r/2} \equiv -1 \pmod{p_j^{\alpha_j}}$. Això indica que $\upsilon_2(r_j) = \upsilon_2(r)$.
Demostrem ara el recíproc. En cas que $\upsilon_2(r_j) = \upsilon_2(r) = 0$, aleshores r és senar. Si suposem que $\upsilon_2(r_j) = \upsilon_2(r) > 0$, aleshores r és parell. Com que $x^r \equiv 1 \pmod{n}$ implica que $x^r \equiv \left(x^{r/2}\right)^2 \equiv 1 \pmod{p_j^{\alpha_j}}$, per a qualsevol $j \in S$. Com que $G(p^\alpha)$ és cíclic, les úniques arrels quadrades de 1 són 1 i -1. Per tant $x^{r/2} \equiv 1 \pmod{p_j^{\alpha_j}}$ o $x^{r/2} \equiv -1 \pmod{p_j^{\alpha_j}}$. Si existeix un j tal que $x^{r/2} \equiv 1 \pmod{p_j^{\alpha_j}}$, aleshores $r_j | \frac{r}{2}$, per la definició d'ordre, contradient que $\upsilon_2(r_j) = \upsilon_2(r)$. Per tant, $x^{r/2} \equiv -1 \pmod{p_j^{\alpha_j}}$, per a tot $j \in S$ i, en conseqüència, $x^{r/2} \equiv -1 \pmod{n}$.
Pel teorema xinès del residu, triar $x \in G(N)$ és equivalent a triar una sèrie d'enters $a_1, \ldots, a_k$ tals que

$$x \equiv a_1 \pmod{p_1^{\alpha_1}},$$

$$\vdots$$

$$x \equiv a_k \ (\mathrm{mod} p_k^{\alpha_k}).$$

Igual que abans, definim r_j com l'ordre de a_j en $G(p_j^{\alpha_j})$. Per tant, en cas que la valoració 2-àdica de tots els r_j sigui la mateixa, no haurem trobat una x adient per factoritzar. A continuació, per cada j, definim $m_j := \upsilon_2(\varphi(p_j^{\alpha_j}))$, on φ és la funció φ d'Euler i $\varphi(p_j^{\alpha_j})$ és l'ordre del grup $G(p_j^{\alpha_j})$. Com que l'ordre d'un element divideix l'ordre del grup, $\upsilon_2(r_j) \leq m_j$. Denotem $m := min(m_1, \ldots, m_k)$. Per tal que x no sigui vàlid, és necessari que $\upsilon_2(r_j) = \upsilon_2(r)$, per a tot $j \in$, i $\upsilon_2(r) \leq m$. Pel lema 5.7.1, podem fer la següent acotació

$$P(x \text{ no vàlid }) \leq P(\upsilon_2(r_j) < m, \forall j) \text{ o } (\upsilon_2(r_j) = m, \forall j) =$$

$$P(\upsilon_2(r_j) < m, \forall j) + P(\upsilon_2(r_j) = m, \forall j) \leq \frac{1}{2^k} + \frac{1}{2^k} = \frac{2}{2^k}.$$

Finalment,

$$P(x \text{ sigui vàlid }) = 1 - P(x \text{ no sigui vàlid }) \geq 1 - \frac{2}{2^k}. \tag{5.1}$$

Com que sempre serà $k \geq 2$, la probabilitat sempre serà més gran o igual $\frac{1}{2}$. □

Corol·lari 5.9. *La probabilitat d'èxit de l'algoritme de factorització és més gran o igual que*

$$\frac{1-\epsilon}{4},$$

on ϵ és la tolerància admesa per l'algoritme d'estimació de fase.

Demostració. La probabilitat de que tingui èxit és la probabilitat que x sigui vàlida i que l'algoritme de cerca d'ordre tingui èxit

$$P((\text{x vàlid}) \text{ i } (\text{obtenir l'ordre})) =$$

$$P(\text{x vàlid})P(\text{obtenir l'ordre}; \text{x vàlid}) \geq$$

$$\frac{1}{2}(1-\epsilon)(1-r^{-1}) \geq \frac{1-\epsilon}{4},$$

on hem utilitzat les cotes inferiors de les probabilitats (4.1) i (5.1). □

5.3 Algoritme quàntic de factorització mitjançant corbes el·líptiques

Siguin $n := p_1 p_2$, un enter producte de dos primers p_1, p_2 diferents entre ells, tal que $mcd(6, n) = 1$ i $L := \lceil n \rceil$.

Notació. Donada una corba el·líptica E sobre $\mathbb{Z}/n\mathbb{Z}$ i $P, Q \in E(\mathbb{Z}/n\mathbb{Z})$, denotem

$$P \equiv Q \ (\mathrm{mod}\ p_i),$$

on $i \in \{1, 2\}$, si la reducció mòdul p_i de P i Q coincideixen.

Codificarem aquests punts $P := (x, y, z)$ com a producte d'estats ben definits,

$$|P\rangle := |x\rangle \otimes |y\rangle \otimes |z\rangle \in \mathbb{C}^{2^{3L}}.$$

Observem que la suma de dos punts a $E(\mathbb{Z}/nZ)$ implica un nombre fitat de sumes, restes, multiplicacions, divisions, per les quals existeixen algoritmes eficaços amb una complexitat $O(L^2)$ (cf. [7]) en el nombre d'operacions bàsiques, i per tant l'exponenciació binària (adició binària, en aquest cas) és pot realitzar de manera eficient amb $O(L^3)$ operacions, ja que només necessitarem $O(L)$ qubits per realitzar-la. Això implica que

$$U_P(|k\rangle \otimes |Q\rangle) = |k\rangle \otimes |kP + Q\rangle,$$

on k és un natural i $P, Q \in E(\mathbb{Z}/n\mathbb{Z})$, es pot implementar eficientment.

El mètode que volem introduir comença amb l'elecció d'una corba i un punt. Per tal de fer això, triem enters $a, x, y \in \{0, \ldots, n-1\}$ a l'atzar i calculem

$$b :\equiv y^2 - x(x^2 + ax) \pmod{n},$$

amb $0 \leq b \leq n-1$. Per tant, el punt $P := (x : y : 1)$ és solució de l'equació de Weierstrass $y^2 \equiv x^3 + ax + b \pmod{n}$. Considerant la reducció mòdul p, on p és un divisor primer de n, se satisfà que $P_p := (x \pmod{p} : y \pmod{p})$ és solució de

$$y^2 \equiv x^3 + ax + b \pmod{p}.$$

La corba serà vàlida sempre que $d := mcd(4a^3 + 27b, n) \neq n$, ja que en aquest $d \equiv 0 \pmod{p}$. En aquest cas haurem de procedir a triar altres $x, y, a \in \{0, \ldots, n-1\}$. Si $1 < d \leq n-1$ hauríem trobar un divisor no trivial de n i hauríem acabat. Si $d = 1$, la corba de Weierstrass defineix una corba el·lítpica sobre $\mathbb{F}_p$, per a tot p divisor primer de n. A més, si un punt $P = (x : y : 1)$ és solució de l'equació, aleshores el punt P_p és solució de l'equació de Weierstrass reduïda mòdul p i, en ser $E(\mathbb{F}_p)$ un grup, podem parlar de l'ordre de P_p. Si suposem que la corba és vàlida i que no hem factoritzat, passem al següent pas de l'algoritme que consisteix en saber l'ordre de P en $E(\mathbb{Z}/n\mathbb{Z})$. Per tal de trobar aquest ordre, utilitzarem l'algoritme de cerca d'ordre. En aquest cas, utilitzarem

$$t := 6L + 1 + \lceil log\left(2 + \frac{1}{2\epsilon}\right)\rceil$$

qubits al primer registre, per tal d'obtenir una estimació de fase amb precisió de $6L + 1$ xifres binàries, amb probabilitat més gran que $1 - \epsilon$.
Suposem que l'algoritme de cerca d'ordre ha tingut èxit, i hem trobat l'ordre de P, que denotarem r. Si r és senar, no podrem factoritzar amb el nostre mètode i haurem de tornar a triar un altre punt i una altra corba. En cas que r sigui parell, definim $Q := \frac{r}{2}P$. Per tant, tindrem que $2Q \equiv O \pmod{p_i}$, per a $i \in \{1, 2\}$, fet que ens portarà a considerar dos casos.

(1) Si $Q \not\equiv O \ (mod\ p_i)$ per a $i \in \{1, 2\}$, aleshores no podrem factoritzar i haurem de tornar a triar un altre punt i una altra corba.

(2) Si $Q \equiv O \ (mod\ p_1)$ i $Q \not\equiv O \ (mod\ p_2)$, o viceversa, i denotem les seves coordenades per $Q = (t : u : v)$, aleshores podem trobar un factor no trivial de n. En efecte, suposem, sense pèrdua de generalitat, que $Q \equiv O \ (mod\ p_1)$ i $Q \not\equiv O \ (mod\ p_2)$. D'una banda, Q_{p_2} és un element d'ordre 2 a $E(\mathbb{Z}/p_2\mathbb{Z})$, amb el que $u \equiv 0 \pmod{p_2}$. D'altra banda $Q_{p_1} = O$ no és un element d'ordre 2 a $E(\mathbb{Z}/p_1\mathbb{Z})$, així que $u \not\equiv 0 \pmod{p_2}$ i, per tant, $u \not\equiv 0 \pmod{n}$. Concloem que $mcd(u, n) = p_1$, amb el que trobem un factor no trivial de n. Direm que en aquest cas, Q *permet factoritzar*.

Observació 9. El cas on $Q \equiv O$ mòdul p_1 i p_2 està exclòs, ja que en aquest cas es contradiu la definició d'ordre de P.

A continuació fem un resum de l'algoritme (cf. [1]).

Algoritme 7 (Alberto Cámara)**.** *Sigui $n := p_1p_2$, producte de dos primers diferents tal que $mcd(6, n) = 1$.*

1. *Triem a l'atzar, de manera equiprobable, $x, y, a \in \{0, \ldots, n-1\}$.*
2. *Calculem $b := y^2 - x(x^2 + a)$ i $d := mcd(4a^3 + 27b^2, n)$. Si $1 < d < n$ retornem d i finalitzem. Si $d = n$, donem un missatge d'error i finalitzem. Si $d = 1$ seguim al següent pas.*
3. *Emprem l'algoritme 3 per a calcular l'ordre r del punt $P = (x : y : 1)$, pertanyent a la corba el·líptica sobre $\mathbb{Z}/n\mathbb{Z}$ amb equació de Weierstrass afí $Y^2 = X^3 + aX^2 + B$.*
4. *Si r és senar donem un missatge d'error i finalitzem. En cas contrari, seguim al pas següent.*
5. *Calculem $Q := \frac{r}{2}$, que té per coordenades $Q = (t : u : v)$.*
6. *Calculem $d' := mcd(u, n)$. Si és $d = n$, retornem un missatge d'error i finalitzem (en aquest cas Q no factoritza n). Si $1 < d' < n$, aleshores retornem d' i finalitzem (Q factoritza n).*

El càlcul dels màxims comuns divisors es realitza per l'algoritme d'Euclides amb $O(L^3)$ operacions bàsiques. La complexitat d'aquest algoritme és $O(L^3)$, corresponent a implementar l'exponenciació binària per al càlcul de r i per al càlcul de Q. Així, necessitem $O(L)$ qubits per guardar el registres i $O(L^3)$ operacions per implementar l'algoritme.

Procedim a calcular la probabilitat de trobar una Q que permeti factoritzar, triant x, y, a de manera equiprobable en $\{0, \ldots, n-1\}$ (sense tenir en compte la probabilitat de fracàs associada a l'algoritme de cerca d'ordre). L'algoritme donarà un missatge d'error si $d = n$ en el pas 2, o r és senar, o r és parell però Q no factoritza n.

Proposició 5.10. *Siguin $x, y, a \in \{0, \ldots, n-1\}$, triats a l'atzar, $b := y^2 - x(x^2+a)$, $\Delta := 4a^3 + 27b^2$ i $d := mcd(\Delta, n)$. Aleshores,*

$$P(d = n) = \frac{1}{n}.$$

Demostració. Veiem que, pel teorema xinès del residu,

$$P(d = n) = P((p_1|d) \text{ i } (p_2|d)) = P((p_1|d)P(p_2|d).$$

D'altra banda,

$$P(p_1 \nmid d) = P((d = p_2) \text{ o } (d = 1)) = P((\Delta \in \{p_2, 2p_2, \ldots, (p_1-1)p_2\}) \text{ o } (\Delta \in G(n))) =$$

$$\frac{p_1 - 1}{n} + \frac{\#G(n)}{n} = \frac{p_1 - 1}{n} + \frac{(p_1 - 1)(p_2 - 1)}{n} = 1 - \frac{p_2}{n} = 1 - \frac{1}{p_1}$$

Per tant, $P(p_1|d) = \frac{1}{p_1}$ i de manera anàloga demostrem que $P(p_2|d) = \frac{1}{p_2}$. Finalment,

$$P(d = n) = P(p_1|d)P(p_2|d) = \frac{1}{p_1}\frac{1}{p_2} = \frac{1}{n}.$$

□

En conseqüència, en l'algoritme passarem al pas 3 de l'algoritme 7 o factoritzarem amb alta probabilitat.

Proposició 5.11. *Siguin p un nombre primer i E una corba el·líptica sobre $\mathbb{F}_p$, aleshores*

$$\left| P(2|\#E(\mathbb{F}_p)) - \frac{2}{3} \right| \leq \frac{1 + 10\sqrt{2}}{\sqrt{p}}.$$

Demostració. La demostració es troba a [5]. □

Teorema 5.12. *Suposem que en el pas 3 de l'algoritme hem trobar l'ordre r d'un punt $P \in E$, on E és una corba el·líptica sobre $\mathbb{Z}/n\mathbb{Z}$. Aleshores, la probabilitat de no trobar una Q que factoritzi és menor o igual que*

$$\left(\frac{2}{3} + \frac{3}{2}\nu\right)^m + \left(\frac{2}{3} + \nu\right)^m,$$

on $\nu := max\left(\frac{1+10\sqrt{2}}{\sqrt{p_1}}, \frac{1+10\sqrt{2}}{\sqrt{p_1}}\right)$

Demostració. L'algoritme fallarà en cas que r sigui senar o en cas que el punt $Q := \frac{r}{2}P$ tingui coordenades $Q = (t : u : v)$ amb $mcd(u, n) = n$. És a dir, la probabilitat de no trobar una Q que factoritzi és

$$P(2 \nmid r) + P(2|r \text{ i Q no factoritza}).$$

Denotem per r_i l'ordre de P mòdul p_i, on $i \in \{1, 2\}$. Ja hem vist, pel teorema 2.23, que r és senar si, i només si, r_i és senar per $i = \{1, 2\}$. Així,

$$P(2 \nmid r) = P(2 \nmid r_1)P(2 \nmid r_2).$$

D'altra banda, se satisfà que

$$P(2 \nmid r_i) = P(2 \nmid \#E_{p_i}) + P(2|\#E_{p_i} \text{ i } 2 \nmid r_i) =$$

$$P(2 \nmid \#E_{p_i}) + P(2|\#E_{p_i})P(2 \nmid r_i; 2|\#E_{p_i}).$$

La proposició 5.11 ens permet acotar aquesta probabilitat,

$$P(2 \nmid r_i) \leq 1 + \left(\frac{1+10\sqrt{2}}{\sqrt{p_i}}\right) + \frac{1}{2}\left(\frac{2}{3} + \frac{1+10\sqrt{2}}{\sqrt{p_i}}\right) \leq \frac{2}{3} + \frac{3\nu}{2}.$$

Així,

$$P(2 \nmid r) = P(2 \nmid r_1)P(2 \nmid r_2) \leq \left(\frac{2}{3} + \frac{3\nu}{2}\right)^2.$$

Finalment, procedim a trobar una fita de $P(2|r$ i Q no factoritza). Aquesta probabilitat coincideix amb

$$P((2|r) \text{ i } (\frac{r}{2}P \neq O(\text{mod } p_j), \forall j \in \{1,2\})).$$

Sigui $s := v_2(r) \geq 1$. Notem que

$$\frac{r}{2}P \neq O \text{ (mod } p_i), \forall i \in \{1,2\}$$

és el mateix que demanar $v_2(r_i) = s$, per a tota i. D'altra banda, r és parell si, i només si, algun dels r_i és parell. Per tant,

$$P((2|r) \text{ i (Q no factoritza)}) = P((\exists j \in \{1,2\}, 2|r_j) \text{ i } (v_2(r_k) = s, \forall k \in \{1,2\})) =$$

$$P(\forall k, v_2(r_k) = s).$$

Si 2^s divideix r_i, aleshores 2^s ha de dividir l'ordre del grup, $\#E_{p_i}(\mathbb{F}_{p_i})$, el que implica que

$$P((2|r) \text{ i (Q no factoritza)}) \leq P(\forall k \in \{1,2\}, 2^s|\#E_{p_k}(\mathbb{F}_{p_k})) \leq$$

$$\leq P(\forall k \in \{1,2\}, 2|\#E_{p_k}(\mathbb{F}_{p_k})) = P(2|\#E_{p_1}(\mathbb{F}_{p_1}))P(2|\#E_{p_2}(\mathbb{F}_{p_2})) \leq \left(\frac{2}{3} + \nu\right)^2,$$

on hem utilitzat la proposició 5.11. Finalment, la probabilitat de no trobar una Q que factoritzi és menor o igual a

$$\left(\frac{2}{3} + \frac{3}{2}\nu\right)^2 + \left(\frac{2}{3} + \nu\right)^2. \tag{5.2}$$

□

Observem, que, en cas que el primers implicats siguin baixos, aquest teorema ens donarà una cota superior a 1, i ens resultaria inútil. Cal que els primers que divideixen n siguin prou grans per tal que ν sigui menyspreable davant de $\frac{2}{3}$. D'aquesta manera, la cota (5.2) seria aproximadament $\frac{8}{9}$, el qual ens indica que, com a mínim, podrem factoritzar un de cada 9 cops que passem del pas 3 de l'algoritme.

5.4 Algoritme quàntic de càlcul del logaritme discret

Siguin $(G,\cdot)$ un grup cíclic finit i e el seu neutre. Sigui un element $a \in G$ tal que $G = \langle a \rangle = \{1, a, a^2, \ldots, a^{r-1}\}$. El problema del logaritme discret consisteix en, donat $b \in G$, determinar quin és l'exponent $s \in \{0, 1, \ldots r-1\}$ que satisfà

$$b = a^s.$$

Observació 10. Degut a que coneixem un algoritme per trobar l'ordre de a, suposarem que coneixem el valor de r.

Per a la resolució del problema ens és útil definir la funció

$$\begin{array}{rccc} f: & \mathbb{Z}^2 & \longrightarrow & \langle a \rangle. \\ & (x_1, x_2) & \mapsto & a^{sx_1+x_2} \end{array}$$

Observem que $f(x_1 + \lambda, x_2 - \lambda s) = f(x_1, x_2)$, per a qualsevol $\lambda \in \mathbb{Z}$. Suposem que podem representar els elements de G amb una codificació en $n := \lceil log(r) \rceil$ qubits. És a dir, a cada element $c \in G$ li assignem un estat ben definit que denotem $|c\rangle \in \mathbb{C}^{2^n}$. Suposem que sabem implementar l'operació

$$U(|x_1\rangle \otimes |x_2\rangle \otimes |y\rangle) = |x_1\rangle \otimes |x_2\rangle \otimes |y \oplus f(x_1, x_2)\rangle, \tag{5.3}$$

on $x_1, x_2 \in \{0, \ldots, 2^t - 1\}$, amb $t := 3n + 1 + \lceil log(2 + \frac{1}{\epsilon}) \rceil$ per a cert ϵ petit, i $y \in G$. L'operació $\oplus$ denota la següent operació entre elements de $(\mathbb{Z}/2\mathbb{Z})^n$,

$$(u_{n-1}, u_{n-2}, \cdots, u_0) \oplus (v_{n-1}, v_{n-2}, \cdots, v_0) :=$$

$$(u_{n-1} + v_{n-1} \pmod 2, u_{n-2} + v_{n-2} \pmod 2, \cdots, u_0 + v_0 \pmod 2).$$

Proposició 5.13. *Sigui:*

$$|\widehat{f}(l_1, l_2)\rangle := C \sum_{x_1=0}^{2^t-1} \sum_{x_2=0}^{2^t-1} exp\left(\frac{-2\pi i(l_1 x_1 + l_2 x_2)}{2^t}\right) |f(x_1, x_2)\rangle,$$

on C és una constant de normalització, aleshores

(1) $|\widehat{f}(l_1, l_2)\rangle = \frac{1}{\sqrt{r}} \sum_{j=0}^{r-1} exp\left(\frac{-2\pi i l_2 j}{r}\right) |f(0, j)\rangle$, *si* $l_1 - l_2 s \equiv 0$ *(mod r) i* $|\widehat{f}(l_1, l_2)\rangle = 0$ *en cas contrari.*

(2) $|f(x_1, x_2)\rangle = \frac{1}{\sqrt{r}} \sum_{l_1=0}^{r-1} \sum_{l_2=0}^{r-1} exp\left(\frac{2\pi i(l_1 x_1 + l_2 x_2)}{r}\right) |\widehat{f}(l_1, l_2)\rangle$.

Demostració. (1)

$$C \sum_{x_1=0}^{r-1} \sum_{x_2=0}^{r-1} exp\left(\frac{-2\pi i(l_1 x_1 + l_2 x_2)}{r}\right) |f(x_1, x_2)\rangle =$$

$$C \sum_{x_1=0}^{r-1} \sum_{x_2=0}^{r-1} exp\left(\frac{-2\pi i(l_1 x_1 + l_2 x_2)}{r}\right) |f(0, sx_1 + x_2)\rangle =$$

$$C \sum_{x_1=0}^{r-1} \sum_{x_2=0}^{r-1} exp\left(\frac{-2\pi i((l_1 - l_2 s)x_1 + l_2(x_2 + sx_1))}{r}\right) |f(0, sx_1 + x_2)\rangle =$$

$$C \sum_{x_1=0}^{r-1} exp\left(\frac{-2\pi i(l_1 - l_2 s)x_1}{r}\right) \sum_{x_2=0}^{r-1} exp\left(\frac{-2\pi i l_2(x_2 + sx_1)}{r}\right) |f(0, sx_1 + x_2)\rangle =$$

$$C \sum_{x_1=0}^{r-1} exp\left(\frac{-2\pi i(l_1 - l_2 s)x_1}{r}\right) \sum_{j=sx_1}^{r-1+sx_1} exp\left(\frac{-2\pi i l_2 j}{r}\right) |f(0, j)\rangle =$$

$$C \sum_{x_1=0}^{r-1} exp\left(\frac{-2\pi i(l_1 - l_2 s)x_1}{r}\right) \sum_{j=0}^{r} exp\left(\frac{-2\pi i l_2 j}{r}\right) |f(0, j)\rangle.$$

En cas que $\frac{l_1 - l_2 s}{r} \in \mathbb{Z}$, aquesta expressió equival a

$$|\widehat{f}(l_1, l_2)\rangle = \frac{1}{\sqrt{r}} \sum_{j=0}^{r-1} exp\left(\frac{-2\pi i l_2 j}{r}\right) |f(0, j)\rangle.$$

En cas contrari, és igual a 0.

(2) Imposant el resultat anterior, s'ha de complir que $l_1 = sl_2 + \alpha_{l_2} r$, per a cert $\alpha_{l_2} \in \mathbb{Z}$. Independentment del valor de s, per cada l_2 només hi ha un únic possible valor de $l_1 \in \{0, \ldots, r-1\}$. Així

$$\frac{1}{\sqrt{r}} \sum_{l_1=0}^{r-1} \sum_{l_2=0}^{r-1} exp\left(\frac{2\pi i(l_1 x_1 + l_2 x_2)}{r}\right) |\widehat{f}(l_1, l_2)\rangle =$$

$$\frac{1}{\sqrt{r}} \sum_{l_2=0}^{r-1} exp\left(\frac{2\pi i(sl_2 x_1 + l_2 x_2)}{r}\right) |\widehat{f}(sl_2 + \alpha_{l_2} r, l_2)\rangle =$$

$$\frac{1}{\sqrt{r}} \sum_{l_2=0}^{r-1} exp\left(\frac{2\pi i(sl_2 x_1 + l_2 x_2)}{r}\right) \frac{1}{\sqrt{r}} \sum_{j=0}^{r-1} exp\left(\frac{-2\pi i l_2 j}{r}\right) |f(0, j)\rangle =$$

$$\frac{1}{\sqrt{r}} \frac{1}{\sqrt{r}} \sum_{j=0}^{r-1} \sum_{l_2} exp\left(\frac{2\pi i l_2(sx_1 + x_2 - j)}{r}\right) |f(0, j)\rangle.$$

Aquesta expressió només és diferent de 0 si $sx_1 + x_2 - j \equiv 0 \pmod{r}$. En aquest cas, el resultat final és

$$|f(0, sx_1 + x_2)\rangle = |f(x_1, x_2)\rangle.$$

□

Algoritme 8. *Aquest algoritme permet trobar el logaritme discret de $b = a^s$ en base a. L'input inicial és el producte tensorial de tres registres: dos registres de $t := 2n + 1 + \lceil log(2 + \frac{1}{\epsilon}\rceil$ qubits cadascun inicialitzats a l'estat $|0\rangle$ i un altre registre amb n qubits inicialitzats a l'estat $|0\rangle$.*

(1) Implementem l'estat inicial

$$|0\rangle \otimes |0\rangle \otimes |0\rangle.$$

(2) Apliquem una porta Hadamard a cadascun dels dos primers $2t$ qubits (dos primers registres):

$$\mapsto \frac{1}{2^t} \sum_{x_1=0}^{2^t-1} \sum_{x_2=0}^{2^t-1} |x_1\rangle \otimes |x_2\rangle \otimes |0\rangle$$

(3) Apliquem la transformació U

$$\mapsto \frac{1}{2^t} \sum_{x_1=0}^{2^t-1} \sum_{x_2=0}^{2^t-1} |x_1\rangle \otimes |x_2\rangle \otimes |f(x_1, x_2)\rangle =$$

$$\frac{1}{2^t} \frac{1}{\sqrt{r}} \sum_{l_2=0}^{r-1} \sum_{x_1=0}^{2^t-1} \sum_{x_2=0}^{2^t-1} exp\left(\frac{2\pi i(sl_2x_1 + l_2x_2)}{r}\right) |x_1\rangle \otimes |x_2\rangle \otimes |\widehat{f}(sl_2, l_2)\rangle =$$

$$\frac{1}{2^t\sqrt{r}} \sum_{l_2=0}^{r-1} \left[\sum_{x_1=0}^{2^t-1} exp\left(\frac{2\pi i s l_2 x_1}{r}\right) |x_1\rangle\right] \otimes \left[exp\left(\frac{2\pi i l_2 x_2}{r}\right) |x_2\rangle\right] \otimes |\widehat{f}(sl_2, l_2)\rangle.$$

(4) Apliquem la inversa de la transformada de Fourier a cadascun dels dos primers registres

$$\mapsto \frac{1}{\sqrt{r}} \sum_{l_2=0}^{r-1} |\phi_{l_2}\rangle \otimes |\phi'_{l_2}\rangle \otimes |\widehat{f}(sl_2, l_2)\rangle,$$

on ϕ_{l_2} i ϕ'_{l_2} són estimacions de $\frac{sl_2}{r}$ i $\frac{l_2}{r}$, respectivament

(5) Fem una observació dels primers dos registres i obtenim les estimacions $(\phi_{l_2}, \phi'_{l_2})$, per a cert $l_2 \in \{1, \ldots, r-1\}$. Si

(6) Utilitzem l'algoritme de fraccions continuades per cada una de les estimacions i utilitzem que coneixem el valor de r per tal d'obtenir s.

En total, hem utilitzat $O(n)$ portes Hadamard, una transformació U i dos cops la inversa de la transformada de Fourier, que requereix de $O(n^2)$ portes simples i l'algoritme de fraccions continuades, que té complexitat $O(n^3)$. D'altra banda només hem requerit $O(n)$ qubits per emmagatzemar els registres. La probabilitat d'èxit de l'algoritme ve donada per la probabilitat d'èxit de la cerca d'ordre que hem vist anteriorment a (4.1).

Exemple 5.14. Si $G = G(p)$, amb p un nombre primer, aleshores podem codificar els seus elements mitjançant $n = \lceil log(p) \rceil$ qubits i implementar (5.3) afegint un quart registre:

$$|x_1\rangle \otimes |x_2\rangle \otimes |y\rangle \otimes |h\rangle,$$

on $x_1, x_2 \in \mathbb{N}$ i $y, h \in G(p)$. Per tal d'implementar-la, apliquem dues vegades exponenciació binària a l'últim registre, utilitzant els dos primers com a control, de la següent manera:

$$|x_1\rangle \otimes |x_2\rangle \otimes |y\rangle \otimes |h\rangle \to |x_1\rangle \otimes |x_2\rangle \otimes |y\rangle \otimes |hp^{x_1}\rangle \to |x_1\rangle \otimes |x_2\rangle \otimes |y\rangle \otimes |hp^{x_1+x_2}\rangle.$$

Per últim utilitzem els qubits de $|hp^{x_1+x_2}\rangle$ com a control de n portes C_{NOT} aplicades al tercer registre per tal de realitzar l'operació

$$|x_1\rangle \otimes |x_2\rangle \otimes |y \oplus hp^{x_1+x_2}\rangle \otimes |hp^{x_1+x_2}\rangle.$$

Per últim desfem els canvis fets de manera reversible en el quart registre i el transformem de nou en l'estat $|h\rangle$.

$$|x_1\rangle \otimes |x_2\rangle \otimes |y \oplus hp^{x_1+x_2}\rangle \otimes |h\rangle.$$

Si $y = 0$ tindrem la transformació desitjada. Necessitem $O(n)$ qubits per les exponenciacions binàries i $O(n)$ portes C_{NOT} i $O(n^3)$ operacions per realitzar l'exponenciació binària, a més de que hem afegit un quart registre de n qubits.

6 Conclusions

La realització del treball ha donat a lloc a diverses conclusions.
En primer lloc, hem vist que un ordinador quàntic és capaç de millorar els ordinadors clàssics a l'hora de resoldre alguns problemes. Per exemple, l'algoritme de Shor millora tots els algoritmes clàssics coneguts de factorització. Mentre que l'algoritme de Shor té una complexitat de $O((log(n)^3)$, on n és el nombre a factoritzar, el millor algoritme clàssic per factoritzar nombres grans, el garbell de cossos de nombres, té una complexitat subexponencial en $log(n)$. No obstant, també hem vist un mètode general per construir qualsevol transformació unitària de n qubits, necessita de $O(4^n n^2)$ portes universals. D'aquesta prescripció no podem concloure que tots els circuits es poden implementar de manera eficient i deixa sospitar que potser hi ha problemes que no es podrien resoldre en temps polinòmic amb un ordinador quàntic.
Per últim, tots els algoritmes que hem treballat parteixen de saber el nombre de qubits amb què treballarem. És a dir, un ordinador quàntic és una construcció específica per a un problema i un registre. De manera que cada cop que vulguem resoldre un problema haurem de tornar a construir un nou circuit.

7 Apèndix

7.1 Fraccions continuades

Definició 7.1. Sigui x un nombre real arbitrari. A partir de x podem definir dos successions per inducció: $\{a_n\}_n$, de nombres enters, i $\{x_n\}_n$, de nombres reals. En primer lloc, $a_0 := [x]$ i $x_0 := x - a_0$. En cas de que $x_0 = 0$ hem acabat i no definim cap més terme. Si $x_0 \neq 0$, aleshores definim $x_1 := \left[\frac{1}{x_0}\right]$ i $x_1 := \frac{1}{x_0} - a_1$. Suposem que hem definit termes de les successions fins a_k i x_k. Aleshores, si $x_k = 0$, acabem i no definim cap més terme. En cas contrari, definim $a_{k+1} := \left[\frac{1}{x_k}\right]$ i $x_{k+1} := \frac{1}{x_k} - a_{k+1}$. Denominarem fracció continuada associada a x la successió de nombres enters $\{a_n\}_n$ determinada d'aquesta manera, la qual pot contenir un nombre infinit de termes.

Observació 11. A través de la definició podem concloure que, per qualsevol $k \geq 0$, se satisfà

$$x = a_0 \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\cdots + \frac{1}{a_k + x_k}}}}.$$

Denotarem aquesta igualtat amb

$$[a_0; a_1, a_2, \ldots, a_k + x_k].$$

Definició 7.2. Sigui x un nombre real, anomenem k-èsim convergent de la fracció continuada associada a x al nombre racional $[a_0; a_1, a_2, \ldots, a_k]$.

Proposició 7.3. *Un nombre real té associada una fracció continuada finita si, i només si, és racional.*

Observació 12. En el cas dels nombres racionals hi ha certa ambigüitat en la definició de fracció continuada. En efecte, considerem un racional x que té associada la fracció continuada $x = [a_0; a_1, \ldots, a_n]$. Si $a_n \neq 1$, aleshores

$$\frac{1}{a_n} = \frac{1}{(a_n - 1) + \frac{1}{1}},$$

amb el que $[a_0; a_1, \ldots, a_n - 1, 1]$ també seria una fracció continuada associada a x. En cas de que $a_n = 1$, podem fer el procés invers i $[a_0; a_1, \ldots, a_{n-2}, a_{n-1} + 1]$ també seria una fracció continuada associada a x. D'aquesta manera, per qualsevol racional, podem escollir una fracció continuada amb un nombre parell o amb un nombre senar de termes.

Proposició 7.4. *Siguin x un nombre real i $[a_0; a_1, a_2, \ldots, a_k, \ldots]$ la fracció continuada associada a x. Definim dos successions de nombres enters, $\{b_n\}_n, \{c_n\}_n$ de manera recursiva,*

$$b_0 := a_0,\ b_1 := a_0 a_1 + 1, \ldots,\ b_{k+2} := a_{k+2} b_{k+1} + b_k,$$

$$c_0 := 1,\ c_1 := a_1, \ldots,\ c_{k+2} := a_{k+2} c_{k+1} + c_k.$$

Llavors,

(a) El k-èsim convergent és $[a_0 : a_1, a_2, \ldots, a_k] = \frac{b_k}{c_k}$.

(b) Per $k \geq 1$ *es compleix que* $b_k c_{k-1} - b_{k-1} c_k = (-1)^{k-1}$.

(c) Se satisfà que $mcd(b_k, c_k) = 1$.

(d) La successió $\{c_n\}_n$ *és creixent.*

7.2 Algoritme d'exponenciació binària

Sigui G un grup abelià finit i e el seu neutre, els elements del qual poden ser codificats per tires de n bits, i que coneixem com implementar el seu producte intern en un ordinador clàssic, el qual denotem per *. Per tal de realitzar un algoritme quàntic d'exponenciació binària, ens basarem en dos senzills algoritme escrits en pseudocodi, per calcular g^a, essent $g \in G$ i a un enter amb representació binària $a = a_{l-1}2^{l-1} + \cdots a_1 2^1 + a_0$.

(1) El primer algoritme és

```
power := e
for i=0 to n-1
        if(a_i == 1)
                power := power * g^(2^i)
        endif
endfor
```

El producte efectuat a la línia 4 es correspon a l'operació pròpia del grup, la qual hem suposat que sabem implementar com a subrutina. D'altra banda podem suposar que els elements g^{2^i} són coneguts i que es troben a la implementació. Aquest algoritme és adient si l'utilitzarem repetides vegades per a diferents exponents de n bits, sense variar la base g. En aquest cas, efectuem, n^2 productes, si tenim en compte el càlcul previ de les potències de g, i utilitzem $O(n^2)$ bits per guardar els valors de g^{2^i} i n bits per a.

(2) El segon algoritme és

```
power := e
for i=0 to n-1
        if(a_i == 1)
                power := power * g
        endif
        g:=g^2;
endfor
```

En aquest cas no suposem coneguts les potències de g. Aquest algoritme serà adient si l'utilitzem per a diferents elements g actuant de base. En total, utilitzem, com a molt, $2n$ productes i emprem n bits per a, n bits per g^a i una de

n bits per les potències de g.
Com que podem adaptar qualsevol algoritme clàssic sense variar la seva complexitat, si codifiquem els elements de G com a estats ben definits d'un sistema de n qubits, donat un $g \in G$ aquest algoritme ens permetrà implementar la porta quàntica

$$|k\rangle \otimes |h\rangle \to |k\rangle \otimes |g^k h\rangle,$$

on $h \in G$ i $k \in \{0, \ldots, 2^n - 1\}$, en $O(n)$ operacions unitàries.

Referències

[1] A. Cámara. *Algoritmes quàntics de factorització*. Treball final del Màster de Matemàtica avançada i Professional. Universitat de Barcelona, 2008.

[2] D. Castelvecchi. *Quantum computers ready to leap out of the lab*. Nat. 541(2017), pàg. 9-10.

[3] H. Cohen. *A course in Computational Algebraic Number theory*. New York; Springer-Verlag 1996.

[4] T. ElGamal. *A public key cryptosystem and a signature scheme based on discrete logarithms.* Workshop on the Theory and Application of Cryptographic Techniques. Springer Berlin Heidelberg, 1984, pàg. 10-18.

[5] E. W. Howe. *On the group orders of elliptic curves over finite fields*. Comp. Math., 85 (1993), pàg. 229-247.

[6] J. Jonsson, B. Kaliski. *Public-key cryptography standards (PKCS)# 1: RSA cryptography specifications version 2.1.* 2003.

[7] D. E. Knuth. *The Art of Programming: Seminumerical Algorithms, vol. 2.* Reading: Addison-Wesley, 1981, pàg. 250-265.

[8] S. Lang. *Álgebra lineal*. Delawere: Addison-Wesley Iberoamericana, 1986.

[9] M. A. Nielsen, I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press, 2010.

[10] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*. SIAM review, 41.2 (1999), pàg. 303-332.

[11] J. Silvermann, *The arithmetic of elliptic curves*. Grad. Texts in Math 106 (1986).

[12] A. Travesa. *Aritmètica*. Barcelona: Edicions de la universitat de Barcelona, 1998.