UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA
Facoltà di Ingegneria “Enzo Ferrari”
Corso di Laurea in Ingegneria Informatica

Tesi di Laurea

QUANTUM COMPUTER: THE EVOLUTION OF ENGINEERING

Relatore: Chiar.ma Prof.sa
Sonia Bergamaschi

Laureando: Alex Gugliotta

ANNO ACCADEMICO 2015/2016

“What we usually consider as impossible are simply engineering problems… there’s no law of physics preventing them.”

(Michio Kaku)

Indice

1. Introduzione

> "La computazione quantistica è […] niente di meno che un nuovo modo distinto per sfruttare la natura […] Sarà la prima tecnologia che permetterà ad universi paralleli di collaborare nello svolgere compiti utili, per poi condividere i risultati." (David Deutsch [CITATION Dav97 \l 1040])

Fin da quando l'umanità ha cominciato ad interrogarsi sull'essenza di ciò che la circonda, ponendosi quesiti sui fenomeni naturali e sul proprio ruolo nel cosmo, ha avuto bisogno di strumenti in grado di supportarla nel raggiungimento di tale obiettivo. L'avanzamento scientifico e quello tecnologico

procedono, da sempre, in parallelo, in un processo che si ripete ciclicamente: il primo porta alla luce nuove possibilità mentre il secondo le sfrutta permettendo a sua volta una miglioria dell'indagine scientifica. D'altra parte è possibile affermare che il più grande strumento a disposizione dell'uomo è il cervello. Non c'è nulla che possa essere paragonato allo strabiliante organo che pare essere il fulcro dell'universo stesso. In quanto componente umana, però, il cervello ha un difetto: è suscettibile del trascorrere del tempo. L'uomo ha creato il cannocchiale, il microscopio, i radar, i satelliti e migliaia di altri mezzi per ampliare i propri sensi e poter così scrutare in modo più accurato i segreti del cosmo, ma tutto ciò non sarebbe stato sufficiente senza strumenti per migliorare la capacità di elaborazione dei dati raccolti. Una volta osservati, i dati, devono essere elaborati attraverso calcoli e processi che portino ad una conclusione ragionevole e coerente con gli esperimenti effettuati e questo è possibile per un umano (o per un gruppo di umani) solo se queste informazioni restano al di sotto di una certa soglia di complessità o al di sotto di una certa quantità. Per tale motivo sono stati sviluppati strumenti di calcolo che supportassero la specie homo nella propria caccia alle risposte. Partendo dagli astrolabi e dai congegni per la stima delle orbite dei pianeti e delle posizioni degli astri e passando per le prime calcolatrici e le prime macchine programmabili, si è infine arrivati alla costruzione del computer nell'accezione più recente del termine. Mentre il cervello si manifesta come lo strumento più potente creato dalla natura, il computer risulta essere lo strumento più potente creato dall'uomo. La scienza ha portato il progresso che ha permesso di sviluppare il calcolatore che oggi è diffuso in ogni angolo del pianeta, ma, esso, non è più sufficiente per indagare a fondo i fenomeni della realtà che ci circonda. Il silicio, che rappresenta la materia di base di cui è costituita una macchina di calcolo, è una tecnologia destinata a raggiungere un limite. La direzione attuale che questa tecnologia segue è legata alla miniaturizzazione dei suoi componenti fondamentali, i transistor. Questi, però, possono essere rimpiccioliti fino a che non ci si imbatte in una serie di fenomeni, definiti dalla teoria della meccanica quantistica, che ne impediscono il funzionamento corretto al di sotto di certe dimensioni. Secondo la Legge di Moore, nel 2020, il limite del silicio sarà raggiunto. Una prima soluzione risiede nella scalabilità dei calcolatori, ovvero nel fatto che è possibile ingrandire un sistema di calcolo o in alternativa farlo lavorare in parallelo con altri, creando quelli che vengono rispettivamente denominati *cluster* e *supercomputer*. Il problema risiede dunque nell'efficienza e nell'impossibilità dei computer classici di effettuare alcuni tipi di operazioni. Per ovviare a ciò si è quindi optato per lo studio di altre tecnologie e per il conseguente sviluppo di computer che le sfruttassero. Tra le varie possibilità esplorate, quella che ha ottenuto il maggiore successo è quella legata alla teoria dell'informazione

quantistica: si arriva così a sfruttare quella stessa proprietà che rappresenta un limite per il calcolo classico. Questa teoria si è diffusa già durante gli anni Sessanta ma solo durante gli ultimi decenni è stato possibile renderla attuabile. Il suo scopo è elaborare e trasmettere le informazioni utilizzando alcuni principi della fisica quantistica quali la sovrapposizione di stati, il parallelismo quantistico, *l'entanglement* e il principio di non-località. La nuova unità base del computer quantistico è il *qubit* che, basandosi su tali principi, permette lo svolgimento di svariati calcoli in parallelo, la possibilità di simulare la fisica atomica e subatomica dell'universo e le relazioni statistiche tra gli avvenimenti. Attualmente i computer quantistici esistono e si contendono il titolo di calcolatori più "veloci" con i supercomputer.

Dominare i *qubit* non è un'impresa semplice, ma esistono le prove matematiche che riuscendo nell'intento si raggiungerebbero livelli di elaborazione dell'informazione fino ad ora solo immaginati. Molti campi verrebbero rivoluzionati dall'introduzione stabile delle macchine quantistiche, in particolare lo studio di grandi quantità di dati e di database, il "machine learning", la crittografia, le simulazioni scientifiche, le previsioni meteo e molti altri. Per il momento queste macchine sono ancora oggetto di studio, poiché si sta ancora cercando di apprendere le migliori tecniche per la realizzazione dei gate quantistici: i circuiti base del calcolo quantistico, gemelli dei gate classici presenti nei computer attuali. Inoltre i computer quantistici devono affrontare il problema della decoerenza ovvero il fenomeno che li priva del loro stato di isolamento e interferisce con le funzioni d'onda dei *qubit* andando a corrompere l'integrità del calcolo.

Anche se alcuni computer di questo tipo esistono già, viene ad essi imputata l'accusa di non essere puramente quantistici, ma ciò rappresenta la norma nell'ambito ingegneristico che sfrutta le migliori tecnologie a disposizione per creare degli ibridi in gradi di far cooperare al meglio le differenze se ciò può portare ad un risultato finale superiore. Dato che la realizzazione di una macchina che utilizzi tutti e soli i principi della meccanica quantistica risulta ancora lontana molte delle ricerche attuali stanno ricercando metodi per far coesistere le due tecnologie (classica e quantistica) in un solo sistema di calcolo. Sono moltissimi i linguaggi di programmazione che permettono di amalgamare algoritmi classici con algoritmi quantistici ma per ora nessuno sembra prevalere sugli altri. Questo però aprirà la strada ai programmatori del futuro che scriveranno i codici di queste macchine permettendo alla tecnologia di fare passi da gigante.

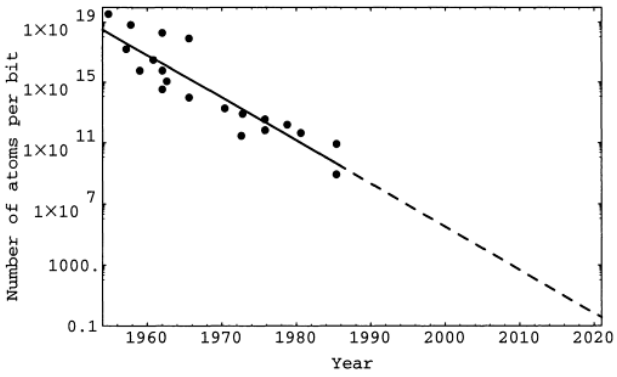
L'idea del computer quantistico è affascinante perché apre letteralmente infinite possibilità e perché la tecnologia per realizzarne dei prototipi che siano al di sopra delle possibilità attuali non sembra lontana.

Basta pensare ad una scienza come quella del “machine learning”, che già di per sé risulta così potente da essere pericolosa: se sfruttata al meglio, cosa che un calcolatore quantico potrebbe fare, permetterebbe all’uomo di vedere l’universo da un punto di vista nuovo, di esplorare i confini di ciò che ancora è distante dal comprendere, di avere un punto di vista differente sul cosmo, non incentrato sulla sua stessa osservazione e di dare risposte a domande che non pensava nemmeno di potersi porre.

2.Oltre il silicio, quando la legge di Moore arriva al termine

Nel 1965 Gordon Moore[1] osservando i dati raccolti negli anni antecedenti propose una legge secondo la quale la complessità dei microcircuiti (misurata ad esempio come numero di transistor per unità di calcolo) raddoppia periodicamente, con un periodo previsto di 12 mesi. Nel corso degli anni successivi tale legge fu corretta adattandola a quello che si dimostrò essere il trend effettivo di tale avanzamento tecnologico. All'inizio degli anni Ottanta il periodo caratteristico della legge si assestò a 18 mesi e tale rimase[2]. Effettuando un'analisi ponderata della legge è facile intuire che tale andamento arriverà prima o poi ad un limite fisico. Tale limite è rappresentato dalla correlazione tra atomo e bit di informazione: si stima infatti che all'incirca nel 2020 i transistor raggiungeranno dimensioni atomiche come si evince dal grafico 1.1. Il problema risiede inoltre nel fatto che a livello atomico le leggi della fisica classica non descrivono più il comportamento delle particelle utilizzate, ma diventa necessario ricorrere all'applicazione delle leggi della fisica quantistica.[CITATION Wil00 \l 1040]

Figura 1.1 *Andamento legge di Moore rispetto al numero di atomi per bit*



Fonte: [CITATION Wil00 \l 1040]

Alla luce di ciò i computer si sono evoluti in un'altra direzione, quella del lavoro in parallelo e/o in serie. Attualmente la maggior parte dei processori possiede difatti più di un core che lavora in parallelo con altri, inoltre, la richiesta sempre superiore di risorse da parte delle nuove branche dell'informatica ha portato ad un ulteriore sviluppo tecnologico di questo tipo: nei moderni pc sono invero presenti, oltre a processori *multi-core*, le GPU che lavorano in "parallelo" alla CPU del computer per risolvere problemi e calcoli specifici. Ad un livello di astrazione più alto troviamo anche i *supercomputer*, utilizzati in campi che richiedono analisi di grandi quantità di dati e lo svolgimento di grandi quantità di calcoli nel minor tempo possibile, dove vengono fatte lavorare in parallelo diverse CPU: tali

1 Gordon Moore, en.wikipedia.org, 17/07/09
2 *Legge di Moore*, treccani.it, 26/07/2016

computer rappresentano attualmente i *device* con la maggior potenza di calcolo esistenti. Questa soluzione, come si può intuire, è attuabile fino a quando non si incorre in problemi di scalabilità, dissipazione termica, consumi energetici etc.[CITATION Wil98 \l 1040] Alla luce di tutto ciò nasce la necessità di ricercare nuove metodologie di calcolo, nuovi sistemi per aggirare l'ostacolo, nuove tecnologie da implementare che consentano un aumento significativo della potenza di calcolo a parità di risorse o addirittura con un ampio risparmio delle stesse. Nel proprio libro del 2008, *Fisica dell'impossibile[CITATION Kak08 \l 1040]*, Michio Kaku[3] scrive:

> "I fisici sono già al lavoro sulla tecnologia del dopo-silicio, che dominerà il mondo dei computer dal 2020 in poi. I risultati ci sono, ma non sono tutti positivi. […] si stanno esaminando diverse tecnologie candidate a prendere il posto di quella del silicio: computer quantistici, a DNA, ottici, atomici e così via." (Kaku, 2008)

Tra le varie tecnologie proposte nel corso degli ultimi anni, quella in cui sono stati fatti i maggiori progressi è quella dei computer quantistici, che in un certo senso rappresentano i diretti discendenti dei computer classici. Negli ultimi 20 anni si sono susseguite molte scoperte riguardo alla computazione quantistica e ad oggi sono presenti persino i primi computer, frutto di molti tentativi, esperimenti e tecnologie innovative. Questo contribuisce ad allontanare i computer quantistici dal mondo ideale avvicinandoli invece a quello reale.

2.1. Breve storia della quantistica: dalle origini al computer quantico

Il termine "computazione quantistica" viene utilizzato per la prima volta nel 1980, anche se già nella decade precedente alcuni studiosi stavano lavorando a ricerche relative all'uso della teoria quantistica legato al campo dell'informazione. Quelli che vengono considerati i fondatori di tale branca di studi sono Paul Benioff[4] e Yuri Manin[5], i quali descrissero l'implementazione teorica di una macchina di Turing con l'utilizzo della meccanica quantistica[CITATION Ben80 \l 1040] e coniarono il termine *quantum computing[CITATION Man80 \l 1040]*. A distanza di due anni dalle loro pubblicazioni, Richard Feynman[6] con i propri studi in materia dimostrò che una normale macchina di Turing non può simulare la meccanica quantistica senza incorrere in un rallentamento esponenziale[CITATION Fey81 \l 1040] e dopo di lui, nel 1985 David Deutsch[7] propose la prima macchina di Turing quantistica correlata al metodo del parallelismo quantico[CITATION Deu85 \l 1040] provando così, seppure in via teorica, l'efficienza di una macchina di questo genere. Le vere potenzialità di questa nuova scienza

3 Michio Kaku, en.wikipedia.org, 17/07/09
4 http://www.phy.anl.gov/theory/staff/pab.html, 17/09/2016
5 https://en.wikipedia.org/wiki/Yuri_Manin, 17/09/2016
6 https://en.wikipedia.org/wiki/Richard_Feynman, 17/09/2016
7 https://en.wikipedia.org/wiki/David_Deutsch, 17/09/2016

furono messe in risalto però dagli studi successivi effettuati nel corso dei primi anni Novanta: un pioniere di tali ricerche fu Richard Jozsa[8] che nel 1991 dopo aver descritto quali sono le funzioni che non possono essere risolte dal parallelismo quantico[CITATION Joz91 \l 1040], collaborò con Deutsch nel proporre il primo problema che una macchina quantistica risolveva in tempi esponenzialmente più rapidi rispetto ad una deterministica. Nel corso dei cinque anni successivi si susseguirono diverse dimostrazioni teoriche della superiorità, nella risoluzione di vari problemi, di una macchina quantistica rispetto ad una deterministica o probabilistica. In particolare tra il 1994 ed il 1996 vennero scoperti rispettivamente da Peter Shor[9] e da Lov Grover[10] gli algoritmi eponimi, i quali dimostravano la possibilità di un ipotetico computer quantistico di fattorizzare grandi numeri a velocità polinomiali e la possibilità di trovare un *record* all'interno di un database in tempi più brevi rispetto agli algoritmi adottati nei computer classici.[CITATION Wil98 \l 1040]

Tra la fine degli anni Novanta e i primi anni Duemila vennero condotti diversi esperimenti che diedero alla luce i primi computer quantistici, basati su un numero esiguo di *qubit* (da 2 a 7) in grado di svolgere gli algoritmi di Deutsch e Shor, nelle università di Stanford, Monaco e nel Laboratorio Nazionale di Los Alamos.

Da lì in poi si susseguirono moltissime scoperte e furono dimostrate altrettante teorie, indirizzate sia allo sviluppo della macchina quantica, sia ai campi scientifici correlati, che hanno portato alla realizzazione di componenti fisici fino ad ora inesistenti in grado di contribuire alla costruzione del primo computer di questo genere. Parallelamente a queste ricerche vennero sperimentate le prime reti quantiche. Nel 2003 venne resa operativa la rete quantistica DARPA e nel 2005 anche gli scienziati del Max Planck Institute sono riusciti a realizzarne un prototipo funzionante. Nello stesso anno venne annunciato il primo *qubyte*. Finalmente nel 2007 fa la sua apparizione pubblica la D-Wave System che produce quello che viene considerato il primo computer quantistico adiabatico[11] a 16 *qubit*. L'11 maggio 2011 la stessa società annuncia il primo computer quantistico ad essere commercializzato: il D-Wave One basato su 128 *qubit*[12]. Questo computer assieme al suo successore, il D-Wave Two (uno è stato acquistato da Google in collaborazione NASA), hanno creato molto scalpore a livello accademico

8 http://www.damtp.cam.ac.uk/people/r.jozsa, 17/09/2016
9 http://www-math.mit.edu/~shor, 17/09/2016
10https://it.wikipedia.org/wiki/Lov_Grover, 17/09/2016
11 Un computer adiabatico è così definito poiché il suo approccio ricorda quello delle trasformazioni adiabatiche della termodinamica, in cui, un sistema scambia e riacquista calore per gradi e non tutto in un unico istante. Ciò significa che l'approccio di un computer quantistico adiabatico consiste nel trovare il risultato al problema in seguito all'elaborazione su diversi gradi di calcolo.
12 *Timeline of quantum computing*, en.wikipedia.org, 26/07/2016

in quanto è nato un forte scontro di idee tra gli studiosi convinti delle sue potenzialità e coloro che invece cercano di dimostrare che non si tratta di vere macchine quantistiche ma soltanto di macchine classiche che sfruttano effetti quantistici per velocizzare i tempi di calcolo. In particolare, riguardo al DW2, uno studio pubblicato su Science dagli scienziati dell'Eth di Zurigo [CITATION Røn14 \l 1040] ha messo in evidenza come la potenza di calcolo di questa macchina non superi quella di un normale supercomputer[13]; bisogna considerare però che si tratta di una tecnologia ancora in fasce nella quale si prevedono molti sviluppi nel corso dei prossimi anni e che le sue prestazioni dipendono anche dal problema specifico con cui vengono confrontate[14]. Inoltre anche se non si riuscisse a creare un computer puramente quantico, un ibrido in grado di sfruttare le migliori caratteristiche di entrambe le macchine potrebbe essere la soluzione vincente. Il dibattito dunque resta aperto nell'attesa di nuovi test, nuovi studi e nuove scoperte da parte degli scienziati e degli ingegneri a lavoro su questa tecnologia.

13 *Defining and detecting quantum speed up*, science.sciencemag.org, 27/07/2016

14 *Il computer quantistico di Google funziona davvero.* Forse, it.ibtimes.com, 27/07/2016

3.La fisica quantistica nell'informazione

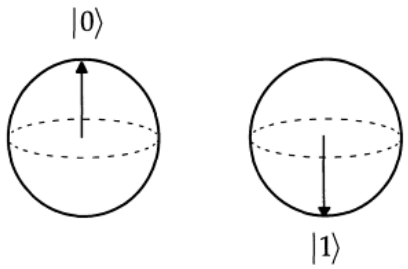
Quando si parla di informatica quantistica è inevitabile nominare i *qubit* ovvero la relativa unità fondamentale di informazione. Che cos'è dunque un *qubit*? Si tratta di un sistema quantistico a due stati definiti rispettivamente con i simboli |0> e |1>: fondamentalmente qualsiasi sistema quantico con almeno due stati può essere utilizzato a tale scopo purché il sistema sia esente dal passare spontaneamente dall'uno all'altro stato[CITATION Bou00 \l 1040]. Questa definizione però non evidenzia la differenza tra i *qubit* e i classici bit, indotta dalle diverse proprietà della fisica che queste nuove unità d'informazione posseggono quali la sovrapposizione, *l'entanglement*, la non-clonabilità etc. Le proprietà che caratterizzano il *qubit* sono le stesse che se utilizzate correttamente portano ad una superiorità di calcolo da parte delle macchine quantistiche rispetto a quelle classiche.

3.1. Sovrapposizione e parallelismo quantistico

Il fenomeno della sovrapposizione è legato direttamente ad uno dei bizzarri effetti della fisica quantistica secondo cui: "se un sistema può essere trovato in uno degli stati di un insieme discreto, esso può anche esistere in una sovrapposizione, o di tutti o di alcuni di quegli stati contemporaneamente" [CITATION Wil00 \l 1040] , quindi il *qubit* può assumere i valori |0> o |1> o un valore dato da $c_0|0> + c_1|1>$ dove i coefficienti c_0 e c_1 sono dei numeri complessi tali che $|c_0|^2$ e $|c_1|^2$ rappresentino rispettivamente le probabilità che il sistema si trovi nello stato |0> o |1> (la somma di tali probabilità è uguale ovviamente ad 1). La rappresentazione più esplicativa del *qubit* è quella che lo descrive come un vettore che ruota all'interno di una sfera di raggio unitario: tale sfera è detta sfera di Bloch. Secondo la rappresentazione comune i due stati fondamentali sono rappresentati dal vettore rivolto verso l'alto o verso il basso lungo l'asse verticale (Figura 3.1) anche in relazione al fatto che in molti casi è lo spin delle particelle fondamentali ad essere utilizzato come unità e tale spin può assumere appunto due soli stati: verso l'alto o verso il basso (in relazione al campo magnetico nel quale la particella è immersa). Lo spin è una grandezza associata alla rotazione delle particelle sul proprio asse ed è stato introdotto matematicamente da W. Pauli[15] per distinguere le particelle che non possono trovarsi in stati energetici affini. Nel caso dell'elettrone, che possiede spin ½ (come tutti i Fermioni), la rotazione può essere rappresentata da un vettore adiacente all'asse dell'elettrone che a sua volta può essere orientato in direzione uguale o opposta a quella del campo magnetico in cui l'elettrone è immerso, da cui si ottengono i due possibili stati logici.

15 https://it.wikipedia.org/wiki/Wolfgang_Pauli

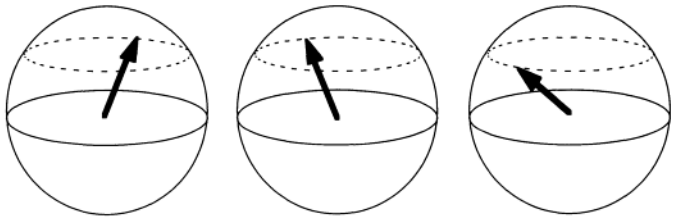
Figura 3.1 *Un qubit negli stati rappresentanti i bit 1 e 0*

(Fonte: [CITATION 

Wil00 \l 1040])

Secondo questa rappresentazione c_0 e c_1 sono i valori delle proiezioni del vettore lungo due degli assi fondamentali e maggiore è la loro lunghezza, maggiore sarà la probabilità che il qubit si trovi nello stato relativo (in un'ottica alternativa, c_0 e c_1 possono essere descritti come il seno ed il coseno del vettore). La necessità di utilizzare una sfera e non una circonferenza deriva dal fatto che è possibile dimostrare come un generico algoritmo vari nel risultato a seconda, ad esempio, che gli sia dato un input di questo tipo $c_0|0> + c_1|1>$ o di questo tipo $c_0|0> - c_1|1>$. Entrambi i qubit in input hanno le stesse probabilità $|c_0|^2$ e $|c_1|^2$, ma devono essere distinti: questo viene reso possibile introducendo una terza variabile ovvero la fase. In Figura 3.2 sono riportati 3 *qubit* equivalenti nelle probabilità ma non nella fase.

Figura 3.2 *Tre qubit con le stesse proporzioni probabilistiche ma fasi differenti*



(Fonte: [CITATION Wil00 \l 1040])

L'altro aspetto fondamentale di questo fenomeno è che ogni qualvolta un *qubit* viene misurato, la sua funzione d'onda collassa in uno dei due stati possibili in relazione alle probabilità precedentemente illustrate, per cui non è possibile osservare un *qubit* in uno stato di sovrapposizione.
Apparentemente la proprietà di sovrapposizione non fornisce risultati utili ai fini informatici.
Ipotizziamo però di utilizzare una coppia di *qubit* come input, il risultato può essere descritto dalla figura seguente.

Figura 3.3 *Rappresentazione del generico stato di due qubit*

(Fonte: [CITATION Wil00 \l 1040])

Quelli che vengono definiti "autostati" dalla fisica quantistica, nel nostro caso, sono rappresentati dalle informazioni 00, 01, 10, 11. Tale risultato è un esempio di quello che viene definito parallelismo quantistico: il calcolo produce contemporaneamente tutti i risultati possibili con diverse probabilità e nel momento della lettura dei valori, i due *qubit* collassano in uno dei quattro autostati. Per fare in modo che tra gli autostati il risultato sia quello desiderato si sfruttano le equazioni di Schrödinger che descrivono l'evoluzione nel tempo del sistema quantistico. Il mezzo con cui vengono programmati i computer è appunto la manipolazione di questa legge attraverso l'uso di "gate quantistici" che permettono di alterare l'andamento evolutivo continuo di tali particelle, facendo sì che il vettore ruoti in ogni qubit fino a rendere il risultato desiderato il più probabile.

3.2. Entanglement

Un'altra caratteristica peculiare della fisica quantistica è l'*entanglement*. Si tratta di uno stato di correlazione tra due o più sistemi quantici tale che, i due sistemi risultino connessi in un rapporto causa-effetto pur non trovandosi a contatto diretto o indiretto. Prendiamo in esame il caso di due particelle elementari A e B, quali gli elettroni, ed utilizziamo lo spin per definire i due stati del sistema (come avviene in alcuni *qubit*). Se le particelle sono in entanglement quantistico si osserverà che effettuando una misura su A collasserà sia la funzione d'onda di A che quella di B; inoltre, ipotizzando di misurarne lo spin, si avrà che nel momento della misura, non appena lo stato di A sarà determinato, si avrà la certezza che B si trovi nello stato opposto o in uno equivalente (a seconda della base) e ciò e valido per ogni altra caratteristica a due autostati delle particelle. La conseguenza diretta di questo fenomeno è il "teletrasporto quantistico" ovvero la trasmissione di uno stato quantistico da un punto ad un altro con velocità istantanea. Il solo *entanglement* però non è sufficiente poiché nel momento della lettura di un qubit, ad esempio, non si ha la certezza di quale sarà l'informazione trasmessa.

"Sebbene non si possa trasmettere informazione attraverso il solo entanglement, l'utilizzo di un canale di comunicazione classico in congiunzione con uno stato *entangled* permette il teletrasporto di uno stato quantistico, che sarebbe altrimenti

impossibile poiché richiederebbe un'infinita quantità di informazione per essere determinato" (it.wikipedia.org, entanglement quantistico, 29/07/2016)

Se da un lato ciò è possibile bisogna tenere conto che il trasferimento dell'informazione con fedeltà assoluta implica la distruzione dell'informazione originale.

L'*entanglement* permette quindi non solo la creazione di reti quantistiche di trasmissione dell'informazione, ma fa sì che gli stessi dati all'interno di un computer quantistico possano essere scambiati in questo modo: questo fenomeno è possibile anche tra moltissime particelle in modi che attualmente sono oggetto di ricerca[16]. Sebbene per molti anni gli scienziati abbiano rifiutato di crederci, questo fenomeno è stato più volte dimostrato, osservato e ricreato in laboratorio[17]. La sua importanza è collegata per di più ad un ulteriore principio quantistico: la non-località. *L'entanglement* non dipende infatti dalla distanza tra le particelle che possono essere lontane migliaia di chilometri l'una dall'altra, rendendo ancora più vasto il numero dei possibili modi in cui l'informatica potrebbe sfruttare questo fenomeno. Tra il 2012 ed il 2013 è stato inoltre dimostrato da alcuni esperimenti[18], che la non-località si estende anche al dominio del tempo.

3.3. Altri principi utili all'informatica

La non-clonabilità è un principio interessantissimo per l'informatica, poiché come esposto in precedenza, non è possibile osservare un *qubit* senza che questo collassi in uno degli autostati, perdendo in questo modo l'informazione trasportata. Ciò è utile ai fini della sicurezza: in questo modo un ipotetico flusso di dati non può essere letto da terzi e quindi si salvaguarda la segretezza dei dati trasferiti.

Per il principio di indeterminazione di Heisenberg non è possibile misurare più caratteristiche osservabili in una particella, poiché, nel momento in cui una di queste caratteristiche assume un valore stabile, le altre non possono essere determinate e ciò influisce oltremodo sulla lettura precisa dei dati. Gli studi però stanno avanzando in modo da poter utilizzare l'*entanglement* multiplo per ovviare al problema.

L'informatica quantistica effettuata con questo tipo di calcoli offre risultati probabilistici e non certi. Un calcolo quantistico collassa nel risultato desiderato con una certa probabilità, quindi, per quanto possa essere resa piccola, nella maggior parte dei casi c'è sempre la possibilità di non ottenere il

16 " *Record 100000 entangled photons detected*", newscientist.com, Jacob Aron, 29/07/2016
17 https://en.wikipedia.org/wiki/Quantum_entanglement
18 https://en.wikipedia.org/wiki/Quantum_entanglement#Other_types_of_experiments

risultato sperato. Come vedremo in seguito però, molti algoritmi quantistici risultano comunque più rapidi di quelli classici.

È necessario poi menzionare la possibilità di un computer quantistico di generare numeri casuali, a differenza di uno classico che genera invece numeri pseudo-casuali basati su algoritmi appositi: ciò consegue dalla casualità intrinseca nell'osservazione dello stato di un registro di *qubit*.

4.Vantaggi del computer quantistico

Per poter dimostrare il miglioramento che i computer quantistici apporterebbero all'informatica e per poter dare fondo alla tesi che ne evidenzia la validità, alla luce delle caratteristiche precedentemente descritte, è necessaria un'analisi dei campi di interesse e delle soluzioni a problemi noti in cui i computer quantici superano in potenza quelli attuali. Secondo un grafico prodotto da IBM, in seguito a diverse analisi, è possibile classificare i computer quantistici in tre categorie[19]:

- Il "quantum annealer computer" rappresenta il tipo di macchina più semplice, in grado di risolvere problemi di ottimizzazione. I computer D-Wave prodotti dall'omonima azienda appartengono, secondo alcuni, a questa categoria e colossi come Google stanno lavorando per dimostrare la validità di questi prodotti in merito alla risoluzione non solo di problemi specifici (come avviene appunto per questo tipo di calcolatori), ma anche per usi più flessibili [20][21]. Osservando soltanto il numero dei *qubit* di un calcolatore D-Wave di ultima generazione esso sembrerebbe appartenere ad una classe più elevata.
- "Analog quantum computer" è il nome della seconda categoria di computer quantici: costituiti da un numero di *qubit* attorno ai 50 o 100, queste macchine possono riprodurre le interazioni quantistiche della materia, inoltre la loro velocità di calcolo è superiore a quella dei computer tradizionali.
- In ultimo vi è la macchina definita "universal quantum computer". Secondo le stime dovrebbe contenere più di 100.000 *qubit*, il che la rende la più difficile da realizzare. In compenso, con gli algoritmi conosciuti un calcolatore di questo tipo potrebbe risolvere problemi di "machine learning", ricerca nei database, crittografia etc.

4.1. Database e Big Data

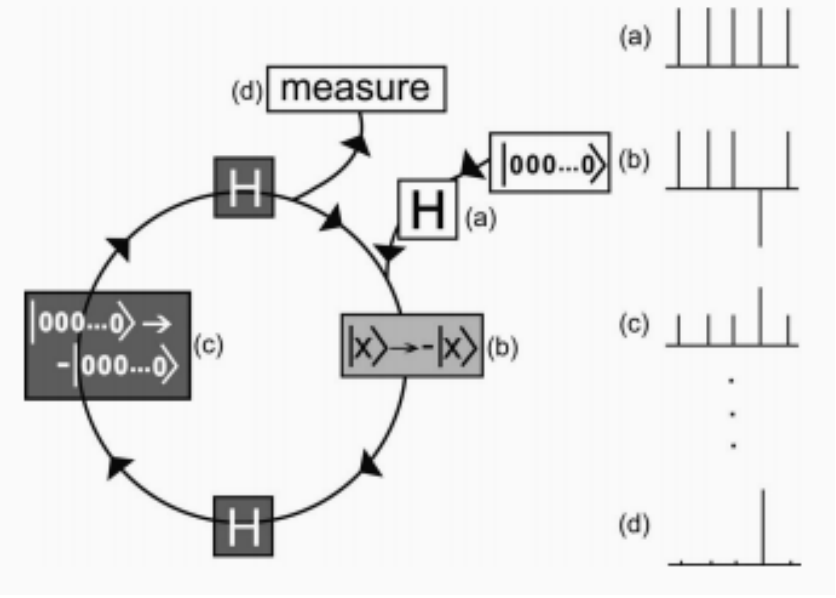
Come anticipato nel Capitolo 2.1, il primo passo verso la raccolta di dati con approccio quantistico fu compiuto da Lov Grover grazie allo sviluppo dell'algoritmo eponimo [CITATION Gro96 \l 1040]. L'algoritmo di Grover permette di trovare un riscontro, in un database di N elementi, in un tempo $O(N^{1/2})$ rispetto a quello impiegato da un computer classico O(N). Ciò avviene con una probabilità di

19 https://www-03.ibm.com/press/us/en/pressrelease/48258.wss, 26/08/2016
20 "*Il computer quantistico di Google funziona davvero.* Forse", it.ibtimes.com, 27/07/2016
21 "The three types of quantum computers and their applications", Jeff Desjardins, visualcapitalist.com, 26/08/2016

poco inferiore ad 1 (Williams e Clearwater, 2000). In quello che viene definito “il problema della rubrica telefonica”, si suppone di voler ricercare un nome in rubrica conoscendo soltanto il numero di telefono; essendo la rubrica ordinata rispetto ai nomi, un computer deterministico impiega in media N/2 tentativi per trovare il riscontro (al più N tentativi). Al contrario, una macchina quantistica, utilizzando l’algoritmo di Grover, impiega $N^{1/2}$ tentativi[22] con una probabilità ≈ 1. Il suo funzionamento può essere descritto nel seguente modo:

- Si parte con un registro a L qubit in uno stato iniziale neutro, es. |00..0> ($N=2^L$) a cui si applica un’operazione quantistica, detta trasformata di Hadamard, che genera una sovrapposizione equiprobabile di tutti i possibili stati del registro;
- Viene effettuata una richiesta a quello che viene definito “oracolo” applicando diversi operatori in modo ciclico. Come si può vedere dall’immagine 4.1a), l’oracolo opera una riflessione basata sul valore dell’obiettivo che ne inverte l’ampiezza provocando un abbassamento della media del medesimo valore a livello globale. Agendo poi sullo stato iniziale, con un’altra inversione si otterrà un abbassamento delle ampiezze generale ad eccezione di quella dell’obiettivo che tendenzialmente verrà triplicata;
- Dopo $N^{1/2}$ cicli vi sarà un’ampia differenza tra l’ampiezza del record desiderato e le altre, ed il risultato corrisponderà a quello richiesto con probabilità prossima ad 1.

Figura 4.1 *Schema di applicazione generica dell’algoritmo in si distinguono le quattro fasi.*

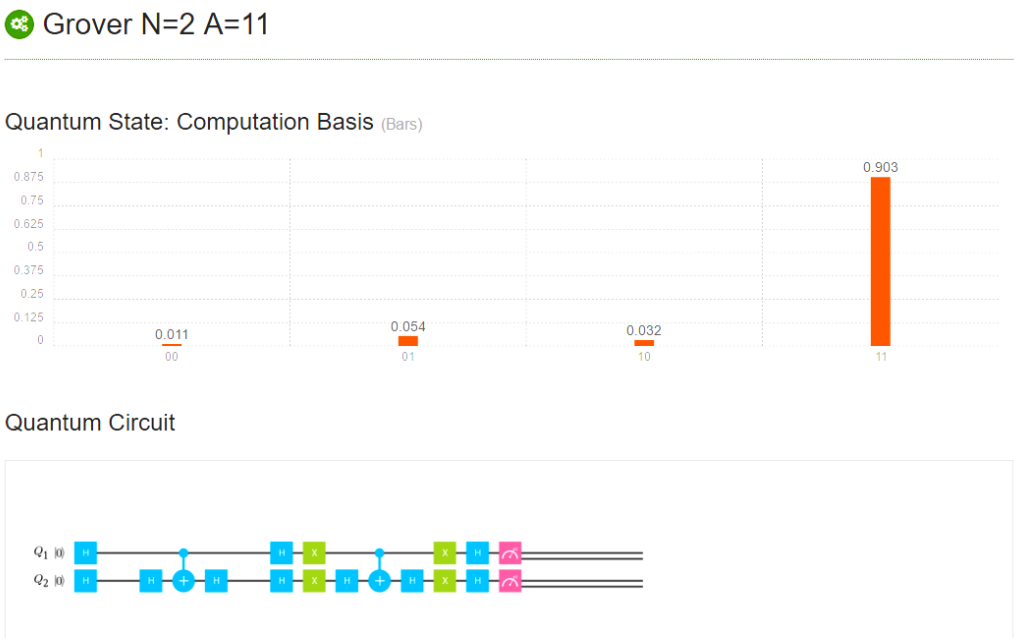


22 È stato dimostrato che nessun algoritmo quantistico può risolvere il problema in meno di $N^{1/2}$ tentativi, ciò rende l’algoritmo di Grover asintoticamente ottimale [CITATION Ben97 \l 1040]

(Fonte[CITATION Bri05 \l 1040])

Sono già state effettuate con successo delle applicazioni reali di tale algoritmo; il problema attuale riguarda la scalabilità in database di grandi dimensioni, dove i risultati sarebbero più evidenti. Tra le applicazioni più famose dell'algoritmo di Grover c'è quella effettuata attraverso un computer quantistico creato nel diamante[23]. Nell'immagine sottostante (figura 4.2) è riportata l'applicazione dell'algoritmo effettuata tramite il computer quantistico IMB, a 5 *qubit*, utilizzabile in cloud. L'esempio rappresenta la ricerca del record "11" all'interno di un registro a 2 *qubit* e mostra la probabilità di ottenere il risultato desiderato.

Figura 4.2 *Esempio di sviluppo dell'algoritmo di Grover su computer IBM. Caso di ricerca del record "11" in un registro a due qubit.*



(Fonte: IBM)

L'applicazione dell'algoritmo affiancata all'utilizzo di database quantistici rappresenterebbe un grosso miglioramento ma, nell'epoca dei Big Data, potrebbe non bastare per dare credito a questa tecnologia.

23 lescienze.it, "*Un computer quantistico nel diamante*", 1/08/2016

Una possibile soluzione giunge con l'utilizzo di una tecnica matematica: la topologia. La topologia analizza un insieme di dati e lo descrive attraverso *buchi e connessioni*. Secondo la visione topologica, ad esempio, l'alfabeto maiuscolo sarebbe composto da sole tre categorie: le lettere senza buchi: I, L, C…, le lettere con un buco: A, O, R…, e la B, unica lettera a presentare due buchi. Secondo questa branca della matematica quindi, un sistema può essere modificato come si vuole fintanto che non vengono né distrutti né creati nuovi buchi: con questo strumento è possibile catalogare, studiare e analizzare una rete i cui nodi rappresentano un tipo qualsiasi di dato o di informazione, semplice o complessa[24].

Il limite attuale di questa tecnica risiede nel fatto che l'analisi di una rete di 300 nodi richiederebbe un numero di unità di calcolo pari a 2^{300}, "un computer delle dimensioni dell'universo" secondo quanto scrive Seth Llyoid in un articolo pubblicato su *Nature Communication*[25]. Al contrario utilizzando un computer quantistico sarebbero necessari "soltanto" 300 *qubit*. I Big Data sono un concetto che riguarda innumerevoli ambiti che richiedono la raccolta e l'analisi di grandi quantità di dati: se questa tecnologia raggiungesse un adeguato livello di complessità verrebbero risolti molti dei problemi che attualmente i super computer non sono in grado di affrontare.

Non è difficile immaginare quali migliorie si otterrebbero dalla possibilità di leggere, conservare ed analizzare un numero esteso di dati. Tra i possibili beneficiari della computazione quantistica legata all'osservazione di grandi quantità di dati troviamo il *processing* di immagini. Normalmente questo processo richiede grandi risorse di calcolo in relazione all'aumento della precisione con cui un'immagine viene mappata e successivamente studiata da un calcolatore. I servizi segreti e gli enti di sorveglianza utilizzano questa tecnica per garantire la sicurezza in vari ambiti. Inoltre il *processing* di immagini è utilizzato per il riconoscimento facciale, lo sviluppo di sistemi intelligenti ed altri innumerevoli progetti. Ebbene, sono stati sviluppati algoritmi di calcolo quantistico che permettono di svolgere queste operazioni in modi molto efficienti[CITATION Ras07 \l 1040][CITATION Car12 \l 1040]. Allo stesso modo si potrebbero migliorare l'analisi di immagini da satellite, terrestri e spaziali; quest'ultime permetterebbero di ottenere informazioni maggiori su pianeti molto lontani dal nostro, scoprendo così, ad esempio, se esistono pianeti in grado di ospitare la vita come la conosciamo. Per questo motivo e per la possibilità di migliorare i calcoli nelle missioni spaziali, la NASA, in collaborazione con Google, possiede un computer quantistico attualmente sotto test nella sezione

24 media.inaf.it, *"Big Data dominati dalla topologia quantistica"*, Marco Malaspina, 1/08/2016
25 News.mit.edu, *"A new quantum approach to big data",* David L. Chandler, 1/08/2016

dell'ente dedicata[26]. Un altro esempio di questo tipo consta nella meteorologia: difatti un computer quantistico potrebbe, in via teorica, analizzare e raccogliere dati ad una velocità tale da rendere molto più accurate le previsioni future.

4.2. Crittografia

La crittografia implementata nei computer classici si basa su molte tecniche differenti mirate a mantenere la segretezza end-to-end del messaggio, di modo che solo la sorgente ed il destinatario possano interpretarlo correttamente mentre chiunque riesca ad intercettarlo non sia in grado di decifrarne il contenuto. Il messaggio crittografato deve poter essere decrittato velocemente dal ricevente mentre deve risultare arduo da decifrare ad un possibile intercettore. I metodi matematici sono quelli che da sempre hanno reso possibile tale sviluppo: in particolare metodi reversibili in cui l'operazione principale risulta molto più semplice del suo inverso. L'esempio più comune è fornito dalla moltiplicazione e dalla fattorizzazione. Mentre risulta facile la moltiplicazione tra due numeri primi molto grandi, la fattorizzazione del prodotto negli stessi richiede una grande quantità di tempo. Lo scambio di messaggi in codice si basa sull'utilizzo di "chiavi" che permettano di rendere il messaggio intelligibile soltanto a sorgente e destinatario. La crittografia può essere quindi classificata in: simmetrica, se si utilizza la stessa chiave per crittografare e decrittare, e in asimmetrica nel caso in cui vi siano due chiavi, una pubblica ed una privata. Quest'ultima è la più utilizzata ad oggi poiché è molto sicura e non richiede lo scambio, tra gli utenti, della chiave segreta, prima dell'inizio delle trasmissioni. Nel caso opposto, invece, un utente possiede una chiave che rende pubblica con cui chiunque può codificare il messaggio da inviargli, ed una chiave privata con cui solo lui può decifrare il messaggio ricevuto. La relazione tra la coppia di chiavi è costituita da una funzione matematica basata sulla moltiplicazione e sulla fattorizzazione, come anticipato prima[CITATION Wil98 \l 1040]. Un estraneo che intercetti la trasmissione crittografata in modo asimmetrico avrebbe bisogno di un calcolatore potentissimo e di un tempo estremamente lungo per ricavare la chiave privata di un utente. Non è impossibile che si trovi un algoritmo di calcolo classico per raggiungere tale scopo ma attualmente non esiste. Nel 1994, Umesh Vazirani[27], informatico dell'Università della California, ha dichiarato, in merito alla difficoltà di fattorizzare un numero di 200 cifre:

> "Non è un solo un caso che tutti i computer del mondo oggi non siano in grado di fattorizzare un numero del genere. È davvero molto più drammatica [la situazione] … Anche immaginando che ogni particella dell'universo

26 "*NASA quantum artificial intelligence Laboratory*", ti.arc.nasa.gov, 04/08/2016
27 https://www.edx.org/bio/umesh-v-vazirani

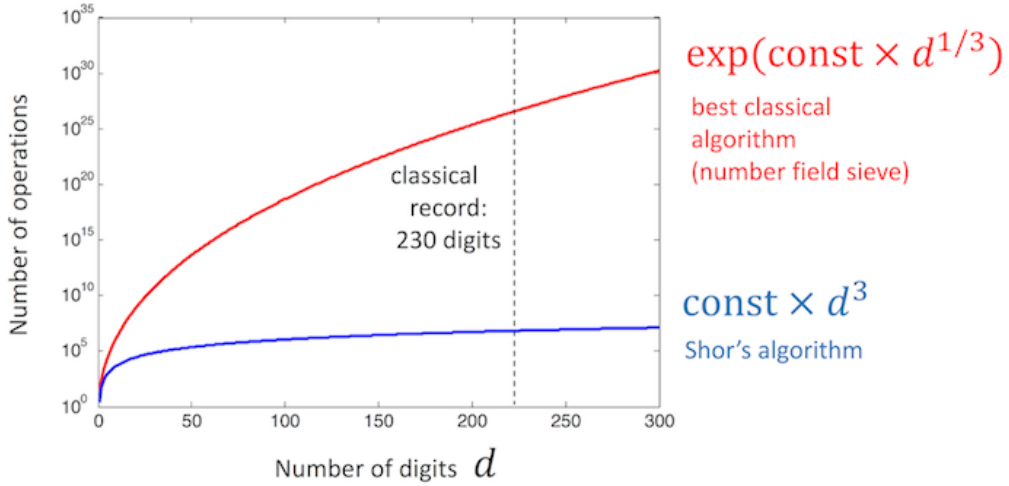
costituisca un computer [classico] e che esso lavori alla velocità massima per tutta la vita dell'universo, questo non basterebbe a fattorizzare quel numero." (Vazirani, 1994)

La differenza tra un computer quantistico ed uno classico risiede nel numero di operazioni necessarie ad ognuno per raggiungere lo scopo. Grazie alla sovrapposizione degli stati e alle altre sue caratteristiche, un computer quantistico può svolgere la fattorizzazione in meno operazioni e di conseguenza in tempi accettabili. Come è noto, per fattorizzare un intero *n* è necessario calcolare i prodotti tra tutte le coppie ordinate di interi minori di $\sqrt{n}$. Se ipotizziamo che gli interi delle coppie siano contenuti in due diversi registri ciò è possibile con l'ausilio dell'algoritmo di Deutsch. In questo caso l'uso di tale algoritmo basato solo sulla sovrapposizione di stati non risulta particolarmente intelligente in quanto vi è la stessa possibilità di ottenere la risposta corretta o una di quelle sbagliate: è necessario far sì che la risposta corretta sia più probabile delle altre. Al contrario l'algoritmo di Shor dimostra come un computer quantico può essere usato per fattorizzare grandi interi in modo efficiente (l'algoritmo è caratterizzato da una complessità polinomiale come si può vedere dalla figura 4.3). Ipotizzando di voler fattorizzare un numero $N = pq$ il funzionamento dell'algoritmo di Shor è diviso in due parti[28][CITATION Hau09 \l 1040]:

- Parte classica: si sceglie un *a* casuale tale che sia minore di *N* e si calcola il MCD(*a*, *N*). Fatto ciò, se il risultato è diverso da 1 si ha che *a* è uno dei fattori triviali di *N*, altrimenti, è necessario procedere diversamente poiché si è scoperto che *a* ed *N* sono coprimi. A questo punto si trova il periodo con cui si ripete la funzione $f(x) = a^x \bmod N$ ovvero, il periodo con cui si ripete il valore del resto della divisione tra le potenze di *a* e *N*. Se il periodo *r* non rispetta determinate condizioni, tra i quali la parità, si deve cercare un nuovo *a*.
- Parte quantistica: si ricerca la periodicità della funzione $f:\{1,0\}^n \rightarrow \{1,0\}^m$ dove il registro in input a *n qubit* contiene i valori della variabile *x* mentre quello in output contiene i valori di *f* associati a *x*. Si vagliano poi tutti i possibili stati dell'output, applicandovi la trasformata di Fourier quantistica, fino a trovarne uno che eguagli i periodi delle funzioni delle due parti. A quel punto si controlla che i due fattori trovati *p* e *q* diano come risultato *N*. In ogni caso questa parte rappresenta un algoritmo di ordinamento quantistico per la ricerca dei fattori.

Figura 4.2 *Differenza di velocità tra il migliore algoritmo classico e l'algoritmo quantistico di Shor*

28 "*Quantum computer*", D. Hautle, G. Genazzi, C. Ferrari, 2008-2009



(Fonte: IBM, 2016)

Attualmente l'implementazione di questo algoritmo è limitata ad un numero esiguo di *qubit*[29], ma in futuro questo tipo di computer potrebbe causare grossi problemi se immesso nella rete poiché permetterebbe di trovare le "chiavi" asimmetriche degli utenti decifrandone i messaggi. Google è già a lavoro in merito alla questione e sta cercando dei modi per difendere il proprio browser da questo tipo di attacchi[30].

Le possibilità di un computer quantistico nella crittografia non sono solo legate alla decrittazione ma anche al suo opposto. Grazie all'uso dei *qubit* e alla caratteristica di non-clonabilità, si palesa la possibilità di nuove tecniche di codifica. La non-clonabilità difatti garantirebbe ad una rete quantistica una sicurezza praticamente inviolabile: in linea di massima un dato quantistico può essere utilizzato per produrre un calcolo ma la sua intercettazione fornirebbe uno stato casuale tra quelli possibili, ovvero un'informazione nulla. Con questa caratteristica ci si può scambiare la chiave di un canale a crittografia simmetrica senza pericoli. È inoltre possibile, anche se risulta ancora molto difficoltoso, utilizzare una codifica basata sulle fasi dei *qubit* all'interno di una fibra ottica, di gran lunga maggiormente controllabili rispetto alla polarizzazione [CITATION Bou00 \l 1040].
Quello della crittografia è l'ambito che, in primo luogo, ha reso i computer quantistici di largo interesse per gli studiosi informatici: la possibilità di intercettare e decodificare una grande quantità delle trasmissioni attuali metterebbe in crisi il sistema di comunicazioni odierno ed al contempo

29 "*Quantum hacking is now possible…"*, Kavita Iyer, www.techworm.net, 29/08/2016
30 "*Google test new crypto in Chrome…"*, Andy Greenberg, www.wired.com, 29/08/2016

costringerebbe tutti a ricorrere a nuovi mezzi. Anche se ancora la scalabilità della tecnica resta un problema, la possibilità di utilizzare la crittografia quantistica appare sempre più vicina. Il massimo traguardo raggiunto nel campo, ad oggi, appartiene a dei ricercatori canadesi che, utilizzando un processore D-Wave 2X, sono riusciti a fattorizzare il numero a 18-bit 200.099 [CITATION Dri16 \l 1040]; considerando che le chiavi attuali sono basate su numeri a 2048-bit, la ricerca è ancora lontana dal mettere in pericolo gli attuali standard di crittografia.

4.3. Il "machine learning"

Con "machine learning" si intende la capacità di una macchina di apprendere in modo automatico reagendo adeguatamente sulla base dei dati raccolti. La macchina deve quindi essere in grado di comportarsi diversamente a fronte di dati differenti. Si tratta di un campo multidisciplinare poiché coinvolge innumerevoli materie di studio[31]. Tra i principali argomenti di ricerca attivi oggi nell'informatica, il "machine learning" prevede diverse modalità di sviluppo che variano a seconda del tipo di algoritmo di apprendimento utilizzato (es. supervisionato se effettua il confronto con un dato desiderato, esperienza con apprendimento continuo se ci si ottiene un "premio" o una "punizione" in base alla correttezza del risultato, etc.), del tipo di approccio (ad esempio: ad albero decisionale, con reti neurali artificiali, con *clustering*, etc.) e appunto in base al campo di utilizzo. Questa tecnica è applicata negli studi in cui sostituisce l'ormai obsoleto concetto di intelligenza artificiale, per misurare le probabilità, per il calcolo della complessità di un compito, nella teoria dell'informazione ed in molti altri settori. Data l'impossibilità per una macchina di analizzare e calcolare tutti i possibili comportamenti, l'apprendimento automatico prevede di determinare, con specifici algoritmi, quelli corretti e di cambiare approccio in base ai dati raccolti. L'obiettivo è quindi quello di spostarsi verso procedure intelligenti che siano in grado di adattarsi ai cambiamenti; d'altra parte l'informatica quantistica dà la possibilità di analizzare grandi quantità di dati e di verificare più scelte grazie al principio della sovrapposizione quantica. Negli ultimi anni sono nati molti algoritmi quantistici di machine learning, tra i quali il più importante sembra essere l'HHL[32]. Basti pensare a quanto esposto in precedenza: la capacità di una macchina di analizzare dati e di elaborarli per decidere quale comportamento tenere verrebbe ampliata di parecchio dalla possibilità di effettuare più calcoli

31 *"Apprendimento automatico"*, wikipedia.it, 10/08/2016
32 *"Quantum machine learning algorithms …"*, www.scottaaronson.com, Scott Aaronson, 13/08/2016

simultaneamente. Nel 2013, Google, in collaborazione con la NASA e l'associazione delle università per la ricerca spaziale, ha addirittura fondato un centro chiamato "Quantum Artificial Intelligence Lab" nella quale viene utilizzato il D-Wave per effettuare ricerche di questo tipo[33]. Anche la Microsoft ha seguito questo esempio avviando molte attività nel campo[34]. Secondo Seth Lloyd[35] del MIT infatti:

> "Viene fuori che molti algoritmi quantici di machine learning attualmente lavorano abbastanza bene sui computer quantistici se si presuppone di avere una Q-RAM [RAM quantistica]. Questo è esattamente il tipo di problemi matematici che stiamo provando a risolvere e noi pensiamo di poterlo fare molto bene con la versione quantistica degli stessi." (Seth Lloyd, conferenza nella Napa Valley[36])

4.4. Simulazioni scientifiche

Simulare un processo scientifico è qualcosa che va al di là delle capacità di un computer classico. Come descritto da Feynman, un computer classico non può simulare una macchina quantistica né tantomeno la natura delle particelle che stanno alla sua base. Dunque per riprodurre un processo scientifico basato sulle interazioni tra particelle è necessario utilizzare un computer di tipo quantico, ovvero una macchina che segue le stesse leggi probabilistiche dei fenomeni che emula. Un computer abbastanza potente sarebbe in grado di simulare i processi che vengono messi in moto all'interno del LHC, portando la fisica a fare innumerevoli passi avanti in tempi esigui. Le simulazioni possono poi essere espanse al campo della chimica: vi sarebbe quindi la possibilità di simulare reazioni ed evoluzioni dei composti. La branca che ne ricaverebbe i maggiori vantaggi sarebbe la medicina: si potrebbero creare nuovi farmaci, vaccini e trattamenti la cui sintetizzazione richiede attualmente tempi lunghissimi. Con un computer quantistico i tempi per la ricerca medica sarebbero ridotti di moltissimo. Inoltre, la potenza di queste macchine è evidenziata da un'ulteriore possibilità: secondo alcune ricerche, un computer quantico riuscirebbe a riprodurre una rete neurale e a simularne il funzionamento. Ovviamente il tipo di rete simulata avrebbe un'estensione dipendente dal numero di *qubit* della macchina, ma partendo da reti semplici, e con molto tempo a disposizione, gli uomini potrebbero essere in grado di apprendere molto sul cervello umano, in particolare ciò che è legato alle malattie neurologiche e ciò che non traspare ancora dalle molteplici macchine per la scansione cerebrale.

---

33 *"Google Quantum A.I. Lab Team – Google+"*, plus.google.com, 13/08/2016
34 "Microsoft Leans on Machine Learning and Quantum Computer …", MIT Technology Review, 13/08/2016
35 http://meche.mit.edu/people/faculty/SLLOYD@MIT.EDU
36 "*How quantum computer and machine learning will revolutionize…"*, Jennifer Ouelette, www.wired.com, 29/08/2016

5.Computer quantistico: realizzazione

Il computer quantistico, secondo le ultime definizioni, è un computer che sfrutta i fenomeni tipici della meccanica quantistica per trattare ed elaborare delle informazioni. L'elaborazione delle informazioni può essere così descritta: si pongono i *qubit* in uno stato iniziale, solitamente uno stato neutro, si fanno passare tali qubit attraverso alcuni componenti definiti "gate quantistici" e successivamente si legge di nuovo il registro, ottenendo con una certa probabilità (dipendente dall'algoritmo implementato) il risultato desiderato.

La realizzazione di un computer quantistico, perché sia tale, deve rispettare almeno i cosiddetti criteri di *Di Vincenzo*[37], ovvero:

- Si devono avere *qubit* ben definiti e identificabili e deve essere possibile poterne aumentare il numero, cioè avere un sistema scalabile che mantenga un certo livello di affidabilità;
- Deve essere possibile inizializzare i *qubit* ad uno stato desiderato prima di eseguire qualsiasi calcolo;
- Si deve avere un tempo di decoerenza maggiore del tempo di calcolo. Idealmente il sistema dovrebbe essere totalmente isolato dall'esterno. (Questo argomento verrà trattato in dettaglio nel capitolo successivo).
- È necessario avere un set universale di gate quantistici ben delineati.
- Infine deve essere possibile effettuare una misurazione del risultato, potendolo trasmettere all'esterno. In particolare deve essere possibile una lettura selettiva dei *qubit*.

A livello fisico, le tecnologie con cui può essere realizzata una macchina quantistica, ovvero le tecnologie con cui si possono realizzare dei *qubit*, sono diverse: ognuna di esse presenta vantaggi e svantaggi in relazione alla *checklist di Di Vincenzo*. Ad esempio il processo NMR (risonanza magnetica nucleare), che prevede l'applicazione di campi elettromagnetici agli spin delle molecole, permette la realizzazione di *qubit* che possono essere posti in uno stato iniziale, lasciati evolvere e poi letti per verificarne il risultato. Questo tipo di realizzazione, così come le altre di tipo microscopico, offre un vantaggio rispetto al tempo di coerenza ma è di difficile integrazione in un circuito complesso. Il problema opposto si ha invece nel realizzare sistemi quantistici macroscopici, dove l'integrazione in circuiti risulta semplice ma l'isolamento e il conseguente aumento del tempo di coerenza sono di difficile attuazione. Oltre al NMR vi sono altre due tecnologie microscopiche principali di

37 http://www.uniroma2.it/ppg/im/repository/Relazione%20Quantum%20Computing.pdf

implementazione dei qubit: il sistema a ioni intrappolati, basato sulla "trappola per ioni" di W. Paul e i reticoli ottici che sfruttano l'intrappolamento di fasci di laser in gabbie costituite da reticoli cristallini artificiali. Per quanto riguarda il NMR e la tecnologia a ioni intrappolati è stato possibile implementarvi diversi algoritmi tra i quali quello di Deutsch-Jozsa e quello di Shor. Inoltre la tecnologia a ioni si è rivelata particolarmente versatile nella realizzazione di gate del tipo C-Not mentre, quella basata sui fotoni, è stata utilizzata nella creazione di stati entangled.
Nell'ultimo periodo si stanno vagliando molte ipotesi tecnologiche da sfruttare nella realizzazione di una macchina quantistica definitiva. Tra tutte, le più studiate sono: quelle riguardanti i *qubit* a stato solido che risultano particolarmente resistenti al fenomeno della decoerenza, i sistemi basati su *superconduttori*, la cui necessità principale è quella di trovare un materiale di questo tipo a temperature vicine a quella ambiente, il Q. C. ottico ed il "cavity QED" che si basa sulla relazione e lo scambio di informazioni tra fotoni intrappolati in una cavità e atomi esterni alla stessa[38].

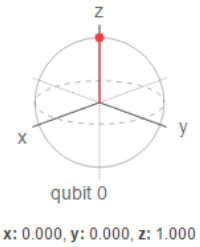
5.1. Gate quantistici

Una delle caratteristiche fondamentali riguardanti i gate quantistici è la loro reversibilità. A differenza dei gate classici, quelli quantistici richiedono un numero di uscite uguale al numero di ingressi, in modo da permettere, appunto, la possibilità di invertire un calcolo. La motivazione alla base di questa caratteristica va ricercata all'interno dell'equazione di Schrödinger che descrive l'evoluzione di un sistema quantico. Dunque, essendo anch'essi dei sistemi quantistici, i gate di questo tipo devono rispettare l'equazione di Schrödinger (anche se in modo differente a seconda del tipo di gate) e di conseguenza devono essere reversibili [CITATION Wil00 \l 1040]. I gate quantistici permettono di variare la probabilità che, una volta letto, un *qubit* collassi in un valore piuttosto che nell'altro.
Attualmente uno dei modi più comodi per comprendere il funzionamento di un *qubit* è quello di simularne o sperimentarne l'evoluzione grazie al computer quantistico a 5 *qubit* messo a disposizione, in "cloud", dal colosso informatico IBM.
Grazie alla "IBM *Quantum experience"*[39] è possibile sia simulare il funzionamento della macchina quantistica sia farne un uso effettivo. Verrà utilizzato il modello della sfera di Bloch per riportare lo stato dei *qubit* in modo da avere un'idea più concreta dello scopo di ogni gate: in figura 5.1 è riportato come esempio lo stato iniziale del *qubit 0*. Si presuppone inoltre che inizialmente tutti i *qubit* si trovino nello stato |0>.

38 *"Cavity quantum electrodynamics"*, en.wikipedia.org, 27/07/08
39 http://www.research.ibm.com/quantum/

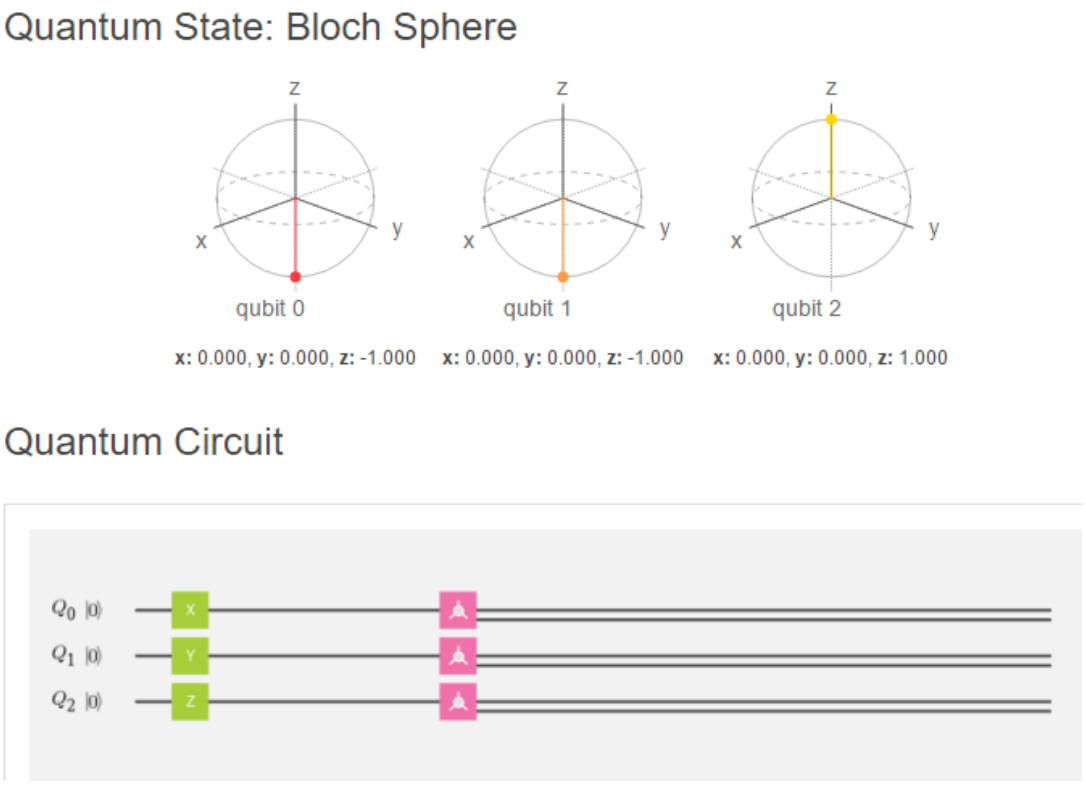
Figura 5.1 *Sfera di Bloch rappresentante il qubit 0 nello stato di riposo |0>*

(Fonte: IBM, 2016)



I cosiddetti “gate di Pauli” sono gate che, considerando la sfera di Bloch rappresentativa di un *qubit*, ci permettono di invertire la direzione, lungo gli assi, del vettore di probabilità. Di conseguenza esistono tre tipi di gate di Pauli: gate di Pauli X, Y e Z. Ognuno di essi, quindi, ruota il vettore probabilistico di 180° rispetto ad uno dei tre assi scelti per descrivere il sistema. In figura 5.2 sono riportati i cambiamenti di 3 *qubit,* inizializzati nello stato base (|0>), dopo aver attraversato ciascuno dei gate di Pauli. L’immagine di esempio si riferisce al caso di un elaboratore quantistico ideale.

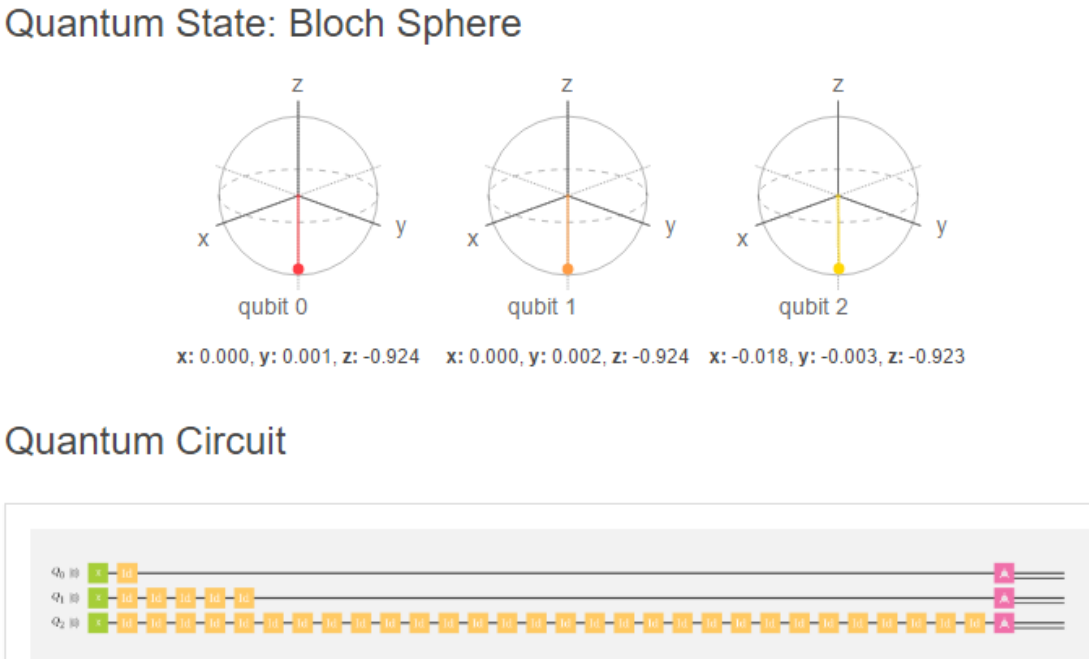
Figura 5.2 *Stato di 3 qubit dopo aver attraversato i gate di Pauli*



(Fonte: IBM, 2016)

Nel caso del gate X e del gate Y, il *qubit* in ingresso ha subìto un cambiamento, spostandosi dallo stato |0> allo stato di massima energia |1>. Al contrario nell'attraversare il gate Z, il *qubit* Q_2 non ha subìto alcun cambiamento in quanto una rotazione dell'asse Z non comporta una variazione per il vettore probabilistico che si trova lungo la stessa. Ovviamente l'utilità di ogni gate di Pauli dipende dalla posizione del vettore nel momento in cui viene attraversato ma, grazie a questi gate, è possibile invertire una delle tre coordinate del vertice del vettore in modo da scambiare, di conseguenza, la probabilità che la funzione d'onda del *qubit* ricada in uno dei due stati logici possibili.
Il gate identità è quello di cui si intuisce più facilmente il funzionamento in quanto, sostanzialmente, non apporta cambiamenti alle probabilità di stato dei *qubit*. Normalmente lo scopo di un circuito di identità è quello di creare un ritardo ai fini di sincronizzare il proprio output con quelli di altri gate [CITATION Wil11 \l 1040]. Nella simulazione ideale di un *qubit* quantistico si otterrà che per ogni applicazione di questo tipo di gate, il risultato in fase di verifica resta lo stesso. Nel caso reale però la situazione appare diversa: il vettore tende a spostarsi verso lo stato meno energetico ad ogni applicazione del gate identità.

Figura 5.3 *Variazioni della sfera di Bloch in seguito a ripetute applicazioni del gate identità su qubit portati allo stato di massima eccitazione*

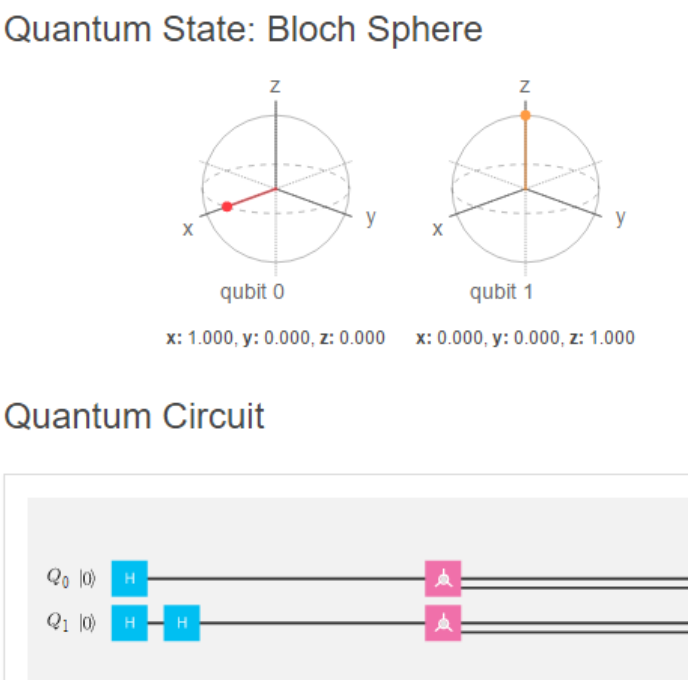


(Fonte: IBM, 2016)

Il fenomeno descritto è quello che verrà trattato in seguito con il nome di *energy relaxation*, parte del processo di decoerenza. Come si può vedere dall'esempio, estrapolato dalla simulazione realistica del calcolatore quantistico IBM (figura 5.3), più tempo trascorre e più il vettore si allontana dallo stato di massima eccitazione.

Per creare la sovrapposizione di stati è necessario introdurre nuovi gate: in primis il gate di Hadamard. Il gate H permette di effettuare uno spostamento di $\pi/2$ rispetto all'asse Y e di π rispetto all'asse X del vettore di probabilità, facendo sì che, dopo un'applicazione, il *qubit* trascorra metà del suo tempo nello stato |0> e metà nello stato |1>.

Figura 5.4 *Funzionamento del gate H*

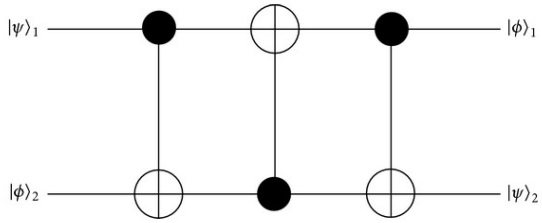


(Fonte: IBM, 2016)

Inoltre è importante evidenziare che l'operatore di Hadamard è legato ad una matrice unitaria poiché $HH^*=I$ ovvero all'operatore identità. Come si evince dall'immagine 5.4, due applicazioni di tale operatore riportano il vettore allo stato iniziale. Vi è poi l'operatore di fase S ed il suo inverso, il cui scopo è appunto quello di variare la fase del vettore. Come il gate di Pauli Z, il gate S permette una rotazione del vettore rispetto all'asse Z, ma in questo caso gli angoli di variazione dipendono dal tipo di gate progettato. Quello messo a disposizione da IBM permette una rotazione di 90°.

Per introdurre un livello di calcolo superiore è necessario però ricorrere all'utilizzo dei cosiddetti gate condizionati. Il più importante di essi, quantomeno in materia di studio, è il CNOT o *controlled NOT*; esso è anche il più semplice da realizzare. Il suo funzionamento è così descritto: "…il valore del cosiddetto *qubit* obiettivo [del gate CNOT] viene negato se e solo se il *qubit* di controllo ha il valore logico "1". Il valore logico del *qubit* di controllo non cambia" [CITATION Bou00 \l 1040]. Dunque questo gate permette la relazione più basilare tra due diversi sistemi quantistici ai fini del calcolo, rendendo così possibile un'evoluzione dello stesso, in termini di complessità. Grazie al suo utilizzo si possono creare ulteriori gate come ad esempio il gate di Swap (figura 5.5) che inverte il valore di due *qubit*.

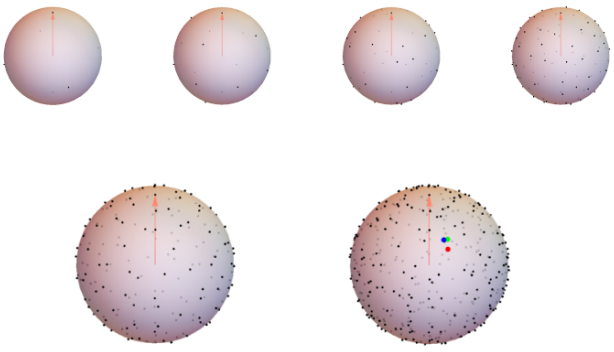
Figura 5.5 *Il gate di Swap realizzabile con una serie di 3 CNOT*



(Fonte: www.research gate.net)

Per poter disporre di un vero computer quantistico in grado di svolgere tutte le computazioni possibili, come enunciato nei criteri di *Di Vincenzo*, è necessario un set di gate "universale". I gate presentati fino ad ora costituiscono il gruppo di gate di Clifford, ma non sono sufficienti per formare un set universale in quanto per renderli tale è necessario introdurre almeno un gate non cliffordiano. Un esempio è il gate T, introdotto da IBM nel suo calcolatore per raggiungere tale scopo. Questo gate, così come il suo inverso, permette di raggiungere, in relazione al numero di volte in cui viene utilizzato, qualsiasi punto della sfera di Bloch (Figura 5.6). In questo modo ogni stato possibile, e di conseguenza ogni rapporto probabilistico tra gli stati logici "0" ed "1", può essere ottenuto. Esistono infine altri gate quantistici, come il gate di Toffoli, conosciuto anche come CCNOT (o controlled controlled NOT) a 3 *qubit*, il cui funzionamento ricalca quello del CNOT, con la differenza che i *qubit* di controllo sono due; questo gate può anche essere utilizzato, dipendentemente dalla tecnica di realizzazione, come una inversione dell'operazione di AND classica. Un ultimo gate da considerare è quello di Fredkin che esegue uno swap controllato.

Figura 5.6 *Punti raggiungibili nella sfera di Bloch dopo ripetute applicazioni del gate T*



(Fonte: IBM)

5.3. Gli ibridi

Già nei capitoli precedenti sono stati proposti argomenti che alludono all'ibridazione tra macchine quantistiche e macchine classiche: uno tra tutti è l'algoritmo di Shor, il quale si divide in due algoritmi minori, uno dei quali è basato sul calcolo classico mentre l'altro su quello quantistico. Questo tipo di ibrido è ampliamente studiato nel Lawrence Barkeley National Laboratory, in particolare da Jarrod McClean[40] e dai suoi colleghi, come soluzione per studi e simulazioni chimiche, in quanto i classici pc non sono abbastanza prestanti per lo scopo e le macchine quantistiche al contempo non sono abbastanza evolute. Come sostiene McClean[41]:

> "Ci stiamo concentrando su un modo per prendere ciò in cui i dispositivi quantistici sono migliori e nell'utilizzarli solo per quelle parti prelevando [invece] le altre parti dell'algoritmo, che sono più banali, e scaricandole ai nostri già molto potenti computer classici" (Jarrod McClean).

Questo tipo di studio, prevede di utilizzare una macchina classica per fornire l'input ed estrapolare l'output da un processore quantistico, anche perché non è possibile ispezionare il valore di un *qubit* senza farlo collassare in uno stato definito ma, con i giusti accorgimenti, è possibile utilizzarne lo stato di sovrapposizione in un calcolo il cui risultato, prima o poi, dovrà essere trasferito ad un computer classico. La Lockheed Martin[42] aggiunge che con ciò non si vuole sostenere il fatto che i processori quantistici non siano indipendenti, bensì che sia necessario un robusto "interprete" (un computer classico) per interpretare quelli che altrimenti rimarrebbero potenziali algoritmi[43]. In un certo senso si arriverebbe ad avere un rapporto CPU e processore quantistico del tutto simile a quello che c'è ora tra CPU e GPU, anche se nell'ambito della ricerca le proposte sono molte e differenti. Un esempio è quello proposto alcuni anni fa nella tesi (M.Sc.) di uno studente dell'università del Cairo[CITATION Elh07 \l 1040]; nel suo lavoro, egli, propone l'inserimento di un'unità di calcolo quantistico all'interno della CPU di un computer classico. Tale unità sarebbe accessibile, parallelamente all'ALU, nella fase di "esecuzione", di modo che, una volta decodificata l'istruzione, se essa risultasse di tipo quantistico, verrebbe inviata all'unità quantistica.

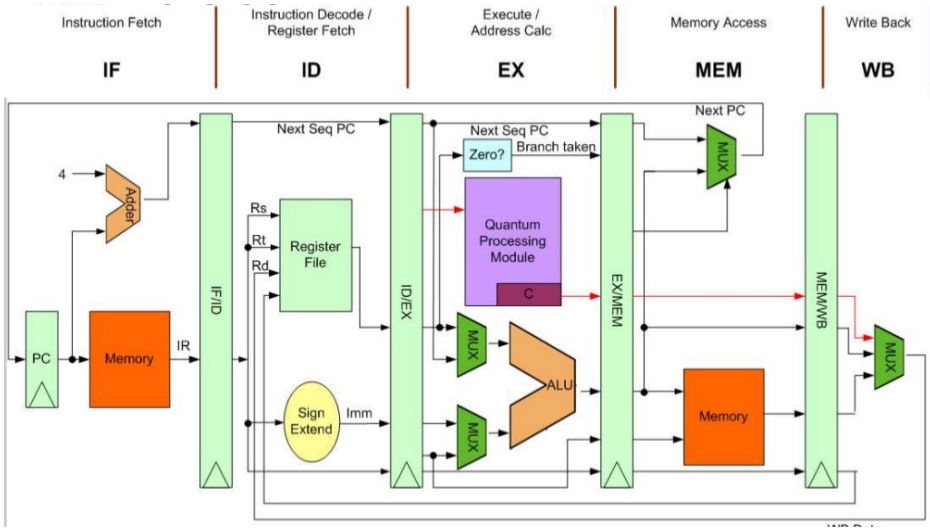
Figura 5.7 *Esempio di cablaggio di un'unità quantistica in una CPU classica*

40 http://jarrodmcclean.com/

41 "*Barkeley lab prepare for quantum-classical computing future",* phys.org, 29/08/2016

42 http://www.lockheedmartin.com/us.html

43 "*The future of quantum computer will be hybrid",* Nicole Hemsoth, nextplatform.com, 29/08/2016



(Fonte[CITATION Elh07 \l 1040])

Nell'immagine 5.7 è presente l'esempio di cablaggio dell'unità quantistica, proposto da M. Elhoushi[44]. La cosiddetta QPU, conterrebbe al proprio interno una ALU quantistica o qALU per l'elaborazione delle operazioni, un registro quantistico contenente i *qubit* con cui svolgere il calcolo, un registro classico della stessa dimensione (in bit) del precedente, ed infine diversi spazi di memoria in cui poter conservare l'istruzione corrente con i relativi parametri. A sua volta la qALU sarebbe composta da un insieme di gate quantistici in grado di eseguire tutte le possibili operazioni quantiche, grazie anche ad un linguaggio di programmazione basato sul C per lo sviluppo degli algoritmi e sul C++ per la creazione di nuove istruzioni.

5.3. Coerenza e decoerenza

Gli scienziati stanno abbattendo, uno dopo l'altro, tutti i muri che ci separano dalla realizzazione di una vera macchina quantistica, ma il grande ostacolo per rendere questi mezzi usufruibili per scopi reali, e non solo di ricerca, è la "decoerenza". Come enunciato in precedenza, esiste una relazione tra le particelle fondamentali, e di conseguenza tra i *qubit*, chiamata *entanglement*. Questo collegamento, oltre a rendere possibile il "teletrasporto quantistico", permette uno *speedup* computazionale. Un computer quantistico ad *n qubit* può trovarsi in 2^n possibili stati, ma questo non basta per ottenere la caratteristica di computazione quantica universale, ovvero la capacità di simulare un qualsiasi altro computer quantistico in quanto, l'*entanglement*, introduce nuovi "path" evolutivi al sistema, non previsti da una computazione priva di tale connessione tra *qubit*. Ad esempio, è possibile codificare un

44 https://ca.linkedin.com/in/mostafaelhoushi

problema computazionale di *n* bit in una sola particella con *k* stati; questo però richiederebbe *k* stati corrispondenti ad un atomo con 2^n livelli energetici e se immaginiamo di voler utilizzare n=32, incorriamo in un problema, poiché attualmente l'uomo non ha strumenti così precisi per distinguere un numero così elevato di stati in un sistema atomico o subatomico. Utilizzando però *n* bit in *entanglement* per effettuare lo stesso calcolo, dato che è possibile misurare i *qubit* uno ad uno, si può ottenere il risultato desiderato. La connessione tra le unità di informazione si rivela quindi indispensabile ma non è facile mantenerla nel tempo. Si dice che due *qubit* sono *entangled* quando si trovano in uno stato di coerenza e ciò può avvenire naturalmente se le due particelle, o sistemi, che costituiscono i *qubit* sono state create da un processo comune o se quest'ultime sono entrate in tale stato in modo spontaneo. Un esempio potrebbe essere rappresentato da una sorgente che emette due fotoni nello stesso istante [CITATION Bou00 \l 1040]. Si intuisce subito la difficoltà che porta alla costruzione di un computer puramente quantistico. In ogni caso la difficoltà non sta tanto nella realizzazione di un registro *entangled* con un numero elevato di qubit, quanto nel mantenere tale condizione. La decoerenza è il processo attraverso il quale due sistemi legati perdono il loro collegamento quantistico a causa di interazioni esterne. Sono sufficienti variazioni ambientali impercettibili per disturbare uno stato di *entanglement*: variazioni termiche, vibrazioni nel terreno, onde elettromagnetiche e persino le radiazioni cosmiche di fondo possono disturbare un *qubit*, facendogli perdere la propria connessione quantica e facendolo entrare invece in risonanza con l'esterno[45]. In generale i tempi di decoerenza variano tra il nanosecondo ed il secondo[46]. Alcuni computer quantici per poter prevenire una decoerenza troppo rapida richiedono di operare ad una temperatura attorno ai 20 millikelvin. Detto ciò si rende necessario lo svolgimento dei calcoli in tempi minori di quelli di decoerenza.

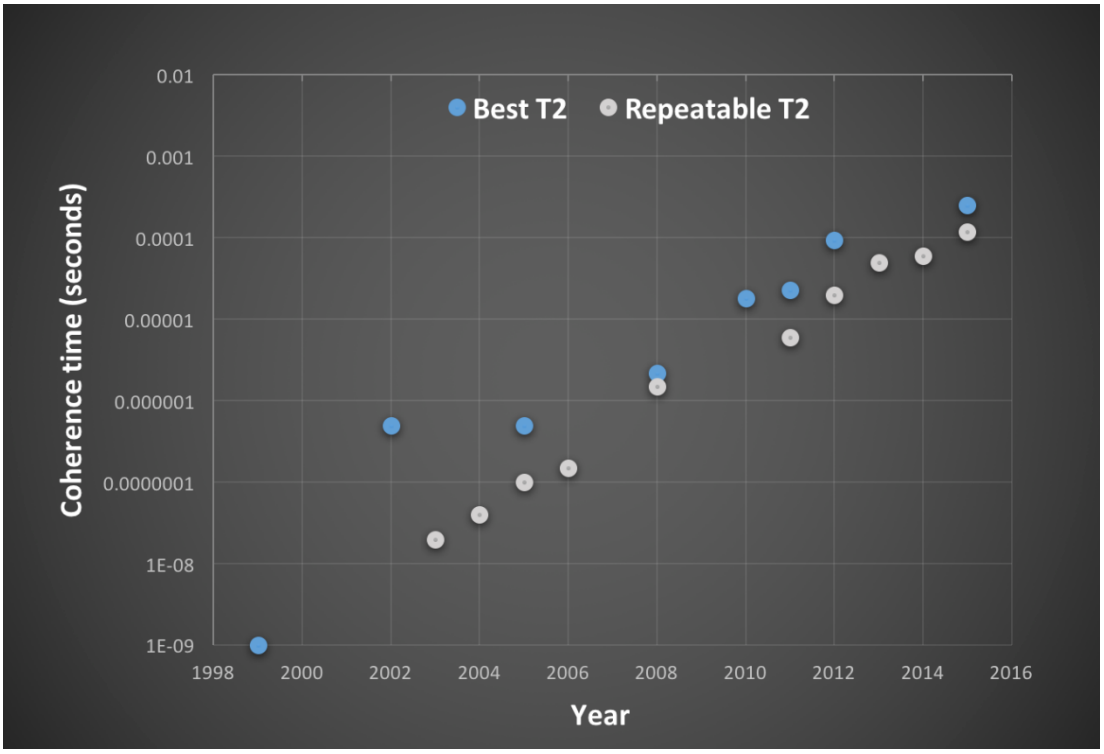
Attualmente grazie agli studi sulla correzione dell'errore quantistico è possibile incrementare il tempo di calcolo oltre tale soglia. Inoltre è possibile realizzare computer quantici più resistenti al fenomeno della decoerenza, per esempio quelli definiti "a stato solido": uno dei casi più interessanti riguarda la prima macchina di questo tipo, costruita nel diamante, in grado di eseguire l'algoritmo di Grover con una precisione del 95%[47].

Secondo una classificazione IBM si possono distinguere due tipi di decoerenza: l'*energy relaxation*, che riguarda la tendenza di un *qubit* nello stato forzato |1> a ritornare allo stato base |0>, la cui costante

45 "*Quantum Computing"*, en.wikipedia.org, 26/08/2016
46 "*Record di "coerenza" per due qubit",* www.lescienze.it, 27/08/2016
47 "*Un computer quantistico di diamante…",* lescienze.it, 26/08/2016

di tempo caratteristica è definita T_1, e il *dephasing*, che riguarda soltanto gli stati di sovrapposizione e non possiede un analogo classico. Entrambe le tipologie vengono incluse in una costante definita T_2 e nella figura seguente è possibile vederne la mappatura, rispetto agli anni, dei progressi, tracciata in relazione ai tempi di coerenza migliori e a quelli medi.

Figura 5.7 *Andamento della costante T_2 in relazione agli anni*

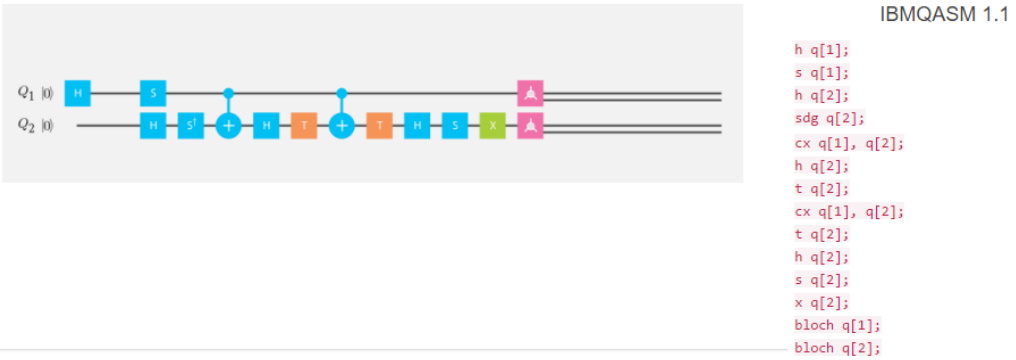


(Fonte: IBM, 2016)

6. La programmazione quantistica

La programmazione quantistica è costituita da un insieme di linguaggi che permettono di esprimere algoritmi quantici usando costrutti di alto livello[48]. Attualmente questo tipo di programmazione non rappresenta uno vero strumento informatico, in quanto la sua diffusione è limitata a scopi di ricerca che rendano più comodo lo studio di algoritmi quantistici: si tratta principalmente di pseudocodici o di formalizzazioni di algoritmi rappresentati da circuiti. Ciò appare evidente da esempi accennati in precedenza: IBM utilizza, infatti, una programmazione di tipo ladder per implementare le istruzioni corrispondenti alle applicazioni di determinati gate, e a questa è associato un codice denominato IBMQASM 1.1. Questo linguaggio rende più comodo l'approccio ai diversi algoritmi e consiste di un'istruzione per ogni tipo di gate applicabile, più altre due istruzioni che rappresentano le due diverse modalità di lettura dei risultati. Nella figura 6.1 è riportato il codice IBMQASM 1.1 relativo all'implementazione di un gate "controlled-Hadamard" con relativa circuitazione. In questo caso, quello che si osserva risulta essere un linguaggio principalmente descrittivo.

Figura 6.1 *Esempio di IBMQASM 1.1, controlled-Hadamard gate*



(Fonte: IBM, 2016)

I linguaggi di programmazione quantistici sono strumenti ancora in via di sviluppo poiché le loro potenzialità dipendono dalla loro applicazione, che a sua volta, dipende dai device quantistici in circolazione ma, come scritto da Dominique Unruh[49] nell'elaborato "Quantum programming languages"[CITATION Unr06 \l 1040], esistono due tipi di approcci a questo problema:

> "Per provare la correttezza di algoritmi e protocolli [di crittografia quantica], sono necessari linguaggi di programmazione [quantistici] con una semantica formale definita in modo esatto, mentre per la sperimentazione e la progettazione euristica di algoritmi e -quando l'hardware per il quantum computing sarà disponibile- le reali

48 En.wikipedia.com, "*Quantum programming*", 30/08/2016
49 http://kodu.ut.ee/~unruh/

applicazioni che richiederanno un linguaggio facile da usare, una specifica intuitiva del comportamento potrebbe essere sufficiente a discapito di una specifica formale della semantica." (Dominique Unruh, 2006)

Egli classifica quindi i linguaggi in pratici e formali. Diverse sono le classificazioni fatte in materia da Simon J. Gay[50] che in un *paper* omonimo[CITATION Gay06 \l 1040] spartisce i linguaggi in imperativi, funzionali ed in una terza categoria che raccoglie i linguaggi non classificabili nelle due precedenti. Nel suo manoscritto egli dimostra la presenza di un'innumerevole quantità di linguaggi, codici e pseudocodici per questo tipo di programmazione, molti dei quali non sono ancora legati a nessun tipo di hardware su cui essere eseguiti.

6.1 Linguaggi imperativi

Il primo linguaggio formalizzato per la descrizione di algoritmi quantistici fu lo pseudocodice quantistico proposto da E. Knill[51], strettamente legato ad un particolare modello di macchina quantistica ad accesso casuale(QRAM). Nell'immagine 6.2 è riportato l'algoritmo di Shor nello pseudocodice di Knill.

Figura 6.2 *Algoritmo di Shor in pseudocodice*

```
FACTORIZE(N)
 1  if N is even
 2      then return (2, N/2)
 3  if N = q^b for prime q ≥ 3 and b ≥ 2
 4      then return (q, N/q)
 5  repeat
 6          repeat choose a ∈ Z_N, a ≥ 2
 7                 d ← gcd(a, N)
 8                 if d > 1
 9                    then return (d, N/d)
10                 r ← FIND-ORDER_N(a)
11             until no failure indicated and r is even
12          d_+ ← gcd(N, a^{r/2} + 1)
13      until d_+ < N
14  d_− ← gcd(N, a^{r/2} − 1)
15  return (d_+, d_−)
16  ▷ the algorithm guarantees 1 < d_+, d_− < N
```

(Fonte:[CITATION Rüd06 \l 1040])

Molto simile al linguaggio C è invece il QCL o "quantum computation language" che con il linguaggio classico ha in comune i tipi primitivi e la sintassi. Idealmente questo codice potrebbe essere utilizzato

50 http://www.dcs.gla.ac.uk/~simon/
51 http://www.eskimo.com/~knill/cv/cv/

per legare un programma classico scritto in linguaggio C ad un algoritmo quantistico. Come si nota nella figura 6.3 la somiglianza tra i codici è evidente.

Figura 6.3 *Esempio di generico codice QLC*

```
qureg x1[2]; // 2-qubit quantum register x1
qureg x2[2]; // 2-qubit quantum register x2
H(x1); // Hadamard operation on x1
H(x2[1]); // Hadamard operation on the first qubit of the register x2
```

(Fonte: Wikipedia, 2016)

Il QLC prevede alcune caratteristiche interessanti: in primo luogo permette, grazie ad un'apposita libreria, di simulare gli stati dei *qubit* (figura 6.4), che non possono essere letti senza distruggerne l'informazione durante l'esecuzione del "programma quantistico". La *feature* più importante è invece la possibilità di inserire nel codice operazioni e funzioni definite dall'utente; in questo modo è inoltre possibile creare librerie che estendano la capacità del linguaggio. Il QLC prevede poi alcune tipologie di dati proprie: quali i *qureg* (registri quantici), le *quconst* (costanti quantistiche), il *quvoid, etc.*; inoltre vi sono tipi speciali di funzioni e la possibilità di utilizzare numeri immaginari.

Figura 6.4 *Esempio di simulazione dello stato dei qubit*

```
qcl> dump
: STATE: 4 / 32 qubits allocated, 28 / 32 qubits free
0.35355 |0> + 0.35355 |1> + 0.35355 |2> + 0.35355 |3>
+ 0.35355 |8> + 0.35355 |9> + 0.35355 |10> + 0.35355 |11>
```

(Fonte: Wikipedia, 2016)

Il secondo linguaggio quantistico imperativo implementato è il "Q Language", sviluppato come estensione del C++. Contiene delle classi per svolgere operazioni quali la trasformata di Fourier quantistica, la trasformata di Hadamard, il not e l'operazione di swap. Come nel linguaggio precedente, grazie alle meccaniche del C++, è possibile definire nuove operazioni. È inoltre presente un simulatore che esegue il processo di calcolo e con la quale si possono emulare ambienti rumorosi. Un esempio di

istruzione è dato dalla funzione Qreg x(2,0) che codifica la creazione di un registro "x" a 2 qubit, inizializzati allo stato logico "0".
Oltre a questi vi sono altri linguaggi imperativi in fase di sviluppo come il qGCL o "quantum guarded command language", ideato da P. Zuliani[52] nella sua tesi di dottorato e basato sul GCL ideato invece da E. Dijkstra[53], o ancora, il LanQ la cui sintassi è anch'essa simile a quella del C.

6.2 Linguaggi funzionali

I linguaggi funzionali sono quelli che negli ultimi anni hanno avuto maggiore successo e maggiore sviluppo. Peter Selinger[54] è l'ideatore di due codici appartenenti a questa categoria: il QFC ed il QPL, che differiscono quasi unicamente per la sintassi. Lo slogan di questo tipo di codici era "classical control, quantum data" [CITATION Rüd06 \l 1040], che lascia intuire la semplicità dell'approccio.
Il QML invece è un tipo di linguaggio con una filosofia differente, dato che prevede anche il controllo di tipo quantico. Esso è definito come un linguaggio funzionale per il calcolo quantistico su tipi finiti[55].
Il QML ha molto in comune con il linguaggio Haskell su cui sembrerebbe essere basato. Una delle sue particolarità principali è la previsione di super-operatori e la relazionalità con i circuiti quantistici. Inoltre questo linguaggio permette un controllo della decoerenza grazie ad alcune meccaniche particolari, riguardanti le regole di ortogonalità tra gli stati dei *qubit*.
Di questa categoria fa anche parte il lambda calcolo quantico, che permette di estendere i linguaggi di programmazione quantistica con la teoria delle funzioni di ordine superiore ovvero, è prevista la presenza di funzioni superiori che usino come parametri ulteriori funzioni. Il primo tipo di lambda calcolo è stato sviluppato da Philip Maymin[56] nel 1996 ed è considerato abbastanza potente da poter svolgere qualsiasi calcolo quantico. Questo linguaggio è stato ulteriormente migliorato anche se, per la sua natura, è improbabile che si riesca ad implementarlo su un dispositivo fisico.
Un ultimo esempio è Quipper, implementato su Haskell come linguaggio *embedded.* Nell'immagine 6.5 è presente un esempio di utilizzo di Quipper nella preparazione di una sovrapposizione di stati.

Figura 6.4 *Esempio di uso del linguaggio Quipper*

52 https://sites.google.com/site/zupaolo/
53 https://it.wikipedia.org/wiki/Edsger_Dijkstra
54 http://www.mathstat.dal.ca/~selinger/
55 "QML", http://sneezy.cs.nott.ac.uk/QML/
56 http://philipmaymin.com/

```
import Quipper

spos :: Bool -> Circ Qubit
spos b = do
    q <- qinit b
    r <- hadamard q
    return r
```

(Fonte: Wikipedia, 2016)

7. Rilevamento e correzione dell'errore quantistico

Come illustrato precedentemente, il più grande ostacolo alla computazione quantistica è la decoerenza, non soltanto in dipendenza ai tempi di calcolo bensì all'introduzione di errore. Il calcolo quantistico, così come quello classico, non è esente dalla presenza di errori, dipendenti sia dalla perdita di coerenza naturale che da quella dovuta agli stati di sovrapposizione e anche alle possibili interferenze fisiche, elettromagnetiche e via discorrendo. Trovare il modo di rilevare e correggere l'errore nel calcolo quantistico è diventato uno dei principali scopi di ricerca per molti studiosi del calcolo quantistico. Secondo Williams e Clearwater [CITATION Wil00 \l 1040] sono quattro i metodi di approccio all'errore quantistico, i quali, in alcuni casi, ricalcano le procedure classiche.

Il primo metodo è detto *laissez-faire* ed è stato anticipato nei capitoli precedenti. Si tratta di un metodo passivo che parte dal presupposto che in un reale computer quantistico la decoerenza sia inevitabile. Con questa premessa l'unica possibilità che resta è quella di svolgere i calcoli prima che i *qubit* vadano incontro a possibili errori. È difatti il metodo d'azione più semplice e non prevede né la rilevazione né la correzione dell'errore in quanto tenta di aggirare il problema. Nella tabella 7.1 è possibile vedere la relazione tra le tecnologie di costruzione dei computer quantistici ed i dati riguardanti il tempo di coerenza, il massimo numero di *step* coerenti di calcolo possibili ed il tempo impiegato da ogni gate per eseguire un'operazione.

Il secondo metodo è quello della correzione di errore che a sua volta prevede diverse tecniche. Le più basilari sono quelle prese dal calcolo classico ed applicate al calcolo quantistico: l'esempio principale è dato dal cosiddetto voto di maggioranza. Si tratta di una tecnica basata sulla ridondanza che però, nel campo quantistico, non può sempre essere applicata dato che funziona soltanto se i *qubit* si trovano già in uno dei due stati logici perché, altrimenti, la ricerca della maggioranza altererebbe il valore della stessa. Da questa tecnica però ne deriva una più efficiente detta "simmetrizzazione" che prevede la presenza di *n* computer indipendenti che lavorano in parallelo sullo stesso calcolo quantico. Se nessun errore affligge nessun computer, ognuno svolge il calcolo secondo la medesima equazione di Schrödinger e di conseguenza, lo stato globale, calcolato come prodotto di tutti gli *n* stati, non cambia anche se vengono scambiati i computer. Questo significa che lo stato "corretto" si trova all'interno di uno spazio computazionale simmetrico. Ipotizzando invece la presenza di errore in un calcolatore, otterremo una rottura della simmetria che cambierà a seconda della posizione della macchina, ma in questo caso, proiettando il risultato errato sullo spazio di calcolo simmetrico, saremo in grado di determinare l'errore e correggerlo.

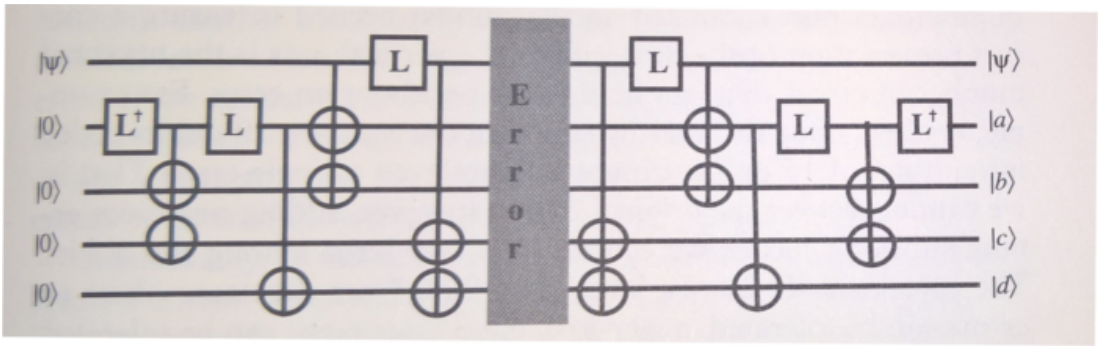
Figura 7.1 *Massimo numero di operazioni computabili senza perdere coerenza*

Quantum System	Time per Gate Operation (sec)	Coherence Time (sec)	Maximal Number of Coherent Steps
Electrons from a gold atom	10^{-14}	10^{-8}	10^{6}
Trapped indium atoms	10^{-14}	10^{-1}	10^{13}
Optical microcavity	10^{-14}	10^{-5}	10^{9}
Electron spin	10^{-7}	10^{-3}	10^{4}
Electron quantum dot	10^{-6}	10^{-3}	10^{3}
Nuclear spin	10^{-3}	10^{4}	10^{7}

(Fonte: [CITATION Wil00 \l 1040])

Della medesima categoria fanno parte anche i codici a correzione di errore derivati da quelli classici. Sostanzialmente vengono inseriti i *qubit* di informazione in delle *codeword* molto distinte tra loro in modo che, nel caso in cui il bit sia errato, sia possibile risalire alla *codeword* originale recuperando così l'informazione. Il problema di questo metodo risiede comunque nell'impossibilità di leggere lo stato errato senza farlo collassare. Questo ha rappresentato un problema fino a quando Shor non ha prodotto un codice che prevede l'*entanglement* del *qubit* di informazione con altri otto; in questo modo non è necessario conoscere l'errore, ma basta leggere i cosiddetti *qubit ancella* per comprendere il tipo di errore e decidere quale operazione applicare a quello contenente l'informazione, per rimuovere lo stato *bug*. Anche se il codice è stato migliorato da un gruppo di scienziati fino a ridurre il numero di "ancelle" a quattro, resta un altro problema, ovvero il fatto che il numero di errori tollerati da questa tecnica è davvero basso e diminuisce drasticamente all'aumentare del numero di *qubit* da controllare, poiché sarebbe richiesto un numero di "ancelle" così alto da vanificare l'efficienza del calcolo. Nell'immagine sottostante è presente un esempio di circuito che effettua la codifica e la decodifica dell'informazione per rilevarne l'errore: l'area grigia rappresenta il circuito nella quale è previsto il verificarsi di un errore.

Figura 7.2 *Esempio di circuito di rilevazione dell'errore con codice a quattro "ancelle"*



(Fonte:[CITATION Wil00 \l 1040])

Una terza tecnica appartenente a questa categoria è chiamata "a codici concatenati", i quali prevedono di legare ogni *qubit* ad un numero definito di simili in modo che ognuno di questi sia protetto dall'errore grazie ad un meccanismo analogo a quello precedentemente illustrato.

La terza categoria è quella dei computer a tolleranza di errore. Questo tipo di approccio è definito da cinque regole secondo John Preskill[57] del CIT[CITATION Pre97 \l 1040] che prevedono la creazione di computer in grado di resistere alla presenza di errori nei calcoli:

- Non utilizzare due volte lo stesso *qubit* ancella per prevenire la propagazione dell'errore;
- Copiare gli errori e non i dati ovvero preparare le ancelle in modo che leggendole si rilevi soltanto l'errore e non si influenzino i dati;
- Quando si effettua la codifica, ovvero si prepara l'*entanglement* tra *qubit* d'informazione e *qubit* ancella, bisogna assicurarsi di non aver commesso errori controllando uno stato quantistico conosciuto;
- Ripetere le operazioni per essere sicuri di non correggere uno stato corretto;
- Usare il codice corretto, perché non tutti sono compatibili con il tipo di operazioni svolte.

In ultimo troviamo la categoria della "topological quantum computer" che prevede un tipo di calcolo quantistico resistente, per sua natura, agli errori. Questo approccio è basato sulla non-località quantistica, in particolare sull'effetto Aharonov-Bohm che prevede, ad esempio, che un elettrone che passi attorno ad un tubo magnetico abbia, una volta raggiunta la sua destinazione, la stessa deviazione di fase. L'unica cosa che determina il cambiamento di fase è data dal numero di volte che l'elettrone gira attorno al tubo; questo porta a considerare che se il calcolo è basato sulla fase dell'elettrone, allora il calcolo non può essere affetto da disturbi.

57 http://www.theory.caltech.edu/~preskill/

8. Conclusioni

Dando speranza ai sogni di molti scienziati, il computer quantistico si è fatto strada tra le nuove tecnologie diventando una realtà tangibile. Un po' alla volta gli studiosi hanno smesso di chiedersi se la macchina quantistica sia effettivamente realizzabile, iniziando a domandarsi invece, quando e come questo strumento diverrà di uso comune. Citando Jungsang Kim, professore di computer engineering alla Duke University del Nord Carolina:

> "Negli anni Quaranta, i ricercatori avevano appena scoperto come usare le valvole a vuoto come semplici interruttori. Tali interruttori hanno poi dato vita ai gate logici, che potevano essere collegati assieme per formare i primi circuiti logici. Ed è qui che siamo arrivati con i processori quantistici. Abbiamo verificato che tutti i componenti funzionano. Il prossimo passo è quello di progettare il più piccolo, ancora più interessante circuito possibile." (Jungsang Kim[58], 2016)

L'eccitazione per la scoperta non deve però essere fuorviante ai fini di una valutazione oggettiva del fenomeno: è difficile che i computer quantistici rimpiazzino quelli classici, quantomeno nel futuro più prossimo. Di contro è inevitabile che, dimostrando prestazioni migliori in diversi campi, questi mezzi non vengano sfruttati appieno. Almeno in una prima fase l'ipotesi più verosimile sarebbe quella di utilizzare le macchine quantistiche come avveniva in passato per i centri di calcolo, dando in pasto ad esse determinati algoritmi per poi catalogare i risultati con i calcolatori più comuni. Molti degli attuali problemi e limiti dell'informatica sarebbero così abbattuti, permettendo lo spalancarsi di nuove porte. Non rimane che attendere i progressi tecnologici e le scoperte che porteranno alla nascita di nuovi componenti chiave per la progettazione delle macchine quantiche, fino a raggiungere la potenza e l'efficienza necessarie per un loro utilizzo costante. Le basi della teoria dell'informazione sono state già stabilite molto tempo fa, lasciando ai posteri l'arduo compito di provarle. Nel corso degli anni queste teorie sono state dimostrate una dopo l'altra, dando il via ad un processo di ottimizzazione inerente ai diversi ambiti di applicazione. È quindi indubbio che il computer quantistico sia una risorsa potente, in grado di far avanzare la scienza e l'ingegneria ad un nuovo livello. Resta da capire come, quando e se i problemi legati alla effettiva realizzazione di una macchina quantistica universale verranno risolti. I computer quantistici costruiti fino ad ora sono solo dei prototipi e non raggiungono molti degli obiettivi che il loro ramo di ricerca si prefigge, tuttavia molti scienziati sono d'accordo nel sostenere una nozione fondamentale: se non si riuscirà a realizzare un calcolatore del genere almeno si apprenderà perché ciò non è possibile.

58 http://ece.duke.edu/faculty/jungsang-kim

In conclusione, non si è in grado di sapere fino a dove l'informatica quantistica si spingerà, ma si può affermare per certo che grazie all'ausilio delle conoscenze e delle tecniche raggiunte nella costruzione delle macchine quantistiche, presto saranno disponibili strumenti in grado di ottimizzare la risoluzione di molti problemi di elaborazione dei dati. Un po' alla volta la barriera della decoerenza sarà abbattuta e le salde fondamenta di questo nuovo ambito scientifico saranno gettate, ed arriverà il momento in cui i fisici e i ricercatori passeranno il testimone agli esperti di informatica e mentre i primi continueranno ad indagare nuove possibilità, i secondi utilizzeranno ciò che fino ad allora sarà stato reso disponibile, scrivendo, bit dopo bit, il codice del futuro della specie umana.

9. Bibliografia

[1] D. Deutsch, The Fabric of Reality, USA: Viking Adult, 1997.

[2] C. P. Williams e S. H. Clearwater, Ultimate zero and one: computing at the quantum frontier, New York: Copernicus, 2000.

[3] C. P. Williams e S. H. Clearwater, Explroration in quantum computing, New York: Telos, 1998.

[4] M. Kaku, Fisica dell'impossibile. Un'esplorazione scientifica nel mondo dei phaser, dei campi di forza, del teletrasporto e dei viaggi nel tempo, New York: Codice, 2008.

[5] P. Benioff, «The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines,» *Journal of Statistical Physics,* vol. 22, 1980.

[6] Y. I. Manin, «Computable and Noncomputable,» *Sov.Radio,* 1980.

[7] R. P. Feynman, «Simulating Physics with Computers,» *International Journal of Theoretical Physic,* vol. 21, 1981.

[8] D. Deutsch, «Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer,» *Proceedings of the Royal Society of London A,* vol. 400, 1985.

[9] R. Jozsa, «Characterizing Classes of Functions Computable by Quantum Paralelism,» *Proceedings Royal Society London A,* vol. 435, 1991.

[10] T. Rønnow F., Z. Wang, J. Job, S. Boixo, S. Isakov V., D. Wecker, J. Martinis M., D. Lidar A. e M. Troyer, «Defining and detecting quantum speedup,» *Science,* vol. 345, 2014.

[11] D. Bouwmeester, A. Ekert e A. Zeilinger, The physics of quantum information: quantum cryptography, quantum teleportation, quantum computation, Vienna: Springer, 2000.

[12] k. L. Grover, «A fast quantum mechanical algorithm for database search,» *Cornell University Library,* 1996.

[13] Brickman, Haljan, Lee, Acton, Deslauriers e Monroe, «Implementation of Grover's quantum search algorithm in a scalable system,» *Physical Review A ,* vol. 27, 2005.

[14] M. Rastegari, «Quantum approach to Image processing,» *Shomal University of Amol,* 2007.

[15] S. Caraiman e M. Vasile, «Image processing using quantum computing,» *IEEE Xplore,* 2012.

[16] D. Hautle, G. Genazzi e C. Ferrari, «Quantum computer,» Liceo Locarno, 2009.

[17] R. Dridi e H. Alghassi, «Prime factorization using quantum annealing and computational algebraic geometry,» *1QB Information Technologies (1QBit),* 2016.

[18] C. P. Williams, Explorations in Quantum Computing, London: Springer-Verlag, 2011.

[19] M. M. M. Elhoushi, «Modeling a Quantum Computer,» *Ain Shams University,* 2011.

[20] D. Unruh, «Quantum programming languages,» *Informatik – Forschung und Entwicklung,* 2006.

[21] J. S. Gay, «Quantum programming languages: Survey and Bibliography,» *Department of Computing Science,* 2006.

[22] R. Rüdiger, «Quantum Programming Languages: An Introductory Overview,» *Oxford University Press,* 2006.

[23] J. Preskill, «Fault-tolerant quantum computation,» 1997.

[24] C. H. Bennet, E. Bernstein, G. Brassard e U. Vazirani, «The strengths and weakness of quantum computation,» *SIAM Journal on Computing,* 1997.