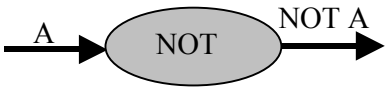
NECESITATEA CALCULULUI CUANTIC

Calculatoarele electronice se bazeaza pe faptul ca toate operatiile matematice se efectueaza asupra bitilor, adica a starilor 0 sau 1, si pe faptul ca toate functiile logice care actioneaza asupra unor registre (siruri de biti) pentru a le converti in alte registre, pot fi descompuse in porti AND si NOT. De aceea, un algoritm care produce un anumit rezultat pornind de la un set de date initiale poate fi implementat prin conectarea corespunzatoare a portilor AND si NOT.

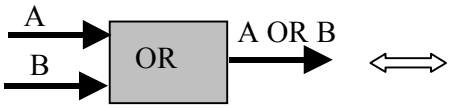
EXEMPLU: operatia logica OR poate fi obtinuta printr-o succesiune de porti AND si NOT:

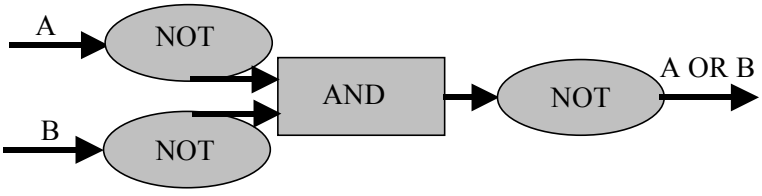
A	B	A AND B
0	0	0
1	0	0
0	1	0
1	1	1

A	NOT A
0	1
1	0



A	B	A OR B
0	0	0
1	0	1
0	1	1
1	1	1

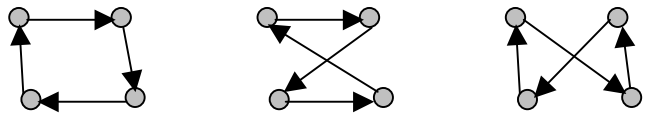




Starile 0 si 1 se pot implementa prin stari fizice distincte ale unui sistem: printr-un capacitor incarcat, respectiv descarcat, printr-o tensiune mica, respectiv mare, la portile unui tranzistor, printr-un comutator deschis, respectiv inchis, sau printr-o magnetizare care poate fi orientata in doua directii diferite. Viteza de calcul depinde de timpul in care starea logica poate fi schimbata (timp de switch). Pentru a reprezenta siruri de biti, un calculator trebuie sa contina o colectie de sisteme fizice, fiecare putand coda un bit.

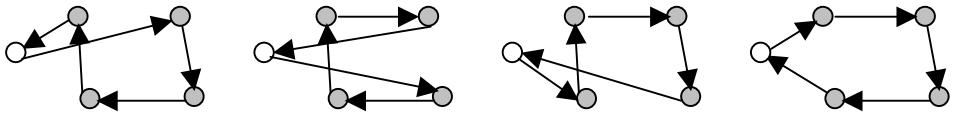
Limitele calculatoarelor electronice sunt evidente atat din punct de vedere tehnologic (pe masura ce componentele electronice ale calculatoarelor moderne devin din ce in ce mai mici, pentru a creste viteza de calcul prin micsorarea timpului de propagare a semnalelor electrice, natura cuantica a purtatorilor de sarcina nu mai poate fi ignorata) cat si din punct de vedere al puterii de calcul. Ultima limitare este evidenta in special in asa-zisele probleme grele din punct de vedere al calculului. O astfel de problema este cea a comis-voiajorului in care, dandu-se pozitiile a N orase se cere sa se calculeze cel mai scurt drum care, incepand dintr-un oras si trecand prin toate celelalte, se reintoarce in orasul de plecare. Numarul de astfel de drumuri este $(N\text{-}1)!/2$, din care trebuie selectat cel mai scurt. Calculatoarele electronice folosesc algoritmi secventiali care necesita un timp de calcul proportional cu $(N\text{-}1)!/2 \sim N^N = e^{N\log N}$. Timpul de rezolvare creste deci exponential cu dimensionalitatea sistemului, problema numindu-se nepolinomiala (NP).

PROBLEMA COMIS-VOIAJORULUI: Drumurile posibile pentru $N = 4$ sunt

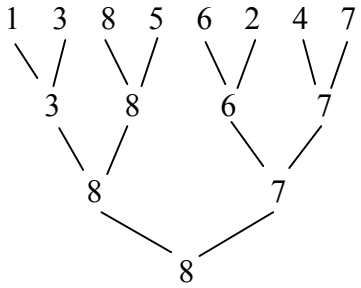


drum optim drumuri ne-optime

Daca mai adaug un oras, astfel incat $N = 5$, fiecare din orasele vechi se poate conecta cu el. Apar astfel inca patru drumuri aditionale; in general, pentru $N>>1$, numarul drumurilor este $(N\text{-}1)!/2$, factorul 1/2 indicand ca doar drumurile unidirectionale se numara. Conexiunile suplimentare pentru drumul optim din figura de mai sus sunt



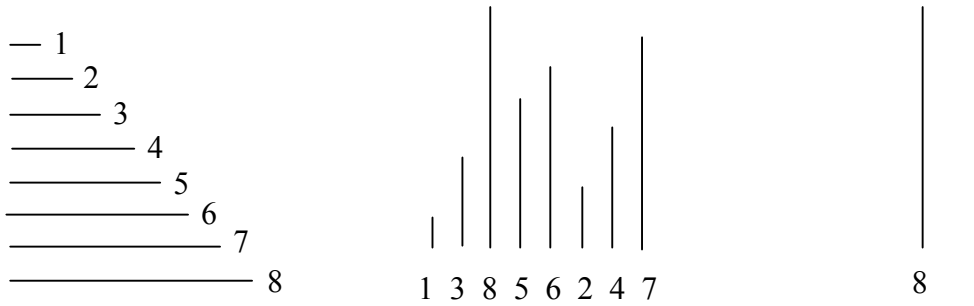
Alternative la calculatoarele electronice se refera la metode noi de calcul care se bazeaza pe paralelismul inerent unor noi medii sau noi algoritmi, in contrast cu calculatoarele si algoritmii secventiali. Interesul este de a se depasi unele limitari ale acestora din urma.



In particular, o modalitate de a evita cresterea exponentiala a timpului de calcul pentru probleme NP este calculul paralel. O problema care ilustreaza avantajul calculului paralel este cea a gasirii maximului din 8 numere, spre exemplu 1, 2, 3, 4, 5, 6, 7, 8. In calculul secvential se grupeaza cele opt numere in perechi, se determina numarul cel mai mare din fiecare pereche, apoi se regrupeaza rezultatele in noi perechi si se repeta procedura. Asa cum se vede din figura de mai sus, este nevoie de 7 pasi (etape) de calcul, timpul de calcul necesar pentru o cautare

secventiala pentru N mare crescand proportional cu N. Acesta este un exemplu de problema polinomiala. Calculul paralel presupune gasirea unei metode de a calcula simultan rezultatul primului pas (3, 8, 6, 7) din algoritmul secvential prezentat mai sus, la care s-ar adauga (pentru a gasi rezultatul final) inca doua etape de calcul corespunzand celui de-al doilea (8, 7) si al treilea (8) pas din calculul secvential. Ar rezulta astfel $\log_2 8=3$ pasi de calcul paralel in cazul problemei discutate, si un timp de calcul paralel T_p care este in general proportional cu $\log_2 N$. In particular, probleme NP, cu un timp de calcul secvential $T_s \sim e^{aN}$, devin probleme polinomiale P cu un timp de calcul paralel $T_p \sim \log_2 T_s \sim N$.

Si mai eficient este calculul paralel analog. De exemplu problema de mai sus poate fi rezolvata intr-un singur pas daca informatia este codata astfel incat numerelor li se asociaza bete cu lungime proportionala cu valoarea lor (vezi figura de mai jos). Atunci, selectarea se face alegand cel mai lung bat din secventa initiala.

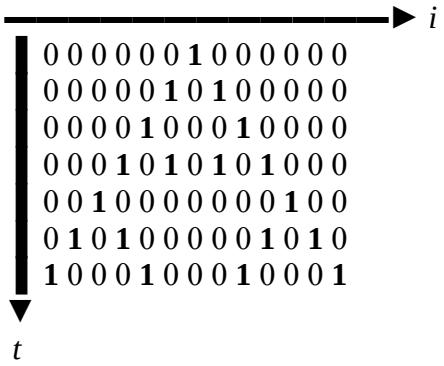


codare secventa initiala rezultat

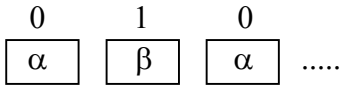
Observatie: Calculul paralel se poate efectua si cu calculatoare clasice. De exemplu, calculatoare de tipul single-instruction multiple-data stream efectueaza aceeasi operatie pe mai multe valori initiale in acelasi timp. Calculatoare mai sofisticate, cum ar fi Gray, sunt sisteme de multiprocesoare, in care fiecare calculator/procesor executa un program diferit pe date initiale proprii. Dar, diferitele procesoare trebuie sa fie sincronizate si interconectate eficient, ceea ce reprezinta o sarcina dificila daca am un numar mare de procesoare. De aceea se cauta solutii alternative.

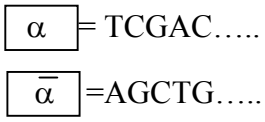
Printre tipurile de calculatoare care efectueaza calcul paralel se numara automatele celulare si calculatoarele moleculare, care le includ pe cele biologice si genetice. Se fac eforturi de asemenea pentru implementarea unor algoritmi de calcul paralel cu ajutorul sistemelor optice, avantajul fiind ca introducerea datelor initiale si prelucrarea acestora in paralel se face nu numai simplu (codand datele in pixelii unei imagini care este apoi prelucrata cu ajutorul unor sisteme optice ce simuleaza actiunea unui algoritm de calcul, astfel incat toti pixelii sunt prelucrati in paralel), dar si foarte rapid (lumina se propaga cu viteza c). Dezavantajul este ca sistemele optice sunt foarte sensibile la dezalinieri.

Automatele celulare se folosesc pentru discretizarea in spatiu si timp a simularii evolutiei unui sistem complex. Pasii de calcul in timp devin numere intregi, iar variabilele spatiale devin puncte, separate din nou de numere intregi. In aceste conditii variabilele campului (in general, continuu) devin concentrate in cateva stari discrete. In figura de mai jos se arata cum, folosind un automat celular a carui stare se reimprospateaza (update) in paralel conform regulii locale $x_i^{t+1} = (x_{i-1}^t + x_{i+1}^t) \bmod 2$, se poate genera o structura in spatiu si timp. Variabila i desemneaza locatiile spatiale discrete iar valoarea 1 corespunde unei valori nenule a campului.

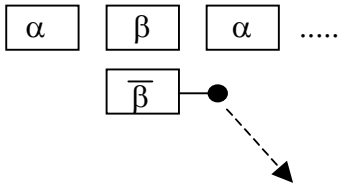


Un exemplu de calculator molecular este cel in care 0 si 1 sunt codate intr-o succesiune de secvente AND α si β, compuse fiecare din baze de tipul A (adenina), C (citozina), G (guanina), si T (timina), astfel incat bazele A si T, respectiv C si G sunt conjugate, in sensul ca se atrag reciproc.



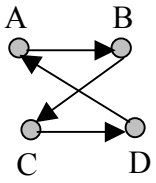


In exemplul de mai sus α si $\overline{\alpha}$ contin perechi de baze conjugate. Calculul paralel se implementeaza prin diverse procese chimice (reactii cu diferite enzime, etc.), iar rezultatul calculului, de exemplu secventa β, poate fi extrasa printr-o secventa care contine $\overline{\beta}$ atasata unui camp magnetic (pentru extragere selectiva cu ajutorul fortelor magnetice).



Calculatoarele electronice efectueaza in jur de 10^9–10^{10} operatii/s. Deoarece calculatoarele moleculare lucreaza cu reactii chimice, ele efectueaza un numar mult mai mic de operatii/s. Dar, pentru ca lucreaza in paralel (aceeasi reactie chimica avand loc simultan pentru foarte multe stari initiale (secvente ADN)), durata calculului = numarul de operatii paralele × timpul necesar fiecarui pas de calcul. Numarul operatiilor paralele fiind proportional cu numarul moleculelor, de ordinul 10^{19}, viteza de calcul a calculatoarelor moleculare poate fi mult mai mare ca a celor electronice (10^{19} fata de 10^{10}).

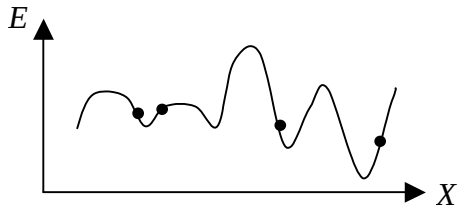
Interesant este ca desi calculatoarele moleculare, in special cele biologice, fac erori, aceste erori (daunatoare din punct de vedere biologic) pot fi folositoare in calcul. Spre exemplu, sa presupunem ca intr-un algoritm genetic al problemei comis-voiajorului drumurile pot fi codate printr-un sir X de 0 si 1. In particular, drumul



poate fi reprezentat prin combinatia

	A	B	C	D
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

unde 1 in primul rand din coloana A inseamna ca orasul A a fost vizitat primul, etc. In aceste conditii X este combinatia tuturor liniilor: X=(1000,0100,0010,0001). Daca lungimea drumurilor este transpusa in energie, care difera pentru diverse drumuri X, solutia problemei o pot gasi cautand printre populatiile diverselor specii care cauta pozitia de minim energetic in paralel pe curba din figura de mai jos (populatiile sunt reprezentate prin puncte pe curba).



Existenta mutatiilor genetice de tipul 10110 → 10010 (in care un 1 se transforma in 0) si a incrucisarilor de tipul
10110 → 10111
11111 → 11110
(in care grupurile 10 si 11 se schimba intre ele) face ca populatiile sa se schimbe si astfel sa poata iesi din pozitia de minim local si sa tinda spre minimul absolut al curbei.

Diversele solutii pentru calcul paralel prezentate mai sus (automate celulare, calcul biologic, genetic, etc.) nu schimba esenta calculului (in sensul ca starile initiale se codeaza tot prin 0 si 1 clar definite) ci doar permit implementarea unor algoritmi paraleli. Schimbarea majora intervine insa prin micsorarea componentelor electronice ale calculatoarelor pana se atinge limita in care dimensiunea sistemului devine comparabila cu lungimea de unda de Broglie $\lambda_{dB} = h/p = h/(mv)$ a electronilor. In acest caz natura cuantica a sistemului predomina. Deci, interesul pentru calculatoare cuantice este impus chiar de limitarile calculatoarelor actuale.

Pe de alta parte, densitatea de porti este in prezent limitata la 10^{10}porti/cm^2 din considerente de racire. In consecinta, o integrare mai avansata a componentelor unui calculator clasic necesita sau procedee mai eficiente de racire sau disipare mai redusa de energie per poarta logica. Aceasta din urma necesitate stimuleaza cercetarea pentru dispozitive logice partial sau total reversibile. Prin urmare, un alt motiv pentru folosirea unui calculator cuantic ar

fi consumul de energie, care poate fi minimizat in acest caz, deoarece in porti logice reversibile energia disipata pe ciclu logic poate tinde spre zero.

Calculul clasic se deosebeste esential de calculul cuantic, dupa cum reiese din tabelul de mai jos, care va fi explicitat in restul cursului

calcul clasic	**calcul cuantic**
bit: reprezentat prin 0 sau 1	qubit: superpozitie de stari $a\|0\rangle + b\|1\rangle$ care transmite doar un bit de informatie
un registru clasic cu *n* biti codeaza o singura stare	un registru cu *n* qubiti codeaza o superpozitie de 2^n stari
pot verifica in orice moment starea bitului in memorie	masuratori asupra qubitului dau ca rezultat $\|0\rangle$ sau $\|1\rangle$
bitii pot fi copiati (clonati)	o stare cuantica necunoscuta nu poate fi clonata
nu exista corelatie la distanta (entanglement) intre biti	exista corelatie la distanta (entanglement) intre qubiti. In consecinta, parti separate ale calculatorului pot sa nu fie neaparat legate fizic intre ele, si pot folosi teleportarea pentru a reproduce o stare cuantica necunoscuta
porti logice: operatori tip Boolean+porti ireversibile+biti de lucru (ancilla)	porti logice: operatori unitari si reversibili. Exista porti logice cuantice fara analog clasic
porti logice universale: NAND sau NOR	porti logice universale: CNOT+porti pentru un qubit
calcul secvential (in general)	calcul paralel
pot citi rezultatul calculului intr-un singur pas	trebuie sa folosesc interferenta cuantica pentru a citi eficient rezultatul

BITI CUANTICI (QUBITI)

Bitii cuantici (qubiti) sunt codati in stari diferite ale unui sistem cuantic (vezi seminarul cu exemple de qubiti) si difera de bitii clasici prin faptul ca starea unui sistem cuantic descris de o functie de unda complexa $|\Psi\rangle$ reprezinta, in general, o superpozitie de stari. Sistemul cuantic nu exista doar in starile "pure" $|0\rangle$ sau $|1\rangle$ ci si in starea de superpozitie $|\Psi\rangle = a|0\rangle + b|1\rangle$. (In calculul clasic sistemul se poate gasi doar in starea 0 sau 1.) Probabilitatea de a gasi sistemul in starile $|0\rangle$ sau $|1\rangle$ este $|\langle\Psi|0\rangle|^2 = |a|^2$, respectiv $|\langle\Psi|1\rangle|^2 = |b|^2$. Evident, $|a|^2 + |b|^2 = 1$, si vectorul de stare cuantic este de modul unitate in spatiul Hilbert. Aceasta proprietate face posibila definitia unei corespondente bijective intre qubitii de forma $|\Psi\rangle = \cos(\theta/2)|0\rangle + \exp(i\phi)\sin(\theta/2)|1\rangle$ si punctele de pe sfera unitate in R^3, unde θ si ϕ sunt coordonate sferice ale unui punct de pe sfera.

Observatie: Un qubit intr-o superpozitie de stari $|0\rangle$ si $|1\rangle$ nu poate fi interpretat ca fiind in $|0\rangle$ sau $|1\rangle$ cu o anumita probabilitate (aceasta ar corespunde definitiei starii mixte a unui ansamblu de sisteme cuantice), ci are proprietati diferite; pe langa modul, si faza relativa a coeficientilor *a* si *b* este importanta.

Starea sistemului cuantic poate fi reprezentata si ca un vector $\begin{pmatrix} a \\ b \end{pmatrix}$ (scris pentru compactitate $(a,b)^T$, unde ^{T} reprezinta operatia de transpunere) de modul unitate in spatiul bidimensional complex C^2 definit de $|0\rangle$ si $|1\rangle$. Starile $|0\rangle$ si $|1\rangle$ sunt ortogonale si formeaza baza de calcul. Baza de calcul nu este unic definita. De exemplu, daca $|\Psi\rangle$ reprezinta starea unui qubit si $|\Phi\rangle$ este perpendiculara pe aceasta, astfel incat $\langle\Phi|\Psi\rangle = 0$, se poate arata ca exista o transformare unitara unica care duce starile $|0\rangle$ si $|1\rangle$ in $|\Psi\rangle$ si $|\Phi\rangle$; acestea din urma pot juca rolul unei alte baze de calcul.

In plus, qubitii trebuie sa interactioneze intre ei, pentru a putea implementa operatii logice, si trebuie sa fie accesibili pentru manipulare externa pentru a permite citirea (masurarea) starilor initiala, finala, si de control.

Sistemele cuantice au un paralelism inerent pentru ca, supuse unei interactii (echivalentul unui algoritm de calcul), descriu evolutia tuturor starilor posibile simultan. Ele sunt deci ideale pentru calculul paralel in conditiile in care comportarea lor dinamica poate fi controlata in izolatie fata de mediul inconjurator si fata de propriile interactii interne care nu sunt utile in calcul. Interactiile nedorite, oricat de irelevante ar fi pentru un calculator clasic, produc perturbari ireversibile in cazul unui calculator cuantic. Aceste interactii produc ceea ce se numeste decoerenta, adica imposibilitatea reprezentarii starii sistemului printr-o superpozitie de $|0\rangle$ sau $|1\rangle$. Pentru a impiedica decoerenta qubitii nu se codeaza in sisteme fizice de dimensiuni macroscopice deoarece astfel de sisteme (de exemplu, superconductorii) nu pot fi izolate de propriile grade interne de libertate care sunt irelevante in calcul (care nu intervin in codare, si care pot interactiona unele cu altele). De aceea, qubitii se codeaza intr-un numar mic de stari ale unui sistem de dimensiuni atomice, in care gradele de libertate interne aditionale sau nu exista sau necesita energii mari pentru a fi excitate (separarea intre nivelele energetice discrete ale unui sistem de dimensiuni atomice este mult mai mare decat in cazul unui sistem de dimensiuni mari). Mai mult, in cazul in care apar erori datorate unor interactii nedorite, acestea pot fi corectate daca rata lor de aparitie este suficient de mica. Corectarea unor astfel de erori este dificila pentru ca, desi operatie de rutina in sisteme clasice, este foarte complicata in cazul cuantic pentru ca trebuie facuta fara a se cunoaste starea originala sau starea incorecta (gresita) a qubitului. Decoerenta limiteaza numarul de operatii permise: un calculator cuantic poate functiona doar atat timp cat decoerenta nu se instaleaza. Deoarece un pas de calcul se desfasoara intr-un interval de timp finit, numarul operatiilor este dat de raportul dintre timpul pana cand se instaleaza decorenta si timpul pasului de calcul. Numarul de operatii permise poate ajunge la 10^{13} intr-un sistem cuantic constand din ioni prinsi in capcana, pentru care timpul de decoerenta este de 0.1 s.

In general, se considera ca un calculator cuantic este mai puternic decat unul clasic in sensul teoriei complexitatii. De exemplu, daca am 20 de qubiti, starea cea mai generala a sistemului este o superpozitie de 2^{20} stari,

$$|\Psi\rangle = \sum_{x=0}^{2^{20}-1} a_x |x\rangle,$$

unde $|x\rangle = |x_1,...,x_{20}\rangle = |x_1\rangle \otimes ... \otimes |x_{20}\rangle$ cu x un numar in reprezentare binara si x_i = 0,1, simbolul $\otimes$ indicand produsul tensorial al qubitilor, care pot fi reprezentati printr-un vector cu doua componente. Evolutia in timp a acestui sistem este descrisa de o matrice de tip $2^{20}\times 2^{20}$.

In general, spatiul Hilbert a *n* qubiti este produsul $C^2 \otimes ... \otimes C^2 = C^{2^n}$ cu vectori de baza $|0...0\rangle, \ldots, |1...1\rangle$ si in acest spatiu $|a\rangle = |a_{n-1}...a_0\rangle$ reprezinta un registru cuantic cu *n* qubiti care codeaza valoarea $a = 2^0 a_0 + 2^1 a_1 + ... + 2^{n-1} a_{n-1}$. Un registru cu *n* qubiti codeaza o superpozitie de 2^n stari. Este greu de simulat un sistem cuantic de dimensiuni finite prin mijloace clasice fara o reducere exponentiala in viteza. Doar un calculator cuantic poate simula un sistem cuantic cu costuri de operatie polinomiale in timp si spatiu.

Atentie: starea unui sistem care codeaza 2 qubiti (o stare de tip 2-qubit) este diferita de produsul starilor care codeaza fiecare cate un qubit!

$$|\Psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle \rightarrow \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

poate fi scris ca produs de stari de tip 1-qubit

$$|\Psi\rangle = (a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle) = a_1a_2|00\rangle + a_1b_2|01\rangle + b_1a_2|10\rangle + b_1b_2|11\rangle) \rightarrow \begin{pmatrix} a_1a_2 \\ a_1b_2 \\ b_1a_2 \\ b_1b_2 \end{pmatrix}$$

doar daca $ad = cb$. Analog pentru stari de tip *n*-qubit. Tipurile de stari *n*-qubit care nu pot fi factorizate in *n* stari 1-qubit se numesc corelate (entangled).

In general, produsul tensorial intre un tensor de rang $n_1 \times m_1$ si un tensor de rang $n_2 \times m_2$ este un tensor de rang $n_1 n_2 \times m_1 m_2$. (Un vector este un caz particular de tensor de rang $n \times 1$, iar vectorul transpus este un tensor de rang $1 \times m$.)

Exemplu:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \otimes \begin{pmatrix} e & f & g \\ h & i & j \\ k & l & m \end{pmatrix} = \begin{pmatrix} ae & af & ag & be & bf & bg \\ ah & ai & aj & bh & bi & bj \\ ak & al & am & bk & bl & bm \\ ce & cf & cg & de & df & dg \\ ch & ci & cj & dh & di & dj \\ ck & cl & cm & dk & dl & dm \end{pmatrix} = \begin{pmatrix} a\begin{pmatrix} e & f & g \\ h & i & j \\ k & l & m \end{pmatrix} & b\begin{pmatrix} e & f & g \\ h & i & j \\ k & l & m \end{pmatrix} \\ c\begin{pmatrix} e & f & g \\ h & i & j \\ k & l & m \end{pmatrix} & d\begin{pmatrix} e & f & g \\ h & i & j \\ k & l & m \end{pmatrix} \end{pmatrix}$$

Calculatoarele cuantice lucreaza prin intermediul operatiilor reversibile *U* care transforma starea initiala a qubitilor intr-o stare finala folosind doar procese a caror actiune poate fi inversata. Aceste operatii sunt in acelasi timp liniare si unitare in sensul ca transforma o stare de modul unitate in spatiul Hilbert intr-o alta stare cu aceeasi proprietate. Singurul proces ireversibil in calculul cuantic este masuratoarea, care este in acelasi timp singura posibilitate de a extrage informatii asupra unui qubit dupa ce starea acestuia ia valoarea finala. Pentru a fi siguri ca rezultatul unui calcul cuantic se citeste doar dupa ce calculul s-a terminat, se foloseste un qubit aditional, care ia o valoare logica bine definita doar la sfarsitul calculului.

Astfel, citind/masurand starea qubitului aditional, se poate sti cand s-a terminat programul de calcul, si se poate apoi extrage informatia utila fara a perturba calculul cuantic.

O masuratoare asupra unei stari cuantice $|\Psi\rangle$ o perturba in sensul ca dupa masuratoare starea sistemului cuantic devine starea masurata, de exemplu $|00..1\rangle$. Pentru ca rezultatul calculului sa poata fi citit eficient (dintr-un numar cat mai mic de masuratori) majoritatea coeficientilor a_x din dezvoltarea $|\Psi\rangle = \sum_{x=0}^{2^n-1} a_x |x\rangle$ trebuie sa fie zero, sau foarte apropiati de zero; informatia utila trebuie sa fie codata in valori *x* care au o probabilitate apreciabila de a fi observate in masuratori. Doar in acest fel informatia poate fi usor interpretata, astfel incat sa nu conteze rezultatele rare si irelevante, cu probabilitate mica. Fenomenul de interferenta cuantica este deseori folosit pentru realizarea acestui scop.

Doar in cazul in care starea sistemului este identica cu una din cele 2^n stari ale bazei de calcul, adica doar daca $|\Psi\rangle = |x\rangle$, cu $a_x = 1$ si $a_y = 0$, $y \neq x$, rezultatul masuratorii este *x* cu probabilitate 1 si starea sistemului dupa masurare este identica cu cea de dinainte. Prin urmare, calculul cuantic poate simula calculul clasic reversibil daca permite ca intrari doar stari din baza de calcul, si daca transformarile unitare utilizate duc stari din baza de calcul in alte stari de acelasi tip (adica, daca nu creez prin transformarile unitare superpozitii netriviale de stari din baza de calcul).

Daca vreau sa masor starea unui singur qubit dintr-o secventa/stare *n*-qubit, functia de unda totala se scrie $|\Psi\rangle = a_0 |0\rangle |\psi_0\rangle + a_1 |1\rangle |\psi_1\rangle$, cu $|a_0|^2 + |a_1|^2 = 1$, $|\psi_{0,1}\rangle$ reprezentand starile normalizate, dar nu neaparat ortogonale ale celor $n-1$ qubiti nemasurati. In urma masuratorii, sistemul se va afla in starea $|x\rangle |\psi_x\rangle$. Probabilitatea masuratorii starii $|x\rangle$ este $|a_x|^2$, cu $\sum_{x=0}^{2^n-1} |a_x|^2 = 1$, qubitii nemasurati ramanand in starea $|\psi_x\rangle$. In general, o stare a $m+n$ qubiti este data de $|\Psi\rangle_{m+n} = \sum_x a_x |x\rangle_m |\psi_x\rangle_n$, cu $\sum_x |a_x|^2 = 1$ si $|\psi_x\rangle_n$ stari normalizate, nu neaparat ortogonale a *n* qubiti. Daca doar cei *m* qubiti din stanga sunt masurati, rezultatul este *x* cu probabilitatea $|a_x|^2$, dupa masuratoare $|\Psi\rangle_{m+n} \rightarrow |x\rangle_m |\psi_x\rangle_n$.

Un exemplu de masuratoare eficienta este: daca am un qubit in starea initiala $|\psi\rangle = 2^{-1/2}(|0\rangle + |1\rangle)$ si aplic poarta Hadamard ca instrument de masura (vezi porti cuantice), rezultatul este $H|\psi\rangle = |0\rangle$: deci obtin valoarea 0 cu probabilitate 1.

O alta diferenta fundamentala intre qubit si bit este ca primul (starea cuantica, in general) nu poate fi copiata. Aceasta proprietate are avantaje si dezavantaje: pe de o parte complica proiectarea unui calculator cuantic, dar pe de alta parte asigura siguranta transmiterii informatiei prin canale cuantice. Imposibilitatea copierii unui qubit este exprimata matematic prin asa-numita teorema "no-cloning". Demonstratia teoremei se bazeaza pe liniaritatea operatorilor. Mai precis, daca U_{copy} este un operator de copiere unitar in spatiul Hilbert $H = H_{orig} \otimes H_{copie}$ al originalului si copiei, actiunea de copiere a unei stari originale $|\Psi\rangle_{orig} = a|0\rangle + b|1\rangle$ intr-o copie a carei stare initiala este $|\Psi_0\rangle$ este reprezentata prin

$$U_{\text{copy}} : |\Psi\rangle_{orig} |\Psi_0\rangle \rightarrow |\Psi\rangle_{orig} |\Psi\rangle_{copie},$$

pentru orice state $|\Psi\rangle_{orig} \in H_{orig}$. Daca U_{copy} este liniar, inseamna ca

$U_{\text{copy}} \mid \Psi\rangle_{orig} \mid \Psi_0\rangle = a \mid 0\rangle \mid 0\rangle + b \mid 1\rangle \mid 1\rangle$.

Pe de alta parte, pentru ca operatorul sa realizeze copia starii originale, este necesar ca

$U_{\text{copy}} \mid \Psi\rangle_{orig} \mid \Psi_0\rangle = \mid \Psi\rangle_{orig} \mid \Psi\rangle_{orig} = a^2 \mid 0\rangle \mid 0\rangle + ab \mid 0\rangle \mid 1\rangle + ba \mid 1\rangle \mid 0\rangle + b^2 \mid 1\rangle \mid 1\rangle$.

Cele doua expresii sunt in general diferite, ceea ce inseamna ca nu exista un operator U_{copy} care sa copieze perfect o stare necunoscuta; se pot face doar copii apropiate de original.

Mai mult, doua stari nu pot fi aproximativ copiate (clonate) decat daca sunt aproximativ identice sau aproximativ ortogonale. Daca as clona starile $\mid \Psi\rangle$ si $\mid \Psi'\rangle$ astfel incat $U_{\text{copy}} \mid \Psi\rangle \mid \Psi_0\rangle = \mid \Psi\rangle \mid \Psi\rangle$ si $U_{\text{copy}} \mid \Psi'\rangle \mid \Psi_0\rangle = \mid \Psi'\rangle \mid \Psi'\rangle$, calculand produsul scalar dintre actiunile operatorului de copiere asupra acestor doua stari, as obtine egalitatea $\langle \Psi \mid \Psi' \rangle = \langle \Psi \mid \Psi' \rangle^2$ care nu poate fi satisfacuta, chiar si aproximativ, decat daca $\langle \Psi \mid \Psi' \rangle \cong 1$ sau $\langle \Psi \mid \Psi' \rangle \cong 0$.

MASINA TURING CLASICA

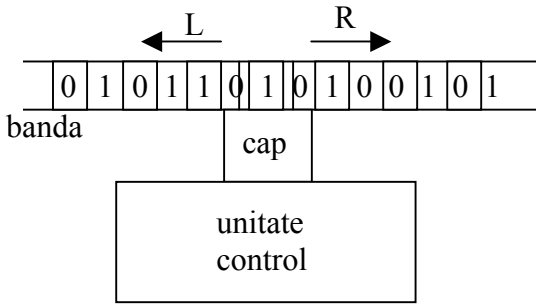
Ce este un calculator (clasic)? Raspunsul la aceasta intrebare vine ca raspuns la problema deciziei, care cauta solutie la intrebarea: Exista, cel putin in principiu, o metoda bine definita sau un proces care poate lua decizii referitor la toate problemele matematice?

Turing (1936), pentru a gasi o definitie la "metoda" din intrebarea de mai sus, incearca sa traduca procesele gandirii umane in actiuni mecanice. Acestea trebuie apoi inglobate intr-o "masina teoretica" care opereaza asupra simbolurilor, desenate ca siruri pe hartie, in acord cu reguli elementare bine definite. Turing afirma ca aceasta masina, denumita masina Turing, este capabila sa cuprinda tot ce s-ar putea intelege prin "metoda bine definita" din problema deciziei, "metoda" care defineste de fapt un algoritm.

O masina Turing functioneaza cu un set finit de stari $S = \{s_1,...,s_S; s_{S+1} = \text{stop}\}$, un alfabet finit de simboluri $A = \{a_1,...,a_A; a_{A+1} = \text{blank}\}$ si un set finit de instructiuni $I = \{i_1,...,i_I\}$, la care se adauga o banda infinit lunga de memorie. Starile s_i corespund la moduri de functionare a masinii, astfel incat masina Turing este in exact una dintre aceste stari la orice moment de timp. Simbolurile din *A* codeaza informatia prelucrata de masina: codeaza datele de intrare/iesire si pastreaza rezultatele operatiilor intermediare. Instructiunile sunt asociate cu stari din *S* si spun masinii ce actiune sa efectueze daca intalneste un simbol dat, si in ce stare sa fie dupa efectuarea acestei actiuni. Exista o stare "stop" in urma careia nu se efectueaza nici o instructiune, aceasta stare nefiind inclusa in numarul total de stari. Exista de asemenea un simbol "blank" care separa secventele de date codate prin restul simbolurilor alfabetice.

O masina Turing consta din trei componente:

1) o banda dublu infinita si divizata in sectiuni sau celule distincte. Fiecare celula contine doar un simbol $a_i \in A$
2) un cap sau cursor de citire/scriere care poate citi sau inscrie simbolul $a_i \in A$ in fiecare celula a benzii
3) o unitate de control, care controleaza miscarile capului de citire/scriere functie de starea curenta a masinii Turing si continutul celulei care este citita/scanata de capul de citire/scriere, adica functie de perechea (s_i, a_i)



Capul de citire/scriere poate efectua trei actiuni:

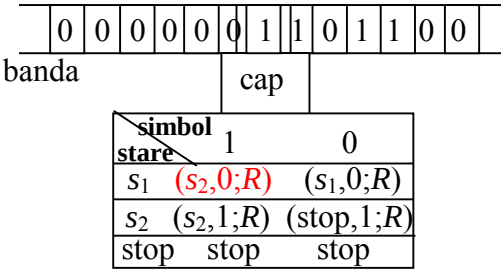
1) scierea sau stergerea de pe banda a continutului celulei care este scanata
2) schimbarea starii interne a masinii
3) miscarea capului cu o celula la stanga sau dreapta. Aceasta variabila o notam $\gamma \in \{L, R\}$

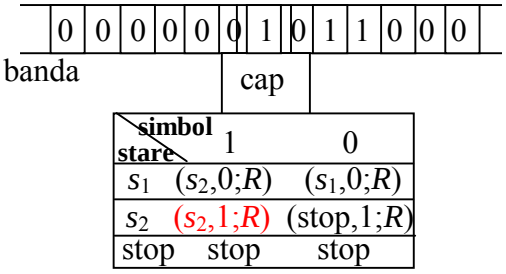
Comportarea masinii Turing este guvernata de setul de instructiuni I, care reprezinta regulile ce descriu tranzitia de la o pereche initiala (s_i, a_i) la o pereche finala (s_f, a_f) si miscarea descrisa de γ. Deci, fiecare instructiune $i \in I$ reprezinta

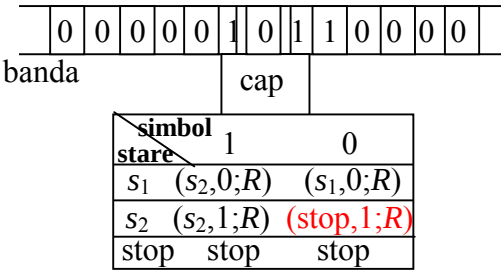
$$i : (s_i, a_i) \rightarrow (s_f, a_f; \gamma)$$

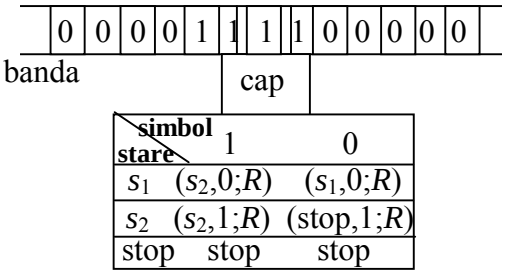
la care se adauga o conditie de consistenta: doua instructiuni $i_1, i_2 \in I$ nu pot avea acelasi set initial (s_i, a_i).

O masina Turing poate calcula functii complicate. Un exemplu simplu este cel din figura de mai jos, care indica modul in care, urmarind instructiunile din unitatea de control (cele care se efectueaza in fapt sunt inrosite) se efectueaza operatia 2+2=4. In acest caz 1 se foloseste pentru simbol, 0 pentru blank; numarul n este reprezentat ca o secventa de n simboluri 1 si nu prin reprezentarea binara!









Ipoteza (pentru ca nu poate fi demonstrata): orice functie ce se presupune in mod natural ca poate fi calculata, poate fi calculata pe o masina Turing.
In consecinta, o functie se numeste calculabila daca poate fi calculata pe o masina Turing si ne-calculabila in rest.

Se poate defini si conceptul de masina Turing universala, ca fiind o singura masina ce cuprinde toate celelalte masini Turing, si care poate deci calcula orice algoritm. In mod analog cu o masina Turing obisnuita, definita pe setul (*S*,*A*,*I*) cu *I* descris de $[(s_i, a_i),(s_f, a_f; \gamma)]$, masina Turing universala este definita pe setul (S_U, A_U, I_U) cu instructiunile $[(s_{Ui}, a_{Ui}),(s_{Uf}, a_{Uf}; \gamma_U)]$, care trebuie sa fie suficient de generale pentru a cuprinde orice masina Turing posibila. Teoretic se arata ca se poate intotdeauna construi o masina Turing universala cu $S_U = 2$ si un numar finit de simboluri, sau cu $A_U = 2$ si un numar finit de stari. In practica, in 1967 s-a construit o astfel de masina cu $S_U = 7$, $A_U = 4$.

Intrebare: se poate calcula orice functie printr-o proiectare corespunzatoare a unei masini Turing?
Raspuns: NU, pentru ca setul functiilor posibile este cu mult mai mare decat setul masinilor Turing posibile (pentru ca orice masina Turing poate fi codata intr-o secventa binara finita, dar exista un set de functii nenumarabile). Exemplu: setul *F* al tuturor functiilor $f : N \to N$. Daca *F* ar fi numarabila, as numerota functiile $F = \{f_0, f_1, ..., f_n, ...\}$. Apoi, as putea construi $g : N \to N$ cu $g(k) = f_k(k) + 1$. Deoarece $g(k)$ nu este continuta in *F* pentru ca difera cu cel putin o valoare a argumentului de fiecare dintre functiile in *F*, rezulta ca *F* nu este complet, deci ipoteza de plecare, ca *F* este numarabila, este gresita.

Desi s-au inventat mai multe tipuri de masini Turing, printre care cea nedeterminista (pentru care, pentru o pereche (s_i, a_i) exista un grup de stari finale posibile in loc de una singura), masina Turing este nepractica. Tot ce se poate calcula cu o masina Turing determinista se poate calcula si cu una nedeterminista si, in general, mai repede, dar masina este greu de implementat. Turing a propus construirea unei masini la sfarsitul celui de-al doilea razboi mondial, dar masina sa nu a fost realizata din cauza complexitatii, dar si din cauze birocratice. De asemenea o masina Turing ireversibila, in sensul ca nu se pot in general reconstrui datele initiale daca se da rezultatul final, poate fi simulata cu o masina Turing reversibila, dar care foloseste mai mult spatiu si mai mult timp de calcul.

Conceptul de masina Turing a fost in cele din urma inlocuit in 1945 de catre von Neumann, care a introdus calculatorul cu program stocat, care executa secvential programul inregistrat in memoria calculatorului. Acest precursor al calculatorului modern consta din:

1) procesor, in care informatia continuta in program este procesata secvential. Procesorul contine: a) o unitate de control, care controleaza fluxul de informatii: extrage date din memorie, executa si interpreteaza instructiuni, etc., b) registre, care contin partea de date care sunt procesate la un anumit moment de timp, si c) o unitate aritmetica si logica, care efectueaza calculele asupra datelor din registre sau memorie la cererea unitatii de control
2) memorie, care stocheaza datele si instructiunile in celule individuale, accesibile prin intermediul unor numere care se numesc adrese

In ciuda faptului ca este nepractica, masina Turing este un concept matematic esential pentru dezvoltarea calculatoarelor. Din fericire, operatiile masinii Turing pot fi inlocuite de

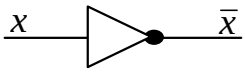
porti logice. O poarta logica este un circuit electronic care implementeaza, in calculatoarele clasice, o operatie logica care transforma o stare *n*-bit de la intrare intr-o iesire *m*-bit. Operatia logica este numita Booleana daca actioneaza doar asupra valorilor logice 0 si 1. Echivalenta intre masinile Turing si circuitele logice clasice este sustinuta de urmatoarea propozitie (in sens matematic): O problema din clasa polinomiala P, se poate rezolva pentru date de lungime *n* de o masina Turing in timp polinomial $p(n)$ daca si numai daca ii este asociata o familie de circuite cu un numar de porti de ordinul $p(n)\log p(n)$.

PORTI LOGICE IN CALCULATOARELE CLASICE

Singurul circuit/operator logic care se poate defini pe o stare de 1-bit este NOT. Tabela de adevar a acestui operator este

x	NOT x
0	1
1	0

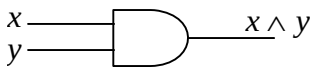
Operatorul logic NOT se mai noteaza si $\text{NOT } x = \bar{x} = 1 - x$ si cicuitul logic se reprezinta prin simbolul

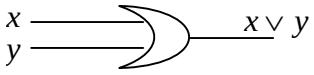


In plus, exista doua operatii logice de baza definite pe doi biti: AND si OR. Tabelele de adevar sunt date mai jos

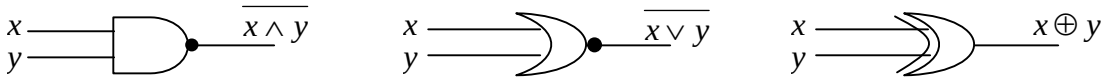
x	y	x AND y	x OR y
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

Aceste operatii logice se mai noteaza: $x \text{ AND } y = xy = x \wedge y$, $x \text{ OR } y = x + y - xy = x \vee y$, circuitele logice respective fiind reprezentate prin simbolurile:





Mai exista de asemenea si combinatii ale acestor porti, a caror reprezentari schematice sunt date in figurile de mai jos:



NAND NOR XOR

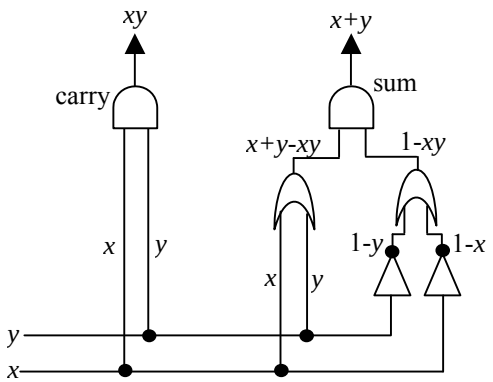
Poarta XOR (exclusive OR) are urmatoarea tabela de adevar:

x	y	x XOR y
0	0	0
0	1	1
1	0	1
1	1	0

si se mai noteaza ca x XOR $y = x \oplus y = x + y$, unde $\oplus$ indica operatia de adunare modulo 2. Descompunerea acestei operatii in operatiile logice de baza este urmatoarea (verificati tabela de adevar!):

$$x + y = (\overline{x} \wedge y) \vee (x \wedge \overline{y})$$

Un exemplu de implementare a operatiei de adunare folosindu-se portile descrise pana acum este ilustrat mai jos.



Din aceasta figura rezulta ca, in calculul clasic, apar porti care copiaza valorile initiale. Acestea se numesc porti COPY si corespund la multiplicarea firelor care poarta informatia.

Se poate arata ca orice functie logica se poate implementa cu ajutorul setului universal de porti logice {NOT, AND, OR}, la care se adauga poarta COPY si biti de lucru. Dar, acest set de porti logice nu este minimal. De exemplu, OR se poate exprima cu ajutorul portilor NOT si AND ca (vezi si primul curs):

$$x \vee y = \overline{(\overline{x} \wedge \overline{y})}.$$

Similar, AND se scrie ca o combinatie de porti OR si NOT in forma

$$x \wedge y = \overline{(\overline{x} \vee \overline{y})}.$$

Deci, pot alege ca set de porti universale {AND, NOT} sau {OR, NOT}. Se poate insa reduce si mai mult acest set, astfel incat doar una dintre portile NAND sau NOR este suficienta pentru a implementa orice functie logica.

Exemplu: $\bar{x} = 1 \text{ NAND } x$, $x \wedge y = \overline{(x \text{ NAND } y)} = 1 \text{ NAND } (x \text{ NAND } y)$

Desi poarta NAND actioneaza local pe doar doi biti, combinand diverse porti se pot efectua orice calcule pe un numar arbitrar de biti. Simplificarea portilor universale si reducerea lor la una permite concentrarea efortului pe crearea unui numar mic de dispozitive (porti) care apoi trebuie conectate corespunzator.

MASINA TURING CUANTICA

Masina Turing cuantica este, ca si varianta sa clasica, o constructie matematica. In anii 1980 s-a incercat folosirea mecanicii cuanticii pentru constructia masinii clasice Turing reversibile. Rezultatul nu este insa echivalent cu un calcul cuantic, deoarece doar o parte din operatorii de evolutie cuantici posibili duc reversibil stari clasice (intelese ca stari din baza de calcul) in alte stari clasice.

In 1985 Deutsch a aratat ca fiecare sistem fizic realizabil si finit poate fi simulat perfect cu un model universal de masina de calcul care opereaza cu resurse finite. O masina Turing cuantica cu un numar finit de stari are trei componente:

1) un procesor finit, care este unitatea de control ce consta dintr-un numar finit p de qubiti. Spatiul Hilbert al starilor procesorului este H_p
2) o unitate (banda) de memorie infinita, din care doar o portiune finita este folosita la orice moment de timp. Aceasta unitate consta dintr-un numar infinit de qubiti cu un spatiu Hilbert asociat H_m
3) un cursor, care este componenta ce asigura interactia intre unitatea de control si banda de memorie. Pozitia cursorului este descrisa de o variabila $x \in H_c = Z$, unde H_c este spatiul Hilbert asociat starilor cursorului

Spatiul Hilbert al starilor asociate unei masini Turing cuantice este $H_T = H_c \otimes H_p \otimes H_m$,

vectorii de baza fiind $|x; \boldsymbol{p}; \boldsymbol{m}\rangle = |x; p_0, ..., p_p; ..., m_{-1}, m_0, m_1, ...\rangle$.

Faptul ca se folosesc resurse finite inseamna ca masina nu poate efectua un numar infinit de operatii intr-un anumit moment de timp sau intr-o pozitie arbitrara de-a lungul benzii de memorie; doar un numar finit de qubiti din memorie participa la un singur pas de calcul.

Corespondenta cu masina Turing clasica este urmatoarea:

$S \rightarrow H_p$

$A \rightarrow$ spatiul qubitilor C^2

$I \rightarrow$ evolutiile unitare in timp a starii cuantice $|\Psi\rangle \in H_T$

Dupa un numar N de pasi de calcul de durata fixa T starea cuantica devine

$$|\Psi(nT)\rangle = U^n |\Psi(0)\rangle .$$

Fiecare operator liniar si unitar U (denumit astfel pentru ca transforma un vector de modul unitate in spatiul Hilbert intr-altul), care satisface relatia $UU^+ = U^+U = 1$ defineste o masina Turing.

Un program cuantic se deruleaza intr-un numar finit de pasi n pe o anumita masina Turing cuantica. Pentru a preciza starea initiala $|\Psi(0)\rangle$ se pune cursorul pe pozitia $x = 0$ si se alege starea procesorului $\boldsymbol{p} = \boldsymbol{0}$. Starile de memorie $\boldsymbol{m}$ se prepara introducand datele de intrare si instructiunile programului, codate intr-un numar finit de secvente de qubiti, ceilalti qubiti ai memoriei fiind setati pe $|0\rangle$. Starea initiala este deci

$$|\Psi(0)\rangle = \sum_{\boldsymbol{m}} a_{\boldsymbol{m}} |0;\boldsymbol{0};\boldsymbol{m}\rangle, \text{ cu } \sum_{\boldsymbol{m}} |a_{\boldsymbol{m}}|^2 = 1$$

Pentru a implementa un mecanism care sa opreasca masina la sfarsitul calculului (analog aducerii ei in stare stop la masina Turing clasica), si deoarece nu pot observa/masura operatia masinii pana cand se termina, aleg unul dintre qubitii procesorului ca sa indice sfarsitul. Astfel, $|q_0\rangle = 1$ cand calculul s-a terminat si $|q_0\rangle = 0$ in timpul operatiei. Programul nu interactioneaza cu $|q_0\rangle$ decat la sfarsit. Pot astfel citi periodic $|q_0\rangle$ fara a afecta functionarea masinii Turing cuantice.

Analog ca in calculul clasic, se poate defini o masina Turing cuantica reversibila, dar nici aceasta nici variante ale ei nu reprezinta puncte de plecare practice pentru proiectarea unui calculator cuantic. Primul pas in acest sens este descompunerea functiei de calculat in cele mai simple operatii primitive sau porti.

PORTI LOGICE CUANTICE

Singura operatie reversibila netriviala pe care o poate efectua un calculator clasic pe un singur bit este NOT. Din contra, operatiile reversibile pe care un calculator cuantic le poate efectua pe un singur qubit sunt toate transformarile liniare si unitare care transforma un vector de modul unitate intr-altul. Fiecare transformare unitara are o inversa, astfel incat aceste transformari sunt reversibile. Operatiile unitare pe starile 1-qubit se numesc porti 1-qubit. Operatia NOT este un caz particular al U care actioneaza asupra starilor 1-qubit. Aceasta are aceeasi tabela de adevar ca si operatia clasica, si poate fi reprezentata de o matrice cu elementele

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

Expresia acestei matrici rezulta din faptul ca $U_{\text{NOT}}|0\rangle = |1\rangle$, $U_{\text{NOT}}|1\rangle = |0\rangle$ si din reprezentarea starilor logice $|0\rangle$ si $|1\rangle$ ca $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$, respectiv $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ (vezi cursul de qubiti). O operatie mai generala, fara analog clasic, care genereaza o superpozitie netriviala de stari in baza de calcul $\{|0\rangle, |1\rangle\}$ este $U_A = 2^{-1/2} \exp(i\pi/4)(I_2 - i\sigma_x)$, cu I_2 matricea unitate (sau identitate) bidimensionala si σ_x una dintre matricile Pauli de spin, care sunt introduse in mecanica cuantica pentru a descrie momentul unghiular asociat unui electron cu spin 1/2:

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

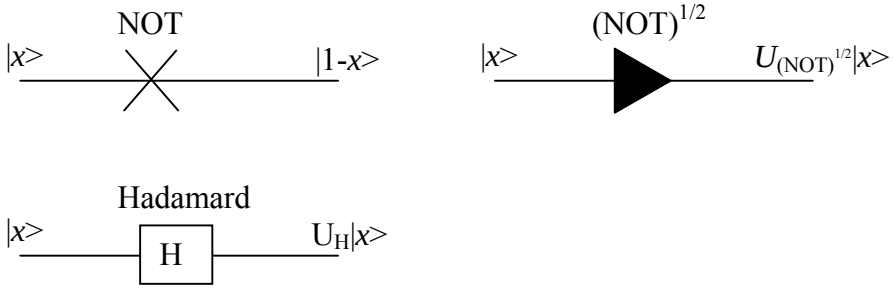
A se remarca ca $U_{\mathrm{NOT}} = \sigma_x$. Deoarece rezultatul aplicarii operatorului U_A de doua ori este NOT, acest operator poate fi identificat ca $U_A = U_{(\mathrm{NOT})^{1/2}}$ (scris si ca $U_{\sqrt{\mathrm{NOT}}}$). Operatorul unitate I_2 si cele trei matrici Pauli formeaza o baza pentru algebra matricilor bidimensionale, in sensul ca orice matrice bidimensionala se poate descompune ca o suma a acestor operatori: $U = U_0 I_2 + U_x \sigma_x + U_y \sigma_y + U_z \sigma_z$.

O alta poarta fara analog clasic care actioneaza asupra starilor 1-qubit este poarta Hadamard, care transforma starea $|0\rangle$ in $2^{-1/2}(|0\rangle + |1\rangle)$ si starea $|1\rangle$ in $2^{-1/2}(|0\rangle - |1\rangle)$. Actiunea acestei porti asupra bazei $\{|0\rangle, |1\rangle\}$ este descrisa de $U_{\mathrm{H}} = 2^{-1/2}(\sigma_x + \sigma_z)$, forma matriciala corespunzatoare fiind

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

Aplicand poarta Hadamard de doua ori se obtine operatorul identitate. Porti unitare 1-qubit se pot implementa usor pe fotoni polarizati cu ajutorul placilor $\lambda/2$ si $\lambda/4$, de exemplu.

Reprezentarile grafice ale portilor 1-qubit mentionate mai sus sunt:



Trebuie remarcat ca aplicand *n* operatori elementari cum ar fi poarta Hadamard asupra unui registru de *n*-qubiti se poate genera o stare care contine toate cele 2^n valori numerice posibile ale registrului, pe cand *n* operatori elementari aplicati unui registru clasic pot prepara doar o singura stare a registrului, care reprezinta un singur numar. Din acest motiv portile logice pot fi folosite nu numai pentru a implementa calculul cuantic ca atare, ci si pentru a defini starea initiala a sistemului, operatie care se numeste preparare. De exemplu, daca aplic fiecarui qubit in starea $|0\rangle|0\rangle = |00\rangle$ transformarea Hadamard, obtin

$$(H \otimes H)(|0\rangle|0\rangle) = (H|0\rangle)(H|0\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle).$$

Daca generalizez aceasta operatie la produsul tensorial de *n* ori a transformarii Hadamard (operatorul $H^{\otimes n}$ aplicat starii $|0\rangle_n$ a *n* qubiti) rezultatul este o superpozitie cu ponderi egale a tuturor starilor posibile a *n* qubiti:

$$H^{\otimes n}|0\rangle_n = \frac{1}{2^{n/2}} \sum_{0 \le x < 2^n} |x\rangle_n$$

Cea mai generala operatie reversibila pe o stare de *n* biti intr-un calculator clasic este permutarea celor 2^n stari distincte, rezultatul fiind $(2^n)!$ operatii posibile distincte. Cea mai generala operatie reversibila pe care un calculator cuantic o poate efectua pe o stare *n*-qubit este o transformare unitara 2^n dimensionala. Mai precis, operatorul actioneaza in spatiul Hilbert C^{2^n} si poate fi reprezentat printr-o matrice unitara $2^n \times 2^n$. Aceste transformari se numesc porti *n*-qubit.

Una dintre cele mai utile porti in calculul cuantic este operatia CNOT (control-NOT) care schimba starea unui qubit (qubit tinta sau qubit semnal) doar daca un alt qubit (qubitul de control) este in starea $|1\rangle$. Aceasta poarta actioneaza asupra starilor de tip 2-qubit conform regulii:

$U_{\text{CNOT}}|00\rangle = |00\rangle$, $U_{\text{CNOT}}|01\rangle = |01\rangle$, $U_{\text{CNOT}}|10\rangle = |11\rangle$, $U_{\text{CNOT}}|11\rangle = |10\rangle$, astfel incat

$$U_{\text{CNOT}} = U_{\text{XOR}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Observatie: forma de mai sus a matricii pentru U_{CNOT} corespunde cazului cand bitul de control este la stanga (ca rezultat, se schimba intre ele starile $|10\rangle$ si $|11\rangle$). Daca bitul de control este cel din dreapta, starile care se schimba intre ele sunt $|01\rangle$ si $|11\rangle$ si forma matriciala a operatorului este

$$U_{\text{CNOT}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

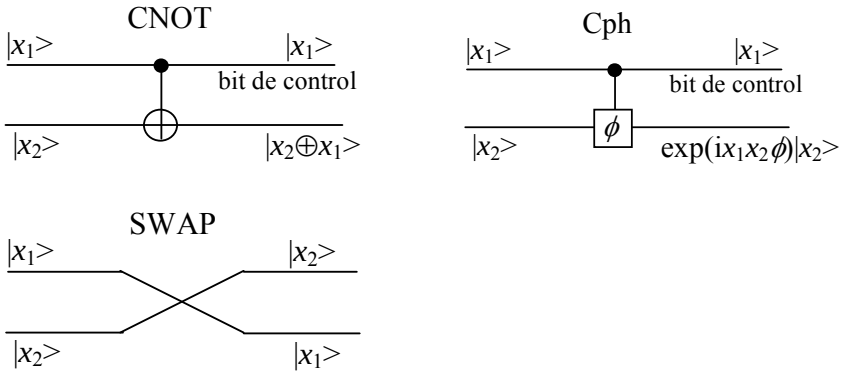
Poarta CNOT poate fi exprimata in functie de matricile Pauli ca
$U_{\text{CNOT}} = |0\rangle\langle 0| \otimes I_2 + |1\rangle\langle 1| \otimes U_{\text{NOT}} = 2^{-1}(1+\sigma_z) \otimes I_2 + 2^{-1}(1-\sigma_z) \otimes \sigma_x$.

Alte porti care actioneaza asupra starilor 2-qubiti sunt poarta Cph (controlled phase) care nu are analog clasic, poarta SWAP, care schimba intre ele starile a doi qubiti, si poarta $(\text{SWAP})^{1/2}$. Matricile corespunzatoare acestor porti sunt:

$$U_{\text{Cph}}(\phi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \exp(i\phi) \end{pmatrix}, \qquad U_{\text{SWAP}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

$$U_{\sqrt{\text{SWAP}}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & (1+i)/2 & (1-i)/2 & 0 \\ 0 & (1-i)/2 & (1+i)/2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Portile 2-qubiti CNOT, Cph si SWAP au urmatoarele reprezentari grafice:

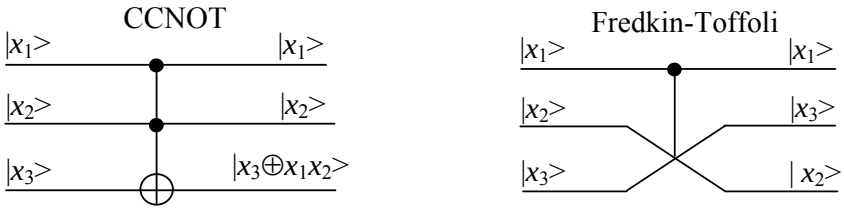


Punctul negru (cerc innegrit) in portile CNOT si Cph indica faptul ca actiunea portii are loc doar daca bitul de control ia valoarea 1; pentru o actiune definita pentru valoarea 0 a bitului de control cercul este alb.

Exemple de porti care actioneaza asupra starilor 3-qubiti sunt poarta Toffoli (numita si CCNOT sau C^2NOT), poarta Fredkin (sau CSWAP (controlled SWAP)) si poarta Fredkin-Toffoli (tot o poarta de tip CSWAP). Poarta Toffoli schimba valoarea logica a bitului semnal (de la 0 la 1 sau de la 1 la 0, respectiv) daca cei doi biti de control au valoarea 1, pe cand poarta Fredkin-Toffoli, a carei tabela de adevar este data mai jos, schimba intre ele valorile logice ale bitilor bit2 si bit3 cand bit1 este 1. Poarta Fredkin, definita printr-o tabela de adevar similara cu cea a portii Fredkin-Toffoli, schimba intre ele valorile logice a doi qubiti de intrare daca valoarea bitului de control este 0.

bit1	bit2	bit3	Fredkin-Toffoli		
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	1	1	1

Reprezentarile grafice ale portilor CCNOT si Fredkin-Toffoli sunt date mai jos:



Un set de porti cuantice este universal daca orice operatie unitara U_N pe N stari cuantice de intrare se poate descompune intr-un produs finit de actiuni successive ale portilor

universale pe diferite sub-seturi de qubiti de intrare. Exemplu: o matrice generica $k \times k$ cu $k \geq 2$ se poate reprezenta ca un produs a $k(k-1)/2$ matrici unitare ce descriu evolutia unor sisteme cu doua nivele (care au in compozitie o matrice 2×2 netriviala).

Deoarece calculul cuantic este reversibil si cel clasic nu este, portile clasice universale nu sunt neaparat universale pentru calculul cuantic. In particular, s-a demonstrat ca orice poarta logica poate fi descompusa intr-o succesiune de doua porti cuantice elementare: o rotatie arbitrara (o poarta) 1-qubit si o poarta CNOT tip 2-qubit. De exemplu, operatia SWAP S_{ij} poate fi descompusa in operatii CNOT de tipul C_{ij} (unde i este bitul de control si j cel tinta/semnal) ca $S_{ij} = C_{ij}C_{ji}C_{ij}$, iar operatia care schimba intre ei bitii de control si tinta in CNOT poate fi descompusa intr-o succesiune de operatii Hadamard H_i aplicate bitului i si operatii CNOT ca $C_{ji} = (H_iH_j)C_{ij}(H_iH_j)$.

Demonstratia ca orice matrice unitara se poate descompune in porti 1-qubit si porti CNOT:
Primul pas: demonstrez ca matricile unui sistem cuantic cu doua nivele sunt universale. Pentru usurinta demonstratiei, luam un caz particular pentru matrici 3×3. Forma generala a unei astfel de matrici unitare este

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

Se poate arata ca pot construi matricile U_1, U_2, U_3 care descriu evolutii ale unui sistem cu doua nivele (carte au in componenta o submatrice netriviala 2×2), astfel incat $U_3U_2U_1U = I_3$, cu I_3 matricea unitate. In aceste conditii U se poate exprima ca produs al celorlalte matrici: $U = U_1^+U_2^+U_3^+$. Matricile U_1, U_2, U_3 se construiesc astfel incat una dintre liniile sau coloanele lui U devine zero (cu exceptia unui termen), procedura continuand pentru matrici de rang mai mare.

Matricea U_1 se construieste astfel: daca $b = 0$, $U_1 = I_3$, altfel, daca $b \neq 0$,

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

In urma acestei alegeri, $U_1U = \begin{pmatrix} a' & d' & g' \\ 0 & e' & h' \\ c' & f' & j' \end{pmatrix}$

Matricea U_2 se construieste astfel: daca $c' = 0$ (care implica $|a'| = 1$ datorita unitaritatii), $U_2 = \begin{pmatrix} a'^* & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$. Altfel, daca $c' \neq 0$,

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

Cu aceasta alegere $U_2U_1U = \begin{pmatrix} 1 & d'' & g'' \\ 0 & e'' & h'' \\ 0 & f'' & j'' \end{pmatrix}$, ceea ce implica, datorita unitaritatii, ca $d''=g''=0$.

Matricea U_3 este atunci $U_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & e''^* & f''^* \\ 0 & h''^* & j''^* \end{pmatrix}$, astfel ca $U_3U_2U_1U = I_3$.

Pasul doi: orice matrice unitara cu doua nivele ce actioneaza asupra a *n* qubiti se poate implementa folosind porti 1-qubit si porti CNOT. Exemplu: daca *U* este de tipul

$$U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a & b \\ 0 & 0 & c & d \end{pmatrix}, \quad M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

U actioneaza netrivial doar asupra ultimului qubit, cand toti ceilalti sunt 1. Deci *U* se implementeaza prin poarta controlled-*M*, unde *M* este o poarta 1-qubit, conditionata de toti ceilalti qubiti, care trebuie sa ia valoarea 1.

De observat ca poarta CNOT clasica nu este universala; de asemenea poarta universala clasica NAND (sau NOR) nu este universala pentru calculul cuantic pentru ca nu este reversibila, in acelasi timp portile 2-bit reversibile nefiind suficiente pentru implementarea calculului clasic. Mai exista de asemenea si porti universale care actioneaza asupra starilor 3-qubit; un exemplu in acest sens este poarta Toffoli sau poarta Fredkin-Toffoli. Portile universale trebuie in general sa fie asociate cu biti de lucru (ancilla) (vezi exemplele de mai jos referitor la poarta Toffoli) si cu biti de iesire (garbage) care nu sunt necesari in restul calculului, pentru a implementa operatii reversibile.

Un exemplu interesant de calcul este implementarea cuantica (deci reversibila) a operatiei AND (care este ireversibila in calculul clasic) prin includerea ei intr-o poarta Toffoli (verificati cu ajutorul tabelei de adevar):

$$U_{\text{CCNOT}} \,|\, x_1, x_2, x_3 = 0\rangle = |\, x_1, x_2, x_1 \wedge x_2\rangle$$

Analog, poarta clasica NAND devine reversibila inclusa intr-o poarta Toffoli de tipul

$$U_{\text{CCNOT}} \,|\, x_1, x_2, x_3 = 1\rangle = |\, x_1, x_2, \overline{x_1 \wedge x_2}\rangle .$$

Chiar si poarta clasica NOT poate fi implementata cu ajutorul portii Toffoli:

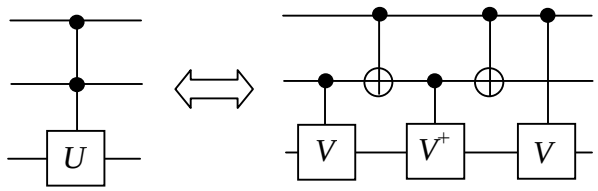
$$U_{\text{CCNOT}} \,|\, 1,1, x_3\rangle = |\, 1,1, \bar{x}_3\rangle ,$$

desi o astfel de implementare a portii NOT este o complicatie inutila.
O poarta COPY clasica poate fi simulata de asemenea de o poarta Toffoli pentru urmatoarea alegere a intrarilor:

$$U_{\text{CCNOT}} \,|\, 1, x_2, 0\rangle = |\, 1, x_2, x_2\rangle .$$

Acest fapt nu contrazice teorema “no-cloning” fiindca poarta Toffoli nu cloneaza superpozitii netriviale de stari ale bazei de calcul. In exemplele de mai sus, bitii de intrare sau iesire cu valori fixe, sunt biti ancilla, respective garbage; ei au fost introdusi doar pentru a obtine rezultatul dorit, si nu sunt utili in restul calculului. Deoarece poarta Toffoli poate simula poarta clasica universala NAND, ea poate simula orice circuit clasic.

Demonstram ca poarta Toffoli se poate implementa cu porti CNOT si porti controlled-*V*, cu *V* operator unitar 1-qubit. Mai exact, daca *U* este un operator unitar 1-qubit, astfel incat $U = V^2$, CCU (double-controlled-*U*) se implementeaza ca



Aceasta echivalenta se demonstreaza urmarind transformarea starii initiale dupa fiecare pas:

$|00x\rangle \to |00x\rangle \to |00x\rangle \to |00x\rangle \to |00x\rangle \to |00x\rangle$

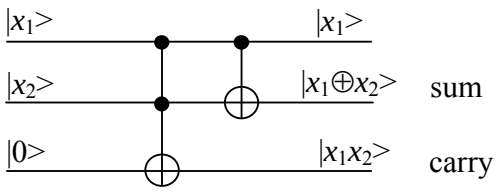
$|01x\rangle \to |01Vx\rangle \to |01V^+Vx\rangle = |01x\rangle \to |01x\rangle \to |01x\rangle \to |01x\rangle$

$|10x\rangle \to |10x\rangle \to |11x\rangle \to |11V^+x\rangle \to |10V^+x\rangle \to |10VV^+x\rangle = |10x\rangle$

$|11x\rangle \to |11Vx\rangle \to |10Vx\rangle \to |10Vx\rangle \to |11Vx\rangle \to |11VVx\rangle = |11Ux\rangle$

Daca $V = (1-i)(I_2 + i\sigma_x)/2$, $V^2 = U = \sigma_x = \text{NOT}$, si obtin circuitul echivalent al portii Toffoli.

O implementare a operatiei de adunare intr-un calculator cuantic:



Observatie: In general nu se poate evalua direct actiunea lui *f* prin intermediul unui operator unitar care sa transforme $|x\rangle$ in $|f(x)\rangle$ datorita unitaritatii calculului care implica conservarea ortonormalitatii in urma unei transformari unitare. Mai exact, este posibil in general ca doua stari $|x\rangle$ si $|x'\rangle$ care sunt initial ortogonale, sa evolueze in doua stari $|f(x)\rangle$ si $|f(x')\rangle$ care nu sunt ortogonale.

De aceea, daca reprezint *x* ca un *n*-qubit si $f(x)$ (rezultatul actiunii asupra lui *x* a functiei *f*) ca un intreg *m*-qubit, calculatorul cuantic are nevoie de cel putin $n+m$ qubiti: am nevoie de un registru de intrare de tip *n*-qubiti ca sa reprezint *x* si de un registru de iesire *m*-qubit ca sa reprezint $f(x)$. (Daca $f(x)$ nu este un numar intreg, pot sa il aproximez ca atare, precizia aproximarii crescand cu *m*; evident $f(x)$ trebuie sa fie mai mic decat 2^m.) In registrul

cuantic cu starea $|\Psi_r\rangle_{n+m} = |x\rangle_n \otimes |y\rangle_m = |x\rangle_n |y\rangle_m$, $|x\rangle_n$ contine datele de intrare si $|y\rangle_m$ codeaza rezultatul evolutiei cuantice a sistemului sau rezultatul aplicarii unor porti logice.

Pentru a defini o transformare unitara *U* trebuie precizata doar actiunea sa pe o baza data, pentru ca orice stare $|\Psi\rangle$ a unui sistem cuantic se scrie ca o superpozitie de stari ale bazei de calcul. In particular, rezultatul calculului functiei $f(x)$ se poate reprezenta ca

$$U_f(|x\rangle_n |y\rangle_m) = |x\rangle_n |y \oplus f(x)\rangle_m,$$

adica $U_f(|x\rangle_n |0\rangle_m) = |x\rangle_n |f(x)\rangle_m$ daca starea initiala a registrului de iesire este 0 si $U_f(|x\rangle_n |1\rangle_m) = |x\rangle_n |1 - f(x)\rangle_m$ daca starea initiala este 1. Alternativ, pentru o functie Booleana *f*, care ia doar valori 0 sau 1, $U_f(|x\rangle_n |y\rangle_m)$ schimba starea qubitilor din registrul de iesire daca *f* aplicat qubitilor din registrul de intrare este 1, si ii lasa neschimbati daca rezultatul lui *f* asupra intrarii este 0. Din aceasta reprezentare rezulta ca, pentru orice valoare initiala *y* a registrului de iesire, registrul de intrare isi pastreaza forma $|x\rangle_n$. In particular, pentru $n = m = 1$, adica pentru functii definite pe setul 1-qubit {0,1} cu valori pe setul {0,1},

$$U_f : \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)|0\rangle = \frac{|0, f(0)\rangle + |1, f(1)\rangle}{\sqrt{2}}$$

deci calculez $f(0)$ si $f(1)$ intr-un singur pas, qubitul de intrare $(|0\rangle + |1\rangle)/\sqrt{2}$ fiind obtinut prin aplicarea portii Hadamard asupra starii $|0\rangle$. Acest lucru nu implica neaparat eficienta calculului cuantic, pentru ca trebuie, in plus, sa ma asigur ca pot extrage informatia eficient, adica rezultatul unei singure masuratori asupra starii sistemului cuantic trebuie sa fie suficient pentru a obtine informatia relevanta.

Analog, pentru $n \neq 1$, $m \neq 1$, daca aplic transformarea asupra superpozitiei cu ponderi egale a tuturor starilor posibilie *n*-qubit obtinute prin aplicarea portii Hadamard obtin

$$U^f (H^{\otimes n} \otimes I_m)(|0\rangle_n |0\rangle_m) = \frac{1}{2^{n/2}} \sum_{0 \le x < 2^n} U_f(|x\rangle_n |0\rangle_m) = \frac{1}{2^{n/2}} \sum_{0 \le x < 2^n} |x\rangle_n |f(x)\rangle_m$$

toate cele 2^n valori ale functiei $f(x)$ fiind generate intr-un singur pas. Rezultatul calculului paralel cuantic asupra starilor $|x\rangle_n$ este similar cu un calcul asupra unei stari clasice $|x\rangle_n$ in registrul de intrare, cu $|x\rangle_n$ aleator. Diferenta este ca selectia aleatoare a starii de intrare asupra careia se aplica functia *f* este facuta in cazul cuantic doar dupa ce calculul a fost efectuat.

Transformarea U_f este inversabila:

$$U_f U_f(|x\rangle |y\rangle) = U_f(|x\rangle |y \oplus f(x))) = |x\rangle |y \oplus f(x) \oplus f(x)\rangle = |x\rangle |y\rangle$$

ALGORITMI CUANTICI

Algoritmii cuantici sunt in general potriviti pentru a studia proprietati globale ale unei functii sau ale unei secvente de date (de exemplu, pentru a gasi perioada unei functii, media unei secvente, etc.), si nu pentru a gasi detalii. De exemplu, daca se doreste gasirea valorii unei functii pentru o valoare data a argumentului, nu exista nici un avantaj in folosirea calculului cuantic pentru ca aceasta valoare trebuie extrasa dintr-o superpozitie, ceea ce implica masurarea repetata a rezultatului pentru a compensa starile de iesire cu probabilitate mica.

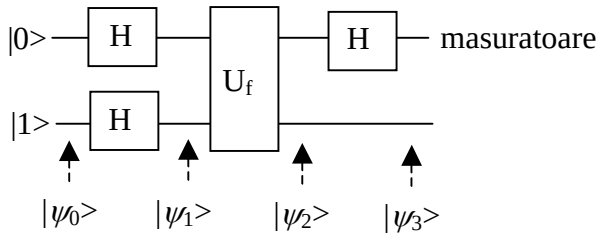
ALGORITMUL DEUTSCH-JOZSA

Cel mai simplu exemplu de algoritm cuantic care intrece in eficienta un algoritm clasic este algoritmul Deutsch-Jozsa (1985-1992). Acest algoritm distinge intre o functie *f*: $\{0,1\}\rightarrow\{0,1\}$ care este constanta si una care este balansata (balanced). O functie se numeste balansata daca ia valoarea 0 pentru exact jumatate din valorile posibile ale argumentului si valoarea 1 pentru cealalta jumatate; in cazul nostru functia este balansata daca $f(0) \neq f(1)$.

Sa presupunem ca avem o masina cuantica (cutie neagra) care calculeaza $f(x)$, adica este descrisa de transformarea 2-qubit unitara

$$U_f(|x\rangle|y\rangle) = |x\rangle|y \oplus f(x)\rangle$$

care schimba valoarea celui de-al doilea qubit daca $f(x) = 1$. Vreau sa stiu daca $f(0) = f(1)$ sau nu. Daca as avea un calculator clasic cu intrari 0 si 1, trebuie sa accesez cutia neagra de doua ori, pentru $x = 0$ si $x = 1$, ca sa gasesc raspunsul (pentru ca sa calculez $f(0)$ si $f(1)$ si apoi sa le compar). Dar, daca am la intrare o superpozitie coerenta a acestor stari, o singura accesare (un singur pas de calcul) este suficienta. Circuitul cuantic care ofera raspunsul la intrebarea asupra constantei lui *f* este dat mai jos (H reprezinta porti Hadamard iar U_f este poarta care calculeaza valoarea functiei *f*).



De-a lungul circuitului, in urma transformarilor suferite, functia de unda incidenta $|\psi_0\rangle$ ia urmatoarele valori (in pozitiile indicate prin sageti in figura de mai sus)

$$|\psi_0\rangle = |0\rangle|1\rangle = |01\rangle,$$

$$|\psi_1\rangle = (H \otimes H)|01\rangle = \frac{1}{2}(|0\rangle + |1\rangle)(|0\rangle - |1\rangle),$$

daca aplic cate o transformare Hadamard fiecarui bit in parte. Valoarea

$$|\psi_2\rangle = \begin{cases} \pm\frac{1}{2}(|0\rangle+|1\rangle)(|0\rangle-|1\rangle), \text{ daca } f(0)=f(1) \\ \pm\frac{1}{2}(|0\rangle-|1\rangle)(|0\rangle-|1\rangle), \text{ daca } f(0)\neq f(1) \end{cases}$$

se obtine tinand cont de faptul ca, in cazul in care al doilea qubit $|y\rangle$ are forma particulara $(|0\rangle-|1\rangle)/\sqrt{2}$, tranformarea unitara U_f poate fi exprimata si ca

$$U_f(|x\rangle,(|0\rangle-|1\rangle)/\sqrt{2}) = (-1)^{f(x)}|x\rangle(|0\rangle-|1\rangle)/\sqrt{2}$$

Acest lucru se verifica usor: daca $f(x)=0$, $U_f(|x\rangle,|y\rangle)=|x\rangle|y\rangle$, iar daca $f(x)=1$, $U_f(|x\rangle,(|0\rangle-|1\rangle)/\sqrt{2}) = |x\rangle(|1\rangle-|0\rangle)/\sqrt{2} = -|x\rangle(|0\rangle-|1\rangle)/\sqrt{2}$) si, tinand cont de linearitatea transformarii: $U_f(|0\rangle+|1\rangle)/\sqrt{2}\,|y\rangle = 2^{-1/2}(U_f(|0\rangle|y\rangle)+U_f(|1\rangle|y\rangle))$.

In final, a treia poarta Hadamard aplicata primului qubit da

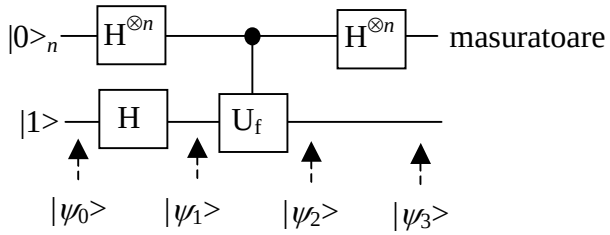
$$|\psi_3\rangle = \begin{cases} \pm|0\rangle(|0\rangle-|1\rangle)/\sqrt{2}, \text{ daca } f(0)=f(1) \\ \pm|1\rangle(|0\rangle-|1\rangle)/\sqrt{2}, \text{ daca } f(0)\neq f(1) \end{cases}$$

expresie care se poate verifica usor, si care se poate scrie si ca

$$|\psi_3\rangle = \pm|f(0)\oplus f(1)\rangle(|0\rangle-|1\rangle)/\sqrt{2}\,.$$

Deci, un algoritm cuantic poate evalua $f(0)\oplus f(1)$ intr-un singur pas, rezultat care nu poate fi obtinut clasic decat in doi pasi. Mai precis, daca in urma masuratorii primului qubit din $|\psi_3\rangle$ se obtine valoarea logica 0, functia este cu certitudine (cu probabilitate unitate) constanta deoarece $f(0)\oplus f(1)=0$ doar daca $f(0)$ si $f(1)$ sunt ambele zero sau 1 (vezi tabela de adevar a operatiei XOR). Din contra, daca masuratoarea primului qubit din $|\psi_3\rangle$ da valoarea logica 1, functia este balansata. Desi stiu daca functia este constanta sau balansata, nu cunosc valoarea functiei pentru cele doua argumente, adica nu stiu separat valoarea $f(0)$ sau $f(1)$.

Eficienta acestui algoritm pentru cazul in care *f* este definit pe stari 1-qubit, nu este spectaculoasa. Adevarata valoare practica a algoritmului cuantic fata de cel clasic se poate vedea pentru functii Booleene *f* definite pe stari *n*-qubiti, cu valori 0 sau 1. Mai precis, ma intereseaza daca $f:\{0,1\}^n\to\{0,1\}$ este constanta sau balansata, unde $\{0,1\}^n$ indica un argument (un registru de intrare) compus din *n* qubiti, fiecare putand lua valoarea 0 sau 1. Intr-un algoritm clasic am nevoie, in general, de $1+2^{n-1}$ pasi ca sa raspund la aceasta intrebare.



Intr-un algoritm cuantic, implementat de circuitul de mai sus, raspund la intrebare intr-un singur pas, cresterea vitezei de calcul fiind exponentiala. Similar cu varianta algoritmului Deutsch-Jozsa pentru cazul $n = 1$, functia de unda la diverse pozitii de-a lungul circuitului este data de:

$$|\psi_0\rangle = |0\rangle_n |1\rangle$$

unde $|0\rangle_n = |\underbrace{00...0}_{n \text{ ori}}\rangle$ este qubitul/registrul de intrare, indicele n indicand ca este vorba despre un n-qubit, iar $|1\rangle$ este starea/registrul de iesire de tip 1-qubit.

$$|\psi_1\rangle = (H^{\otimes n} \otimes H)|0\rangle_n |1\rangle = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle_n (|0\rangle - |1\rangle)/\sqrt{2}$$

deoarece poarta Hadamard $H^{\otimes n}$ aplicata starii $|0\rangle_n$, astfel incat fiecarui qubit ii este aplicat individual poarta Hadamard H produce starea

$$H^{\otimes n}|0\rangle_n = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle_n$$

si poarta Hadamard aplicata starii $|1\rangle$ produce $(|0\rangle - |1\rangle)/\sqrt{2}$.

Expresia $H^{\otimes n}|0\rangle_n = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle_n$ se poate demonstra usor, pornind de la cazuri particulare. De exemplu, pentru $n = 2$,

$$H^{\otimes 2}|0\rangle_2 = (H|0\rangle)(H|0\rangle) = 2^{-1}(|0\rangle + |1\rangle)(|0\rangle + |1\rangle) = 2^{-1}(|00\rangle + |01\rangle + |10\rangle + |11\rangle),$$

care este de forma de mai sus, cei patru termeni din suma fiind reprezentari binare ale valorilor x egale cu 0, 1, 2, si 3, respectiv. Indicele n pus dupa starea $|x\rangle_n$, care este una din starile bazei de calcul, arata ca valoarea numerica x, cuprinsa intre 0 si $2^n - 1$ se reprezinta prin n qubiti. Aceasta stare este un registru de tip n-qubit. Indicele n poate sa lipseasca daca $n = 1$ sau daca nu exista confuzii cu privire la numarul de qubiti. De asemenea, de cele mai multe ori simbolul produsului direct (sau tensorial) $\otimes$ intre starile mai multor qubiti se omite.

Exemplu: $|00\rangle = |0\rangle \otimes |0\rangle = |0\rangle |0\rangle$ (am folosit aceasta notatie si pana acum)

$$|\psi_2\rangle = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle_n (|0\rangle - |1\rangle)/\sqrt{2}$$

unde s-a folosit forma particulara a transformatei U_f in cazul in care al doilea qubit este $(|0\rangle - |1\rangle)/\sqrt{2}$ (vezi mai sus). Pentru a afla $|\psi_3\rangle$ folosesc faptul ca transformata Hadamard se mai poate exprima si ca

$$H|x\rangle = \sum_{y=0,1}(-1)^{xy}|y\rangle/\sqrt{2}$$

pentru o stare 1-qubit (expresia se poate usor verifica pentru $|x\rangle = |0\rangle$ si $|x\rangle = |1\rangle$) si

$$H^{\otimes n}|x\rangle_n = \prod_{i=1}^{n}\left(\sum_{y_i=0,1}(-1)^{x_i y_i}|y_i\rangle/\sqrt{2}\right) = \sum_{y=0}^{2^n-1}(-1)^{x\cdot y}|y\rangle_n/\sqrt{2^n},$$

pentru o stare *n*-qubit, unde $x\cdot y$ reprezinta produsul dintre *x* si *y* modulo 2, iar indicele *i* se refera la bitul *i* din numarul format din *n* biti. Cu aceste notatii $|\psi_3\rangle$ devine

$$|\psi_3\rangle = 2^{-n}\sum_{y=0}^{2^n-1}\sum_{x=0}^{2^n-1}(-1)^{f(x)}(-1)^{x\cdot y}|y\rangle_n(|0\rangle - |1\rangle)/\sqrt{2}.$$

Daca *f* este constant, suma $2^{-n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}(-1)^{x\cdot y}$ din $|\psi_3\rangle$ devine

$$(-1)^{f(x)}\left(2^{-n}\sum_{x=0}^{2^n-1}(-1)^{x\cdot y}\right) = (-1)^{f(x)}\delta_{y,0},$$

deci este nula cu exceptia cazului $y = 0$ pentru ca in suma dupa *x* am un numar egal de valori *x* pare si impare. In consecinta, daca masor registrul de *n* qubiti obtin $|0\rangle_n$ cu probabilitate 1. Din contra, daca functia *f* este balansata, probabilitatea de a obtine $|y\rangle_n = |0\rangle_n$ este zero, pentru ca suma devine $2^{-n}\sum_{x=0}^{2^n-1}(-1)^{f(x)} = 0$ deoarece jumatate din termeni sunt pozitivi si jumatate sunt negativi. Din nou, obtin raspunsul la intrebarea daca functia *f* este constanta sau balansata, fara sa cunosc valoarea functiei pentru fiecare argument in parte. Anihilarea reciproca a unor termeni in sumele dupa *x* reprezinta un exemplu de interferenta cuantica, folosita pentru a obtine un rezultat cu probabilitate semnificativa pentru o singura conditie indeplinita (aici, pentru $y = 0$).

O varianta a algoritmului Deutsch-Josza este problema Bernstein-Vazirani, care isi propune sa determine factorul *a*, reprezentat ca o succesiune de *n* biti, din definitia functiei $f_a(x) = a\cdot x$ calculata de o cutie neagra. Algoritmul cuantic poate rezolva problema intr-o singura rulare, pentru o singura masuratoare asupra unui *n*-qubit. In acest caz starea cuantica a qubitului care se masoara (vezi expresia lui $|\psi_3\rangle$ de mai sus) este

$$2^{-n}\sum_{y=0}^{2^n-1}\sum_{x=0}^{2^n-1}(-1)^{a\cdot x}(-1)^{x\cdot y}|y\rangle_n,$$

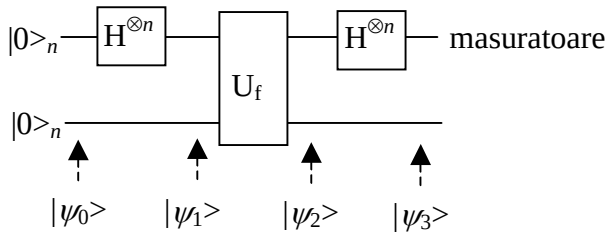
dar, deoarece $2^{-n}\sum_{x=0}^{2^n-1}(-1)^{a\cdot x}(-1)^{x\cdot y} = \delta_{a,y}$, starea *n*-qubit pe care o masor este de fapt $|a\rangle$. Un algoritm clasic ar necesita *n* rulari pentru a gasi fiecare bit din registrul de *n* biti care reprezinta *a* (rezultatul unei rulari clasice este un bit de informatie).

ALGORITMUL SIMON

Algoritmul Simon este inrudit cu algoritmul Deutsch-Jozsa. Sa presupunem ca avem functia $f:\{0,1\}^n \to \{0,1\}^n$, definita pe un registru *n*-qubiti, cu valori intr-un registru *n*-qubiti, care este periodica cu perioada *a* reprezentata ca un numar cu *n* biti, adica $f(x)=f(y)$ daca $y=x\oplus a$, sau, alternativ $x\oplus y=a$, unde $\oplus$ este operatia XOR, adica operatia de adunare modulo 2. Ne propunem sa gasim *a* daca functia *f* este calculata de o cutie neagra.

Pentru un calculator clasic problema este grea. Trebuie in general sa calculez functia de un numar exponential de ori pentru a avea o probabilitate semnificativa de a gasi *a*; nu avem nici o informatie decat daca alegem intamplator doua numere *x* si *y* care satisfac $x\oplus y=a$. Numarul de astfel de perechi pentru care cutia neagra ofera valoarea functiei este mai mic decat $(2^{n/4})^2$, iar pentru fiecare pereche {*x*,*y*} probabilitatea ca $x\oplus y=a$ este 2^{-n}. In consecinta, probabilitatea de a gasi *a* cu succes este mai mica decat $2^{-n}(2^{n/4})^2=2^{-n/2}$; chiar daca am un numar exponential de rulari probabilitatea de succes este exponential mica.

Intr-un circuit cuantic, folosim o varianta a algoritmului Deutsch-Jozsa in care ambii registri contin *n* qubiti; reprezentarea schematica a algoritmului Simon este data in figura de mai jos.



Functiile de unda in pozitiile indicate prin sageti sunt

$$|\psi_0\rangle = |0\rangle_n |0\rangle_n,$$

$$|\psi_1\rangle = (H^{\otimes n}\otimes I_{2^n})|0\rangle_n|0\rangle_n = \frac{1}{2^{n/2}}\sum_{x=0}^{2^n-1}|x\rangle_n|0\rangle_n$$

cu I_{2^n} matricea unitate de dimensiune 2^n.

$$|\psi_2\rangle = U_f\left(\frac{1}{2^{n/2}}\sum_{x=0}^{2^n-1}|x\rangle_n|0\rangle_n\right)=\frac{1}{2^{n/2}}\sum_{x=0}^{2^n-1}|x\rangle_n|f(x)\rangle_n.$$

Dupa ultima poarta Hadamard aplicata primului registru, folosind notatia utilizata la algoritmul Deutsch-Jozsa, am

$$|\psi_3\rangle = 2^{-n}\sum_{y=0}^{2^n-1}\sum_{x=0}^{2^n-1}(-1)^{x\cdot y}|y\rangle_n|f(x)\rangle_n = 2^{-(n+1)}\sum_{x,y=0}^{2^n-1}[(-1)^{x\cdot y}+(-1)^{(x\oplus a)\cdot y}]|y\rangle_n|f(x)\rangle_n$$

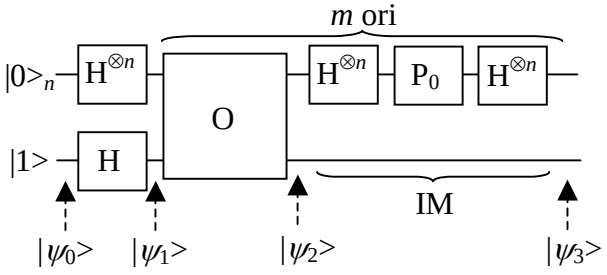
Acum masuram al doilea registru. Daca $a\cdot y=1$ termenii din coeficientul lui $|y\rangle$ interfera distructiv (un alt exemplu de interferenta cuantica), doar starile $|y\rangle$ cu $a\cdot y=0$ ramanand in

suma peste *y*. Rezultatul masuratorii este deci selectat aleator dintre toate valorile posibile *y* ce satisfac $a \cdot y = 0$, fiecare valoare avand probabililatea $2^{-(n+1)}$. Ruland algoritmul de mai multe ori, de fiecare data se obtine o alta valoare *y* care satisface $a \cdot y = 0$. Odata ce am gasit *n* astfel de valori independente se pot rezolva equatiile $a \cdot y_i = 0$, *i* = 1,.., *n* pentru a determina valoarea unica a lui *a* (care are *n* biti). Cu *n* repetitii probabilitatea de succes este exponential apropiata de 1. Deci, problema se rezolva in timp polinomial cuantic, si exponential clasic.

ALGORITMUL GROVER

In exemplele de pana acum algoritmii cuantici au fost aplicati la probleme ce se refereau la functii cu o proprietate data (constanta, balansata, periodica, etc.). Un algoritm cuantic in care functia *f* nu are proprietati speciale, si care este in acelasi timp superior analogului clasic este algoritmul de cautare, descoperit de Grover in 1996. Acesta ofera solutia rapida problemei urmatoare: sa se gaseasca unul dintre obiectele care satisfac o anumita cerinta dintr-o lista nesortata de *N* obiecte. Daca $N = 2^n$ obiectele din lista pot fi reprezentate prin secvente de *n* biti, numarul obiectelor/solutiilor care satisfac o cerinta data fiind *M*; in particular, daca ma intereseaza un anumit obiect, *M* = 1.

Un exemplu este: daca am $f(x) = a$ pentru $x = x_0$, ma intereseaza valoarea lui *x* pentru care valoarea functiei este *a*. Daca lista ar fi sortata, eventual dupa valoarile functiei *f*, as putea gasi valoarea $x = x_0$, uitandu-ma doar la $\log_2 N$ intrari in lista. De exemplu, pentru $N = 2^n$, caut $f(x)$ pentru $x = 2^{n-1} - 1$, si verific daca este mai mare ca *a*. Daca da, verific *f* pentru $x = 2^{n-2} - 1$, etc. Un numar de *n* cautari este suficient sa verific toti cei 2^n termeni sortati. Daca numerele nu sunt sortate, trebuie sa verific $N/2$ termeni inainte sa am o probabilitate de 1/2 pentru a gasi raspunsul. In general, daca am *M* obiecte care satisfac cerinta data, e nevoie in medie de N/M rulari ale algoritmului clasic pentru a gasi o solutie. In algoritmul Grover solutia este gasita in $\sqrt{N/M}$ rulari, ceea ce constituie o imbunatatire fata de cazul clasic.



Circuitul care implementeaza algoritmul de cautare Grover este reprezentat in figura de mai sus. Functiile de unda initiala si dupa primul set de porti Hadamard sunt, ca si in algoritmii studiati pana acum,

$$|\psi_0\rangle = |0\rangle_n |1\rangle ,$$

$$|\psi_1\rangle = (H^{\otimes n} \otimes H)|0\rangle_n |1\rangle = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle_n (|0\rangle - |1\rangle)/\sqrt{2} = |\psi\rangle(|0\rangle - |1\rangle)/\sqrt{2} ,$$

unde starea $|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle_n$ este obtinuta prin aplicarea transformatei $H^{\otimes n}$ asupra lui $|0\rangle_n$; aceasta stare/registru cuantic reprezinta superpozitia tuturor starilor posibile *x* din *n* qubiti. Starea $|1\rangle$, de tip 1-qubit, este numita qubitul oracolului.

Transformarea care urmeaza a fi aplicata, notata *O*, se numeste "oracol", si este o procedura care determina daca o stare *x* este o solutie x_0 a problemei sau nu. Un oracol poate fi reprezentat de o functie Booleana $f_0 : \{0,1\}^n \to \{0,1\}$, cu $f_0(x) = 1$ daca $x = x_0$ este o solutie si $f_0(x) = 0$ in caz contrar. Forma generala a actiunii oracolului,

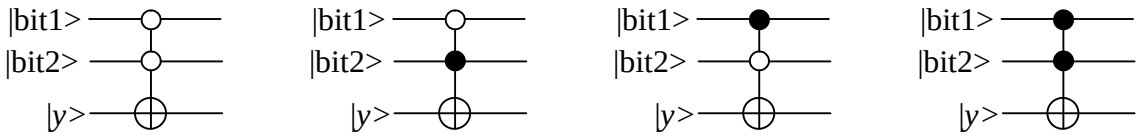
$$O : |x\rangle_n |y\rangle \to |x\rangle_n |y \oplus f_0(x)\rangle,$$

se poate scrie, analog cazului intalnit la algoritmul Deutsch-Jozsa, ca

$$O : |x\rangle_n (|0\rangle - |1\rangle)/\sqrt{2} \to (-1)^{f_0(x)} |x\rangle_n (|0\rangle - |1\rangle)/\sqrt{2},$$

daca qubitul oracolului este preparat in starea $(|0\rangle - |1\rangle)/\sqrt{2}$ prin aplicarea unei porti Hadamard starii $|1\rangle$.

Exemplu: pentru *M* = 1, *N* = 4, *n* = 2, circuitele care pot fi folosite ca oracol (adica circuitele pentru care $f_0(x) = 1$ daca $x = x_0$ si $f_0(x) = 0$ in caz contrar) pentru x_0 = 0, 1, 2, si 3, sunt reprezentate in figurile de mai jos (de la stanga la dreapta). Bit1 si bit 2 din figurile de mai jos reprezinta cei doi biti din care este formata starea 2-qubit x_0.



Deoarece ultimul qubit (cel al oracolului) ramane in starea $(|0\rangle - |1\rangle)/\sqrt{2}$ de-a lungul intregului calcul, nu il vom mai scrie explicit, pentru simplificare. Deci, actiunea oracolului este de fapt $O : |x\rangle_n \to (-1)^{f_0(x)} |x\rangle_n$, (in general $O : \sum_{x=0}^{2^n-1} a_x |x\rangle_n \to \sum_{x=0}^{2^n-1} (-1)^{f_0(x)} a_x |x\rangle_n$). De asemenea, pentru simplificare, vom omite in continuare indicele *n* care indica numarul de qubiti intr-un registru/stare cuantica; trebuie insa sa retinem ca lucram cu stari *n*-qubit in registrul de intrare.

Conform definitiei lui *O*, transformarea oracolului este echivalenta cu

$$U_{f_0} |x\rangle = (I_N - 2|x_0\rangle\langle x_0|)|x\rangle = \begin{cases} -|x_0\rangle, & x = x_0 \\ |x\rangle, & x \neq x_0 \end{cases},$$

adica $O = I_N - 2|x_0\rangle\langle x_0|$, unde I_N este matricea unitate de dimensiune *N*. Echivalent, $|\psi_2\rangle = [(I_N - 2|x_0\rangle\langle x_0|) \otimes I]|\psi_1\rangle$ unde matricea unitate *I* arata ca qubitul oracolului ramane nemodificat.

Oracolul inverseaza semnul starii necunoscute $|x_0\rangle$, dar lasa neschimbata orice stare ortogonala pe $|x_0\rangle$, adica, din punct de vedere geometric reflecta orice vector in spatiul Hilbert de dimensiune $N = 2^n$ fata de hiperplanul ortogonal pe $|x_0\rangle$. Chiar daca nu stiu valoarea lui x_0, stiu ca $|x_0\rangle$ este o stare din baza de calcul, astfel incat $|\langle x_0|\psi\rangle| = 1/\sqrt{N}$, cu $|\psi\rangle = N^{-1/2}\sum_{x=0}^{N-1}|x\rangle$, independent de valoarea lui x_0. Daca am masura starea $|\psi\rangle$, proiectand-o pe baza de calcul, am gasi starea marcata cu x_0 cu o probabilitate de doar $1/N$. Pentru a favoriza amplitudinea de probabilitate a starii necunoscute $|x_0\rangle$, suprimand amplitudinea de probabilitate a celorlalte stari, repet de mai multe ori actiunea oracolului si a transformarii *IM* (inversare fata de medie) care urmeaza. Deoarece transformarea *IM* lasa invariant qubitul oracolului, functia de unda se transforma ca

$$|\psi_3\rangle = (IM \otimes I)|\psi_2\rangle = [(H^{\otimes n}P_0H^{\otimes n}) \otimes I]|\psi_2\rangle,$$

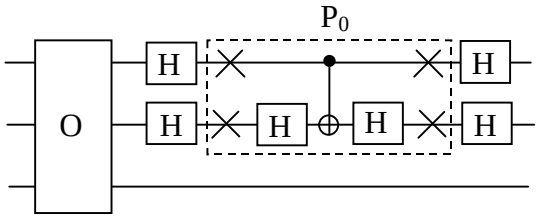
transformarea *IM* fiind echivalenta cu succesiunea de transformari $H^{\otimes n}P_0H^{\otimes n}$. P_0 este operatia de schimbare de faza conditionala, definita ca

$$P_0 : |x\rangle \rightarrow \begin{cases} |x\rangle, x = 0 \\ -|x\rangle, x > 0 \end{cases}$$

pentru orice stare $|x\rangle$ din baza de calcul (cu $0 \le x \le N-1$). Deoarece P_0 este similar (identic, dar cu semn schimbat) cu operatorul U_{f_0} definit mai sus, pentru $x_0 = 0$, actiunea acestui operator asupra starilor bazei se poate exprima ca $P_0 = 2|0\rangle\langle 0| - I_N$. In particular, pentru *n* = 1

$$P_0 = 2|0\rangle\langle 0| - I_2 = 2\begin{pmatrix}1\\0\end{pmatrix} \otimes (1\,0) - \begin{pmatrix}1 & 0\\0 & 1\end{pmatrix} = 2\begin{pmatrix}1 & 0\\0 & 0\end{pmatrix} - \begin{pmatrix}1 & 0\\0 & 1\end{pmatrix} = \begin{pmatrix}1 & 0\\0 & -1\end{pmatrix}.$$

Exemplu: pentru cazul particular *M* = 1, *N* = 4, *n* = 2, considerat in exemplul de mai sus pentru implementarea unui oracol, circuitul echivalent care implementeaza transformarea Grover este



Cele trei intrari reprezinta cei doi biti ai starii de intrare de tip 2-qubit si qubitul oracolului, iar implementarea se face cu porti 1-qubit Hadamard, NOT (vezi notatia X de la cursul de porti cuantice), si poarta CNOT. Operatorul oracol nu a mai fost aratat explicit; vezi exemplul de mai sus pentru forma sa.

Deoarece operatia Hadamard este propria sa inversa,

$H^{\otimes n}P_0H^{\otimes n} = H^{\otimes n}(2|0\rangle\langle 0| - I_N)H^{\otimes n} = 2|\psi\rangle\langle\psi| - I_N$,

cu $|\psi\rangle = N^{-1/2}\sum_{x=0}^{N-1}|x\rangle$, operatorul $2|\psi\rangle\langle\psi| - I_N$ numindu-se si operator de inversie fata de medie, deoarece actiunea sa asupra unei stari cuantice are ca rezultat

$(2|\psi\rangle\langle\psi| - I_N)\sum_{x=0}^{N-1}a_x|x\rangle = \sum_{x=0}^{N-1}(-a_x + 2\langle a\rangle)|x\rangle$, cu $\langle a\rangle = \sum_{x=0}^{N-1}a_x/N$.

Similar cu actiunea oracolului *O*, operatorul de inversie fata de medie pastreaza neschimbata starea $|\psi\rangle$ dar schimba semnul unui vector perpendicular pe $|\psi\rangle$ (il reflecta). (Pentru un vector oarecare, *IM* pastreaza neschimbata componenta unei stari de-a lungul lui $|\psi\rangle$, si inverseaza cealalta componenta).

In consecinta, primul qubit/registru format din *n* biti este supus transformarii Grover

$G = H^{\otimes n}P_0H^{\otimes n}O = (2|\psi\rangle\langle\psi| - I_N)(I_N - 2|x_0\rangle\langle x_0|)$,

Aceasta transformare combina reflexia oracolului fata de un hiperplan ortogonal pe starea necunoscuta $|x_0\rangle$ cu o reflexie in jurul unei stari $|\psi\rangle$ stiute ($|x_0\rangle$, fiind un vector din baza de calcul, este ortogonal/perpendicular pe toti ceilalti $N-1$ vectori din baza de calcul si pe orice combinatie a acestora). Daca am o singura stare $|x_0\rangle$ care trebuie cautata, in planul definit de $|x_0\rangle$ si $|\psi\rangle$, succesiunea celor doua reflexii este echivalenta cu o rotatie θ a starii initiale $|\psi\rangle$ inspre $|x_0\rangle$. Mai precis, definind $\sin(\theta/2) = 1/\sqrt{N} = |\langle\psi|x_0\rangle|$, oracolul *O* reflecta vectorul initial $|\psi\rangle$ fata de hiperplanul perpendicular pe $|x_0\rangle$, iar *IM* il reflecta in jurul lui $|\psi\rangle$, efectul net fiind o rotatie de θ in planul determinat de $|x_0\rangle$ si $|\psi\rangle$, inspre $|x_0\rangle$.

In general, daca solutia consta din *M* stari, transformarea Grover poate fi considerata ca o rotatie in planul generat de starea initiala $|\psi\rangle$ si starea obtinuta ca superpozitie uniforma a solutiilor bazei de stari *M*. Daca $T = \{x \in \{0,1\}^n;\ f_0(x) = 1\}$ este spatiul solutiilor problemei, cu f_0 functia ce defineste oracolul, si $S = \{0,1\}^n - T$ spatiul complementar acestuia fata de spatiul Hilbert al starilor de *n*-qubiti, orice combinatie $|x_{0\perp}\rangle = \frac{1}{\sqrt{N-M}}\sum_{x\in S}|x\rangle$ este perpendiculara pe $|x_{0\|}\rangle = \frac{1}{\sqrt{M}}\sum_{x\in T}|x\rangle$, starea initiala descompunandu-se ca

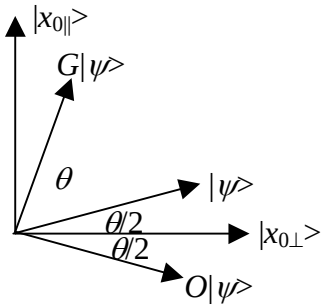
$|\psi\rangle = \sqrt{(N-M)/N}\,|x_{0\perp}\rangle + \sqrt{M/N}\,|x_{0\|}\rangle$.

Deci $|\psi\rangle$ este in planul definit de $|x_{0\perp}\rangle$ si $|x_{0\|}\rangle$, actiunea operatorului *O* asupra oricarei stari $a|x_{0\perp}\rangle + b|x_{0\|}\rangle$ din acest plan fiind $O(a|x_{0\perp}\rangle + b|x_{0\|}\rangle) = a|x_{0\perp}\rangle - b|x_{0\|}\rangle$. Ca urmare, *O* reprezinta o reflexie fata de $|x_{0\perp}\rangle$ (care este perpendicular pe spatiul solutiilor) si, analog, actiunea operatorului $2|\psi\rangle\langle\psi| - I_N$ in planul definit de $|x_{0\perp}\rangle$ si $|x_{0\|}\rangle$ este o reflexie fata de $|\psi\rangle$. Deoarece vectorul de stare initial $|\psi\rangle$ se poate scrie ca

$|\psi\rangle = \cos(\theta/2)\,|x_{0\perp}\rangle + \sin(\theta/2)\,|x_{0\|}\rangle$,

cu $\cos(\theta/2)=\sqrt{(N-M)/N}$, $\sin(\theta/2)=\sqrt{M/N}$, rezulta ca *G* roteste vectorul de stare $|\psi\rangle$ in planul definit de $|x_{0\perp}\rangle$ si $|x_{0\|}\rangle$) cu unghiul θ inspre $|x_{0\|}\rangle$, unde $\theta/2$ este unghiul dintre $|\psi\rangle$ si $|x_{0\perp}\rangle$ (vezi figura de mai jos). Daca baza de calcul aleasa ar fi $\{|x_{0\perp}\rangle, |x_{0\|}\rangle\}$, reprezentarea matriciala a operatorului Grover in aceasta baza de calcul ar fi

$$G=\begin{pmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{pmatrix}.$$



Daca aplic operatorul Grover de *m* ori vectorul de stare initial $|\psi\rangle$ devine

$$G^m|\psi\rangle=\cos[(2m+1)\theta/2]\,|x_{0\perp}\rangle+\sin[(2m+1)\theta/2]\,|x_{0\|}\rangle.$$

O masuratoare a vectorului de stare in acest moment da o solutie $|x_{0\|}\rangle$ cu probabilitate $|\langle x_{0\|}|G^m|\psi\rangle|^2=\sin^2[(2m+1)\theta/2]$. Aceasta probabilitate trebuie sa fie cat mai apropiata de 1 pentru ca sa gasesc cu succes solutia dorita. Pentru $(2m+1)\theta/2\cong\pi/2$, adica dupa un numar de iteratii dat de cel mai apropiat numar intreg *m* fata de $\pi/2\theta-1/2$, vectorul de stare se afla in interiorul unui unghi $\theta/2\le\pi/4$ fata de $|x_{0\|}\rangle$ si $|\langle x_{0\|}|G^m|\psi\rangle|^2\ge\cos^2(\pi/4)=1/2$ (initial a fost la un unghi $\theta/2$ fata de $|x_{0\perp}\rangle$; $\theta/2$ este considerat mai mic decat $\pi/4$ pentru ca in general se considera ca $M<N/2$). Daca $M << N$, $\theta\cong\sin\theta\cong2\sqrt{M/N}$, astfel incat masurarea vectorului de stare in acest caz produce o solutie cu probabilitate de cel putin $\cos^2(\theta/2)\cong1-M/N$. Pentru ca *m* este cel mai apropiat intreg fata de $\pi/2\theta-1/2$, rezulta ca in general *m* este cel mai apropiat intreg fata de $\pi/2\theta$, si deoarece pentru orice numar real θ avem $\theta/2\ge\sin(\theta/2)=\sqrt{M/N}$, numarul de iteratii *m* ale operatorului *G* necesare pentru a gasi o solutie probabila in cazul $M << N$ este mai mic sau egal cu intregul cel mai apropiat de $\pi/2\theta=(\pi/4)\sqrt{N/M}$. Dupa *m* iteratii pot sa ma opresc.

Intuitiv, algoritmul Grover poate fi inteles in modul urmator: oracolul marcheaza starea (sau starile) cautata prin inversarea fazei acesteia, operatia *IM* convertind informatia de faza in informatie de amplitudine. *IM* amplifica amplitudinea de probabilitate a starii cautate intr-un mod iterativ, elementul cautat din lista fiind gasit cand amplitudinea de probabilitate asociata este aproape de unitate.

TRANSFORMATA FOURIER CUANTICA

Transformata Fourier este una dintre cele mai utilizate transformari in fizica, in special pentru inlocuirea functiilor dependente de timp cu functii dependente de frecventa, astfel incat o functie periodica cu perioada $T > 0$ este transformata intr-o functie a carei amplitudine este diferita de zero doar pentru frecvente care sunt multipli intregi de $1/T$. Transformata Fourier cuantica este analoaga transformarii Fourier discrete in fizica clasica. Mai exact, daca am un vector complex (a carui elemente sunt numere complexe) la intrare $(x_0, x_1, ..., x_{N-1})$, transformata Fourier discreta, notata

F: $(x_0, x_1, ..., x_{N-1}) \to (y_0, y_1, ..., y_{N-1})$,

da la iesire vectorul $(y_0, y_1, ..., y_{N-1})$ ale carui elemente sunt numere complexe, astfel incat

$$y_k = N^{-1/2} \sum_{j=0}^{N-1} x_j \exp(2\pi i jk / N) .$$

In particular, pentru $N = 4$, transformata Fourier discreta poate fi reprezentata prin matricea

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \exp(2\pi i/4) & \exp(2\pi i/2) & \exp(3\times 2\pi i/4) \\ 1 & \exp(2\pi i/2) & \exp(2\pi i) & \exp(2\pi i/2) \\ 1 & \exp(3\times 2\pi i/4) & \exp(2\pi i/2) & \exp(2\pi i/4) \end{pmatrix},$$

in sensul ca $(y_0, y_1, ..., y_{N-1})^{\mathrm{T}} = M(x_0, x_1, ..., x_{N-1})^{\mathrm{T}}$, $^{\mathrm{T}}$ indicand transpunerea vectorilor.

Daca ma refer la o baza de calcul $|j\rangle_n$, cu $j = 0,\ldots,N-1$, astfel incat $N = 2^n$, transformarea Fourier cuantica se defineste prin

$$F : \sum_{j=0}^{N-1} x_j |j\rangle_n \to \sum_{k=0}^{N-1} y_k |k\rangle_n .$$

Analog, inversa transformarii F este definita prin

$$F^+ : |j\rangle_n \to N^{-1/2} \sum_{k=0}^{N-1} \exp(-2\pi i jk / N) |k\rangle_n .$$

Intr-o forma mai explicita, transformata Fourier cuantica actioneaza asupra intrarii n-qubit ca

$$\begin{aligned} |j\rangle_n &\to N^{-1/2} \sum_{k=0}^{N-1} \exp(2\pi i jk / N) |k\rangle_n = 2^{-n/2} \sum_{k_1=0,1} \ldots \sum_{k_n=0,1} \exp(2\pi i j \textstyle\sum_{l=1}^{n} k_l 2^{-l}) |k_1 \ldots k_n\rangle \\ &= 2^{-n/2} \sum_{k_1=0,1} \ldots \sum_{k_n=0,1} \otimes_{l=1}^{n} \exp(2\pi i j k_l 2^{-l}) |k_l\rangle = 2^{-n/2} \otimes_{l=1}^{n} \sum_{k_l=0,1} \exp(2\pi i j k_l 2^{-l}) |k_l\rangle \\ &= 2^{-n/2} \otimes_{l=1}^{n} [|0\rangle + \exp(2\pi i j 2^{-l}) |1\rangle] \end{aligned}$$

unde am folosit explicit faptul ca o stare *n*-qubit este egala cu produsul tensorial ⊗ al *n* stari 1-qubit, dupa care am inversat ordinea produsului tensorial cu cea a sumei. Expresia de mai sus, desi aparent complicata, este usor de demonstrat luand un caz particular.

Conform acestei definitii, transformata Fourier cuantica transforma starea $|0\rangle_n$ intr-o superpozitie egal probabila a tuturor starilor $|k\rangle_n$ posibile, cu valori *k* de la 0 la *N* – 1, deci poate prepara un registru de qubiti analog actiunii portilor Hadamard asupra fiecarui qubit dintr-un registru.

Ultima expresie din definitia de mai sus se poate pune sub forma

$$2^{-n/2} \otimes_{l=1}^{n} [|0\rangle + \exp(2\pi i \sum_{m=n-l+1}^{n} j_m 2^{n-m-l}) |1\rangle],$$

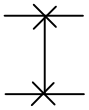
deoarece in produsul $j2^{-l} = 2^{-l} \sum_{m=1}^{n} j_m 2^{n-m}$ care apare in argumentul exponentialei (numarul *j*, reprezentat binar prin $j = j_1 2^{n-1} + ... + j_n 2^0$ simbolizeaza starea cuantica $|j_1 ... j_n\rangle$) termenii care contin 2^{n-m-l} cu $n-m-l$ pozitiv nu contribuie la argumentul exponentialei pentru ca in acest caz 2^{n-m-l} este un numar intreg si termenul exponential asociat este 1. Ca urmare, ma intereseaza doar termenii cu $n \geq m > n-l$, pentru care $j2^{-l} = \sum_{m=n-l+1}^{n} j_m 2^{n-m-l}$. Cu aceste simplificari, termenii din suma pentru *l* = 1, 2, …, *n* sunt, respectiv, $j_n 2^{-1}$, $j_{n-1} 2^{-1} + j_n 2^{-2}$, …, $j_1 2^{-1} + j_2 2^{-2} + ... + j_n 2^{-n}$. Toti acesti termeni sunt cuprinsi intre 0 si 1 deoarece $j_m = 0,1$ pentru orice *m*. Daca starii cuantice $|j_1 j_2 ... j_n\rangle$ ii asociez numarul intreg $j = \sum_{k=1}^{n} j_k 2^{n-k}$, sumei $\sum_{k=1}^{n} x_k 2^{-k}$ ii asociez numarul fractionar (cuprins intre 0 si 1 pentru x_k = 0, 1) $0.x_1 x_2 ... x_n$. Cu aceste notatii efectul transformatei Fourier asupra unei stari de intrare *n*-qubit este

$$F : |j_1 j_2 ... j_n\rangle \rightarrow$$
$$2^{-n/2}[|0\rangle + \exp(2\pi i 0.j_n)|1\rangle][|0\rangle + \exp(2\pi i 0.j_{n-1} j_n)|1\rangle]...[|0\rangle + \exp(2\pi i 0.j_1 j_2 ... j_n)|1\rangle]$$

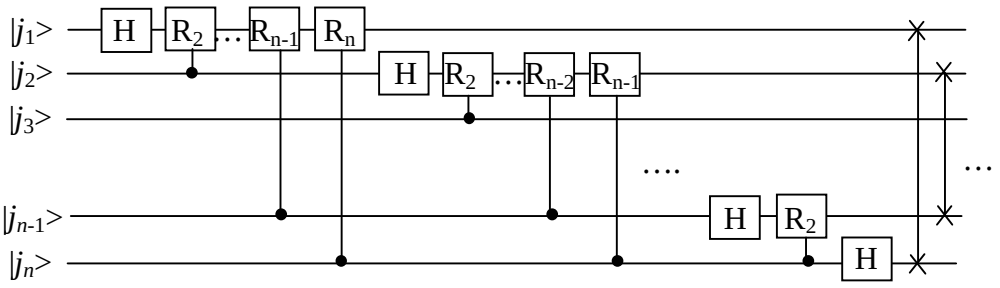
Transformata Fourier cuantica este folosita intr-un numar mare de algoritmi, printre care si in algoritmul Shor. Cu ajutorul operatorului 1-qubit

$$R_k = \begin{pmatrix} 1 & 0 \\ 0 & \exp(2\pi i / 2^k) \end{pmatrix}$$

transformata Fourier cuantica se poate implementa cu circuitul descris mai jos unde



reprezinta operatia SWAP.



Pentru a ne convinge ca acest circuit realizeaza transformata Fourier cuantica, sa observam ca actiunea portii Hadamard asupra primului qubit $|\, j_1\rangle$ se poate scrie ca

$$2^{-1/2}[|\,0\rangle + \exp(2\pi i 0.j_1)\,|1\rangle]\,|\,j_2 \ldots j_n\rangle \;= 2^{-1/2}[|\,0\rangle + \exp(2\pi i j_1/2)\,|1\rangle]\,|\,j_2 \ldots j_n\rangle,$$

deoarece $\exp(2\pi i j_1/2) = 1$ daca $j_1 = 0$, si $= -1$ daca $j_1 = 1$. Aplicand apoi poarta control-R_2, am

$$2^{-1/2}[|\,0\rangle + \exp(2\pi i 0.j_1 j_2)\,|1\rangle]\,|\,j_2 \ldots j_n\rangle,$$

iar dupa aplicarea tuturor portilor control-R_k pana la R_n, obtin

$$2^{-1/2}[|\,0\rangle + \exp(2\pi i 0.j_1 j_2 \ldots j_n)\,|1\rangle]\,|\,j_2 \ldots j_n\rangle.$$

Al doilea qubit, $|\,j_2\rangle$, trece prin succesiunea de porti Hadamard si control-R_k pana cand se transforma in

$$2^{-1/2}[|\,0\rangle + \exp(2\pi i 0.j_2 \ldots j_n)\,|1\rangle],$$

starea finala la iesire dupa transformarea primilor doi qubiti fiind

$$2^{-1}[|\,0\rangle + \exp(2\pi i 0.j_1 j_2 \ldots j_n)\,|1\rangle][|\,0\rangle + \exp(2\pi i 0.j_2 \ldots j_n)\,|1\rangle]\,|\,j_3 \ldots j_n\rangle.$$

Analog, starea sistemului n-qubiti dupa succesiunea de porti Hadamard si control-R_k este

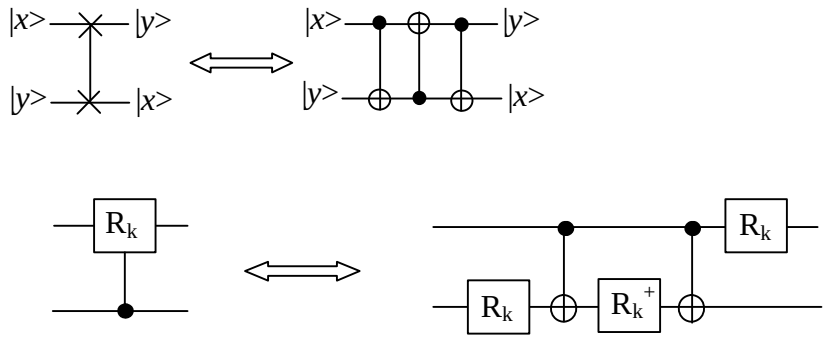
$$2^{-n/2}[|\,0\rangle + \exp(2\pi i 0.j_1 j_2 \ldots j_n)\,|1\rangle][|\,0\rangle + \exp(2\pi i 0.j_2 \ldots j_n)\,|1\rangle] \ldots [|\,0\rangle + \exp(2\pi i 0.j_n)\,|1\rangle],$$

ordinea bitilor schimbandu-se datorita succesiunii portilor SWAP pana obtin rezultatul dorit

$$2^{-n/2}[|\,0\rangle + \exp(2\pi i 0.j_n)\,|1\rangle] \ldots [|\,0\rangle + \exp(2\pi i 0.j_2 \ldots j_n)\,|1\rangle][|\,0\rangle + \exp(2\pi i 0.j_1 j_2 \ldots j_n)\,|1\rangle].$$

Deoarece toate portile care formeaza circuitul ce implementeaza operatia F sunt unitare, transformata Fourier cuantica F este o operatie unitara. Circuitul cuprinde n porti Hadamard, $n(n-1)/2$ porti control-R_k, deci $n(n+1)/2$ porti elementare, la care se adauga $[n/2]$ porti SWAP, unde [.] indica partea intreaga. In consecinta, algoritmul pentru calculul transformatei Fourier cuantice este de complexitate $O(n^2)$ (este polinomial in n, deci este un algoritm eficient). Pot renunta la portile SWAP daca citesc bitii de iesire in ordine inversa!

Portile SWAP si control-R_k se pot implementa cu setul de porti universale CNOT si porti care rotesc un singur qubit, asa cum se vede din figura de mai jos.



ALGORITMUL SHOR

Algoritm Shor cuantic, propus in 1994, este utilizat pentru factorizarea eficienta a unui numar mare. Factorizarea (gasirea factorilor primi ai unui numar) este un exemplu de problema in care solutia poate fi usor verificata, odata ce este gasita, dar este dificil de gasit. Adica, daca p si q sunt numere prime mari, produsul $N=p\cdot q$ poate fi usor calculat (numarul de operatii elementare necesare este de ordinul $\log_2 p\cdot\log_2 q$) dar, dandu-se N este greu de gasit p si q. Mai precis, algoritmul de multiplicare este rapid, cel de factorizare este lent.

Un algoritm este rapid sau eficient daca timpul necesar executiei (numarul de pasi de calcul) creste mai incet decat o functie polinomiala de numarul de biti de intrare; un numar N necesita $n = \log_2 N$ biti. Un algoritm eficient se executa intr-un timp mai mic decat o functie polinomiala de log$N$ pentru toti N.

Timpul necesar pentru a gasi factorii unui numar N intr-un algoritm clasic este suprapolinomial in log$N$ (daca N creste, timpul necesar creste mai repede decat orice putere a log$N$). Cel mai bun algoritm de factorizare clasic cunoscut necesita un timp de $\exp[2(\log_2 N)^{1/3}(\log_2\log_2 N)^{2/3}]$, astfel incat factorii de 65 digiti ai unui numar de 130 digiti pot fi gasiti cam intr-o luna de o retea care cuprinde sute de calculatoare, iar factorizarea unui numar de 400 digiti ar lua cam 10^{10} ani (varsta universului). In 1990, 1000 de computere din intreaga lume au calculat timp de doua luni pentru a factoriza cel de-al 9-lea numarul Fermat $F_9 = 2^{2^9}+1$ care are 155 digiti. Problema factorizarii este importanta in practica pentru ca dificultatea factorizarii este schema de baza pentru criptografie.

Algortimul cuantic Shor factorizeaza un numar N intr-un timp polinomial, de ordinul $O(\log_2 N)^3$, folosind $O(\log_2 N)$ resurse, un numar de 400 digiti putand fi factorizat in mai putin de trei ani. Ca si algoritmul Grover, algoritmul Shor este probabilistic, in sensul ca ofera raspunsul corect cu probabilitate mare, probabilitate ce creste prin repetarea algoritmului.

Algoritmul Shor este mai sofisticat decat algoritmii studiati pana acum pentru ca are mai multe parti (clasice si cuantice). Algoritmul Shor se bazeaza pe faptul ca problema factorizarii lui N se poate reduce la problema gasirii ordinului unui numar intreg x mai mic decat N. Aceasta procedura, care poate fi rezolvata pe un calculator clasic, este apoi implementata printr-un algoritm cuantic de gasire a ordinului, care este mai rapid.

ALGORITMUL EUCLID

Se foloseste pentru gasirea celui mai mare divizor comun al numerelor intregi p si q, notat cmdc(p,q). Algoritmul lui Euclid se bazeaza pe rezultatul urmator: daca p si q sunt intregi si $r \neq 0$ este restul impartirii lui p la q, cmdc(p,q) = cmdc(q,r).

Demonstratie: arat ca fiecare parte a acestei egalitati o imparte pe cealalta. Daca r este restul impartirii lui p la q, atunci exista a astfel incat $r = p - aq$. Deci, cmdc(p,q), care il divide pe p si pe q il divide si pe r, astfel incat cmdc(p,q) il divide pe cmdc(q,r). In plus, daca cmdc(q,r) il divide pe q, deoarece $p = aq + r$, il divide si pe p. Ca urmare cmdc(q,r) il divide pe cmdc(p,q).

Algortimul lui Euclid de a gasi cmdc al intregilor p si q cu $p > q$ este: se divide p la q pentru a se gasi restul $r_1 < q$. Datorita rezultatului de mai sus, cmdc(p,q) = cmdc(q,r_1). Se divide apoi q la r_1 pentru a gasi restul $r_2 < r_1$, cu cmdc(q,r_1) = cmdc(r_1,r_2). Procedand iterativ, divid r_{n-2} la r_{n-1} pentru a gasi restul $r_n < r_{n-1}$, cu cmdc(p,q) = cmdc(q,r_1) = … = cmdc(r_{n-1},r_n). Deoarece r_j este o secventa strict descrescatoare de numere intregi pozitive, exista n cu $r_{n+1} = 0$, adica $r_{n-1} = ar_n$. Algoritmul se termina daca cmdc(p,q) = cmdc(r_{n-1},r_n) = r_n.

Exemplu: cmdc(1071,1029) = 21 se gaseste in pasii urmatori: 1071=1029+42; primul rest este 42; restul impartirii lui 1029 la 42 este 21; restul impartirii lui 42 la 21 este 0; deci cmdc = 21.

Daca p si q sunt intregi reprezentati prin numere cu n biti, r_j este un intreg din n biti pentru toti j. Fiecare divizare costa resurse de ordinul $O(n^2)$. Deoarece $r_{j+2} \leq r_j/2$, numarul de diviziuni necesare este de ordinul $O(n)$, astfel incat costul total este de ordinul $O(n^3)$.

PROBLEMA DE GASIRE A ORDINULUI UNUI NUMAR INTREG

Daca x si N au factori comuni, cmdc(x,N) este un factor al lui N. De aceea ma limitez la cazul cand x este coprim cu N (cmdc al lor este 1). In particular, daca N este par, il impart la 2 repetat, pana obtin un numar impar, dupa care aplic algoritmul.

Ordinul lui x modulo N este cel mai mic intreg pozitiv pentru care

$x^r \equiv 1 \bmod N$,

deci N este divizorul/factorul lui $x^r - 1$.

Pentru numere intregi pozitive x si N, restul impartirii lui x la N se numeste x modulo N si se scrie x mod N; mai precis x se poate scrie unic ca $x = kN + r$, unde k este pozitiv si $0 \leq r \leq N-1$ este restul $r = x \bmod N$.

Existenta ordinului este asigurata de faptul ca puterile intregi ale x^j formeaza un grup ciclic, finit, si deci exista un cel mai mic intreg $1 < r < N$, numit ordinul lui lui x mod N pentru care $x^r = 1 \bmod N$.

Observatie: pentru calculul eficient al x^j se foloseste metoda ridicarii la patrat repetate, care este un algoritm recursiv pentru a calcula x^j pentru un numar pozitiv intreg j. Algoritmul este

$$\text{power}(x, j) = \begin{cases} 1, \text{daca } j = 0 \\ x \times \text{power}(x^2, (j-1)/2), \text{daca } j \text{ este impar} \\ \text{power}(x^2, j/2), \text{daca } j \text{ este par} \end{cases}$$

Fata de metoda standard de multiplicare a x cu el insusi de $j-1$ ori, acest algoritm foloseste doar $O(\log j)$ multiplicari si astfel creste viteza calcularii lui x^j.

Daca r este par, definesc y prin $x^{r/2} \equiv y \bmod N$, adica y este restul impartirii lui $x^{r/2}$ la N, si $0 \le y < N$. Deoarece y satisface $y^2 \equiv 1 \bmod N$, adica

$$x^r - 1 = (x^{r/2} - 1)(x^{r/2} + 1) = (y - 1)(y + 1) \equiv 0 \bmod N,$$

rezulta ca N divide $(y - 1)(y + 1)$. Daca $1 < y < N - 1$, factorii $y - 1$ si $y + 1$ satisfac

$$0 < y - 1 < y + 1 < N,$$

si deci N nu poate divide $y - 1$ sau $y + 1$ separat. Singura alternativa este ca N sa aiba un factor comun netrivial cu fiecare dintre $(y - 1)$ si $(y + 1)$, factori care sa dea N prin multiplicare. In consecinta, cmdc$(y - 1,N)$ si cmdc$(y + 1,N)$ sunt factori netriviali ai lui N care pot fi calculati intr-un numar de $O[(\log N)^3]$ operatii. Daca N are alti factori, ei se pot calcula aplicand algoritmul recursiv.

Exemplu: $N = 21$. Aleg $x = 2$. Din setul de echivalente $2^1 \equiv 2 \bmod 21$, $2^2 \equiv 4 \bmod 21$, $2^3 \equiv 8 \bmod 21$, $2^4 \equiv 16 \bmod 21$, $2^5 \equiv 11 \bmod 21$, $2^6 \equiv 11 \times 2 \equiv 1 \bmod 21$, ordinul lui 2 modulo 21 este 6. Atunci $y \equiv 2^3 \equiv 8 \bmod 21$, $y - 1 = 7$, $y + 1 = 9$, cmdc(7,21) = 7, si cmdc(9,21) = 3, care, prin multiplicare dau 21.

Observatie: Acest algoritm de factorizare nu functioneaza in toate cazurile. De exemplu, algoritmul nu functioneaza, in sensul ca da factori triviali ai lui N, daca $y = 1 \bmod N$ si $y = (N - 1) \bmod N$.

Exemplu: pentru $N = 3 \times 23 = 69$, $x = 14$, secventa $x^j \bmod N$ este formata din 1, 14, 58, 53, 38, 49, 65, 13, 44, 64, 68, 55, 11, 16, 17, 31, 20, 4, 56, 25, 1. Deci $r = 20$, $x^{r/2} = y \bmod N$, rezulta $y = 68$, si cmdc(67,69) = 1, cmdc(69,69) = 69.

Alt exemplu: Daca $N = 21823$ si $x = 12083$, ordinul lui x este $r = 3588$, $12083^{1794} = 4866 \bmod 21823$, si cmdc$(12083^{1794} \pm 1, 21823) = \{157, 139\}$ sunt factori ai 21823. Daca insa $x = 14335$, $r = 1794$, dar $14335^{897} = -1 \bmod 21823$, (echivalent = $(21823 - 1) \bmod 21823$), si cmdc$(14335^{897} \pm 1, 21823) = \{21823, 1\}$, deci nu se obtine un factor netrivial al lui N.

Sumarizand, problema clasica a factorizarii, bazata pe gasirea ordinului unui numar x, consta in:

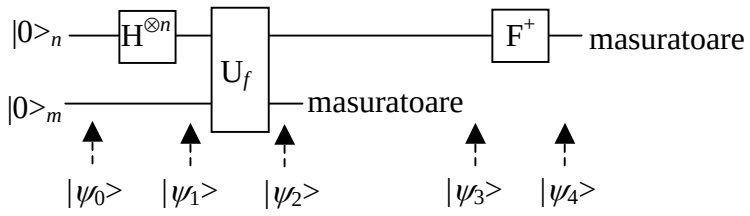
1) se alege un numar aleator $x < N$
2) se calculeaza cmdc(x,N), cu ajutorul algoritmului Euclid
3) daca cmdc$(x,N) \neq 1$, atunci exista un factor netrivial al lui N, si stop
4) in caz contrar, se foloseste subrutina de gasire a ordinului r al lui x detaliata mai sus
5) daca r este par, si daca $0 < y - 1 < y + 1 < N$, factorii lui N sunt cmdc$(y \pm 1, N)$. Stop.
6) daca una dintre conditiile de mai sus nu este indeplinita, aleg alt numar x, si ma intorc la pasul 1)

Daca N este impar probabilitatea de a gasi un intreg x, cu $1 < x < N$, coprim cu N si satisfacand conditiile de la 5) este egala cu sau mai mare decat $1/(2\log N)$ daca N nu este o putere a unui numar prim impar; aceasta probabilitate se mai poate exprima ca $1 - 2^{1-k}$, unde k este numarul factorilor primi ai lui N. In cel mai rau caz, in care N are 2 factori, aceasta probabilitate este mai mare sau egala cu 1/2. (Daca N este o putere a unui numar prim impar, algoritmul de mai sus nu da rezultate bune si trebuie folosit un alt algoritm clasic eficient.) Valorile x alese aleator pentru care se obtine un factor netrivial al lui N cu probabilitate mai mare decat $1-\varepsilon$ sunt in numar de $O[\log(1/\varepsilon)\log N]$.

Desi algoritmul clasic descris mai sus pare eficient, in realitate nu este pentru ca nu se cunoaste un algoritm clasic eficient pentru a calcula ordinul unui intreg x modulo N (daca $N \le 2^L$, nu exista algoritmi clasici polinomiali in L care sa rezolve problema) . Este in special dificil de a calcula ordinul r al x mod N pentru N mare. In schimb, algoritmul Shor rezolva problema eficient, in $O(L^3)$.

ALGORITMUL SHOR CUANTIC

Algoritmul Shor calculeaza ordinul r al unui intreg pozitiv x mai mic decat N si coprim cu N. El poate fi implementat cu ajutorul circuitului de mai jos



unde U_f este operatorul unitar $U_f(|j\rangle_n|k\rangle_m) = |j\rangle_n|k \oplus f(x,j)\rangle_m = |j\rangle_n|k \oplus (x^j \bmod N)\rangle_m$, cu $f(x,j) = x^j \bmod N$ si $|j\rangle_n$, $|k\rangle_m$ stari n-qubit, respectiv m-qubit din registrul de intrare si de iesire. Operatiile fiind efectuate modulo N, $0 \le x^j \bmod N < N$, iar F^+ este inversa transformatei Fourier cuantice. Primul registru (registrul de intrare) are n qubiti, unde n este ales pentru motive care vor deveni clare ulterior astfel incat $N^2 \le 2^n < 2N^2$. Doar daca r este o putere a lui 2 putem lua $n = m$; consideram pentru inceput acest caz, dar pastram indicii diferiti. Starea calculatorului cuantic la inceputul algoritmului este (vezi figura de mai sus)

$$|\psi_0\rangle = |0\rangle_n|0\rangle_m$$

care, dupa aplicarea operatorului Hadamard asupra fiecarui qubit din registrul de intrare devine

$$|\psi_1\rangle = 2^{-n/2}\sum_{j=0}^{2^n-1}|j\rangle_n|0\rangle_m$$

Registrul de intrare este acum intr-o superpozitie cu ponderi egale, $2^{-n/2}$, a tuturor starilor din baza de calcul.

Dupa aplicarea operatorului U_f starea cuantica devine

$$|\psi_2\rangle = U_f\,|\psi_1\rangle = 2^{-n/2}\sum_{j=0}^{2^n-1} U_f\,(|\,j\rangle_n\,|0\rangle_m) = 2^{-n/2}\sum_{j=0}^{2^n-1} |\,j\rangle_n\,|\,x^j \bmod N\rangle_m$$

Deoarece U_f este liniar, actioneaza asupra tuturor starilor $|\,j\rangle_n\,|0\rangle_m$ simultan pentru toate cele 2^n valori ale *j*, si genereaza toate puterile lui *x* simultan (clasic, asa cum am aratat mai sus, acest lucru se face calculand succesiv $x^j \bmod N$ pentru *j* de la 2 pana cand se ajunge la *r*); acesta este un exemplu de paralelism cuantic. Unele dintre aceste puteri sunt egale cu 1, cum ar fi cele care corespund, dupa aplicarea lui U_f, starilor $|0\rangle|1\rangle$ (deoarece $x^0 = 1$), $|\,r\rangle|1\rangle$, $|\,2r\rangle|1\rangle,\ldots,|(2^n/r-1)r\rangle|1\rangle$ (deoarece daca $x^r = 1 \bmod N$, $(x^r)^n = 1 \bmod N$ deoarece, daca *N* divide $x^r - 1$, divide si $(x^r)^n - 1$, care are ca factor pe $x^r - 1$) (nu am mai folosit indicii *n* si *m* pentru cele doua registre pentru ca acum ne referim la valoarea numerica, si nu la scrierea binara a numerelor din cele doua registre). Pentru a gasi valorile lui *j* pentru care $x^j \equiv 1 \bmod N$, nu se fac masuratori asupra registrului de intrare, deoarece toate starile din superpozitia de mai sus au aceeasi probabilitate. Pentru a gasi *r*, folosesc faptul ca starile $|0\rangle|1\rangle$, $|\,r\rangle|1\rangle$, $|\,2r\rangle|1\rangle,\ldots,|\,2^n - r\rangle|1\rangle$ sunt periodice (amintim ca folosim aici *n* = *m* si *r* o putere a lui 2).

$x^j \bmod N$ este periodica cu perioada *r* pentru ca $x^j = x^{j+r}$ (modulo *N*).

Demonstratie: daca $x^j = y \bmod N$, cu $y < N$ arbitrar, deci $x^j - y = k_1 N$, si $x^r - 1 = k_2 N$, cu k_1, k_2 intregi, $x^{j+r} = x^j x^r = (y + k_1 N)(1 + k_2 N) = y + kN$, cu $k = k_1 + yk_2 + k_1 k_2 N$ numar intreg, deci restul impartirii la *N* este acelasi ca si pentru x^j. Cele doua numere sunt echivalente modulo *N*.

Pentru a vedea ce se obtine daca am masura starile din registrul de iesire (al doilea registru), rescriem $|\psi_2\rangle$ colectand termenii egali din registrul al doilea. Deoarece $x^j \bmod N$ are perioada *r*, inlocuim *j* cu *ar* + *b* in expresia lui $|\psi_2\rangle$, unde $0 \le a \le (2^n/r) - 1$, $0 \le b \le r - 1$, si putem scrie $x^b \bmod N$ in loc de $x^{ar+b} \bmod N$ (vezi demonstratia de mai jos), astfel incat

$$|\psi_2\rangle = 2^{-n/2}\sum_{b=0}^{r-1}\sum_{a=0}^{2^n/r-1} |\,ar + b\rangle_n\,|\,x^b \bmod N\rangle_m$$

x^b si x^{ar+b} sunt echivalente modulo *N*.

Demonstratie: daca $x^b = y \bmod N$, cu $y < N$ arbitrar, deci $x^b - y = c_1 N$, si $x^r - 1 = c_2 N$, cu c_1, c_2 intregi, $x^{ar+b} = x^{ar} x^b = (x^r)^a x^b = (1 + c_2 N)^a (y + c_1 N) = y + cN$, cu *c* intreg.

Acum se masoara starea registrului de iesire. Orice iesire x^0, x^1, …, x^{r-1} poate fi obtinuta cu egala probabilitate. Daca rezultatul este $x^{b'}$, starea calculatorului cuantic este

$$|\psi_3\rangle = \sqrt{r/2^n}\sum_{a=0}^{2^n/r-1} |\,ar + b'\rangle_n\,|\,x^{b'} \bmod N\rangle_m$$

(dupa masuratoare constanta de normare devine $\sqrt{r/2^n}$ deoarece in suma de mai sus sunt $2^n/r$ termeni). Probabilitatile de a obtine starile din baza de calcul dupa masuratoare, in

registrul de intrare, formeaza o functie periodica cu perioada *r*. Probabilitatea acestor stari este zero cu exceptia starilor $|b'\rangle$, $|r+b'\rangle$, $|2r+b'\rangle$,…, $|2^n - r + b'\rangle$.

In pasul urmator se gaseste perioada *r*, folosind pentru aceasta transformata Fourier cuantica. Transformata Fourier a unei functii periodice cu perioada *r* este o alta functie periodica cu perioada proportionala cu $1/r$. Transformata Fourier cuantica gaseste perioada mult mai eficient decat un algoritm clasic.

Observatie: Algoritmul Simon de gasire a perioadei nu se aplica in cazul nostru deoarece acest algoritm gaseste doar perioada modulo 2; imi trebuie un algoritm care sa gaseasca perioada modulo *N*.

In particular, in algoritmul Shor se aplica transformata Fourier cuantica inversa registrului de intrare. Definitia generala a acestei transformari:

$$F^+ |x\rangle_n = 2^{-n/2} \sum_{y=0}^{2^n-1} \exp(-2\pi i xy/N) |y\rangle_n$$

se particularizeaza in cazul nostru ca

$$|\psi_4\rangle = F^+ |\psi_3\rangle = \sqrt{r/2^n} \sum_{a=0}^{2^n/r-1} \left(2^{-n/2} \sum_{j=0}^{2^n-1} \exp[-2\pi i j(ar+b')/2^n] |j\rangle_n \right) |x^{b'} \bmod N\rangle_m$$

O inversare a ordinii de sumare da

$$|\psi_4\rangle = r^{-1/2} \left[\sum_{j=0}^{2^n-1} \left((r/2^n) \sum_{a=0}^{2^n/r-1} \exp(-2\pi i jar/2^n) \right) \exp(-2\pi i jb'/2^n) |j\rangle_n \right] |x^{b'} \bmod N\rangle_m$$

Deoarece, datorita identitatii

$$N^{-1} \sum_{j=0}^{N-1} \exp(2\pi i jk/N) = \begin{cases} 1, \text{ daca } k \text{ este multiplu de } N \\ 0, \text{ in caz contrar} \end{cases}$$

expresia din parantezele rotunde mari in formula lui $|\psi_4\rangle$ este diferita de zero doar daca $j = k2^n/r$, cu $k = 0,\ldots, r-1$, obtinem

$$|\psi_4\rangle = r^{-1/2} \left(\sum_{k=0}^{r-1} \exp(-2\pi i kb'/r) |k2^n/r\rangle_n \right) |x^{b'} \bmod N\rangle_m .$$

Algoritmul pentru gasirea perioadei se bazeaza pe abilitatea unui calculator cuantic de a fi in mai multe stari simultan (de a fi intr-o superpozitie de stari). Pentru a gasi perioada functiei calculez valoarea functiei in toate punctele simultan. Transformata Fourier cuantica are rolul de a creste probabilitatea raspunsului corect, in comparatie cu alte stari posibile.

Daca masuram acum starea cuantica a registrului de intrare, obtinem valoarea $k'2^n/r$, unde k' poate fi orice numar intre 0 si $r-1$, cu egala probabilitate. Daca obtin $k' = 0$ (adica $j =$

0), nu am nici o informatie despre *r* si algoritmul trebuie rulat din nou. Daca insa $k' \neq 0$, impart $k'2^n / r$ la 2^n si obtin raportul k'/r, fara sa cunosc separat k' sau *r*. Pot exista doua variante:

1) k' este coprim cu *r*, caz in care selectez numitorul,
2) k' si *r* au un factor comun. In ultimul caz numitorul fractiei k'/r este un factor al lui *r*, dar nu *r*. Notand acest numitor cu r_1, avem $r = r_1 r_2$, scopul fiind de a gasi r_2, care este ordinul lui x^{r_1} (deoarece $x^r = x^{r_1 r_2} = (x^{r_1})^{r_2} = 1 \bmod N$). Deci, trebuie sa rulez inca o data algoritmul pentru a gasi ordinul lui x^{r_1}. Daca gasesc r_2 dupa prima rulare, algoritmul se opreste, in caz contrar il aplic recursiv, numarul iteratiilor necesare fiind mai mic decat sau egal cu $\log_2 r$.

Exemplu: $N = 15$. Setul numerelor mai mici decat 15 si coprime cu 15 este {1,2,4,7,8,11, 13,14}. Numerele din setul {4,11,14} au ordinul 2, iar numerele din setul {2,7,8,13} au ordinul 4, in ambele cazuri *r* fiind o putere a lui 2, astfel incat factorii lui $N = 15$ se pot gasi intr-un calculator cuantic cu registrii de 8 biti ($n + m = 2n = 2[\log_2 15] = 8$).

Pana acum am considerat cazul special in care *r* este o putere a lui 2 si $n = m$, cu *n* numarul qubitilor din primul registru ales astfel incat 2^n este intre N^2 si $2N^2$. In general, $m = [\log_2 N]$, unde [.] este cel mai mic intreg mai mare sau egal cu argumentul. Pentru a vedea cum se poate generaliza algoritmul Shor daca *r* nu este o putere a lui 2 si *n* nu este egal cu *m*, sa luam un exemplu, si anume sa consideram factorizarea lui $N = 21$. In acest caz cea mai mica valoare a lui *n* este 9 si $m = 5$. Primul pas este sa aleg aleator un intreg *x* cu $1 < x < N$ si sa verific daca *x* este coprim cu *N*. Daca nu este, pot gasi usor un factor al lui *N* calculand clasic cmdc(*x*,*N*). Daca *x* este coprim cu *N*, incep rularea algoritmului Shor. Sa presupunem ca aleg $x = 2$, si vreau sa-i gasesc ordinul (care este 6). Calculatorul cuantic este initializat in starea

$$|\psi_0\rangle = |0\rangle_9 |0\rangle_5$$

unde primul registru (registrul de intrare) are 9 qubiti si al doilea registru (cel de iesire) are 5 qubiti. Urmatoarea etapa consta in aplicarea operatorului $H^{\otimes 9}$ asupra primului registru, ceea ce are ca rezultat starea

$$|\psi_1\rangle = (1/\sqrt{512}) \sum_{j=0}^{511} |j\rangle_9 |0\rangle_5$$

Dupa aplicarea lui U_f starea cuantica devine

$$\begin{aligned} |\psi_2\rangle &= (1/\sqrt{512}) \sum_{j=0}^{511} |j\rangle_9 |2^j \bmod N\rangle_5 \\ &= (1/\sqrt{512})(|0\rangle|1\rangle + |1\rangle|2\rangle + |2\rangle|4\rangle + |3\rangle|8\rangle + |4\rangle|16\rangle + |5\rangle|11\rangle \\ &\quad + |6\rangle|1\rangle + |7\rangle|2\rangle + |8\rangle|4\rangle + |9\rangle|8\rangle + |10\rangle|16\rangle + |11\rangle|11\rangle \\ &\quad + |12\rangle|1\rangle + ...) \end{aligned}$$

Forma in care este scrisa aceasta stare sugereaza ca starile celui de-al doilea registru din fiecare coloana sunt aceleasi. In consecinta, putem rearanja termenii in forma

$$|\psi_2\rangle = (1/\sqrt{512})[(|0\rangle + |6\rangle + |12\rangle + ... + |504\rangle + |510\rangle)|1\rangle$$
$$+ (|1\rangle + |7\rangle + |13\rangle + ... + |505\rangle + |511\rangle)|2\rangle$$
$$+ (|2\rangle + |8\rangle + |14\rangle + ... + |506\rangle)|4\rangle$$
$$+ (|3\rangle + |9\rangle + |15\rangle + ... + |507\rangle)|8\rangle$$
$$+ (|4\rangle + |10\rangle + |16\rangle + ... + |508\rangle)|16\rangle$$
$$+ (|5\rangle + |11\rangle + |17\rangle + ... + |509\rangle)|11\rangle)]$$

Deoarece ordinul, 6, nu este o putere a lui 2, primele doua linii au 86 de termeni iar celelalte au cate 85 de termeni. Daca masor acum al doilea registru, obtin cu egala probabilitate unul din urmatoarele numere: {1,2,4,8,16,11}. Daca rezultatul masuratorii este 2,

$$|\psi_3\rangle = (1/\sqrt{86})(|1\rangle + |7\rangle + |13\rangle + ... + |505\rangle + |511\rangle)|2\rangle = (1/\sqrt{86})\left(\sum_{a=0}^{85} |6a+1\rangle\right)|2\rangle .$$

Starea $|\psi_3\rangle$ a fost renormalizata la unitate. Nu conteaza care este rezultatul masuratorii registrului al doilea, ci doar structura periodica a functiei de unda din primul registru in urma masuratorii. Perioada starilor primului registru este solutia problemei, iar pentru a gasi perioada se foloseste transformata Fourier inversa. Rezultatul dupa rearanjarea termenilor este

$$|\psi_4\rangle = \frac{1}{\sqrt{512}}\left[\sum_{j=0}^{511}\left(\frac{1}{\sqrt{86}}\sum_{a=0}^{85}\exp(-2\pi ij6a/512)\right)\exp(-2\pi ij/512)|\,j\rangle\right]|2\rangle .$$

Expresia lui $|\psi_4\rangle$ este similara cu cazul in care *r* era o putere a lui 2, dar cu o diferenta: deoarece 6 nu divide 512, nu mai pot folosi identitatea

$$N^{-1}\sum_{j=0}^{N-1}\exp(2\pi ijk/N) = \begin{cases} 1, \text{ daca } k \text{ este multiplu de } N \\ 0, \text{ in caz contrar} \end{cases}$$

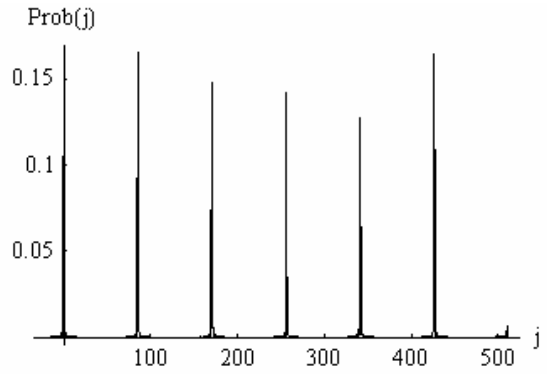
pentru a simplifica forma functiei de unda cuantice. In particular, desi valoarea 0 a sumei de mai sus nu se mai obtine, suma este in continuare semnificativa pentru valori *j* = 0, 85, 171, 256, 341, 427, care se obtin aproximand la un numar intreg $512k'/6$, pentru k' de la 0 la 5. Probabilitatea de a obtine rezultatul *j* (intre 0 si 511) masurand primul registru al $|\psi_4\rangle$ este

$$\text{Prob}(j) = \frac{1}{512\times 86}\left|\sum_{a=0}^{85}\exp(-2\pi ij6a/512)\right|^2$$

Aceasta probabilitate este cu atat mai mare cu cat $6j/512 = j/85.33$ este mai apropiat de un numar intreg. O reprezentare a acestei functii (vezi figura de mai jos) arata maxime in jurul *j* = 0, 85, 171, 256, 341, 427, valorile intre aceste maxime fiind foarte aproape de zero. Latimea maximelor depinde de *n* (numarul de qubiti in primul registru). Limita de jos $2^n \geq N^2$ asigura o probabilitate mare de a masura o valoare *j* care poarta informatia utila.

Daca in urma masuratorii primului registru obtin *j* = 0 (primul maxim), algoritmul nu da informatia dorita (vezi discutia de mai sus), si trebuie rulat inca o data. Mentinand *x* = 2, si ruland din nou algoritmul (partea sa cuantica), obtinem o probabilitate mica de a regasi *j* = 0: Prob(0) = 86/512 ≅ 0.167 (din formula de mai sus). Daca gasesc in urma masuratorii *j* = 85

(sau orice alta valoare in maximul al doilea), il impart la 512, valoarea 85/512 fiind o aproximatie rationala a $k'/6$, pentru $k' = 1$. Cum obtin r din 85/512?



Pentru aceasta se poate folosi metoda aproximatiei prin fractii continue. O dezvoltare generala in fractii continue a numarului rational j_1/j_2 are forma

$$\frac{j_1}{j_2} = a_0 + \cfrac{1}{a_1 + \cfrac{1}{\ldots + \cfrac{1}{a_p}}},$$

reprezentata ca $[a_0,a_1,\ldots,a_p]$, unde a_0 este un intreg ne-negativ, si $a_1,\ldots,a_p$ sunt numere intregi pozitive. Convergenta lui j_1/j_2 de ordin q, cu $0 \le q \le p$, definita ca $[a_0,a_1,..,a_q]$, are un numitor mai mic decat j_2. Aceasta metoda se aplica in cazul nostru inversand fractia 85/512, pentru a obtine 512/85, urmata de descompunerea acestei valori in fractii rationale sub forma 6 + 2/85. Repet procedura cu 2/85 pana obtin numaratorul 1:

$$\frac{85}{512} = \cfrac{1}{6 + \cfrac{1}{42 + \cfrac{1}{2}}}$$

Deci, convergentele lui 85/512 de ordin 1, 2, si 3 sunt, respectiv, 1/6, 42/253 si 85/512. Trebuie sa alegem convergentele care au un numitor mai mic decat $N = 21$ (deoarece $r < N$). Prin aceasta metoda selectez 1/6, din care rezulta $r = 6$. Verificam ca $2^6 \equiv 1 \bmod 21$, si partea cuantica a algoritmului se termina, obtinandu-se raspunsul dorit. Ordinul $r = 6$ este un numar par, deci cmdc($2^{(6/2)}\pm1$,21) sunt cei noi factori netriviali ai 21. Un calcul analog arata ca orice rezultat masurat in al doilea maxim (care se intinde, cu valori j semnificative, intre 81 si 89), da ca rezultat convergenta 1/6.

Observatie: daca r este o putere a lui 2, nu mai este necesar sa folosesc metoda aproximatiei prin fractii continue pentru a obtine r (sau un factor al acestuia) deoarece in acest caz maximele Prob(j) sunt functii delta pentru $2^n k'/r$ si k'/r obtinute prin impartire la 2^n sunt deja in forma unei fractii rationale cu numitorul mai mic decat N.

Masuratori asupra celui de-al treilea maxim, cu $167 \le j \le 175$, corespund la $k'/6$, cu $k' = 2$. Metoda aproximatiei cu fractii continue da 1/3 pentru orice j in interiorul maximului, in acest caz obtinandu-se un factor al lui r ($r_1 = 3$), deoarece $2^3 \equiv 8 \neq 1 \bmod 21$. Trebuie sa rulam din nou partea cuantica a algoritmului pentru a gasi ordinul lui 8. Se obtine $r_2 = 2$, deci $r = r_1 r_2 = 3 \times 2 = 6$.

Masuratori asupra maximului al patrulea si al cincilea dau din nou factori ai lui r. Ultimul maxim este similar cu al doilea, in sensul ca r se obtine direct. Probabilitatea de succes este urmatoarea: aria de sub toate maximele este aproximativ aceeasi, de $\cong 0.167$. Primul si al patrulea maxim, spre deosebire de celelalte, nu sunt extinse. Pentru a calcula contributia lor la probabilitatea totala, luam baza acestora egala cu 1. Aria de sub maximele al doilea, al treilea, al cincilea si ultimul se calculeaza adunand Prob(j), pentru j in jurul centrului fiecarui maxim. Rezulta ca in aproximativ 17% din cazuri algoritmul nu da rezultatul corect (primul maxim). In aproximativ 33% din cazuri se obtine r din prima rulare (al doilea si al saselea maxim), iar in aproximativ 50% din cazuri obtin r din a doua sau din mai multe rulari ale algoritmului (maximele 3, 4, 5). Probabilitatea de a gasi r la rularea a doua este: pentru maximele 3 si 5 factorul r_2 este 2, care se obtine in 50% din cazuri, iar pentru maximul 4 factorul r_2 este 3, obtinut in 66.6% din cazuri. In total (2×50% + 66.6%)/3 din 50%, care este egal cu 22%. Deci probabilitatea de succes pentru $x = 2$ in doua rulari este de 55% (obtinut din 33% + 22%).

Exemplu: in urma aplicarii transformarii $x^j \bmod N$ se obtine un sir de numere care, pentru $N = 55$, si pentru $x = 13$, de exemplu, este:
1, 13, 4, 52, 16, 43, 9, 7, 36, 28, 34, 2, 26, 8, 49, 32, 31, 18, 14, 17,
1, 13, 4, 52, 16, 43, 9, 7, 36, 28, 34, 2, 26, 8, 49, 32, 31, 18, 14, 17,.....
deci se obtine un sir de 20 numere care se repeta. Pentru a gasi perioada, adica 20, se aplica transformata Fourier inversa. Urmatorul pas este de a gasi numarul din mijlocul sirului de 20 de valori. Aceasta a zecea valoare este y. In cazul nostru $y = 34$, $y - 1 = 33$, $y + 1 = 35$. cmdc(33,55) = 11, cmdc(35,55) = 5. Deci cei doi factori p si q ai lui 55 sunt $p = 11$ si $q = 5$.

COMPLEXITATEA ALGORITMULUI SHOR

Prima etapa, de porti Hadamard aplicata starii $|0\rangle_n$, are ca si cost $O(\log_2 N) = O(n)$. Partea de ridicare la puteri modulo N are un cost in timp de $O[(\log_2 N)^2 \log_2\log_2 N \log_2\log_2\log_2 N] = O[n^2 \log_2 n \log_2\log_2 n]$. Transformata Fourier inversa are un cost de $O[(\log_2 N)^2] = O(n^2)$. Deci, costul total pentru a determina ordinul r cu probabilitate de succes $O(1)$ este $O[(\log_2 N)^{2+\varepsilon}]$, cu $\varepsilon > 0$. Odata ce determin r, trebuie sa calculez cmdc($x^{r/2} \pm 1, N$) pentru a gasi un factor al lui N, operatie ce necesita $O[(\log_2 N)^3]$ pasi daca aplic algoritmul Euclid (clasic!). In total, costurile pentru algoritmul complet de factorizare cu probabilitate mare sunt $O[(\log_2 N)^3] = O(n^3)$, reprezentand un castig subexponential fata de cel mai bun algoritm clasic cunoscut.

CONSIDERATII GENERALE REFERITOARE LA ALGORITMII CUANTICI

Un calculator clasic trebuie programat intr-un anumit limbaj pentru a rezolva o problema data. Un limbaj de programare face usoara citirea, scrierea si modificarea programului de calcul. Pe de alta parte, asa cum am vazut din exemplele de algoritmi cuantici, un calculator cuantic nu necesita programare. Daca trebuie rezolvata o problema, trebuia intai descrisa in detaliu ca o functie definita pe qubitii calculatorului cuantic, si apoi trebuie construita aplicatia specifica

(setul de porti) care actioneaza asupra qubitilor si toate interconexiunile necesare. Flexibilitatea calculatoarelor clasice este deci mult mai mare.

Una dintre cele mai importante probleme in calculul cuantic este gasirea algoritmilor cuantici. Foarte putini dintre ei sunt cunoscuti, deoarece nu exista principii generale de gasire a versiunii cuantice a unui algoritm clasic. Desi bazele calculului cuantic sunt bine intelese, nu exista nici o indicatie sau metoda generala pentru a construi algoritmi. Pentru a analiza strategiile clasice de gasire a unui algoritm, cu aplicatii in calculul cuantic, sa consideram problema calculului lui a^n mod p. Avem urmatoarele tipuri posibile de algoritmi:

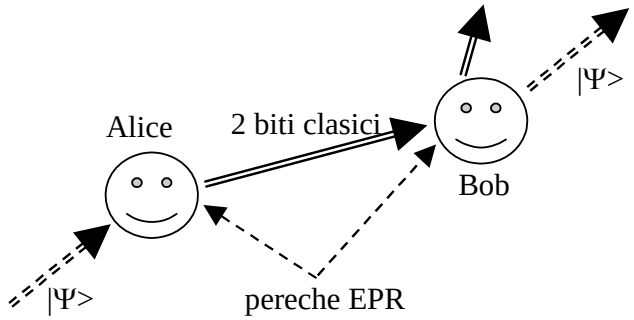
1) direct (brute-force), care rezolva problema inmultind direct a cu el insusi de n ori. Exemplu: $a^n = a \cdot a \cdot \ldots \cdot a$
2) divizeaza-si-cucereste (divide-and-conquer), in care problema originala este impartita intr-un numar de sub-probleme mai mici, de acelasi tip. Acestea sunt rezolvate iar solutiile lor sunt combinate pentru a gasi solutia problemei mai complicate. Aceasta strategie implica recursivitate pentru a obtine o eficienta mai mare. Exemplu: $a^n = a^{[n/2]} \cdot a^{[n/2]} \cdot a^{n-2[n/2]}$
3) scade-si-cucereste (decrease-and-conquer). Problema originala este redusa la una mai simpla, care este rezolvata de obicei recursiv si solutia astfel obtinuta este aplicata pentru a gasi solutia problemei originale. Exemple: a) $a^n = a^{n-1} \cdot a$ (scade-cu-unul), b) $a^n = (a^{[n/2]})^2$ daca n este par, $a^n = (a^{[n/2]})^2 \cdot a$ daca n este impar (scade-la-jumatate)
4) transforma-si-cucereste (transform-and-conquer). Problema initiala este transformata in alta echivalenta care este mai usor rezolvabila utilizand tehnici mai simple. Exemplu: a^n este calculat folosind reprezentarea binara a lui n

Dintre algoritmii cuantici pe care i-am studiat, algoritmul Deutsch-Jozsa, Simon si Grover apartin tipului 1), pe cand algoritmul Shor apartine tipului 4) deoarece reduce problema factorizarii la problema gasirii perioadei unei functii date. Algoritmul clasic Euclid apartine tipului 3), ca si metoda de calcul a puterii prin ridicare repetata la patrat. Diferenta intre algoritmii clasici si cuantici de acelasi tip este ca operatia cuantica este realizata printr-o transformare unitara care implementeaza calculul cuantic reversibil, dar strategiile sunt aceleasi. Desi tipul 1) de algoritmi are de obicei eficienta mica, exista cazuri (ca in problema Grover a cautarii) in care are performante superioare strategiilor mai sofisticate de tipul 2), de exemplu. Este interesant ca nu se cunosc algoritmi cuantici de tipul 2) sau 3), desi tipul 2) este cea mai generala si mai raspandita strategie pentru algoritmii clasici. In plus, un registru cuantic contine o superpozitie de mai multe stari in acelasi timp. Aceasta inseamna ca qubitii registrului cuantic sunt puternic corelati, starea cuantica totala putand sa nu fie separabila in produsul starilor unor sub-registrii mai mici. In consecinta, paralelismul cuantic si corelatia cunatica face ca orice incercare de a implementa o strategie de tipul divizeaza si cucereste sa nu fie potrivita/aplicabila registrelor cuantice, cel putin intr-un mod direct.

TELEPORTARE

Fenomenul de teleportare presupune existenta unui emitator si unui receptor de informatie (numiti generic Alice si Bob), aflati la distanta unul fata de celalalt, care sunt initial in posesia unei particule cuantice dintr-o pereche de particule corelate de tip EPR (vezi seminarul despre corelatia cuantica) dar care nu-si pot trimite stari cuantice mai tarziu, in timpul teleportarii (nu au acces ulterior la qubitii celuilalt). Teleportarea consta din transferarea unei stari cuantice necunoscute de la Alice la Bob cu ajutorul unui canal clasic de informatie (de exemplu, o convorbire telefonica); starea cuantica necunoscuta este distrusa de Alice si apoi reconstruita de Bob din informatie pur clasica si corelatii EPR pur cuantice. Teleportarea poate fi realizata chiar daca Alice nu cunoaste pozitia lui Bob (in acest caz comunicarea clasica se poate realiza printr-un telefon mobil, prin difuzarea informatiei printr-o statie radio, sau prin publicarea informatiei clasice intr-un ziar).

Mai exact, procedura consta in urmatoarele: Alice face intai o masuratoare in baza Bell pe sistemul constand din starea necunoscuta si particula ei EPR, adica transforma baza Bell in baza de calcul aplicand porti CNOT si Hadamard, si apoi masoara qubitii cu ajutorul portilor de masura. Apoi, in pasul al doilea, transmite rezultatul masuratorii (reprezentat prin doi biti clasici) lui Bob printr-un canal clasic de comunicare, iar ulterior, in pasul al treilea, Bob este capabil sa reproduca starea necunoscuta efectuand una dintre cele patru transformari unitare posibile (depinzand de rezultatul masuratorii lui Alice) asupra particulei sale EPR. In timpul acestui transfer (mai precis, dupa masuratoarea facuta de Alice), starea cuantica necunoscuta initiala, care este in posesia lui Alice, este distrusa (altfel as fi in conflict cu teorema no-cloning), si reapare in locatia unde se afla Bob, fara a trece prin stari intermediare. Teleportarea implica deci transmiterea unui qubit via doi biti clasici, si necesita un sistem cuantic care nu poate fi clonat si o comunicare clasica care nu se propaga mai repede decat lumina. Aceasta procedura se numeste teleportare, ca si in scrierile SF, cu toate ca doar informatia despre starea cuantica si nu starea cuantica in sine (sub forma de materie sau energie) trece de la Alice la Bob. Fenomenul de teleportare se poate reprezenta grafic ca in figura de mai jos.



Intr-o formulare matematica, Alice are o particula intr-o stare pe care nu o cunoaste,

$|\Psi\rangle = a|0\rangle + b|1\rangle,$

cu a, b numere complexe arbitrare necunoscute, si doreste sa-i transmita aceasta stare lui Bob; mai exact, doreste sa transfere starea in qubitul aflat in posesia lui Bob. Deoarece clonarea unei stari cuantice necunoscute este imposibila, Alice nu poate copia starea pentru a-i trimite copia lui Bob. De asemenea, ea nu poate masura starea cuantica pentru a-i transmite lui Bob informatii asupra lui a, b pentru ca masuratoarea nu ofera suficiente informatii asupra lui a si b si in acelasi timp distruge starea cuantica necunoscuta (obtin starile $|0\rangle$ si respectiv $|1\rangle$ cu

probabilitati $|a|^2$ si $|b|^2$, daca as putea repeta de mai multe ori masuratoarea). Solutia este: Alice si Bob impart o pereche de stari corelate EPR (sau Bell) aflate, de exemplu, in starea de singlet

$$|\Phi\rangle = (|01\rangle - |10\rangle)/2^{1/2} = (|0\rangle_A|1\rangle_B - |1\rangle_A|0\rangle_B)/2^{1/2},$$

unde indicii A, B indica persoana (Alice sau Bob) care ia particula respectiva. Perechea EPR plus starea necunoscuta sunt in starea

$$|\Delta\rangle = |\Psi\rangle\otimes|\Phi\rangle = [(a|0\rangle + b|1\rangle)\otimes(|01\rangle - |10\rangle)]/2^{1/2}$$
$$= (a|00\rangle_A|1\rangle_B - a|01\rangle_A|0\rangle_B + b|10\rangle_A|1\rangle_B - b|11\rangle_A|0\rangle_B)/2^{1/2}$$

In aceasta etapa nu exista nici o corelatie intre particula necunoscuta si perechea EPR, si deci nici o masuratoare asupra oricarei particule din perechea corelata nu poate da informatii asupra $|\Psi\rangle$; pasul urmator din protocolul de teleportare are ca scop crearea corelatiei intre particulele lui Alice. Exprimand starile lui Alice in functie de starile din baza Bell (vezi seminarul despre corelatia cuantica !)

$$|\psi^{\pm}\rangle = (|01\rangle \pm |10\rangle)/2^{1/2}, \quad |\phi^{\pm}\rangle = (|00\rangle \pm |11\rangle)/2^{1/2},$$

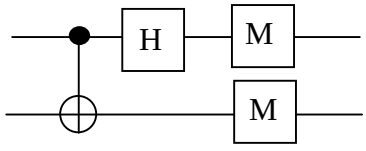
obtinem

$$|00\rangle = (|\phi^+\rangle + |\phi^-\rangle)/2^{1/2}, |01\rangle = (|\psi^+\rangle + |\psi^-\rangle)/2^{1/2}, |10\rangle = (|\psi^+\rangle - |\psi^-\rangle)/2^{1/2}, |11\rangle = (|\phi^+\rangle - |\phi^-\rangle)/2^{1/2},$$

astfel incat starea sistemului total se scrie

$$|\Delta\rangle = [|\phi^+\rangle_A(a|1\rangle - b|0\rangle)_B + |\phi^-\rangle_A(a|1\rangle + b|0\rangle)_B + |\psi^+\rangle_A(-a|0\rangle + b|1\rangle)_B + |\psi^-\rangle_A(-a|0\rangle - b|1\rangle)_B]/2.$$

Daca acum Alice masoara cele doua particule ale sale in baza Bell (creaza o corelatie intre particula in starea necunoscuta si particula ei din perechea EPR), obtine patru posibile rezultate cu probabilitate 1/4, si functie de acestea ii poate spune lui Bob ce sa faca cu particula sa ca sa obtina starea initiala $|\Psi\rangle$, reprezentata si ca vectorul $\begin{pmatrix} a \\ b \end{pmatrix}$. O masuratoare in baza Bell presupune aplicarea portilor CNOT si Hadamard urmate de porti de masura *M*, ca in circuitul de mai jos, in care cele doua linii reprezinta cei doi qubiti ai starii corelate EPR.



In urma aplicarii portii CNOT si Hadamard, starile corelate ortogonale din baza Bell devin (vezi seminarul despre corelatia cuantica; alternativ, verificati singuri):

$$|\psi^+\rangle \to |01\rangle, \; |\psi^-\rangle \to |11\rangle, \; |\phi^+\rangle \to |00\rangle, \; |\phi^-\rangle \to |10\rangle,$$

Deci, in urma masuratorii in baza Bell, Alice obtine ca rezultat doi qubiti in una din urmatoarele stari posibile: $|00\rangle$, $|01\rangle$, $|10\rangle$ sau $|11\rangle$. Aceasta informatie este transmisa lui Bob printr-un canal de informatie clasic (telefon), folosind doi biti clasici (fiecare de valoare logica

0 sa 1) care reprezinta rezultatul masuratorii asupra qubitilor din posesia lui Alice. De exemplu, daca Alice masoara $|00\rangle$, Bob aplica asupra particulei lui, aflata (conform formei starii sistemului total) in starea $(a|1\rangle - b|0\rangle)$, transformarea unitara

$$\sigma_{00} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

(identica cu $Y = i\sigma_y$), pentru a obtine

$$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}\begin{pmatrix} -b \\ a \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}.$$

Daca Alice masoara $|10\rangle$, ii transmite lui Bob rezultatul si acesta aplica asupra particulei sale $(a|1\rangle + b|0\rangle)$ transformarea unitara

$$\sigma_{10} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

(identica cu $X = \text{NOT} = \sigma_x$), pentru a obtine

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

Analog, daca Alice masoara $|01\rangle$, Bob aplica asupra starii $(-a|0\rangle + b|1\rangle)$ transformarea unitara

$$\sigma_{01} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

(identica cu $-Z = -\sigma_z$), pentru a obtine

$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} -a \\ b \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix},$$

iar daca Alice masoara $|11\rangle$, transformarea aplicata de Bob asupra starii $(-a|0\rangle - b|1\rangle)$ este

$$\sigma_{11} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

(identica cu $-I_2$), rezultatul fiind

$$\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}\begin{pmatrix} -a \\ -b \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}.$$

Teleportarea este o consecinta a corelatiilor nelocale care exista intre particulele ce constituie perechea EPR; corelatiile la distanta (entanglement) a perechii EPR impreuna cu

transmisia clasica a rezultatului masuratorii inlocuiesc transmisia fizica a qubitului intre doua locatii separate spatial. Ca efect secundar al teleportarii se produc si doi biti de informatie clasica aleatoare (rezultatul masuratorilor lui Alice), necorelata cu starea initiala $|\Psi\rangle$, care raman ca atare la sfarsitul procesului.

Experimente de teleportare cuantica au fost efectuate de diverse grupuri; de exemplu, grupul lui Boschi, in 1998 la Roma, a teleportat cu o rata de succes ideala de 100% o stare necunoscuta care a fost preparata intr-unul dintre fotonii EPR ca si qubit de polarizare. In acest experiment care foloseste doi fotoni corelati starea cuantica necunoscuta nu poate fi arbitrara, starile cuantice fiind codate in diferite grade de libertate ale aceleiasi particule. Un experiment de teleportare cu patru fotoni a fost realizat in 1997 de catre grupul lui Bouwmeester la Innsbruck, dar nu a fost posibila realizarea unei masuratori Bell complete; doar una dintre cele patru stari Bell a putut fi obtinuta cu o rata de succes de cel mult 25%. O schema eficienta de teleportare folosind trei qubiti de pozitie, care este replica exacta a propunerii originale de teleportare, a fost propusa in 2001. In aceasta schema nu este necesara folosirea a trei fotoni (se folosesc doar doi fotoni), dar trebuie sa contina trei qubiti, doi dintre ei generand perechea corelata. Avantajul fata de alte scheme de teleportare este ca starea cuantica necunoscuta poate fi arbitrara si toate cele patru stari Bell pot fi accesate. In afara de aceste scheme de teleportare care folosesc fotoni corelati creati printr-un proces parametric cu ajutorul unui cristal neliniar, mai exista propuneri de teleportare a particulelor cu masa, in particular atomi.

Folosirea starii corelate de singlet in experimentul de teleportare nu este esentiala; se poate folosi orice stare maximal corelata a unei perechi de particule, de exemplu

$$|\Phi\rangle = (|0\rangle|0\rangle + |1\rangle|1\rangle)/2^{1/2}.$$

Starea initiala 3-qubit este acum

$$|\Delta\rangle = (a|0\rangle_A + b|1\rangle_A)(|0\rangle_A|0\rangle_B + |1\rangle_A|1\rangle_B)/2^{1/2}.$$

Pentru a efectua masuratoarea in starea Bell, ca si in exemplul de mai sus, Alice aplica intai poarta CNOT folosind qubitul ei necunoscut ca si qubit de control si particula din perechea corelata ca si qubit tinta. Aceasta produce starea

$$a|0\rangle_A(|0\rangle_A|0\rangle_B + |1\rangle_A|1\rangle_B)/2^{1/2} + b|1\rangle_A(|1\rangle_A|0\rangle_B + |0\rangle_A|1\rangle_B)2^{1/2}.$$

Dupa aceea ea aplica transformarea Hadamard qubitului ei necunoscut (partea stanga din ecuatie), ceea ce ii pune pe toti cei trei qubiti in starea

$$\begin{aligned}
&a(|0\rangle_A + |1\rangle_A)(|0\rangle_A|0\rangle_B + |1\rangle_A|1\rangle_B)/2 + b(|0\rangle_A - |1\rangle_A)(|1\rangle_A|0\rangle_B + |0\rangle_A|1\rangle_B)/2 \\
&= |0\rangle_A|0\rangle_A(a|0\rangle_B + b|1\rangle_B)/2 + |1\rangle_A|0\rangle_A(a|0\rangle_B - b|1\rangle_B)/2 \\
&+ |0\rangle_A|1\rangle_A(a|1\rangle_B + b|0\rangle_B)/2 + |1\rangle_A|1\rangle_A(a|1\rangle_B - b|0\rangle_B)/2.
\end{aligned}$$

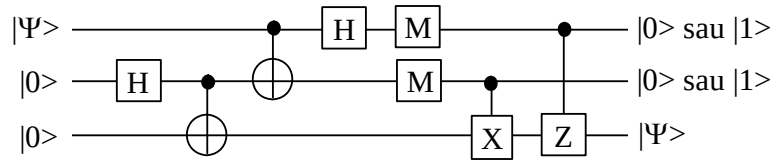
Alice masoara acum cei doi qubiti in posesia ei. Daca rezultatul este $|00\rangle$, qubitul lui Bob este in starea originala a qubitului necunoscut al lui Alice (a carui stare se reduce la $|0\rangle$), si Bob nu trebuie sa mai aplice nici o transformare deoarece $\sigma_{00} = I_2$. Daca rezultatul masuratorii lui Alice este $|10\rangle$, $|01\rangle$ sau $|11\rangle$, starile qubitului lui Bob devin, respectiv,

$$(a|0\rangle_B - b|1\rangle_B),\ \ (a|1\rangle_B + b|0\rangle_B),\ \ (a|1\rangle_B - b|0\rangle_B).$$

In fiecare din aceste trei cazuri, in urma unei transformari unitare starea qubitului lui Bob se modifica, devenind identica cu starea originala a qubitului lui Alice. In primul caz aplic $\sigma_{10} = Z$

(care lasa $|0\rangle$ neschimbat dar schimba semnul lui $|1\rangle$), in cazul al doilea aplic $\sigma_{01} = X$ (care inverseaza $|0\rangle$ si $|1\rangle$), iar in cazul al treilea, $\sigma_{11} = ZX$. Transformarile pe care trebuie sa le aplice Bob depind de alegerea starii perechii EPR.

Prin masuratorile pe care le face, Alice obtine informatia necesara pentru a reconstrui o stare cuantica, fara a sti care este aceasta stare. O stare necunoscuta a unui qubit este descrisa de doua numere complexe *a* si *b*, care pot lua valori continue cu singura restrictie $|a|^2 + |b|^2 = 1$. Dar, cu ajutorul unei perechi corelate, Alice ii transmite lui Bob un qubit in aceasta stare cu doar doi biti de informatie clasica (rezultatul celor doua masuratori ale sale) si cu ajutorul distrugerii corelatiei perechii initiale. Mai mult, Bob nu stie de asemenea valorile lui *a* si *b*. Un circuit care ilustreaza teleportarea, fara a putea reprezenta corelatiile la distanta dintre particulele din perechea EPR sau convorbirea clasica intre Alice si Bob, ci numai succesiunea de porti logice, este cel din figura de mai jos.



Cele doua porti din stanga transforma qubitii de jos in starea corelata (de tip EPR). Urmatoarele porti CNOT si *H* aplicate celor doi qubiti de sus sunt cele descrise in text; ele sunt urmate de doua porti de masura *M*. Deoarece Alice stie rezultatul masuratorii, ea stie daca portile care urmeaza, CNOT si control-*Z* sunt aplicate sau nu, si poate inlocui aceste porti printr-un telefon dat lui Bob in care ii spune daca sa aplice sau nu portile *X* si/sau *Z* asupra qubitului sau.

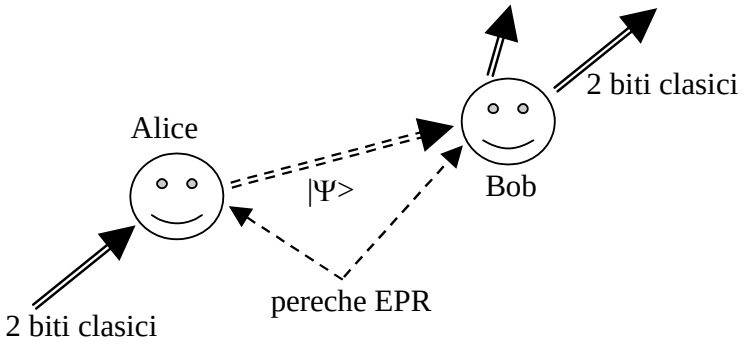
Atentie: Poarta X este de fapt o poarta NOT, si control-X (penultima operatie in circuitul de sus) este CNOT!

Teleportarea poate fi aplicata pentru transferarea informatiei cuantice la distanta mare, fara ajutorul sistemelor cuantice care sa conecteze fizic pozitiile initiale si finale ale qubitului. Teleportarea poate fi generalizata la qubiti corelati. Daca qubitul lui Alice nu are o stare proprie, ci este corelat cu alti qubiti, Alice poate teleporta rolul sau in starea corelata in qubitul lui Bob, acesta din urma devenind corelat asa cum a fost qubitul lui Alice, pe cand qubitul lui Alice devine total necorelat. In acest caz teleportarea se mai numeste si inversare a corelatiei, pentru ca avem initial doua perechi de fotoni corelati: 1-2 si 3-4, perechea de fotoni impartita de Alice si Bob fiind 3-4. Daca fotonii 2 si 3 (ai lui Alice) sunt implicati intr-o masuratoare in baza Bell, in urma acestui proces fotonii 1 si 4 devin corelati.

CODAREA DENSA A INFORMATIEI

In general, o stare cuantica intr-un spatiu Hilbert 2-dimensional poate fi folosita doar pentru a transmite 1 bit clasic de informatie. Mai precis, desi este necesara o cantitate infinita de informatie pentru a specifica starea unui singur bit $|\Psi\rangle = a|0\rangle + b|1\rangle$, (in sensul ca *a* si *b* pot lua o infinitate de valori) cineva care poseda acest bit nu poate afla in ce stare este. Daca Alice prepara un qubit in starea $|\Psi\rangle$ si il trimite lui Bob, tot ce poate face Bob este sa aplice o transformare unitara la alegere si apoi sa masoare qubitul, obtinand ca rezultat valoarea 0 sau 1 (cu probabilitati $|a|^2$ si respectiv $|b|^2$), adica un bit de informatie. Dupa aceasta starea qubitului este fie $|0\rangle$, fie $|1\rangle$, si nici o masuratoare ulterioara nu aduce informatii asupra starii originale $|\Psi\rangle$. Alice poate sa-i comunice lui Bob in acest fel doar un singur bit de informatie.

Din contra, in codarea densa (numita uneori si superdensa) a informatiei este posibil sa transmit 2 biti clasici de informatie trimitand un singur qubit. Codarea densa este o comunicatie clasica asistata de corelatia cuantica, scopul fiind transmiterea unei cantitati mai mari de informatii decat este posibil prin mijloace pur clasice. Intre teleportare si codarea densa exista o stransa legatura, dupa cum se poate vedea comparand schema teleportarii cu cea a codarii dense, ilustrata in figura de mai jos.



Pentru a coda dens informatia, Alice si Bob trebuie sa imparta o pereche EPR, de exemplu

$|\Phi\rangle = (|0\rangle_A|1\rangle_B - |1\rangle_A|0\rangle_B)/2^{1/2}$

inainte ca sa aiba loc comunicarea. (Daca perechea de particule pe care o imparte Alice si Bob nu este maximal corelata, cantitatea de informatie clasica transmisa este mai mica decat 2 biti.) Apoi, Alice aplica unul dintre operatorii unitari de mai sus, σ_{00}, σ_{01}, σ_{10} sau σ_{11}, asupra partii ei din sistem, depinzand de mesajul pe care vrea sa il trimita. Dupa aceea, Alice ii trimite lui Bob particula ei, starea ambelor particule fiind una din:

$$(\sigma_{00}\otimes I_2)\,|\Phi\rangle = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}\otimes\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}|\Phi\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ -1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 \\ 0 \\ 0 \\ -1 \end{pmatrix} = -|\phi^+\rangle$$

$$(\sigma_{10}\otimes I_2)\,|\Phi\rangle = -|\phi^-\rangle,\ (\sigma_{01}\otimes I_2)\,|\Phi\rangle = -|\psi^+\rangle,\ (\sigma_{11}\otimes I_2)\,|\Phi\rangle = -|\psi^-\rangle$$

Bob efectueaza acum o masuratoare Bell asupra ambelor particule, similar ca la experimentul de teleportare. Rezultatul lui Bob este una dintre starile $|00\rangle$, $|01\rangle$, $|10\rangle$, sau $|11\rangle$, adica Bob obtine exact rezultatul clasic constand din doi biti pe care Alice intentioneaza sa i-l trimita: 00, 01, 10 sau, respectiv, 11. Cei doi biti clasici ai lui Alice raman ca atare la sfarsitul procesului.

Daca starea cuantica pe care o impart la inceput Alice si Bob nu este maximal corelata, dupa ce Alice aplica transformarile unitare, starile cuantice obtinute nu sunt mutual ortogonale, dar sunt liniar independente; starile neortogonale nu pot fi distinse perfect, ci doar cu o anumita probabilitate de succes, egala cu $p = 2|\varepsilon|^2/(1 + |\varepsilon|^2)$, unde ε este gradul de corelatie. $\varepsilon = 1$ corespunde starilor maximal corelate. In acest fel, Bob obtine un numar $1+p$ de biti de informatie, cantitate de informatie care este (inca) mai mare decat cea care poate fi transmisa prin mijloace pur clasice.

De mentionat ca, spre deosebire de teleportare, unde era permisa transmiterea de informatie clasica intre Alice si Bob, dar acestia nu aveau acces la qubitul celuilalt, in cazul codarii dense Alice ii transmite lui Bob o particula cuantica, dar nu comunica clasic cu acesta.

Similar, daca perechea de qubiti corelati se afla initial in starea

$|\Phi\rangle = (|00\rangle + |11\rangle)/2^{1/2} = (|0\rangle_A|0\rangle_B + |1\rangle_A|1\rangle_B)/2^{1/2}$,

Alice prepara corespunzator particula ei, aplicand transformarile I_2, X, Z sau ZX qubitului ei din perechea corelata, depinzand de mesajul clasic pe care vrea sa il trimita lui Bob: 00, 01, 10 sau 11. Aceste transformari duc starea perechii corelate intr-una dintre cele patru stari mutual ortogonale, numite stari Bell:

$I_A|\Phi\rangle = (|0\rangle|0\rangle + |1\rangle|1\rangle)/2^{1/2}$, $X_A|\Phi\rangle = (|1\rangle|0\rangle + |0\rangle|1\rangle)/2^{1/2}$,
$Z_A|\Phi\rangle = (|0\rangle|0\rangle - |1\rangle|1\rangle)/2^{1/2}$, $Z_AX_A|\Phi\rangle = (|0\rangle|1\rangle - |1\rangle|0\rangle)/2^{1/2}$.

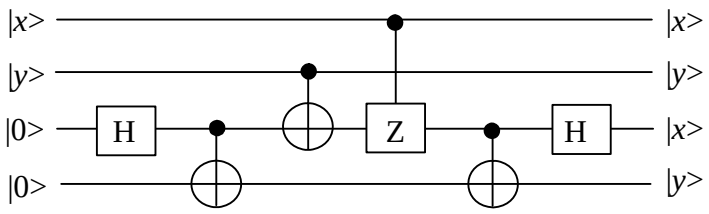
Dupa aceasta, ea trimite qubitul ei lui Bob. Acesta efectueaza o masuratoare Bell asupra ambelor particule, aplicand intai perechii transformarea CNOT C_{AB} in care foloseste qubitul pe care-l primeste de la Alice ca bit de control, si obtine

$C_{AB}I_A|\Phi\rangle = (|0\rangle + |1\rangle)|0\rangle/2^{1/2}$, $C_{AB}X_A|\Phi\rangle = (|0\rangle + |1\rangle)|1\rangle/2^{1/2}$,
$C_{AB}Z_A|\Phi\rangle = (|0\rangle - |1\rangle)|0\rangle/2^{1/2}$, $C_{AB}Z_AX_A|\Phi\rangle = (|0\rangle - |1\rangle)|1\rangle/2^{1/2}$

Dupa aceea aplica transformarea Hadamard, rezultatul fiind

$H_AC_{AB}I_A|\Phi\rangle = |0\rangle|0\rangle$, $H_AC_{AB}X_A|\Phi\rangle = |0\rangle\,|1\rangle$, $H_AC_{AB}Z_A|\Phi\rangle = |1\rangle|0\rangle$, $H_AC_{AB}Z_AX_A|\Phi\rangle = |1\rangle|1\rangle$.

In final, masoara cei doi qubiti, obtinand 00, 01, 10 sau 11, adica mesajul 2-bit pe care Alice vrea sa il trimita. Procedura de codare densa poate fi implementata cu circuitul din figura de mai jos care reprezinta succesiunea de porti logice aplicate, nu si corelatia cuantica la distanta a perechii EPR sau trimiterea particulei cuantice de la Alice catre Bob:



Daca se numeroteaza qubitii de sus in jos cu 1, 2, 3, 4, primele doua porti din partea stanga a figurii de mai sus actioneaza asupra starii $|0\rangle|0\rangle$ pentru a produce starea de tip EPR. Qubitul din perechea de jos, adica 4, este dat lui Bob, si qubitul 3 ii este dat lui Alice, care poseda si cele doua linii de sus, qubitii 1 si 2 ce contin mesajul. Cele doua porti care urmeaza, CNOT si control-Z actioneaza ca I_2, X, Z sau ZX asupra qubitului 3, in functie de starile qubitilor 1 si 2: $|0\rangle|0\rangle$, $|0\rangle|1\rangle$, $|1\rangle|0\rangle$ sau, respectiv, $|1\rangle|1\rangle$. Aceasta actiune reproduce transformarea pe care Alice o aplica particulei sale din perechea corelata, depinzand de valorile celor doi biti pe care vrea sa-i transmita lui Bob. Alice trimite apoi qubitul 3 lui Bob. Transformarea finala (ultimele doua porti) este transformarea pe care Bob o aplica perechii corelate reunite inainte de a efectua o masuratoare, care are ca rezultat valorile x, y pe care Alice a vrut sa i le transmita.

NOTIUNI DE CRIPTOGRAFIE CLASICA

Criptografia clasica presupune transmiterea unui mesaj codat de la un emitator de informatie (numit generic Alice) catre un receptor de informatie (numit generic Bob), fara ca mesajul sa fie interceptat de un spion (numit generic Eva). Mai precis, in criptografia clasica Alice codeaza mesajul *M* intr-un mesaj cifrat *C* cu ajutorul unei chei *K*, care este cunoscuta doar de Bob, si trimite mesajul cifrat *C* printr-un canal nesecurizat de informatie, mesajul putand fi interceptat de Eva. Bob primeste mesajul cifrat *C*, foloseste cheia *K*, si decodeaza mesajul *M*. In 1940 Shannon a aratat ca informatia care poate fi introdusa intr-un mesaj codat nu poate fi mai mare decat cantitatea de informatie in cheia de criptografie cu care s-a codat mesajul. Procedura de codare descrisa mai sus este vulnerabila pentru ca Alice si Bob trebuie sa comunice intai printr-un canal securizat pentru a stabili cheia *K*, inainte de a-si trimite alte mesaje. Deci, principala problema a criptografiei clasice este gasirea unei chei de comunicatie securizata, cu care sa codeze mesajul. In plus, mai este nevoie de o procedura de autentificare (Alice trebuie sa fie sigura ca vorbeste cu Bob si nu, de exemplu cu Eva) si de o procedura de detectie a interceptarii (Alice trebuie sa poata detecta daca Eva asculta convorbirea). Niciuna dintre aceste probleme nu este rezolvata satisfacator in criptografia clasica.

Un sistem de comunicatie criptografic este securizat din punct de vedere practic daca schema de incriptare poate fi sparta dupa *X* ani, unde *X* este determinat de nevoile de securitate si de tehnologia existenta. Multe coduri considerate securizate in trecut sunt usor de spart in prezent.

Exemple: 1) cifrul bazat pe transpunere, atribuit lui Iulius Cezar, in care literele dintr-un mesaj sunt deplasate cu un numar cunoscut (si secret) de pozitii in alfabet; nu este un cifru foarte sigur. Cifrul YORQRP EXP X HKFCB poate fi decodificat in BRUTUS HAS A KNIFE prin simpla incercare a celor 25 de posibile deplasari intre alfabetul mesajului si al cifrului, pana cand se obtine un mesaj cu sens. Metoda este nesigura pentru ca exista un numar limitat de moduri in care mesajul poate fi codat.
2) cifrul de substitutie, in care fiecare litera este inlocuita de un simbol; cifrul nu este sigur pentru ca spargerea sa se bazeaza pe frecventa cu care simbolurile apar, astfel ca, daca secventa este suficient de lunga, codul poate fi spart.

Sistemele de securitate moderna folosesc doua tehnici criptografice pentru a asigura confidentialitatea convorbirii: cu chei simetrice (secrete) si cu chei asimetrice (publice). Cele mai bune sisteme folosesc ambele tehnici pentru autentificare si pentru generarea unor chei secrete, folosite inr-o singura sesiune de comunicatie.

Sistemul criptografic cu cheie asimetrica foloseste un tip de securizare criptografica, numita securizare computationala/din punct de vedere al calculului, care defineste ca securizat un canal de comunicatie daca resursele de calcul necesare pentru a-l sparge sunt mai mari decat tot ce poate fi prevazut in prezent sau viitor. De exemplu, un cifru este securizat din punct de vedere al calculului daca numarul de biti din memoria calculatorului necesar pentru a-l sparge este mai mare decat numarul de atomi din univers, sau daca timpul de calcul necesar pentru a-l sparge este mai mare decat varsta universului. Pot fi create sisteme criptografice astfel incat sa fie nerealist din punct de vedere al calculului sa se gaseasca o cheie de decriptare *D* chiar daca cheia de codare *E* este cunoscuta. Pentru a crea un astfel de sistem criptografic este nevoie de o functie capcana (trap-door) *f* definita in felul urmator:
1) *f* se poate calcula usor (este calculabila intr-un timp polinomial), si
2) daca se da functia *f*, inversa acesteia, f^{-1} nu poate sa fie calculata intr-un timp polinomial (poate fi calculata intr-un timp suprapolinomial sau mai lung, sau poate sa nu fie calculabila).

Pana acum nimeni nu a demonstrat existenta unui sistem criptografic perfect securizat cu o functie capcana.

O functie capcana E poate fi folosita pentru a crea un sistem criptografic cu cheie publica, in sensul ca toate persoanele care vor sa comunice in secret isi aleg functia capcana E pe care o pot face publica (pot, de exemplu, sa o publice intr-un ziar, in "pagini galbene", etc.) dar trebuie sa tina secreta cheia de decriptare $D = E^{-1}$. Deci D este securizata chiar daca E este publica. Daca acum Alice vrea sa trimita un mesaj secret lui Bob, cauta in "paginile galbene" cheia de codare a lui Bob, E_B, codeaza mesajul M cu aceasta si produce textul cifrat $C = E_B(M)$, pe care il trimite printr-un canal public. Bob il poate apoi decripta cu cheia de decriptare D_B si gaseste mesajul $M = D_B(C)$. In principiu, Alice poate face mai mult: poate autentifica mesajul (il poate semna) astfel incat Bob sa stie cu certitudine ca mesajul pe care il primeste vine de la Alice si nu de la Eva, care s-ar putea da drept Alice. Alice poate face acest lucru codandu-si semnatura ALICE cu cheia de decriptare secreta D_A(ALICE), dupa care cripteaza mesajul M si semnatura D_A(ALICE) folosind cheia de codare a lui Bob E_B, care este cunoscuta public, pentru a produce textul codat semnat $C_S = E_B(M + D_A(\text{ALICE}))$ pe care il trimite lui Bob printr-un canal public. Bob decripteaza, ca si mai inainte, mesajul si semnatura, pe care o poate verifica uitandu-se dupa cheia de codare a lui Alice, E_A, in "paginile galbene", si folosind-o pentru a gasi $E_A(D_A(\text{ALICE})) = \text{ALICE}$. El autentifica astfel faptul ca Alice a trimis mesajul pentru ca doar ea stie cheia de decriptare secreta D_A, si ca urmare doar ea poate semna mesajul.

Un alt protocol folosit in criptografia cu cheie publica este cel de transfer necunoscut de informatie unul-din-doi (denumit si codare conjugata/conjugate coding), in care Alice are doua mesaje, M_1 si M_2 pe care i le trimite lui Bob, dar Bob alege doar unul din ele. Alice nu poate afla pe care mesaj l-a primit Bob. (Alternativ, Bob primeste doar jumatate din mesajele pe care i le trimite Alice, si stie daca a primit sau nu un mesaj, pe cand Alice nu stie ce se intampla cu mesajele trimise.) Acest protocol poate fi folosit in calcule securizate daca Alice si Bob calculeaza o functie cu doua argumente pe date pe care vor sa le tina secrete unul fata de celalalt, astfel incat ei stiu rezultatul functiei dar nu si argumentele respective.

Pe langa faptul ca rezolva problema autentificarii, sistemul de criptare cu cheie publica pare sa rezolve si problema unui mod securizat de comunicare a cheii. Un exemplu de astfel de sistem criptografic cu cheie publica este RSA (de la Rivest, Shamir, Adleman, in 1978, care foloseste algoritmul de factorizare a numerelor intregi mari ca functie capcana; dupa cum am vazut la algoritmul Shor problema factorizarii ar putea fi rezolvata mult mai repede cu un calculator cuantic, ceea ce pune in pericol siguranta sistemului RSA). O mare parte din sistemele de siguranta de pe internet folosesc sisteme de criptare cu cheie publica. Totusi, de obicei cheile de criptare si decriptare a unui sistem criptografic cu cheie publica sunt managerizate de o banca centrala de chei, care este o posibila sursa de scurgere a informatiei.

In sistemele de criptografie simetrice se utilizeaza o singura cheie pentru criptare si decriptare. Un exemplu de sistem criptografic simetric este cifrul Vernam, sau protectie folosita o singura data (titlul acestui sistem: one-time-pad, sau caiet de unica folosinta, folosit pentru prima data in al doilea razboi mondial, provine din faptul ca cheile erau scrise pe un caiet si de fiecare data cand acestea erau schimbate, foaia respectiva din caiet era rupta. Vom folosi in acest curs denumirea de protectie de unica folosinta pentru aceasta schema). Mesajul M este o secventa binara de 0 si 1 (obtinut, de exemplu, prin transpunerea textului original intr-un cod binar de tip ASCII): $M = M_1, M_2,\ldots, M_n,\ldots$, iar cheia consta dintr-o secventa aleatoare binara de aceeasi lungime: $K = K_1, K_2,\ldots, K_n,\ldots$. Mesajul cifrat este atunci secventa binara $C = C_1, C_2,\ldots, C_n,\ldots$ obtinuta prin adunarea bitilor corespunzatori ai secventelor M si K modulo 2: $C_i = M_i \oplus K_i$, pentru $i = 1, 2, 3, \ldots$

Exemplu: M = 0110 0101 1101
K = 1010 1110 0100
$C = M \oplus K$ = 1100 1011 1001

Bob, care are o copie a cheii, poate decodifica mesajul adunand modulo 2 mesajul cifrat primit cu o copie a cheii. El o obtine astfel $M \oplus K \oplus K = M$, deoarece $K \oplus K = 0$.

Cifrul este perfect securizat daca cheia K este total aleatoare si stiuta doar de Alice si Bob. Daca insa aceeasi cheie K este folosita pentru codarea a doua mesaje M si M' in texte cifrate C si C', $C \oplus C' = M \oplus M'$ si sistemul de cifrare se schimba dintr-un cifru perfect securizat intr-unul care poate fi usor spart datorita redundantei prezente de obicei in mesaj. Chiar daca folosesc protectia codurilor doar o singura data, problema este ca secventa lunga de biti trebuie trimisa printr-un canal securizat inainte de a putea fi utilizata.

Sistemul criptografic cu cheie protejata de unica folosinta poate fi vazut ca teleportare clasica, in sensul ca Alice foloseste cheia pentru a comunica securizat informatia dorita lui Bob, care poate apoi reconstrui (o copie a) mesajului/starii initiale. Alternativ, teleportarea este versiunea complet cuantica a sistemului criptografic cu protectie de unica folosinta; daca sistemul cuantic initial este un mesaj codat in forma unei secvente de qubiti, acesta poate fi transmis cu securitate maxima lui Bob, fara nici o interceptare (pentru ca doar Alice si Bob impart perechea EPR initiala). In acest caz insa nici Alice si nici Bob nu cunosc mesajul.

Criptografia cu protectie folosita o singura data este unicul sistem criptografic demonstrat ca fiind perfect securizat in sensul urmator: o comunicatie criptata este perfect securizata daca textul cifrat C nu ofera nici o informatie despre mesajul initial M chiar daca se cunoaste modelul/planul (designul) sistemului criptografic. Mai precis, aceasta conditie impune ca probabilitatea de a gasi mesajul M daca se da un text cifrat C este egala cu probabilitatea de a gasi mesajul M. Pentru a obtine securizarea perfecta cheia trebuie sa fie cel putin la fel de lunga ca si mesajul si nu trebuie folosita decat o data.

Datorita problemelor cu distribuirea secventelor lungi de biti cheie, protectiile cu folosire unica sunt utilizate in prezent doar in aplicatiile cele mai critice. Sistemele de criptografie simetrica folosite pentru aplicatii de rutina cum ar fi comertul electronic utilizeaza chei relativ scurte. Un astfel de sistem este DES (data encryption standard), care lucreaza cu chei semnificativ mai scurte decat mesajul codat (DES lucreaza cu o cheie de 56 biti si un algoritm public de codare; desi poate fi spart printr-o cautare a cheii asemanatoare cu codul de transpozitie, numarul de chei posibile este 7×10^{16} ($= 2^{56}$), in comparatie cu doar 25 in codul de transpozitie, si chiar daca un calculator ar verifica 10^6 chei pe secunda, i-ar lua 2000 ani pentru o cautare completa a codului. Sisteme similare sunt IDEA (international data encryption sytem), si AES (advanced encryption standard). Ca si sistemele criptografice asimetrice, sistemele simetrice cu chei mai scurte decat mesajul ofera doar securitate din punct de vedere al calculului. Dar, pentru o cheie de o anumita lungime, sistemele simetrice sunt mai sigure decat cele asimetrice. In implementari practice, algoritmii asimetrici sunt utilizati nu neaparat pentru criptare, datorita incetinelii lor, cat mai ales pentru distribuirea cheilor pentru diverse sesiuni in sistemele criptografice simetrice, ca DES.

CRIPTOGRAFIE CUANTICA

Criptografia cuantica nu ofera o solutie completa pentru toate problemele criptografice: chei securizate, algoritmi de criptare bazati pe acestea, autentificarea mesajelor si gasirea unor modalitati de prevenire (sau cel putin de detectare) a interceptarii; in acest sens criptografia cuantica poate fi vazuta ca un complement la sistemele criptografice simetrice standard. In particular, in criptografia cuantica nu exista un mecanism specific pentru autentificare. Avantajul criptografiei cuantice este ca ofera, in schimb, o procedura de distributie/schimb a cheilor cuantice care poate detecta automat daca are loc interceptarea comunicatiei. Securitatea

transmiterii mesajelor presupune nu doar existenta unor chei securizate, ci si securizarea celorlalte proceduri mentionate mai sus (algoritmi de criptare, autentificare). Deoarece algoritmi de criptare perfect securizati exista in criptografia clasica (vezi cifrul Vernam), si deoarece pentru autentificare se poate folosi o cheie secreta mai mica, adica o parte dintr-o cheie securizata comunicata printr-un protocol cuantic (eventual pastrata pentru comunicarea viitoare daca autentificarea se face la inceputul comunicarii), criptografia cuantica s-a concentrat indeosebi pe generarea de chei securizate.

Qubitii, datorita faptului ca se supun legilor mecanicii cuantice, pot oferi o baza securizata pentru schimbul de mesaje secrete. In realizarile experimentale ale criptografiei cuantice qubitii sunt in general fotoni, a caror stare de polarizare sau faza codeaza starile logice 0 sau 1. Astfel de qubiti care pot fi usor generati si care sunt transportabili prin fibre optice sunt foarte utili in comunicatii secrete deoarece Alice si Bob pot avea un cod care nu poate fi spart, daca impart intre ei secvente identice nou create de biti aleatori, numite coduri protejate de folosinta unica (one-time codepads). Daca Alice si Bob poseda secvente aleatoare identice, Alice poate lua mesajul ei, in forma unei secvente lungi de valori 0 si 1 si il poate transforma intr-o suma modulo 2 aplicata fiecarui bit cu o secventa aleatoare de valori 0 si 1 de aceeasi lungime luata din codul sau protejat de unica folosinta (aplica operatia XOR (exclusive OR) celor doua siruri de biti). Prin inversarea sau ne-inversarea fiecarui bit al unui mesaj coerent in functie de valoarea, 0 sau 1, a bitului corespunzator din secventa aleatoare, mesajul este convertit intr-o alta secventa aleatoare. Mesajul original nu poate fi recuperat fara a sti secventa aleatoare cu care a fost codat. Ca urmare, doar Bob, care are o copie a acestei secvente aleatoare, poate decodifica mesajul. El poate face asta aplicand XOR asupra secventei de 0 si 1 primite de la Alice, care acum nu mai are sens, si asupra copiei sale a aceleiasi secvente aleatoare pe care Alice a folosit-o la codare. Secventa pe care el o obtine in acest fel este $M \oplus S \oplus S$, unde M este mesajul, S este secventa aleatoare si $M \oplus S$ mesajul codat de la Alice. Deoarece $S \oplus S = 0$, Bob regaseste mesajul original.

Problema cu codurile protejate de unica folosinta este ca pot fi folosite doar o singura data. Daca un spion care asculta (Eva) receptioneaza doua mesaje codate cu acelasi cod protejat, poate aplica XOR asupra celor doua mesaje codate, ca si in cazul clasic discutat mai sus. Secventa aleatoare folosita pentru a coda cele doua mesaje dispare in urma acestui proces, ramanand doar XOR al celor doua mesaje necodate. Dar XOR a doua mesaje cu sens, combinat cu procedurile de spargere a erorilor obisnuite, bazate pe frecventa de aparitie a literelor, poate fi folosit (cu ceva mai multa sofisticare decat pentru un singur mesaj) pentru a separa si decoda ambele texte. Deci, pentru a fi in perfecta siguranta, Alice si Bob trebuie sa inlocuiasca continuu codurile protejate cu secvente aleatoare noi de biti.

PROTOCOLUL CUANTIC BB84

Spre deosebire de codul RSA, sau alte coduri cu chei publice asimetrice, BB84 isi bazeaza securitatea pe legi fizice (ale mecanicii cuantice) mai degraba decat pe complexitati matematice, si de aceea securitatea sa este independenta de eventualele progrese in tehnicile de calcul. Problema schimbului securizat al unor secvente aleatoare de biti este identica cu problema originala a schimbului de mesaje cu sens intr-un mod securizat. Cu ajutorul mecanicii cuantice un astfel de schimb securizat este posibil doar daca secventa de biti este aleatoare. Schema care asigura acest schimb securizat se numeste BB84 dupa inventatorii sai, Charles Bennett si Gilles Brassard, care au propus protocolul in 1984. BB84 consta in: Alice ii trimite lui Bob o lunga secventa de qubiti in forma unor fotoni polarizati aleator in una dintre patru stari posibile:

1) polarizare liniara orizontala (starea $|0\rangle$),
2) polarizare liniara verticala (starea $|1\rangle$),
3) polarizare diagonala la $\pi/4$ fata de verticala (starea $H|0\rangle$, obtinuta aplicand poarta Hadamard $H|0\rangle = (|0\rangle + |1\rangle)/2^{1/2}$ asupra starii $|0\rangle$),
4) polarizare diagonala la $-\pi/4$ fata de verticala (starea $H|1\rangle$, obtinuta aplicand poarta Hadamard $H|1\rangle = (|0\rangle - |1\rangle)/2^{1/2}$ asupra starii $|1\rangle$).

Aceste patru stari se obtin plasand un filtru polaroid in calea unui fascicul de lumina, si pot fi transmise la distante mare prin fibre optice, protejate de interactiile cu mediul inconjurator. Primele doua stari formeaza baza orizontal-verticala si ultimele doua stari formeaza baza diagonala de polarizare. Acestea sunt doua baze ortonormale cu care se poate descrie complet starea de polarizare a uni foton. (O alta posibilitate este sa aleg bazele orizontal-vertical si circulara dreapta-stanga daca lucrez cu fotoni polarizati.)

Pentru fiecare foton, Alice alege aleator un tip de polarizare (orizontal-verticala sau diagonala) si in cadrul fiecarui tip alege aleator o valoare a polarizarii (una dintre cele doua stari ortogonale asociate cu acel tip de polarizare. In limbaj formal, cele patru tipuri egal probabile de qubiti trimise de Alice lui Bob se pot imparti in doua categorii: starile $|0\rangle$ si $|1\rangle$, formeaza qubitii tip-S (standard), si starile $H|0\rangle$ si $H|1\rangle$ formeaza qubitii tip-H. Observatiile/masuratorile in baza tip-S sunt incompatibile cu masuratorile in baza tip-H (fotonii polarizati liniar orizontal-vertical sunt aleatorii in baza H, si invers); de aceea folosesc doua baze. Pe masura ce primeste un qubit, Bob decide aleator daca sa il trimita direct spre o poarta de masura sau sa ii aplice o poarta Hadamard inainte de a-l masura. Aceste doua optiuni reprezinta, respectiv, masuratori de tip-S si de tip-H. Qubitii trebuie sa poata fi individual identificabili (de exemplu, prin ordinea in care sosesc), astfel incat Bob si Alice sa-i poata identifica corect si sa poata compara informatiile despre ei.

Dupa ce Bob a masurat qubitii lui, Alice ii comunica printr-un canal nesecurizat care qubit este de tip-S si care de tip-H, dar nu-i spune in care dintre cele doua stari a preparat ea qubitii. Pentru acei qubiti (aproximativ jumatate dintre ei) pentru care alegerea aleatoare a lui Bob a tipului de masuratoare este in acord cu alegerea aleatoare a lui Alice a tipului de qubiti pe care-i trimite, Bob afla din rezultatul masuratorii sale care este valoarea aleator selectata (0 sau 1) pe care a trimis-o Alice. Pentru acei qubiti (cealalta jumatate) pentru care alegerea lui Bob a tipului de masuratoare nu este in acord cu alegerea lui Alice a tipului de qubit trimis, rezultatul masuratorii lui Bob nu ii spune nimic despre valoarea selectata aleator de Alice (vezi figura de mai jos, care arata si cazul erorilor de transmisie sau detectie, in care qubitul 6, de exemplu, nu este receptionat de Bob desi a fost trimis de Alice).

		1	2	3	4	5	6	7	8	9	10	….
Alice	preparare	S	H	H	H	S	S	H	S	H	H	….
	valoare	0	1	0	1	1	0	1	0	0	1	….
Bob	masuratoare	H	H	H	S	S		S	S	S	H	….
	valoare	1	1	0	0	1		1	0	0	1	….
	cod	1	0	0	1….							

In final, Bob ii transmite lui Alice, printr-un canal nesecurizat, care din qubitii pe care el i-a supus unui anumit tip de masuratoare coincide cu alegerea ei in ce tip sa-i prepare, adica care qubiti le ofera biti aleatori identici. Ei ignora ceilalti qubiti, si sunt astfel capabili sa construiasca coduri protejate de unica folosinta din secventa identica de biti aleatori pe care i-

au obtinut. Daca Eva nu a interceptat comunicarea pe cai clasice (un canal clasic este de obicei nesecurizat pentru ca bitii clasici pot fi copiati), secventele de biti ramase ale lui Alice si Bob sunt identice, altfel sunt diferite. Deoarece interventia Evei implica perturbarea sistemului si deci posibilitatea de a o prinde, mecanica cuantica asigura ca nimeni nu a intervenit daca bitii ramasi ai lui Alice si Bob sunt identici. In acest caz cheia este securizata. Alice si Bob nu folosesc canalul cuantic pentru a transmite informatia/mesajul, ci doar pentru a transmite o secventa aleatoare de biti (cheia!). In protocolul BB84, nici Alice si nici Bob nu decid individual asupra cheii ce rezulta din protocol; alegerile aleatoare ale ambilor produc cheia.

Motivul pentru care Alice si Bob trebuie sa renunte la jumatate din qubiti este securitatea fata de interceptarea comunicatiei. Daca Alice ar trimite toti qubitii de acelasi tip si Bob ar face intotdeauna acelasi tip de masuratori si ar obtine intotdeauna un bit care sa fie in acord cu cel al lui Alice (sau daca Alice ar trimite cu fiecare qubit informatia clasica despre tipul acestuia), Eva ar putea intercepta aceeasi informatie ca si Bob fara sa fie detectata. De exemplu, daca Alice si Bob cad de acord ca toti qubitii sa fie tip-S si Eva afla despre aceasta, ea poate intercepta fiecare qubit inainte ca sa ajunga la Bob si poate sa il masoare fara sa ii altereze starea, dupa care il trimite lui Bob. Ar putea in acest fel sa procure aceeasi informatie ca Bob, fara ca acesta sa-si dea seama. Dar daca Alice produce secret si aleator qubiti de tip-S si de tip-H, Eva nu mai poate folosi aceeasi strategie. In cel mai bun caz ea poate, ca si Bob, sa faca aceleasi masuratori aleatoare tip-S sau tip-H. Dar facand aceasta se tradeaza, pentru ca Bob si Alice pot afla daca Eva a compromis securitatea bitilor lor, folosind acelasi protocol ca si mai inainte, dar renuntand la cativa qubiti identici. Mai precis, Alice si Bob iau o parte din acesti qubiti si ii compara (printr-un canal clasic nesecurizat) pentru a vedea daca sunt identici, asa cum ar trebui sa fie in absenta intercepterii. Daca Eva a interceptat qubitii, si face masuratori aleatoare de tip-S sau de tip-H inainte de a-i trimite lui Bob, pentru jumatate din qubitii utili alegerea ei va diferi de alegerea comuna a lui Alice si Bob. In aproape jumatate din aceste cazuri interventia lui Eva va avea ca rezultat faptul ca masuratorile lui Bob vor fi diferite de ceea ce Alice ii trimite. Daca, de exemplu, Eva efectueaza o masuratoare de tip-S asupra unui qubit pe care Alice l-a preparat in statea $H|0\rangle$, ea va schimba starea acestuia in $|0\rangle$ sau $|1\rangle$, cu aceeasi probabilitate. In ambele cazuri, daca ulterior Bob aplica transformarea Hadamard inainte de masuratoare, va obtine rezultatul 0 doar jumatate din timp. Deci, daca Eva intercepteaza sistematic qubitii, rezultatele lui Bob nu mai corespund cu qubitii preparati de Alice pentru o patrime din date; aceasta indica o transmisie nesecurizata.

Daca Alice si Bob compara printr-un canal public portiuni mici din secventele de biti, pot estima rata de eroare (rata de interceptare de catre Eva, pentru care bitii lui Alice nu sunt identici cu cei ai lui Bob), si apoi pot renunta la bitii interceptati pentru a produce secventa finala de biti. Daca toate rezultatele concorda, cu exceptia unui numar mic de cazuri, Alice si Bob pot stabili o limita maxima a fractiunii din qubiti pe care Eva a interceptat-o, si pot evalua gradul de securitate cu care pot fi folositi qubitii ramasi. Ca urmare, Alice si Bob pot verifica daca Eva a ascultat comunicarea, iar in acest caz pot decide sa produca noi coduri protejate de unica folosinta.

O posibila intrebare legata de acest protocol de producere a codurilor protejate este de ce Bob nu asteapta sa decida ce tip de masuratoare sa faca pe fiecare qubit pana cand Alice ii transmite tipul fiecarui foton, dubland astfel numarul bitilor aleatori pe care ii au in comun. Cel putin in implementarea optica a BB84 (cu fotoni polarizati), aceasta ar presupune ca Bob sa memoreze qubitii pe care-i primeste de la Alice, ceea ce este in prezent dificil tehnologic pentru ca nu este usor sa se conserve polarizarea fotonului. Din acest motiv Bob trebuie sa ia decizia sa si sa masoare polarizarea fiecarui foton pe masura ce ii primeste.

Protocolul BB84 a fost experimental demonstrat pentru prima data in 1992 la IBM, folosind lumina de la un LED pulsat, astfel incat fiecare puls sa aiba in medie 0.1 fotoni. Cu ajutorul unui modulator de polarizare, prin modificarea tensiunii pe o celula Pockels, Alice poate alege polarizarea orizontala, verticala, circular stanga, sau circular dreapta. Bob are un alt

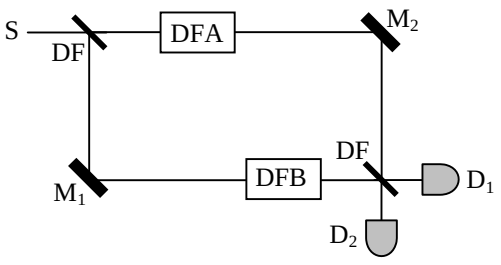
modulator si un detector pentru a determina polarizarea liniara sau circulara a fotonilor. In acest fel s-a demonstrat realizarea unei chei securizate din 105 biti intr-o transmisiune pe o distanta de 32 cm in spatiu liber care a durat 10 minute. Pentru comunicatii optice in spatiul liber se prefera surse de lumina cu lungimea de unda 800 nm, pentru care exista detectori eficienti ce masoara un singur foton (fotodiode cu avalansa cu eficienta cuantica de 80%), si absorbtia aerului este mica. Dar o astfel de transmisie presupune sa nu existe nici un obstacol intre Alice si Bob, deoarece altfel lumina sufera imprastieri. In plus, transmisia depinde de conditiile atmosferice, si trebuie luata in calcul si divergenta fasciculului la receptor.

Pentru transmiterea fotonilor pe distante de ordinul kilometrilor se pot utiliza fibre optice, insa folosirea fotonilor polarizati pe distante lungi are dezavantaje deoarece birefringenta in partile (inevitabil) curbate ale fibrei transforma starea liniar polarizata in polarizare eliptica la care se adauga pierderi in fibra si dispersia modurilor ortogonale de polarizare (care au viteze de grup diferite). De aceea se foloseste codificare in faza in loc de polarizare a fotonilor. In 1994 British Telecom a demonstrat protocolul BB84 transmitand fotoni codati in faza la peste 30 km cu ajutorul fibrelor optice de telecomunicatii, care au doua lungimi de unda standard: 1550 si 1300 nm (pentru care absorbtia este minima). Experiente similare au fost facute in 1999 la Los Alamos, reusindu-se transmisia fotonilor la 50 km, pentru transmisii peste 100 km fiind necesari repetitori de semnal. La lungimile de unda de 1300 si 1550 nm eficienta detectorilor care pot masura un foton este insa mai mica (in jur de 20% pentru fotodetectori de InGaAs sau Ge raciti prin efect Peltier pana la –20 C); alternativ, pentru detectie se pot utiliza scheme de interferenta cu un singur foton.

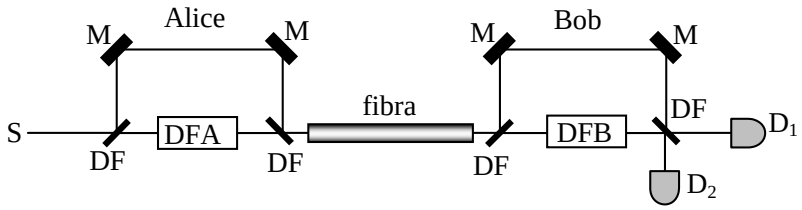
In implementarile protocolului BB84 se utilizeaza pulsuri care au cel mult un foton per puls (marea majoritate au in jur de 0.1 fotoni per puls). Este important sa avem un singur foton in fiecare moment de timp, pentru ca altfel Eva ar putea separa fotonii (cu ajutorul unui divizor de fascicul) si ar putea sa masoare polarizarea circulara sau liniara. In acest caz interventia Evei ramane nedetectata dar intensitatea fasciculului care ajunge la Bob este mai mica. Dezavantajul acestei implementari este ca fluctuatiile statistice in numarul fotonilor fac ca ocazional sa am doi sau mai multi fotoni in acelasi timp. Acest dezavantaj nu mai exista daca lucrez cu perechi de fotoni corelati.

Un sistem de distribuire a cheilor trebuie sa fie suficient de rapid pentru ca dispozitivele de criptare sa nu ajunga sa-si epuizeze stocul de biti cheie; exista deci o competitie intre rata de generare a bitilor codului si rata la care sunt consumati pentru criptare si decriptare. Sistemele actuale folosesc fibre optice sau chiar propagarea in spatial liber pentru a produce coduri cu rate de pana la 5000 biti/s la distante de 10–20 km in spatiu liber sau 50 km prin fibre optice standard de telecomunicatie. Aceste rate de generare a codurilor nu sunt suficient de rapide pentru a permite traficul cu coduri protejate de unica folosinta, dar permit generarea de noi chei pentru algoritmi de criptografie conventionali, cum ar fi AES. Viteza de generare a codurilor este limitata in prezent de rata cu care se poate detecta un singur foton. Exista insa detectoare noi bazate pe materiale superconductoare criogenice care ar putea inlatura acest obstacol.

Un exemplu de implementare a unui cod securizat bazat pe interferenta si pe folosirea a patru stari neortogonale (codate in faza) ale unui singur foton, este ilustrat in figura de mai jos.



Sursa S genereaza fotoni care sunt trimisi de primul divizor de fascicul DF in cele doua brate ale interferometrului Mach-Zehnder. Alice aplica un defazaj/deplasare de faza aleator de 0, $\pi/2$, π sau $3\pi/2$ intr-unul din brate, prin DFA (valoarea logica 0 este reprezentata prin defazaj 0 in baza S sau $\pi/2$ in baza H, iar valoarea logica 1 este reprezentata prin defazajul π in baza S sau $3\pi/2$ in baza H). Bob, prin DFB, genereaza un defazaj aleator de 0 sau $\pi/2$ (isi defineste una dintre cele doua baze de masura posibile) in celalalt brat. Dupa transmisia cuantica, Alice si Bob cad de acord sa pastreze doar acele masuratori in care defazajele lor difera prin 0 sau π, caz in care folosesc baze compatibile pentru masuratori si fotonii detectati la unul sau la celalalt dintre cei doi detectori plasati dupa al doilea divizor de fascicul DF pot fi atribuiti cu certitudine valorilor 0 sau 1 logic. Daca diferenta de faza dintre fotonii lui Alice si Bob este de $\pi/2$ sau $3\pi/2$, bazele folosite de ei sunt incompatibile si fotonii ajung la unul dintre cele doua detectoare aleator.



O varianta (implementata in practica daca Alice si Bob sunt separati spatial) este aratata mai sus. Alice si Bob sunt fiecare in posesia unui interferometru identic, nebalansat (cu unul din brate mai scurt decat celalalt) definit de divizorii de fascicul DF care impart pulsul incident de lumina in pulsuri de egala amplitudine. Deoarece bratele interferometrelor sunt inegale, in fiecare dintre acestea un puls initial este transformat la iesire in doua pulsuri, care corespund parcurgerii bratului lung, respectiv scurt al interferometrelor. Sursa S este un laser de lungime de unda 1550 nm foarte atenuat, care produce pulsuri coerente de lumina cu un numar mediu de fotoni < 1. Transmisia intre cele doua interferometre este asigurata de o fibra de telecomunicatii standard, care transmite pulsuri la 1550 nm si 1300 nm. Pulsurile trimise prin bratele scurte ale interferometrului sunt modulate ca si mai sus de Alice, respectiv Bob. La iesire, se obtin trei pulsuri. Primul este dat de pulsurile care strabat bratele scurte ale interferometrelor. Al doilea puls, intarziat fata de primul, este compus din suma pulsurilor care trec prin bratul lung al interferometrului lui Alice si prin bratul scurt al interferometrului lui Bob si, respectiv, prin bratul scurt al interferometrului lui Alice si prin bratul lung al interferometrului lui Bob. Acest puls contine termenul de interferenta. Al treilea puls, intarziat fata de primele doua, contine pulsurile care parcurg bratele lungi ale ambelor interferometre. Semnalul util este pulsul din mijloc, identificarea cheii protejate facandu-se similar ca in montajul anterior. Inaintea trimiterii pulsului slab de 1550 nm Alice trimite un puls puternic de 1300 nm, pentru a-l avertiza pe Bob sa-si pregateasca detectorii si modulatorii de faza in vederea prelucrarii impulsului semnal care urmeaza.

Din decembrie 2002 exista protocoale cuantice de generare a cheilor, bazate principial pe sistemul din figura de mai sus, care functioneaza continuu cu scopul protejarii traficului pe internet. Astfel de sisteme, care functioneaza in USA si Elvetia la scara redusa, folosesc modularea in faza a fotonilor pentru codare si contin fotodetectori raciti, cu avalansa, care detecteaza cate un singur foton.

AMPLIFICAREA CONFIDENTIALITATII

In criptografia cuantica se utilizeaza doua canale de comunicare: un canal unidirectional cuantic (prin care Alice ii trimite qubitii lui Bob), si un canal bi-directional public, care poate fi ascultat si de Eva. Daca sistemul de comunicare criptografic are erori/zgomot (toate au!), Bob nu poate distinge intre erorile cauzate de zgomot (de exemplu, nereceptarea unui qubit din cauza erorilor de transmisie sau detectie) si cele datorate interceptarii de catre Eva; de aceea el presupune ca toate erorile se datoreaza Evei si in acest sens o cheie de comunicare este doar partial securizata. Pentru a tine cont de prezenta zgomotului, se foloseste o metoda, numita amplificarea confidentialitatii (privacy amplification), pentru a extrage o cheie secreta mai mica dintr-o cheie mai mare partial secreta. Protocolul este similar cu BB84, cu urmatoarele modificari: daca rata de eroare, cauzata de bitii pe care Bob ar trebui sa-i primeasca si nu i-a primit fie din cauza Evei fie din cauza zgomotului/erorii de receptie din sistem, este mai mica decat o valoare maxima, Alice si Bob renunta la bitii cu erori pentru a produce o cheie de comunicare fara erori, cheia acceptata. (Daca erorile sunt mai mari decat o valoare maxim admisa, se reia procedura si se genereaza un nou cod protejat de unica folosinta). Deoarece cheia acceptata este doar partial secreta fata de Eva, urmeaza amplificarea confidentialitatii: Alice si Bob estimeaza, din rata de eroare gasita anterior, limita maxima k a numarului de biti ai cheii acceptate pe care Eva ii stie. Daca numarul total de biti ai cheii acceptate este n si s este un parametru de securitate, Alice si Bob selecteaza public $n - k - s$ subseturi aleatoare ale cheii acceptate, fara sa faca public continutul lor. Paritatile nedivulgate ale acestor subseturi devin cheia secreta finala. Se poate arata ca informatia medie a Evei despre cheia secreta finala este mai mica decat $2^{-s}/\ln 2$ biti.

Paritatea unei secvente de biti este reprezentata printr-un bit care indica daca numarul de valori 1 din secventa este par sau impar. Exista doua tipuri de biti de paritate: pari, care iau valoarea 1 daca numarul de valori 1 din secventa este impar, si impari, care iau valoarea 1 daca numarul de biti 1 din secventa este par. O alternativa la bitul de paritate este bitul ce rezulta daca se efectueaza operatia XOR pe toti bitii din secventa.

O varianta a acestui protocol este urmatoarea: Alice si Bob cad de acord asupra unei permutatii aleatoare a pozitiei bitilor in secventele lor (pentru ca pozitia erorilor sa devina aleatoare), dupa aceea impart secventa permutata in blocuri de dimensiune k astfel incat un singur bloc este presupus a nu avea mai mult de o eroare. Pentru fiecare din aceste blocuri Alice si Bob compara paritatea lor. Blocurile cu paritati identice sunt considerate corecte, cele neidentice sunt supuse unei noi cautari, fiind impartite in $\log(k)$ sub-blocuri a caror paritate este comparata, pana cand eroarea este gasita si corectata. Pentru a preveni scurgerea de informatie catre Eva in timpul procesului, Alice si Bob cad de acord sa nu foloseasca ultimul bit al fiecarui bloc sau sub-bloc a carui paritate a fost facuta publica. Unele erori raman nedetectate daca sunt in blocuri sau sub-blocuri cu numar par de erori. Pentru a le elimina, permutarea aleatoare a blocurilor si comunicarea paritatii acestora se poate repeta de cateva ori, cu blocuri de dimensiuni mai mari.

DETECTAREA INTERCEPTARII TRANSPARENTE A INFORMATIEI

Pana acum s-a presupus ca prezenta Evei se manifesta prin interceptare opaca (in sensul ca Eva poate doar intercepta si masura fotonii lui Alice si, dandu-se drept Alice, poate trimite fotoni in starile masurate lui Bob). Dar, exista si interceptare transparenta: Eva poate interactiona unitar

cu fotonii trimisi de Alice, si ii poate retrimite lui Bob intr-o stare usor modificata, sau poate corela un foton cu cel al lui Alice, pe care il redirectioneaza apoi spre Bob. Astfel de interceptari sunt mult mai greu detectabile.

Sa presupunem ca $|\phi_m\rangle$, m = 1, 2, 3, 4 sunt cele patru stari posibile ale qubitului lui Alice: $|0\rangle$, $|1\rangle$, $H|0\rangle$, si $H|1\rangle$. Daca $|\Psi\rangle$ este starea initiala a celor n qubiti din calculatorul Evei, si daca U este o transformare unitara $(n + 1)$-qubit, care actioneaza asupra qubitilor Evei si a qubitului lui Alice, conditia ca qubitul lui Alice sa ramana in starea originala (si deci ca prezenta Evei sa nu fie detectata) este:

$$U(|\phi_m\rangle \oplus |\Psi\rangle) = |\phi_m\rangle \oplus |\Psi_m\rangle.$$

Eva ar trebui sa gaseaca o transformare U care sa dea patru stari $|\Psi_m\rangle$ distincte, care sa ii permita sa cunoasca in ce stare a fost $|\phi_m\rangle$. Acest lucru este insa imposibil deoarece nici unul dintre $|\phi_1\rangle$ sau $|\phi_2\rangle$ nu este ortogonal pe $|\phi_3\rangle$ sau $|\phi_4\rangle$, si de aceea toti $|\Psi_m\rangle$ sunt identici. Aceasta concluzie rezulta din unitaritatea lui U, care conserva produsele scalare, si din care rezulta ca produsul scalar a oricaror doua stari de intrare din calculatorul Evei trebuie sa fie egal cu produsul scalar al starilor de iesire corespunzatoare, adica

$$\langle\phi_m|\phi_n\rangle\langle\Psi|\Psi\rangle = \langle\phi_m|\phi_n\rangle\langle\Psi_m|\Psi_n\rangle.$$

Dar, deoarece $\langle\Psi|\Psi\rangle = 1$ si $\langle\phi_m|\phi_n\rangle \neq 0$ pentru mn = 13, 14, 23, 24, rezulta ca $\langle\Psi_m|\Psi_n\rangle = 1$ pentru mn = 13, 14, 23, 24. Produsul scalar a doua stari normalizate este 1 doar daca starile sunt identice, ceea ce implica

$$|\Psi_1\rangle = |\Psi_2\rangle = |\Psi_3\rangle = |\Psi_4\rangle.$$

Pretul platit de Eva pentru eliminarea urmelor interceptarii este ca ea nu poate cunoaste, din starile de iesire ale calculatorului ei cuantic, in care dintre cele patru stari posibile este qubitul lui Alice. Este deci imposibil sa obtin informatie de la un sistem fizic fara sa-l perturb intr-un mod aleator, incontrolabil; acest principiu asigura securitatea schimbului de chei cuantice intre utilizatori. Protocoale cuantice ca BB84, si altele similare, sunt securizate neconditionat, adica chiar in prezenta unui intrus cu putere de calcul nelimitata. Pe de alta parte, criptografia cu chei publice devine nesigura daca se fac progrese in algoritmii de factorizare a intregilor, si in particular daca se va construi un calculator cuantic.

PROTOCOLUL CUANTIC B92

Pentru a construi un protocol cuantic mai simplu se poate folosi doar un singur tip de stari, cu doua valori: de exemplu, starea $|\theta_+\rangle$ cu valoarea logica 1 pentru un foton polarizat liniar la un unghi θ fata de verticala, cu $0 < \theta < \pi/4$, si starea $|\theta_-\rangle$ cu valoarea logica 0 pentru un foton polarizat la un unghi $-\theta$. Acest exemplu face parte din procedura/protocolul B92 (Bennett, 1992) de generare a codurilor protejate care foloseste doua stari neortogonale ale unui sistem cuantic, notate $|u\rangle$ si $|v\rangle$. Daca

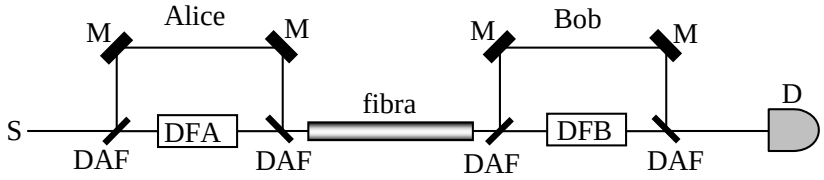
$$P_u = 1 - |v\rangle\langle v|, \quad P_v = 1 - |u\rangle\langle u|$$

sunt operatori de proiectie pe subspatiile ortogonale lui $|v\rangle$ si, respectiv, $|u\rangle$, care indica raspunsul detectorului aflat in posesia lui Bob cand Alice ii trimite starile $|u\rangle$, respectiv $|v\rangle$, Alice incepe generarea cheii preparand si trimitand lui Bob o secventa binara aleatoare de sisteme cuantice folosind starile $|u\rangle$ si $|v\rangle$ ca bitii 0 si 1. Dupa aceea Bob masoara fotonii primiti de la Alice, si ii spune lui Alice in ce momente masuratorile lui dau rezultat pozitiv (dar nu ce masuratori face). Prin rezultat pozitiv intelegem rezultatul unei masuratori prin care Bob sa poata identifica cu certitudine starea qubitului trimis de Alice. Bitii receptionati in momentele in care se obtin rezultate pozitive devin cheia de comunicare, amandoi cazand de acord sa nu ia in considerare bitii sositi in celelalte momente. Daca nu exista interceptare, momentele ramase, care formeaza o fractie de aproximativ $(1 - |\langle u|v\rangle|^2)/2$ din toate incercarile, trebuie sa fie perfect corelate, constand doar din momentele in care Alice trimite $|u\rangle$ si Bob masoara P_u, sau Alice trimite $|v\rangle$ si Bob masoara P_v.

Fractiunea $(1 + |\langle u|v\rangle|^2)/2$ reprezinta momentele in care masuratorile lui Bob nu dau un rezultat pozitiv. Spre deosebire de protocolul BB84, B92 are intotdeauna erori, deoarece doua stari neortogonale nu pot fi distinse neambiguu fara a fi perturbate.

Inainte ca Alice si Bob sa aiba incredere in aceasta cheie, trebuie, ca si in alte scheme de generare de chei securizate, sa sacrifice o parte din date ca sa verifice daca versiunile lor ale cheii sunt intr-adevar identice. Aceasta procedura asigura comunicatia impotriva interceptarii, care ar perturba starile $|u\rangle$ si $|v\rangle$, respectiv P_u si P_v.

Un exemplu de implementare a protocolului B92 este dat in figura de mai jos. Notatiile sunt aceleasi ca si in figura anterioara, cu exceptia DAF, care semnifica un divizor asimetric de fascicul, care imparte pulsul incident in pulsuri de amplitudine diferita; ne putem referi la pulsul de amplitudine mica ca fiind atenuat de DAF.



In aceasta figura starile neortogonale $|u\rangle$ si $|v\rangle$ sunt pulsuri slabe de lumina (cu un numar mediu de fotoni < 1), care difera in faza fata de un puls de referinta de intensitate mare (cu un numar mediu de fotoni > 1) care acompaniaza aceste doua stari. Sursa de lumina S produce pulsuri coerente de lumina care sunt impartite de DAF intr-un puls de intensitate ridicata A care se propaga de-a lungul bratului lung al interferometrului, si un puls de intensitate scazuta, a, care trece prin DFA si poate fi defazat aleator cu 0 sau π pentru a coda bitii 0 si 1. Rezultatul este deci un puls scazut in intensitate urmat de un puls cu intensitate ridicata. Aceste pulsuri sunt transmise printr-o fibra pana la interferometrul lui Bob, unde sufera transformari similare. Daca suma defazarilor introduse de Alice si Bob este 0 sau π, pulsurile slabe interfera constructiv, respectiv distructiv, pulsul de interferenta, de intensitate $2a$, respectiv 0, ajungand la fotodetector inaintea pulsului de referinta de intensitate mare, intarziat de Alice si Bob, dar neatenuat de nici unul dintre ei. Pulsul de interferenta este format din suprapunerea fasciculului intarziat de Alice (pentru ca parcurge bratul lung) si atenuat de Bob (pentru ca trece prin bratul scurt si este atenuat de DAF) si a pulsului intarziat de Bob si atenuat de Alice. Pulsul de referinta, intens, este de asemenea monitorizat pentru a fi sigur ca pulsul de interferenta a sosit deja si pentru a confirma absenta interceptarii de catre Eva. (Eva nu poate suprima pulsul de referinta fara a fi detectata, deoarece reducerea intensitatii acestuia este observabila, dar ar putea intercepta pulsul de interferenta, producand erori la detectorul lui Bob.) Inaintea pulsului de interferenta mai apare un puls de intensitate foarte mica, atenuat de ambele interferometre, care nu este luat in considerare.

Prima implementare a protocolului B92 a avut loc in 1995, folosindu-se fotoni polarizati trimsi printr-o fibra de 22.8 km intre Geneva si Nyon in Elvetia. In 1998 protocolul B92 cu fotoni codati in polarizare a fost utilizat pentru transmisia unui cod secret cu rata de 5 kHz in spatiu liber pe o distanta de peste 0.5 km; cheia astfel generata a fost folosita pentru codarea unei fotografii cu opt biti per pixel.

PROTOCOALE CARE FOLOSESC STARI CORELATE

Exista diferite protocoale cuantice de criptografie. Unele, mai sofisticate, sunt bazate pe perechi EPR. Intr-o versiune care pare diferita de BB84, dar este identica cu aceasta, sa presupunem ca exista o sursa care produce qubiti in starea corelata

$$|\Psi\rangle = (|00\rangle + |11\rangle)/2^{1/2},$$

una dintre aceste particule fiindu-i trimisa lui Bob si cealalta lui Alice. Aplicand transformarea Hadamard asupra fiecarei particule in parte obtinem

$$(H\otimes H)(|00\rangle + |11\rangle)/2^{1/2} = (|00\rangle + |11\rangle)/2^{1/2},$$

deci daca Alice si Bob fac masuratori de acelasi tip, obtin aceleasi rezultate. Acest protocol pare la prima vedere mai sigur decat BB84 pentru ca qubitii sunt corelati pana cand Alice si Bob fac masuratorile. Daca Eva ar intercepta un qubit inainte ca acesta sa ajunga la Alice sau Bob, bitii corelati apar in momentul masuratorii Evei. Acest moment este dupa cel din protocolul BB84 (cand fiecare bit exista din momentul in care Alice face propriile masuratori) dar destul de devreme pentru a o ajuta pe Eva in interceptare.

Daca Alice si Bob decid sa produca bitii aleatori perfect corelati facand intotdeauna masuratori tip-S, si daca Eva afla, poate intercepta un membru al perechii corelate facand propria masuratoare tip-S, decoreland starea initiala prematur, astfel incat sa afle ce bit aleator este, fara a altera corelatia intre valorile pe care Alice si Bob le vor masura. Aceasta posibilitate poate fi exclusa daca Alice si Bob aleator (si independent) alterneaza masuratori tip-S cu masuratori tip-H, si apoi folosesc procedura BB84 in care qubitii nu erau corelati. In consecinta, ne intoarcem la protocolul orginal, care nu folosea perechi corelate.

Daca Alice masoara particula ei din perechea corelata (tip-S sau tip-H) inainte ca Bob sa masoare, aceasta este echivalenta cu actiunea lui Alice de a-i trimite lui Bob un qubit cu o stare selectata aleator, pe care ea o cunoaste. Diferenta este ca nu Alice prepara starea, ci aceasta este rezultatul masuratorii. Deci, orice diferenta aparenta intre protocolul care foloseste corelatia cuantica EPR si BB84 este doar superficiala. Experimente de generare a unei chei securizate cu perechi de fotoni corelati in polarizare, au fost efectuate de mai multe grupuri in 1999-2000, cheile fiind produse cu rate de pana la 0.8 kHz cu o rata de eroare de 3%.

Intr-o varianta a unui sistem criptografic bazat pe corelatia cuantica Alice si Bob, separati spatial, testeaza inegalitatile Bell pe perechi de particule corelate EPR pentru a genera numere aleatoare identice. Daca masuratorile nu satisfac inegalitatile Bell (vezi seminarul despre corelatia cuantica), perechile EPR nu au fost interceptate de o a treia persoana. De exemplu, Alice si Bob pot alege aleator intre trei axe coplanare pentru masuratori de spin aleatoare pe qubiti (de spin) separati spatial. Dupa ce o serie de perechi EPR au fost produse si masurate, Alice si Bob anunta public (astfel incat chiar si adversarul poate asculta) care axe au folosit, dar nu si rezultatul masuratorilor. Dupa aceea, cad de acord sa neglijeze toate masuratorile in care au masurat dupa axe diferite, si masuratorile in care rezultatele au fost

eronate datorita detectorilor imperfecti. Toate masuratorile ramase trebuie sa fie perfect corelate; pentru a verifica aceasta, Alice si Bob compara public rezultatul masuratorilor pe un subset aleator suficient de mare (mai mult decat jumatate) din masuratorile ramase. Daca acest subset este perfect corelat, se presupune ca subsetul ramas netestat este de asemenea perfect corelat, si deci reprezinta o sursa pentru numere aleatoare comune.

In particular, daca un adversar ar intercepta comunicatia, masurand una sau ambele particule in drumul lor de la sursa EPR la observatorii legitimi, inegalitatea Bell sau nu mai este satisfacuta, sau este satisfacuta dar cu alti coeficienti. Dar nu se poate sti in acest protocol daca sursa de perechi EPR nu este inlocuita de o sursa falsa, care da informatii inamicului. Alice si Bob nu pot exclude cazul in care Eva foloseste o sursa EPR care produce trei particule corelate. Aceasta nu o ajuta insa pe Eva sa intercepteze informatia fara a fi detectata pentru ca, daca ea genereaza o stare corelata de tipul cel mai general

$$|\Psi\rangle = |00\rangle|A\rangle + |11\rangle|B\rangle + |01\rangle|C\rangle + |10\rangle|D\rangle$$

(0 spin sus, 1 spin jos), cu $|A\rangle$, $|B\rangle$, $|C\rangle$, $|D\rangle$ arbitrare, pentru a se sustrage detectiei, starea $|\Psi\rangle$ trebuie sa fie o stare proprie a operatorului $(\sigma_z)_A(\sigma_z)_B$ cu valoare proprie -1, pentru ca orice pereche (cu spini antiparaleli) are o sansa ca ambii membri sa fie masurati de-a lunugl axei z si apoi sa fie inclusa in setul test. (Indicii A si B indica faptul ca primul operator este aplicat de Alice asupra particulei ei din starea corelata, iar al doilea operator este aplicat de Bob asupra particulei aflate in posesia sa.) Deci pentru a nu fi detectata prin datele masuratorilor dupa z Eva trebuie sa se limiteze la starile

$$|\Psi\rangle = |01\rangle|C\rangle + |10\rangle|D\rangle.$$

Printr-un rationament similar, fiecare pereche poate fi masurata de-a lunugl axei x de ambii observatori, deci trebuie sa fie o stare proprie a $(\sigma_x)_A(\sigma_x)_B$ cu valoare proprie -1. Aceasta cerinta impune

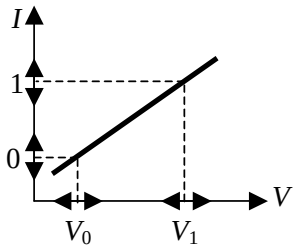
$$|\Psi\rangle = (|01\rangle - |10\rangle)|C\rangle,$$

adica particula Evei este complet necorelata cu perechea EPR; in acest caz orice masuratoare a Evei nu divulga nimic despre perechea corelata. Alternativ, in loc ca Alice si Bob sa aleaga aleator intre trei axe coplanare pentru masuratori de spin, pot alege intre trei baze de polarizare liniara neortogonala pentru fotoni corelati. Ca functie de unghiul de polarizare fata de axa verticala, de exemplu, se poate defini baza 1, cu stare logica 0 pentru polarizare liniara la unghi 0 si stare logica 1 pentru polarizare la $3\pi/6$, baza 2, cu stare logica 0 pentru polarizare la $\pi/6$ si 1 pentru $4\pi/6$, respectiv baza 3, cu stare logica 0 pentru $2\pi/6$ si 1 pentru $5\pi/6$.

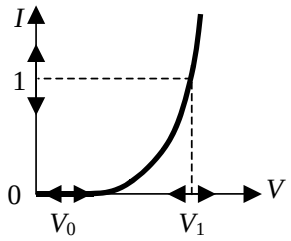
Vulnerabilitatea protocolului EPR la substituirea sursei nu exista in BB84. Pe de alta parte, vulnerabilitatea schemei BB84, absenta in protocolul EPR, este ca informatia exista in momentul in care poate fi interceptata, pe cand la protocolul EPR este creata mai tarziu. Dar nici in protocolul EPR nici in BB84 spionul nu poate citi informatia fara sa fie detectat. In protocolul EPR adversarul poate crea informatie pe care Alice si Bob pot sa o ia drept reala, dar asta desconspira adversarul. In BB84 informatia exista, spionul o poate intercepta dar, deoarece este codata in stari neortogonale, o poate citi doar daca perturba corelatiile. Spionul are insa sansa sa afle o portiune mica din cheie, fara a fi detectat, interceptand doar cateva particule care nu sunt in subsetul testat. Intr-o versiune moderna a BB84 subsetul test este inlocuit de tehnici mai sofisticate de corectare a erorilor si mixare care ii permit lui Alice si Bob sa gaseasca o cheie ultrasecreta daca datele lor au fost compromise de interceptare la limita statistica a detectibilitatii sau in prezenta altor surse de pierderi.

METODE DE CORECTARE A ERORILOR DE CALCUL CUANTIC

Intr-un calculator clasic problema corectarii erorilor este simpla deoarece sistemele fizice care codeaza bitii individuali (componente electronice de tip tranzistor, capacitor, etc.) sunt imense la scara atomica. In consecinta, cele doua stari ale unui bit care reprezinta 0 si 1 logic sunt atat de diferite incat probabilitatea inversarii (a trecerii dintr-o stare in alta) ca rezultat al fluctuatiilor termice, vibratiilor mecanice, sau alte interactii, este infinitezimala. In particular, in cazul calculatoarelor electronice erorile de calcul se pot minimiza prin folosirea componentelor electronice cu prag (neliniare) pentru definirea starilor logice 0 sau 1. In acest caz diferenta intre starile logice este una calitativa (de tip trece curent sau nu trece) si nu una cantitativa (curent mai mare sau mai mic, ca la dispozitivele liniare), la fluctuatii ale tensiunii, de exemplu, diferenta intre starile logice fiind mai bine definita decat la dispozitivele liniare (vezi figura de mai jos).



dispozitiv liniar



dispozitiv neliniar

Chiar daca apar erori de inversare a bitilor, ele sunt corectate la cea mai apropiata valoare digitala 0 sau 1 imediat ce sunt detectate. Problema corectarii erorilor apare clasic doar in cazul transmiterii informatiei la distante mari deoarece cu cat semnalul se propaga pe distante mai mari, cu atat se atenueaza si diferenta intre valorile logice se reduce. Exista mai multe moduri de a rezolva aceasta problema. Cea mai primitiva este de a coda fiecare bit logic in trei biti, astfel incat 0 si 1 sa fie inlocuiti de cuvintele cod (codewords) 000 si, respectiv, 111. Ulterior aceste cuvinte cod sunt monitorizate pentru a se gasi eventualele inversari ale bitilor individuali, care apoi pot fi corectate folosind regula majoritatii. Monitorizarea trebuie sa fie suficient de des efectuata astfel incat probabilitatea ca mai mult de un singur bit sa fie inversat intr-un cuvant cod intre inspectii succesive sa fie neglijabila.

Din contra, problema corectarii erorilor de calcul este una conceptual dificila in mecanica cuantica deoarece erorile le pot detecta doar prin masuratori, care perturba starea qubitilor masurati, si deci produce erori si mai mari. Primul algoritm cuantic de corectare a erorilor a aparut in 1995, si a fost propus de Shor. Principiul de a folosi cuvinte cod multi-bit pentru corectarea erorilor se poate folosi si in calculatoarele cuantice pentru a face improbabila aparitia unor tipuri de erori. Mai precis, un cod de corectare a erorilor cuantice este o transpunere a k qubiti pe care dorim sa-i ferim de erori, care ocupa un spatiu Hilbert de dimensiune 2^k si sunt numiti “qubiti logici” sau “qubiti codati”, in n qubiti (intr-un spatiu de dimensiune 2^n), cu $n > k$. Cei $n – k$ qubiti ramasi ne ajuta sa memoram cei k qubiti logici intr-un mod redundant astfel incat informatia codata sa nu fie usor perturbata. Insa corectia erorilor intr-un calculator cuantic este diferita de problema clasica analoaga deoarece:

1) intr-un calculator cuantic corectarea erorilor este esentiala, deoarece qubitii fizici sunt sisteme la scara atomica (atomi, fotoni, ioni in capcana, momente nucleare magnetice, etc.). In consecinta, orice cuplare la alte grade de libertate care nu este explicit controlata de calculator si programul sau (de succesiunea de porti) poate perturba substantial starea

asociata acestor qubiti, corelandu-i cu grade de libertate irelevante din punct de vedere al calculului, care devine incorect. Pentru un calculator cuantic nu se poate lucra fara corectia erorilor deoarece altfel fiecare qubit ar fi imposibil de izolat fata de interactii cu alte parti ale calculatorului irelevante din punct de vedere al calculului sau fata de orice fel de interactii cu mediul inconjurator.

2) a cauta erorile intr-un calculator cuantic este problematic pentru ca monitorizarea qubitilor implica masurarea acestora, ceea ce altereaza starea qubitilor si, in particular, distruge eventualele corelatii cuantice cu alti qubiti. Este deci necesara gasirea altor metode de monitorizare a qubitilor.
3) inversarea bitilor nu este singura eroare posibila, deoarece qubitii pot exista intr-o superpozitie de stari de tip 0 sau 1. De aceea exista erori care nu se intalnesc in calculatoare clasice, digitale. Un exemplu ar fi erorile de faza de tipul alterarii starii $|0\rangle + |1\rangle$ si transformarii ei in $|0\rangle - |1\rangle$.
4) spre deosebire de inversarea bitilor, care este o eroare discreta clasica de tipul totul-sau-nimic si care apare brusc cand perturbatiile asupra sistemului fizic ating un prag destul de ridicat, erorile unei stari cuantice cresc progresiv fata de starea necorupta/corecta, perturbatiile neintentionate asupra starii qubitului acumulandu-se pana cand calculul nu mai da rezultate corecte.

CODUL 3-QUBIT DE CORECTARE A ERORILOR

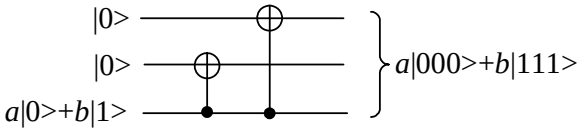
O ilustrare a procedurii de corectare a erorilor in cazul calculatoarelor cuantice presupune ca singura eroare posibila este inversarea aleatoare a qubitilor individuali. Acest tip de eroare poate fi modelata de un circuit care difera de cel ideal, in care nu se iau in considerare posibilele erori, doar prin prezenta ocazionala a unor porti aditionale NOT de tip 1-qubit (portile NOT inverseaza valorile logice 0 si 1). Aceste porti simuleaza interactiile nedorite care duc la inversarea unui bit. Daca portile NOT sunt suficient de rare, perturbatiile pe care le introduc in circuit pot fi corectate tripland numarul de qubiti (analog cu cazul discutat mai sus pentru calculatoarele clasice) si folosind un cod de tip 3-qubit. Intr-o procedura clasica acest cod presupune inlocuirea fiecarei stari din baza de calcul x, cu $x = 0$ sau 1, cu cuvantul de cod $\overline{x} = xxx$. O stare cuantica arbitrara insa nu poate fi codata intr-un cod repetitiv de tipul

$$|\overline{\psi}\rangle = |\psi\rangle\,|\psi\rangle\,|\psi\rangle = |\psi\rangle \otimes |\psi\rangle \otimes |\psi\rangle$$

datorita teoremei no-cloning. De aceea, daca ne referim la qubiti, pot presupune doar ca superpozitia $a\,|0\rangle + b\,|1\rangle$ poate fi automat codata de un circuit in starea

$$a\,|\overline{0}\rangle + b\,|\overline{1}\rangle = a\,|0\rangle\,|0\rangle\,|0\rangle + b\,|1\rangle\,|1\rangle\,|1\rangle = a\,|000\rangle + b\,|111\rangle,$$

pentru a si b arbitrari, in absenta oricarei indicatii asupra valorilor a si b. Aceasta codare se poate realiza cu doua porti CNOT care actioneaza asupra a doi qubiti aditionali (ancilla), ambii in starea $|0\rangle$ (vezi figura de mai jos). In toate figurile de mai jos ordinea qubitilor este de sus in jos (qubitul 1 este prima linie, qubitul 2 a doua, etc.).



Dupa ce starea $a|\bar{0}\rangle + b|\bar{1}\rangle$ a fost produsa, trebuie s-o protejam impotriva perturbarii datorate unei posibile actiuni ale unei porti NOT aditionale care actioneaza asupra a cel mult unul dintre cei trei qubiti. Aceasta se realizeaza usor pentru biti clasici, unde exista doar doua stari necorupte posibile initiale: 000 si 111, si examinarea acestora se poate face fara a perturba sistemul. Daca toti cei trei biti sunt in aceeasi stare, nu s-a produs nici o eroare. Daca unul dintre ei este intr-o stare diferita de a celorlalti doi biti, acest bit a fost supus actiunii neintentionate a unei porti NOT aditionale, si poate fi corectat prin aplicarea unei alte porti NOT. Daca p este probabilitatea inversarii bitului si $1 - p$ probabilitatea ca acesta sa ramana necorupt, metoda descrisa functioneaza daca unul sau nici un bit este inversat si da gres daca doi sau toti trei biti sunt inversati. Deci, probabilitatea de eroare a procedurii este probabilitatea ca doi sau trei biti sunt inversati, care este

$$p_{\text{eroare}} = 3p^2(1 - p) + p^3 = 3p^2 - 2p^3.$$

Avem $p_{\text{eroare}} < p$ daca $3p^2 - 2p^3 < p$, adica daca $p(1 - p)(2p - 1) < 0$, sau $p < 1/2$. Daca $p < 1/2$, multiplicarea de trei ori a bitului original reduce probabilitatea de eroare.

In cazul qubitilor starea nu poate fi citita fara a fi perturbata. Singurul mod de a extrage informatia dintr-un set de qubiti este prin intermediul portilor de masura, care au ca rezultat distrugerea superpozitiei $|\Psi\rangle = a|000\rangle + b|111\rangle$ si transformarea ei in starea $|000\rangle$ cu probabilitate $|a|^2$ sau in starea $|111\rangle$ cu probabilitate $|b|^2$. Toate formele posibile ale cuvantului cod necorupt format din 3 qubiti (toate valorile posibile ale a si b) se afla intr-un subspatiu 2-dimensional al spatiului $2^3 = 8$-dimensional care contine toate starile posibile de tip 3-qubit.

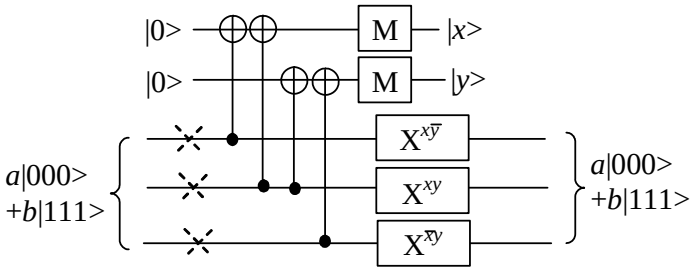
Un efect similar, de distrugere a superpozitiei datorita masuratorii, se intalneste la fiecare dintre cele trei stari corupte posibile

$$|\Psi_1\rangle = \text{NOT}_1|\Psi\rangle = a|100\rangle + b|011\rangle$$
$$|\Psi_2\rangle = \text{NOT}_2|\Psi\rangle = a|010\rangle + b|101\rangle$$
$$|\Psi_3\rangle = \text{NOT}_3|\Psi\rangle = a|001\rangle + b|110\rangle$$

(indicele operatorului NOT indica qubitul asupra caruia actioneaza) astfel incat orice dependenta post-masurare de amplitudinile complexe a si b nu este evidenta. Fiecare dintre aceste trei stari corupte se afla la randul ei intr-un subspatiu 2-dimensional al spatiului starilor 3-qubit. Cele trei subspatii care contin starile corupte posibile sunt fiecare ortogonale pe subspatiul ce contine cuvantul cod necorupt si ortogonale una pe alta; aceasta proprietate este esentiala pentru corectarea erorilor.

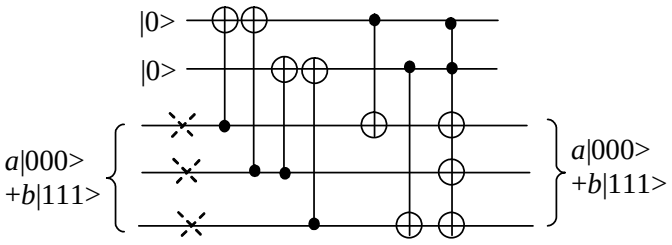
In general, daca vrem sa folosim un cuvant cod de n qubiti intr-un model in care singurele erori posibile sunt inversari ale unui singur qubit, am avea nevoie de $2(1 + n)$ dimensiuni pentru cele $n + 1$ subspatii 2-dimensionale mutual ortogonale asociate cu starea necorupta si cele n coruptii diferite de tip 1-qubit. Deoarece toate starile posibile ale n qubiti formeaza un spatiu 2^n dimensional, o conditie necesara pentru o metoda de corectare a erorilor n-qubit este $2^n \geq 2(1 + n)$, sau $2^{n-1} \geq (1 + n)$. Cel mai mic n care satisface aceasta relatie este $n = 3$, pentru care conditia este de egalitate. Deci, codul 3-qubit este suficient pentru corectarea erorilor de tip inversare a unui singur qubit.

Pentru a extrage informatia, astfel incat efectele perturbatoare nu mai sunt resimtite de cuvintele cod ci de qubiti aditionali (ancilla), cuvintele de cod se cupleaza cu qubitii aditionali prin porti unitare 2-qubiti. Masurand apoi qubitii aditionali pot extrage informatii asupra relatiilor existente intre qubitii din cuvintele cod. Aceasta informatie limitata este suficienta pentru a diagnostica si corecta anumite erori pastrand superpozitia, fara a cunoaste starea necorupta originala a cuvintelor cod. Necunoasterea starii necorupte este o restrictie necesara a oricarei proceduri de corectare a erorilor capabila sa refaca exact starea necorupta. In particular, pentru cuvintele cod 3-qubit, detectarea si corectarea erorilor necesita doi qubiti aditionali care sunt initial in starea $|0\rangle$, reprezentati in figura de mai jos prin liniile de sus.



X-urile punctate din stanga reprezinta porti NOT neintentionale, care creaza inversari/erori ale qubitilor (vezi cursul despre porti logice pentru notatia X pentru poarta NOT), iar operatiile M reprezinta masuratori. O linie deasupra unui qubit, *x* sau *y*, reprezinta negarea valorii: $\overline{0}=1$, $\overline{1}=0$. Portile de corectie X din partea dreapta se aplica dupa ce se masoara qubitii aditionali; daca starea initiala nu este corupta, qubitii aditionali raman in starea initiala $|0\rangle$, starea lor fiind deci $|00\rangle$, si nu se aplica porti de corectie deoarece $X^0 = 1$ (adica identitate). Cele trei stari corupte posibile care indica inversarea unui qubit, enumerate mai sus, produc stari finale diferite pentru qubitii aditionali: $|10\rangle$, $|11\rangle$ si respectiv $|01\rangle$ (primul qubit aditional este reprezentat de prima linie de sus, al doilea fiind indicat prin a doua linie) daca primul, al doilea sau al treilea qubit din cuvantul cod este inversat. Dupa masurarea qubitilor aditionali pot vedea care qubit din cuvantul cod a fost inversat si apoi il pot corecta, fara sa stiu nimic despre forma starii initiale (fara sa stiu nimic despre amplitudinile *a* si *b*). Aceasta procedura extrage doar informatia asupra corelatiilor intre qubitii din cuvantul cod.

Procedura ilustrata mai sus de gasire si corectare a erorilor necesita porti de masura a qubitilor aditionali, rezultatul acestor porti trebuind sa fie cunoscut inainte de a corecta erorile. Portile de masura pot fi eliminate si astfel se pot gasi circuite care corecteaza automat erorile, cum ar fi cel din figura de mai jos (verificati urmarind evolutia fiecarui bit). Succesiunea portilor CNOT si CCNOT din figura de mai jos lasa nemodificati qubitii necorupti si ii corecteaza doar pe cei corupti.



Acest circuit poate fi folosit de mai multe ori pentru corectarea erorilor daca bitii auxiliari sunt resetati la $|0\rangle$, procedura ce implica masurarea lor si eventual corectarea prin folosirea unei

porti NOT. Aceasta procedura de corectie (automata sau nu) este eficienta chiar daca qubitii din cuvantul cod sunt corelati cu alti qubiti din alte cuvinte cod. In acest ultim caz qubitii din cuvantul cod nu au o stare proprie, starea qubitilor din cuvintele cod corelate fiind $a\,|000\rangle\,|\Psi\rangle + b\,|111\rangle\,|\Phi\rangle$, iar corectia erorilor se aplica doar celor trei qubiti din stanga.

FORMA GENERALA A ERORILOR CUANTICE

Deoarece eroarea pe care o suporta un qubit este in general mult mai complexa decat inversarea unui bit, avem nevoie de cuvinte cod cu mai mult de trei qubiti pentru corectarea erorilor unui singur qubit, si avem nevoie de proceduri de diagnostic si corectare mult mai sofisticate. Forma generala a erorii unui qubit poate fi reprezentata, pe langa porti NOT (sau σ_x) neintentionate, de porti σ_z neintentionate sau porti $\sigma_z\sigma_x = i\sigma_y$ neintentionate. Pentru simplitate, portile σ_x, $i\sigma_y$ si σ_z le vom nota in continuare cu *X*, *Y* si respectiv *Z*. Amintim ca

$$I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},\ X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},\ Y = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix},\ Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

In general, un qubit aflat in starea $|x\rangle$, cu $x = 0,1$, trebuie considerat ca parte a unui sistem mai cuprinzator, constand din mediul inconjurator in starea $|e\rangle$ si qubit, starea totala fiind $|e\rangle|x\rangle$. In cazul ideal, daca qubitul evolueaza doar sub actiunea portilor unitare 1-qubit sau interactioneaza controlat cu alti qubiti prin intermediul portilor unitare de tip 2-qubit, acesta ramane necorelat fata de mediul inconjurator. In consecinta, partea ce reprezinta mediul inconjurator din starea totala a sistemului poate fi ignorata in studiul functionarii calculatorului cuantic, asa cum am facut pana acum. Din nefericire, interactia coreleaza starile qubitului si ale mediului inconjurator, si transforma starea initiala in una din cele doua stari posibile

$|e\rangle|0\rangle \rightarrow |e_0\rangle|0\rangle + |e_1\rangle|1\rangle,$
$|e\rangle|1\rangle \rightarrow |e_2\rangle|0\rangle + |e_3\rangle|1\rangle,$

unde $|e_i\rangle$ sunt stari posibile finale ale mediului inconjurator. Aceste stari nu sunt neaparat ortogonale sau normalizate, ele trebuie doar sa satisfaca cerinta ca cele doua stari din partea dreapta sa fie ortogonale deoarece interactia qubit-mediu inconjurator este unitara (duce la o dezvoltare unitara in timp a starii totale). Perturbarea rezultatului calculului datorita corelatiei intre starea qubitului si starea mediului inconjurator este numita decoerenta, si este principalul factor care limiteaza calculul cuantic.

In majoritatea cazurilor interactia cu mediul inconjurator are un efect slab asupra qubitilor, exprimat prin:

$|e_0\rangle \cong |e_3\rangle \cong |e\rangle; \qquad \langle e_1|e_1\rangle, \langle e_2|e_2\rangle << 1$

(produsul scalar/norma unei stari cuantice este 1 doar daca aceasta este izolata!). Modificarea starii sistemului datorita interactiei se poate exprima cu ajutorul operatorilor de proiectie

$P_x = [I_2 + (-1)^x Z]/2,$

care proiecteaza o stare arbitrara pe starile 1-qubit $|x\rangle$, $x = 0,1$, ca

$|e\rangle|x\rangle \to [(|e_0\rangle I_2 + |e_1\rangle X)P_0]|x\rangle + [(|e_2\rangle X + |e_3\rangle I_2)P_1]|x\rangle$.

(I_2 este operatorul identitate 2-dimensional). Aceasta transformare se mai poate pune sub forma

$|e\rangle|x\rangle \to 2^{-1}[(|e_0\rangle + |e_3\rangle)I_2 + (|e_0\rangle - |e_3\rangle)Z + (|e_2\rangle + |e_1\rangle)X + (|e_2\rangle - |e_1\rangle)Y]|x\rangle$.

In general, orice stare cuantica $|e\rangle|\Psi\rangle = |e\rangle(\alpha|0\rangle + \beta|1\rangle)$ se transforma ca

$|e\rangle|\Psi\rangle \to (|a\rangle I_2 + |b\rangle X + |c\rangle Y + |d\rangle Z)|\Psi\rangle$

actiunea operatorilor X, Z si Y fiind descrisa ca eroare de inversare a qubitului, eroare de faza, si, respectiv, eroare combinata de inversare si faza.

Daca $|\Psi\rangle$ este o stare a unui numar mic n de qubiti care formeaza un cuvant cod n-qubit, probabilitatea coruperii cuvantului cod este atat de mica incat putem considera ca doar una dintre erorile posibile actioneaza asupra fiecarui qubit, si deci

$$|e\rangle|\Psi\rangle \to \left(|a\rangle I + \sum_{i=1}^{n}(|b_i\rangle X_i + |c_i\rangle Y_i + |d_i\rangle Z_i)\right)|\Psi\rangle .$$

(n-am mai specificat prin indice dimensiunea operatorului identitate/unitate I !). Procedura de corectare a erorilor trebuie sa inlocuiasca starea corupta

$$\left(|a\rangle I + \sum_{i=1}^{n}(|b_i\rangle X_i + |c_i\rangle Y_i + |d_i\rangle Z_i)\right)|\Psi\rangle$$

cu starea originala $|e\rangle|\Psi\rangle$.

Pentru a corecta starea cuantica, cuplam qubitii logici cu qubiti auxiliari, ale caror stari, in urma interactiei, sa depinda de erorile posibile, dar nu si de starea cuantica care trebuie corectata. Aceasta cerinta poate fi in general indeplinita pentru anumite stari si pentru anumite tipuri de erori. Daca A este operatorul care descrie interactia unitara intre qubitii logici si cei auxiliari, si $|s\rangle$ este starea qubitilor auxiliari in urma interactiei, presupunand ca starea lor initiala este $|0\rangle$, procedura de corectare a erorilor este eficienta daca valoarea s este in corespondenta bijectiva cu tipul de eroare a starii cuantice $|\Psi\rangle$, modelata prin aplicarea operatorului E_s asupra starii neperturbate $|\Psi\rangle$. Daca aceasta cerinta este indeplinita, starea cuantica este corectata prin aplicarea operatorului E_s^{-1} asupra qubitilor logici. Qubitii auxiliari pot fi folositi repetitiv pentru corectarea erorilor daca sunt reactualizati in starea $|0\rangle$. Problema centrala a procedurii de corectare a erorilor cuantice este gasirea formei qubitilor logici $|\overline{0}\rangle$ si $|\overline{1}\rangle$ si a operatorului A care permite identificarea erorilor E_s. Corectarea erorilor presupune proiectia sistemului pe spatiul Hilbert al starii logice necorupte sau pe un spatiu Hilbert ortogonal pe acesta, urmat de corectarea erorii, care poate fi vazuta ca rotatie pe spatiul Hilbert al starii necorupte.

In general, daca A este un operator hermitic n-qubit arbitrar unitar, al carui patrat este operatorul unitate, $A^2 = I$, rezulta ca $A^+ = A$, si ca valorile proprii ale lui A pot fi doar +1 sau –1 deoarece, daca $A|\Psi\rangle = a|\Psi\rangle$, $A^2|\Psi\rangle = |\Psi\rangle = a^2|\Psi\rangle$, adica $a^2 = 1$. Operatorii de proiectie pe subspatiul starilor cu valori proprii +1 sau –1 sunt, respectiv,

$P_{A+} = (I + A)/2, \quad P_{A-} = (I - A)/2$

si, deoarece $P_{A+} + P_{A-} = 1$, orice stare $|\Psi\rangle$ se poate exprima ca o superpozitie a proiectiilor pe aceste doua subspatii:

$|\Psi\rangle = P_{A+}|\Psi\rangle + P_{A-}|\Psi\rangle$.

Pe langa cei n qubiti logici asupra carora actioneaza A, avem qubiti auxiliari si deci avem nevoie de operatorul control-A, CA, care actioneaza asupra celor n qubiti daca starea qubitilor auxiliari este $|1\rangle$, actiunea fiind identitate daca starea qubitilor auxiliari este $|0\rangle$. Daca starea qubitilor auxiliari este o superpozitie de $|0\rangle$ si $|1\rangle$, CA actioneaza liniar.

Daca initial qubitul auxiliar este in starea $|0\rangle$ si se aplica transformarea Hadamard H asupra sa inainte si dupa aplicarea CA asupra celor $n + 1$ qubiti, starea initiala a celor n qubiti logici, $|\Psi\rangle$, devine corelata cu qubitul auxiliar, starea finala totala fiind

$(H \otimes I_2)CA(H \otimes I_2)|0\rangle|\Psi\rangle = 2^{-1/2}\,(H \otimes I_2)CA\,(|0\rangle + |1\rangle)|\Psi\rangle$
$= 2^{-1/2}\,(H \otimes I_2)\,(|0\rangle|\Psi\rangle + |1\rangle A|\Psi\rangle) = 2^{-1}(|0\rangle + |1\rangle)|\Psi\rangle + 2^{-1}(|0\rangle - |1\rangle)A|\Psi\rangle$
$= |0\rangle 2^{-1}(I + A)|\Psi\rangle + |1\rangle 2^{-1}(I - A)|\Psi\rangle = |0\rangle P_{A+}|\Psi\rangle + |1\rangle P_{A-}|\Psi\rangle$

Daca se masoara acum qubitul auxiliar, starea celor n qubiti devine proiectia lui $|\Psi\rangle$ pe subspatiul starilor proprii pozitive (valoare proprie +1) sau negative ale lui A (valoare proprie – 1) in cazul in care masuratoarea indica 0, respectiv 1.

Rezultatul de mai sus sugereaza ca, pentru a corecta o eroare generala a unui cuvant cod n-qubit trebuie sa facem o masuratoare care proiecteaza un cuvant cod posibil corupt pe unul din cele $1 + 3n$ spatii ortogonale 2-dimensionale: un subspatiu 2-dimensional pentru cuvantul cod necorupt $|\Psi\rangle$, si $3n$ subspatii 2-dimensionale pentru fiecare din termenii eronati 1-qubit $X_i|\Psi\rangle$, $Y_i|\Psi\rangle$ si $Z_i|\Psi\rangle$, cu $i = 1,\ldots, n$. Deci, spatiul 2^n dimensional al starilor a n qubiti trebuie sa fie suficient de mare pentru a contine cele $1 + 3n$ subspatii ortogonale 2-dimensionale, adica

$2^{n-1} \geq (3n + 1)$

pentru un cod n-qubit capabil sa corecteze o eroare generala 1-qubit. Cel mai mic n care satisface aceasta conditie este $n = 5$, pentru care se obtine egalitate. Un astfel de cod 5-qubit exista si este cel mai compact cod de corectare a erorilor cuantice, dar este dificil de construit generalizarile portilor 1-qubit si 2-qubit intre cuvintele cod. De aceea s-a inventat un al doilea cod, care necesita 7 qubiti.

CODUL 5-QUBIT DE CORECTARE A ERORILOR

Pentru a distinge cele $1 + 3\times5 = 16$ subspatii 2-dimensionale mutual ortogonale ne trebuie patru operatori hermitici ($2^4 = 16$) care comuta intre ei si al caror patrat este unitate, pentru ca fiecare poate avea independent cate doua valori proprii (±). Acesti operatori (operatorii A de mai sus) sunt

$M_1 = Z_2X_3X_4Z_5$, $M_2 = Z_3X_4X_5Z_1$, $M_3 = Z_4X_5X_1Z_2$, $M_4 = Z_5X_1X_2Z_3$.

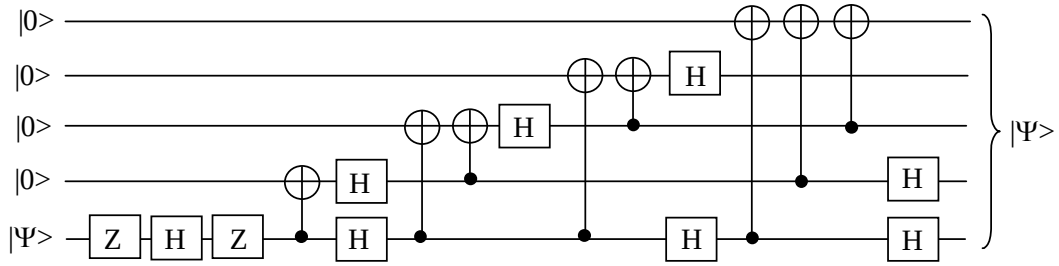
Acesti operatori ridicati la patrat dau operatorul unitate pentru ca sunt produse de operatori al caror patrat este unitate. Operatorii M_i comuta intre ei deoarece X_i si Z_j comuta daca $i \neq j$, si $X_iZ_i = -Z_iX_i$, dar astfel de schimbari de semn se intalnesc intotdeauna de doua ori in calculul

produsului dintre orice doi M_i. (Operatorul $M_5 = Z_1X_2X_3Z_4$, care ar putea fi adaugat la ceilalti patru datorita simetriei, nu este independent de acestia deoarece $M_5 = M_1M_2M_3M_4$.)

Cuvintele cod 5-qubit sunt definite in functie de M_i ca:

$$|\overline{0}\rangle = (1/4)(1+M_1)(1+M_2)(1+M_3)(1+M_4)\,|00000\rangle$$
$$|\overline{1}\rangle = (1/4)(1+M_1)(1+M_2)(1+M_3)(1+M_4)\,|11111\rangle$$

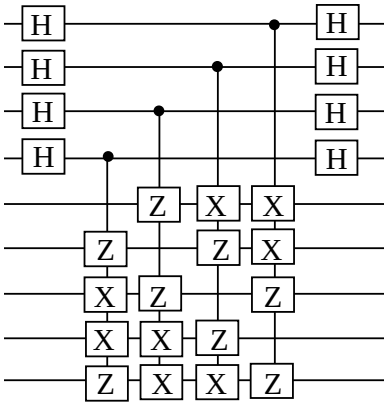
Deoarece fiecare M inverseaza doi qubiti (avand in componenta doua porti NOT), $|\overline{0}\rangle$ este o superpozitie de stari din baza de calcul cu un numar impar de valori 0 (si un numar par de valori 1), pe cand $|\overline{1}\rangle$ este o superpozitie de stari cu un numar impar de valori 1 (si un numar par de 0); in consecinta cele doua cuvinte cod sunt ortogonale. Un circuit care implementeaza $|\Psi\rangle = a\,|\overline{0}\rangle + b\,|\overline{1}\rangle$ daca la intrare am $|\Psi\rangle = a|0\rangle + b|1\rangle$ este ilustrat in figura de mai jos.



Deoarece $(M_i)^2 = 1$, rezulta ca $(1 + M_i)^2 = 2(1 + M_i)$ si

$$\langle\overline{0}\,|\,\overline{0}\rangle = \langle 00000\,|\,(1+M_1)(1+M_2)(1+M_3)(1+M_4)\,|\,00000\rangle$$
$$\langle\overline{1}\,|\,\overline{1}\rangle = \langle 11111\,|\,(1+M_1)(1+M_2)(1+M_3)(1+M_4)\,|\,11111\rangle$$

Daca dezvolt produsul dintre $1 + M_i$ in 16 termeni, obtin termenul 1, care contribuie cu 1 la ambele produse scalare, si alti 15 termeni, care contribuie cu 0 la produsele scalare pentru ca inverseaza un numar par de qubiti (doi sau patru pentru ca pot reduce cei 15 termeni fie la un M_i fie la produse de doi M_i).



Deoarece toti M_i comuta si $M_i(1 + M_i) = (1 + M_i)$, starile $|\overline{0}\rangle$, $|\overline{1}\rangle$ si superpozitiile lor $|\Psi\rangle = a\,|\overline{0}\rangle + b\,|\overline{1}\rangle$ sunt stari proprii ale fiecarui M_i cu valoare proprie 1. Cele 15 posibile stari

corupte ale $|\Psi\rangle = a|\overline{0}\rangle + b|\overline{1}\rangle$ sunt de asemenea stari proprii, care se disting prin cele 15 = $2^4 - 1$ alte seturi posibile de valori proprii ± 1 pe care le pot avea cei patru M_i. In fiecare caz de eroare starea originala poate fi corectata prin aplicarea unei transformai unitare corespunzatoare, X_i, Y_i sau Z_i, asupra qubitului corupt. Un circuit care masoara cei patru M_i este prezentat in figura de mai sus (observati portile Hadamard inainte si dupa aplicarea operatorilor $A = M_i$). Cei cinci qubiti logici sunt reprezentati prin cele cinci linii de jos, cele patru linii de sus fiind qubiti auxiliari, care masoara operatorii $M_1 = Z_2X_3X_4Z_5$, $M_2 = Z_3X_4X_5Z_1$, $M_3 = Z_4X_5X_1Z_2$, si respectiv $M_4 = Z_5X_1X_2Z_3$.

Asa cum am afirmat si mai sus, cele 16 seturi distincte de valori proprii pentru cei patru operatori care comuta M_i descompun spatiul 32-dimensional al celor 5 qubiti in 16 subspatii 2-dimensionale mutual ortogonale, corespunzatoare superpozitiei de $|\overline{0}\rangle$ si $|\overline{1}\rangle$ si fiecareia dintre cele 15 perechi de erori 1-qubit. Degenerarea celor patru M_i in fiecare din aceste 16 subspatii este ridicata de operatorul $N = Z_1Z_2Z_3Z_4Z_5$, care comuta cu toti M_i. Deoarece $|00000\rangle$ si $|11111\rangle$ sunt stari proprii ale N cu valori proprii +1 si –1, si deoarece N comuta cu Z_i si anticomuta cu X_i si Y_i, rezulta ca

$N|\overline{0}\rangle = |\overline{0}\rangle$, $N|\overline{1}\rangle = -|\overline{1}\rangle$, $NZ_i|\overline{0}\rangle = Z_i|\overline{0}\rangle$, $NZ_i|\overline{1}\rangle = -Z_i|\overline{1}\rangle$,
$NX_i|\overline{0}\rangle = -X_i|\overline{0}\rangle$, $NX_i|\overline{1}\rangle = X_i|\overline{1}\rangle$, $NY_i|\overline{0}\rangle = -Y_i|\overline{0}\rangle$, $NY_i|\overline{1}\rangle = Y_i|\overline{1}\rangle$.

In consecinta, daca masuram cei patru M_i plus N, proiectam practic cei cinci qubitii intr-una din cele 32 de stari

$|\overline{0}\rangle$, $X_i|\overline{0}\rangle$, $Y_i|\overline{0}\rangle$, $Z_i|\overline{0}\rangle$, $|\overline{1}\rangle$, $X_i|\overline{1}\rangle$, $Y_i|\overline{1}\rangle$, $Z_i|\overline{1}\rangle$

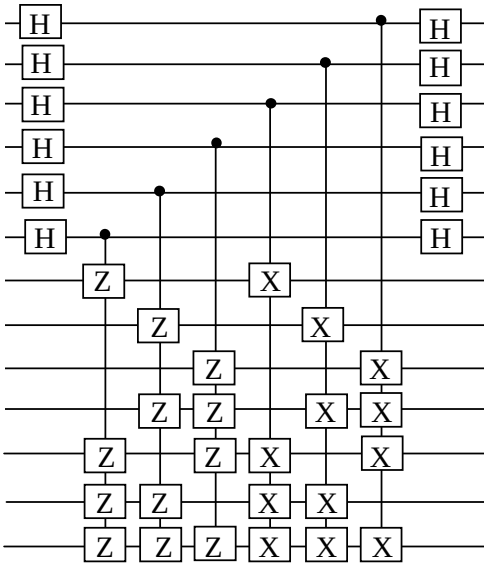
si din masuratori gasim in ce stare se afla. Exact ca si in procedura de corectare a erorilor 3-qubit, daca starea nu este $|\overline{0}\rangle$ sau $|\overline{1}\rangle$, o putem corecta aplicand operatorii corespunzatori X_i, Y_i sau Z_i. Daca vrem sa initializam cei 5 qubiti in starea $|\overline{0}\rangle$, putem aplica transformrea $X_1X_2X_3X_4X_5$ in cazul in care masuratoarea indica starea corectata ca fiind $|\overline{1}\rangle$.

CODUL 7-QUBIT DE CORECTARE A ERORILOR

Codul 5-qubit este teoretic ideal, dar sufera de problema ca circuitele care implementeaza operatiile logice elementare asupra cuvantului cod de 5 qubiti sunt complicate. De aceea in prezent se foloseste un cod 7-qubit, propus de Steane in 1996, care permite implementarea simpla a operatiilor NOT sau CNOT asupra cuvintelor cod. Codul Steane foloseste sase operatori care comuta intre ei si al caror patrat este operatorul identitate pentru a diagnostica eroarea:

$M_1 = X_1X_5X_6X_7$, $M_2 = X_2X_4X_6X_7$, $M_3 = X_3X_4X_5X_7$,
$N_1 = Z_1Z_5Z_6Z_7$, $N_2 = Z_2Z_4Z_6Z_7$, $N_3 = Z_3Z_4Z_5Z_7$.

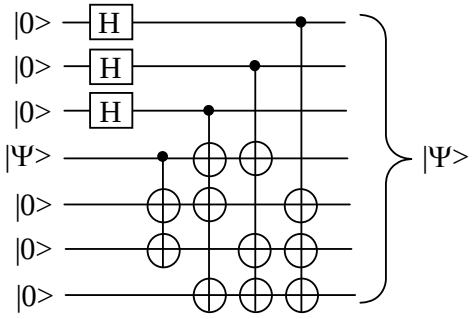
Un circuit care masoara acesti sase operatori este ilustrat in figura de mai jos (este valabila aceeasi conventie cu privire la liniile care reprezinta cei sapte qubiti si qubitii aditionali ca si in celelalte figuri).



Cuvintele cod 7-qubit sunt definite ca

$$|\overline{0}\rangle = 2^{-3/2}(1+M_1)(1+M_2)(1+M_3)\,|0\rangle_7$$
$$|\overline{1}\rangle = 2^{-3/2}(1+M_1)(1+M_2)(1+M_3)\,\overline{X}\,|0\rangle_7$$

unde $\overline{X} = X_1X_2X_3X_4X_5X_6X_7$, astfel incat $|1111111\rangle = \overline{X}\,|0000000\rangle$.



Cele doua stari normalizate la unitate, $|\overline{0}\rangle$ si $|\overline{1}\rangle$, sunt din nou ortogonale deoarece fiecare M inverseaza patru qubiti si $\overline{X}$ ii inverseaza pe toti sapte, astfel incat $|\overline{0}\rangle$ este o superpozitie de stari 7-qubit cu un numar impar de valori 0, si $|\overline{1}\rangle$ este o superpozitie cu un numar par de valori 0. Un circuit care implementeaza $|\Psi\rangle = a\,|\overline{0}\rangle + b\,|\overline{1}\rangle$ daca la intrare am $|\Psi\rangle = a|0\rangle + b|1\rangle$ este ilustrat in figura de mai sus (acest circuit este mult mai simplu decat circuitul analog pentru codul 5-qubit).

Deoarece $\overline{X}$ comuta cu toti M_i, o superpozitie generala a doua cuvinte cod se poate scrie ca $|\Psi\rangle = a\,|\overline{0}\rangle + b\,|\overline{1}\rangle = (aI + b\overline{X})\,|\overline{0}\rangle$ si o stare corupta are forma generala

$$|e\rangle\,|\Psi\rangle \rightarrow \left(|a\rangle I + \sum_{i=1}^{7}(|b_i\rangle X_i + |c_i\rangle Y_i + |d_i\rangle Z_i)\right)|\Psi\rangle\,.$$

Deoarece toti M_i comuta si $M_i(1 + M_i) = 1 + M_i$, si deoarece N_j comuta cu M_i si cu $\overline{X}$ si au $|0000000\rangle$ ca stare proprie cu valoare proprie 1, rezulta ca $|\overline{0}\rangle$, $|\overline{1}\rangle$ si $|\Psi\rangle$ sunt stari proprii ale fiecarui M_i si N_i cu valoarea proprie 1. Cele 21 stari posibile corupte ale $|\Psi\rangle$ sunt de asemenea stari proprii, care pot fi distinse prin seturile posibile de valori proprii ±1 pe care le pot avea cei trei M_i si cei trei N_i. Motivul este ca, ca si in cazul 5-qubit, fiecare X_i, Y_i si Z_i comuta sau anticomuta cu fiecare din M_i si N_i, fiecare stare corupta fiind o stare proprie a fiecarui M_i si N_i cu valori proprii +1 sau –1.

Rezultale masuratorilor asupra M_i si N_i determina un singur termen din cei 22 posibili (functia originala + functiile de unda corupte). Mai precis, o eroare de tip X_i se caracterizeaza prin toate trei masuratori M_i cu rezultat +1, valorile –1 din masuratorile asupra lui N_i determinand care din cele 7 posibile X_i reprezinta eroarea (daca toate masuratorile asupra lui N_i dau +1, nu avem eroare). Analog, erorile de tip Z_i corespund la toti trei N_i cu valori +1, valorile –1 ale M_i determinand care din cele 7 posibile Z_i caracterizeaza eroarea. In final, erorile de tip Y_i se disting prin aceea ca cel putin cateva din masuratorile M_i si N_i dau –1, cele sapte posibile valori ale lui Y_i fiind caracterizate de diferite seturi de astfel de masuratori. In concluzie, cele sase masuratori proiecteaza starea corupta intr-una singura din cele 22 posibilitati, si determina care este aceasta; starea corupta poate fi apoi corectata prin aplicarea corespunzatoare a unuia din cei 22 operatori $I, X_1,\ldots,Z_7$. Cuvintele cod 7-qubit si cele 21 coruptii 1-qubit formeaza doar 44 stari mutual ortogonale, in timp ce spatiul celor 7 qubiti are dimensiunea $2^7 = 128$. Degenerarea operatorilor in spatiile ortogonale se ridica similar ca la codul 5-qubit.

Proprietatea codului 7-qubit, care il face preferabil celui 5-qubit, desi am nevoie de un numar mai mare de qubiti, este ca multe dintre portile fundamentale 1-qubit sunt extinse intr-un mod trivial la portile 7-qubit ce actioneaza asupra cuvintelor cod. Deoarece, de exemplu, $\overline{X}$ comuta cu toti M_i si inverseaza toti cei sapte qubiti, implementeaza operatia logica NOT pe cuvintele cod:

$$\overline{X}\,|\overline{0}\rangle = |\overline{1}\rangle, \quad \overline{X}\,|\overline{1}\rangle = |\overline{0}\rangle.$$

De asemenea, $\overline{Z} = Z_1Z_2Z_3Z_4Z_5Z_6Z_7$ comuta cu M_i, anticomuta cu $\overline{X}$, si lasa invariant $|0\rangle_7$, deci implementeaza operatia logica $Z = \sigma_z$ asupra cuvantului cod:

$$\overline{Z}\,|\overline{0}\rangle = |\overline{0}\rangle, \quad \overline{Z}\,|\overline{1}\rangle = -|\overline{1}\rangle.$$

Aceste relatii sunt de asemenea valabile si pentru codul 5-qubit. Dar, in plus, transformata Hadamard pentru codul 7-qubit: $\overline{H} = H_1H_2H_3H_4H_5H_6H_7$ implementeaza de asemenea poarta logica Hadamard asupra cuvintelor cod:

$$\overline{H}\,|\overline{0}\rangle = (|\overline{0}\rangle + |\overline{1}\rangle)/\sqrt{2}, \quad \overline{H}\,|\overline{1}\rangle = (|\overline{0}\rangle - |\overline{1}\rangle)/\sqrt{2},$$

proprietate care nu este valabila pentru codul 5-qubiti. Similar, este usor sa se implementeze poarta logica 14-qubit CNOT, care transforma perechea de cuvinte cod $|\bar{x}\rangle|\bar{y}\rangle$ in $|\bar{x}\rangle|\overline{x \oplus y}\rangle$ prin aplicarea portilor obisnuite CNOT asupra fiecareia dintre cele 7 perechi de qubiti corespunzatori din covintele cod.

Datorita simplitatii acestor porti, se poate folosi corectarea erorilor pentru a elimina functionarea necorespunzatoare a insasi portilor elementare (de tip 1-qubit si CNOT), daca rata de functionare necorespunzatoare este atat de mica incat doar una singura dintre cele sapte porti elementare nu functioneaza normal. Un alt avantaj al portilor aplicate cuvintelor cod este ca, daca functioneaza corect, nu pot converti erori 1-qubit in erori multi-qubit, cum s-ar putea

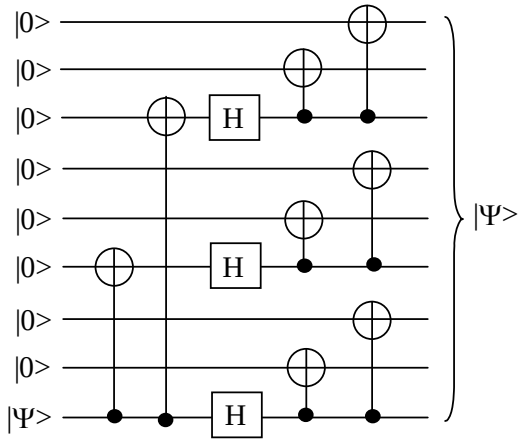
intampla daca utilizam porti mai complexe asupra cuvintelor cod. Aceasta proprietate se numeste toleranta la eroare, si codul 7-qubit are marele avantaj ca majoritatea portilor logice importante pot fi implementate intr-o maniera toleranta la erori.

CODUL 9-QUBIT DE CORECTARE A ERORILOR

Acest cod, propus de Shor in 1995, este primul cod de corectare a erorilor din punct de vedere cronologic. Actualmente prezinta insa doar interes istoric. Cele doua cuvinte cod 9-qubit ortogonale sunt

$$|\bar{0}\rangle = 2^{-3/2}(|000\rangle+|111\rangle)(|000\rangle+|111\rangle)(|000\rangle+|111\rangle)$$
$$|\bar{1}\rangle = 2^{-3/2}(|000\rangle-|111\rangle)(|000\rangle-|111\rangle)(|000\rangle-|111\rangle)$$

codul putand fi vazut ca o extensie a codului simplu 3-qubit care poate corecta erori de faza 1-qubit si erori de inversare 1-qubit. Majoritatea valorilor din fiecare set de trei qubiti corecteaza impotriva inversarii bitilor, iar majoritatea celor trei semne corecteaza erorile de inversare de semn. Cele doua stari din baza de calcul, $|\bar{0}\rangle$ si $|\bar{1}\rangle$, pot fi distinse prin observabilele 3-qubit $X_1X_2X_3$, ale caror stari proprii sunt, cu valori proprii +1 si –1, dar nu putem distinge $|\bar{0}\rangle$ si $|\bar{1}\rangle$ observand unul sau doi din qubitii din blocul de 9 qubiti. Deci qubitii logici (ca si in cazul celorlalte coduri de corectare a erorilor) sunt codati nelocal, prin corelatiile dintre qubitii din bloc. Aceasta proprietate nelocala a informatiei codate ofera protectie impotriva zgomotului, daca zgomotul este local (in sensul ca actioneaza independent pe fiecare qubit din bloc).



Un circuit care sa codeze cuvintele cod este prezentat mai sus (observati similaritatea cu circuitul corespunzator de la codul 3-qubit). Pentru un cuvant cod $|\Psi\rangle$ care este o superpozitie a celor doua stari $|\bar{0}\rangle$ si $|\bar{1}\rangle$ de mai sus avem

$$Z_1|\Psi\rangle = Z_2|\Psi\rangle = Z_3|\Psi\rangle,\ Z_4|\Psi\rangle = Z_5|\Psi\rangle = Z_6|\Psi\rangle,\ Z_7|\Psi\rangle = Z_8|\Psi\rangle = Z_9|\Psi\rangle$$

forma cea mai generala a starilor corupte 1-qubit a lui $|\Psi\rangle$ continand doar 22 termeni independenti (in loc de $28 = 3\times9 + 1$):

$$|e\rangle|\Psi\rangle \rightarrow \left(|a\rangle I + |b_1\rangle Z_1 + |b_2\rangle Z_4 + |b_3\rangle Z_7 + \sum_{i=1}^{9} (|c_i\rangle X_i + |d_i\rangle Y_i) \right) |\Psi\rangle$$

Diagnosticarea erorilor se face cu 8 operatori hermitici care comuta intre ei si a caror patrat este unitatea:

Z_1Z_2, Z_2Z_3, Z_4Z_5, Z_5Z_6, Z_7Z_8, Z_8Z_9,
$X_1X_2X_3X_4X_5X_6$, $X_4X_5X_6X_7X_8X_9$.

Din definitia starilor $|\bar{0}\rangle$ si $|\bar{1}\rangle$ rezulta ca orice superpozitie a lor, $|\Psi\rangle$, este invarianta in raport cu toti acesti operatori. Fiecare dintre cei 21 termeni corupti este de asemenea o stare proprie a celor 8 operatori cu valori proprii +1 sau –1, si fiecaruia ii corespunde un numar diferit de valori negative a celor opt operatori:

1) cele trei erori Z_1, Z_4 si Z_7 se disting de X_i si Y_i prin aceea ca ele comuta cu fiecare dintre cei sase operatori Z din lista de mai sus. Acesti trei Z_i se disting intre ei deoarece Z_1 anticomuta cu unul dintre cei doi operatori X, Z_7 anticomuta cu celalalt si Z_4 anticomuta cu ambii
2) toate cele 9 erori X_i se disting de Z_i si Y_i prin aceea ca ele comuta cu toti operatorii X. Se disting unele de altele pentru ca X_1, X_3, X_4, X_6, X_7 si X_9 anticomuta fiecare cu unul singur dintre cei 6 operatori Z din lista de mai sus (respectiv, cu Z_1Z_2, Z_2Z_3, Z_4Z_5, Z_5Z_6, Z_7Z_8, Z_8Z_9), in timp ce X_2, X_5 si X_8 anticomuta fiecare cu doi operatori Z diferiti (respectiv, cu Z_1Z_2 si Z_2Z_3, Z_4Z_5 si Z_5Z_6, si Z_7Z_8 si Z_8Z_9)
3) cele 9 erori Y_i au acelasi mod de comutare cu operatorii Z din lista de mai sus cum il au si X_i, si se pot distinge unele de altele in acelasi fel. Se pot distinge de operatorii X_i prin necomutarea cu cel putin unul dintre cei doi operatori X din lista de mai sus.

Deci, ca si in celelalte metode, masurarea simultana a celor 8 operatori din lista de mai sus care comuta intre ei proiecteaza starea corupta intr-una singura si o identifica. Dupa aceea se aplica un circuit care implementeaza transformarea unitara inversa pentru a corecta starea. Circuitul care identifica erorile din codul 9-qubit este figurat mai jos.