Alma Mater Studiorum · Università di Bologna

Scuola di Scienze
Dipartimento di Fisica e Astronomia
Corso di Laurea in Fisica

Quantum Error Correction e Toric Code

Relatore:

Prof.ssa
Elisa Ercolessi

Presentata da:
Andrea Pondini

Anno Accademico 2019/2020

Abstract

L'elaborato studia la Quantum Error Correction, ovvero quella branca del Quantum Computing che studia gli errori nella computazione e come correggerli. Questo campo è di fondamentale importanza nella costruzione di computer quantistici, in cui l'interazione con l'ambiente porta rumore alla computazione e perdita di coerenza degli stati del sistema. Particolare attenzione è posta nello studio degli Stabilizer Codes, una particolare categoria di Quantum Error Correcting Codes. Tra questi si studia il Toric Code, esempio peculiare di stabilizer code ordinato topologicamente. Le peculiarità del codice sono conseguenza della sua definizione su un reticolo immerso in una superficie toroidale, come suggerisce il nome.

Indice

Introduzione

I computer quantistici sembrano promettere un'enorme svolta al modo in cui si risolvono problemi attraverso computer. Essi si basano su qubit i quali, al contrario dei bit classici che possono essere solo 0 o 1, possono essere in sovrapposizioni di stati [6]. Questa particolare sovrapposizione termina col collassare su un particolare stato al momento della misura. Questa sostanziale differenza è ciò che rende profondamente diversi tutti gli aspetti di un computer quantistico dal corrispettivo classico. La natura quantistica delle operazioni in questi processi permette algoritmi profondamente differenti da quelli classici che sbloccano possibilità e risultati di particolare interesse [15]. Purtroppo gli stati dei qubit sono fragili e, siccome in pratica non esistono sistemi isolati, interagendo con l'ambiente che li circonda possono essere disturbati. Dunque l'ostacolo principale che blocca la realizzazione di computer quantistici con un numero elevato di qubit è l'incorrere degli errori in computazione in seguito a stati corrotti dalle interazione con l'esterno.

Anche nella computazione classica questo succede, ed il problema è risolto codificando le informazioni ridondantemente in modo che nell'eventualità di errori si possa risalire comunque all'informazione codificata attraverso algoritmi e protocolli specifici. Non è possibile fare lo stesso nel caso quantistico soprattutto a causa del No Cloning Theorem [13]. La Quantum Error Correction studia gli errori nella computazione quantistica, così come le condizioni e i modi di correggerli. Un caso particolare di Error Correcting Code sono gli Stabilizer Codes, codici composti da stati che rimangono invariati sotto determinate trasformazioni.

Viene presentato l'esempio del Toric Code, uno stabilizer code definito secondo un ordine topologico attraverso complessi di catene Z_2 su un reticolo immerso in una superficie toroidale. Lo spettro energetico del codice può essere descritto conoscendo le caratteristiche dello stato fondamentale e tenendo conto delle eccitazioni del reticolo su cui il codice è definito. Queste eccitazioni possono essere identificate come creazione e distruzione di quasiparticelle, che sono trasportate lungo il reticolo da operatori di trasporto. Il codice è rilevante in quanto presenta caratteristiche che sono conseguenza diretta della definizione sulla semplice superficie del toro. Il Toric Code è una delle principali alternative tra i codici di correzione di errori da implementare nella costruzione dei moderni computer quantistici.

Nel Capitolo 1 vengono poste le basi della computazione quantistica introducendo i qubit e i gates agenti su di essi. Nel Capitolo 2 viene descritto il formalismo per descrivere gli errori e definiti gli Stabilizer Codes. Infine nel Capitolo 3 viene descritto il Toric Code.

Capitolo 1

Introduzione alla Computazione Quantistica

In questo capitolo verranno introdotti i concetti fondamentali della computazione quantistica, a partire dai qubit e dalle possibili operazioni attraverso gates. Verrà introdotto il procedimento di misura su un codice quantistico e poi studiata l'universalità di set discreti di gates. Infine si studierà come poter approssimare un generico codice quantistico attraverso il cosiddetto set standard. Per una trattazione più esaustiva si rimanda a [15], [4].

1.1 Operazioni su qubit singolo

Un *qubit* è un vettore $|\psi\rangle$ in $\mathbb{C}^2$

$$|\psi\rangle = a\,|0\rangle + b\,|1\rangle \tag{1.1.1}$$

parametrizzato da due costanti complesse tale che $|a|^2 + |b|^2 = 1$. La base $\{|0\rangle, |1\rangle\}$ è detta *base computazionale* ed è definita:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix},\ |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{1.1.2}$$

Al contrario di un bit classico, vincolato univocamente agli stati 0 *o* 1, un qubit è in sovrapposizione degli stati citati: se si effettua una misu-

ra nella base computazionale si ha la possibilità di ottenere i risultati 0 e 1 rispettivamente con probabilità $|a|^2$ e $|b|^2$. Un singolo qubit può essere immaginato come un punto (θ, φ) su una sfera di raggio unitario, dove $a = \cos(\theta/2)$, $b = e^{i\varphi}\sin(\theta/2)$, con una fase complessiva dello stato che viene trascurata in quanto non rilevante. Questa rappresentazione è chiamata rappresentazione sferica di Bloch (Fig. 1.1).

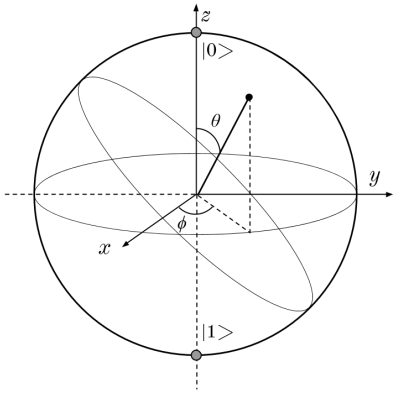


Figura 1.1.1: Rappresentazione sferica di Bloch di un qubit. Il qubit è individuato da un punto sulla sfera unitaria parametrizzato dagli angoli (θ, φ).

Le operazioni su un qubit devono mantenere la norma del vettore pari a 1 e dunque appartengono al gruppo delle trasformazioni $U(2)$. Tra queste le più importanti sono le matrici di Pauli, che vengono mostrate di seguito:

$$X \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix};\ Y \equiv \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix};\ Z \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{1.1.3}$$

Altri quantum gates di rilevante interesse sono rappresentati dal gate di Hadamard (denotato con H) e il gate di fase (denotato con S):

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix};\ S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}. \tag{1.1.4}$$

Si può verificare la validità delle seguenti *code identities*

$$HXH = Z;\ HYH = -Y;\ HZH = X. \tag{1.1.5}$$

La base $\left\{|+\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}, |-\rangle = \frac{|0\rangle-|1\rangle}{\sqrt{2}}\right\}$ costituita dagli autovettori dell'operatore H sarà di particolare interesse nelle trattazioni future.

Gli esponenziali di matrice delle matrici di Pauli danno luogo a rotazioni nello spazio tridimensionale rispettivamente intorno all'asse x, y, z:

$$R_x(\theta) \equiv e^{-i\theta X/2} = \begin{bmatrix} \cos\theta/2 & -i\sin\theta/2 \\ -i\sin\theta/2 & \cos\theta/2 \end{bmatrix} \tag{1.1.6}$$

$$R_y(\theta) \equiv e^{-i\theta Y/2} = \begin{bmatrix} \cos\theta/2 & -\sin\theta/2 \\ -\sin\theta/2 & \cos\theta/2 \end{bmatrix} \tag{1.1.7}$$

$$R_z(\theta) \equiv e^{-i\theta Z/2} = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}. \tag{1.1.8}$$

Ricordando che $U(2) = \langle SU(2) \cup U(1) \rangle$ [7], si enuncia il seguente teorema:

Teorema 1.1.1 (Scomposizione $Y - Z$ per un singolo qubit)**.** Sia U un operatore unitario generico agente su un singolo qubit. Esistono allora i numeri reali α, β, γ e δ tale che

$$U = e^{i\alpha} R_z(\beta) R_y(\gamma) R_z(\delta).$$

Corollario 1.1.2. Sia U un operatore unitario generico agente su un singolo qubit. Esistono allora gli operatori unitari A, B, C agenti su un singolo qubit tale che $ABC = I$ e

$$U = e^{i\alpha} AXBXC$$

con α fattore di fase.

Per rappresentare circuiti quantistici è comune rappresentare i qubit con fili e i gate con box contrassegnati dalla trasformazione applicata. Se ne può vedere un esempio in Figura 1.1.2 [11]. Il tempo scorre convenzionalmente da sinistra verso destra.

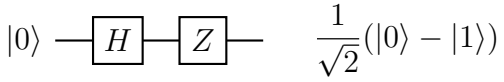


Figura 1.1.2: Circuito in cui un qubit nello stato $|0\rangle$ attraversa un Hadamard gate e in seguito un gate Pauli-Z trasformandosi nello stato $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.

1.2 Operazioni su qubit multipli

Uno stato a n-qubit può essere descritto come combinazione lineare del prodotto tensoriale delle basi degli stati dei singoli qubit

$$|\Psi\rangle = \sum_{i_1,i_2,\ldots,i_n} C_{i_1 i_2 \ldots i_n} |i_1 i_2 \ldots i_n\rangle \,, \tag{1.2.1}$$

dove $i_k = 0, 1$ e $|i_1, i_2, \ldots, i_n\rangle \equiv |i_1\rangle \otimes |i_2\rangle \otimes \ldots \otimes |i_n\rangle$. Un gate che agisce sul k-esimo singolo qubit si può dunque esprimere

$$A_k = \underbrace{I \otimes \ldots \otimes I}_{k-1} \otimes A \otimes \underbrace{I \otimes \ldots \otimes I}_{n-k-1} . \tag{1.2.2}$$

Il primo esempio di gate agente su multipli qubit è il *controlled not gate* (CNOT). Questo gate ha come input due qubit, quello di *controllo* e quello *target*: come suggerisce il nome, se il qubit di controllo si trova nello stato $|1\rangle$ il target quibit subisce un "flip". La rappresentazione matriciale con $n = 2$ in base computazionale e la rappresentazione grafica del gate CNOT sono rappresentati in Figura 1.2.1.

Si premette che per semplicità di notazione in generale nello scrivere gli stati di n-qubit viene omesso il simbolo di prodotto tensoriale: $|k_1\rangle \ldots |k_n\rangle \equiv |k_1\rangle \otimes \ldots \otimes |k_n\rangle$.

Può sorgere spontaneo il dubbio su quale sia l'effetto del gate CNOT su una base che non sia quella computazionale. Viene riportata la trasforma-

$$\Lambda_{1,2}(X) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

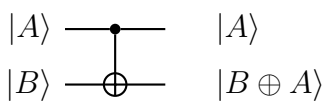


Figura 1.2.1: Rappresentazione matriciale e grafica del gate CNOT. Il simbolo $\oplus$ indica la somma con modulo 2: si può considerare il controlled not gate come un gate che applica lo XOR sui qubit $|A\rangle$ e $|B\rangle$ e ne conserva il risultato nel target qubit.

zione sulla base $\{|+\rangle, |-\rangle\}$:

$$|+\rangle|+\rangle \rightarrow |+\rangle|+\rangle \tag{1.2.3}$$
$$|-\rangle|+\rangle \rightarrow |-\rangle|+\rangle \tag{1.2.4}$$
$$|+\rangle|-\rangle \rightarrow |-\rangle|-\rangle \tag{1.2.5}$$
$$|-\rangle|-\rangle \rightarrow |+\rangle|-\rangle \tag{1.2.6}$$

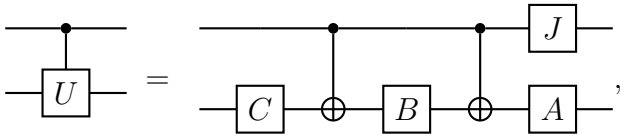
Si noti come i target e control qubit si siano scambiati i ruoli: avviene infatti un flip di fase al control qubit solo nel caso in cui il target qubit sia nello stato $|-\rangle$.

In generale, per qualsiasi operatore unitario U è possibile definire il gate *controlled-U*

$$\Lambda_{c,t}(U) = |0\rangle\langle 0|_c I_t + |1\rangle\langle 1|_c U_t, \tag{1.2.7}$$

dove c, t indicano rispettivamente il control e il target qubit. Ricordando il

Corollario 1.1.2, si può esprimere un generico $\Lambda(U)$ come:



(1.2.8)

dove $J = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{bmatrix}$.

Si introduce ora il Toffoli gate, che permette di fare un parallelo con la computazione classica. La rappresentazione grafica è visibile in Figura 1.2.2 ed è descritto dal seguente operatore:

$$\Lambda^2_{c_1,c_2,t}(X) = (I_{c_1} I_{c_2} - |1\rangle\langle 1|_{c_1} |1\rangle\langle 1|_{c_2}) I_t + |1\rangle\langle 1|_{c_1} |1\rangle\langle 1|_{c_2} X_t. \tag{1.2.9}$$

Per un input in base computazionale del tipo $|i_1\rangle_{c_1} |i_2\rangle_{c_2} |j\rangle_t$ il Toffoli gate

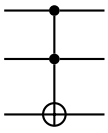


Figura 1.2.2: Rappresentazione grafica del Toffoli gate.

applica una trasformazione del tipo

$$\Lambda^2_{c_1,c_2,t}(X) |i_1\rangle_{c_1} |i_2\rangle_{c_2} |j\rangle_t = |i_1\rangle_{c_1} |i_2\rangle_{c_2} |j \oplus (i_1 \cdot i_2)\rangle_t . \tag{1.2.10}$$

Si noti come nel caso $|j\rangle = |1\rangle$ il risultato del target qubit sia equivalente all'operazione classica NAND tra i due control qubit. Questa porta logica è ritenuta fondamentale in quanto qualsiasi funzione booleana è realizzabile attraverso quest'ultima. Un altro caso di particolare interesse è quello mostrato in Figura 1.2.3, dove si mostra come attraverso il Toffoli gate è possibile riprodurre l'operazione classica di FANOUT. Infatti $\Lambda^2_{c_1,c_2,t}(X) |1\rangle_{c_1} |x\rangle_{c_2} |0\rangle_t = |1\rangle_{c_1} |x\rangle_{c_2} |x\rangle_t$. Questa operazione non viola il no-cloning Theorem [13] in quanto lo stato $|x\rangle$ è un stato copiato grazie ad un

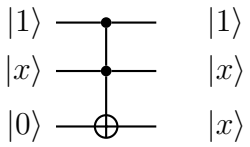


Figura 1.2.3: Operazione di FANOUT attraverso il Toffoli gate. Dopo aver applicato il gate lo stato del secondo control bit è riprodotto anche nel target qubit.

particolare input specificatamente preparato, che necessita che il primo control qubit sia nello stato $|1\rangle$ e che il target qubit sia nello stato $|0\rangle$, tutt'altro che uno stato arbitrario. Essendo gli operatori unitari invertibili, attraverso il Toffoli gate la computazione quantistica può simulare la computazione classica in modo reversibile.

1.3 Misura quantistica

Operatori $\{M_m\}$ sono detti *operatori di misura* e agiscono sullo spazio degli stati del sistema di cui si effettua la misura. Sia $|\psi\rangle$ lo stato del sistema, allora la probabilità che si presenti il risultato m è

$$p(m) = \langle\psi| M_m^\dagger M_m |\psi\rangle \tag{1.3.1}$$

e lo stato successivo alla misura

$$\frac{M_i |\psi\rangle}{\sqrt{\mathrm{Tr}\left[M_i^\dagger M_i |\psi\rangle \langle\psi|\right]}}. \tag{1.3.2}$$

Gli operatori di misura soddisfano la relazione di completezza

$$\sum_m M_m^\dagger M_m = I \tag{1.3.3}$$

assicurando dunque $\sum_m p(m) = 1$.

Due stati non sono distinguibili a meno che essi non siano ortogonali. Siano $|\psi_1\rangle$ e $|\psi_2\rangle$ due stati non ortogonali, allora $|\psi_1\rangle$ può essere decomposto

in una componente non nulla parallela a $|\psi_2\rangle$ e dunque effettuando misure su $|\psi_1\rangle$ si otterranno risultati analoghi a misure su $|\psi_2\rangle$ con probabilità non nulla.

Si può definire

$$E_m \equiv M_m^\dagger M_m, \tag{1.3.4}$$

allora secondo la (1.3.1) vale $\sum_m E_m = I$ e $p(m) = \langle\psi| E_m |\psi\rangle$. Il set completo $\{E_M\}$ è detto POVM (positive-operator-valued measure) e i singoli E_m elementi POVM.

Sia M un *osservabile* [6], ovvero un operatore hermitiano agente sullo stato degli spazi del sistema in considerazione. Allora M può essere decomposto spettralmente

$$M = \sum_m m P_m, \tag{1.3.5}$$

dove P_m è il proiettore sull'autospazio associato all'autovalore m di M. I possibili risultati di una misura corrispondono agli autovalori m e si presentano con probabilità

$$p(m) = \langle\psi| P_m |\psi\rangle \tag{1.3.6}$$

e lo stato successivo alla misura

$$\frac{P_m |\psi\rangle}{\sqrt{p(m)}}. \tag{1.3.7}$$

Questo tipo di misura è detta *misura di proiezione*.

L'atto di misura è rappresentato graficamente in Figura 1.3.1 e in generale si assume che al termine di un circuito quantistico i fili vengano misurati. In generale dopo l'atto di misura si perde la sovrapposizione di stati e il bit può essere considerato di tipo classico, rappresentato da una doppia linea. Le misure in computazione quantistica sono di vitale importanza in quanto non sono utilizzate solo per ottenere informazioni sugli stati dei qubit ma anche come metodo di interferenza sul circuito, per esempio per scegliere quale gate utilizzare.

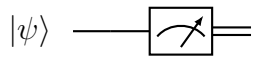


Figura 1.3.1: Rappresentazione grafica dell'atto di misura. In generale si assume che al termine di ogni circuito i fili vengano misurati e quindi il simbolo del misuratore viene omesso.

1.4 Quantum gates universali

Così come le porte logiche AND, OR, NOT costituiscono un set universale per realizzare qualsiasi funzione in computazione classica, si cerca ora un set discreto di quantum gates *universali* per la computazione quantistica. Con set di quantum gate universali si intende un set che permette di realizzare qualsiasi funzione con livello di precisione arbitraria.

1.4.1 Two-level operators universali

Sia U un operatore unitario su uno spazio di Hilbert n-dimensionale. Si chiamano *two-level operators* unitari gli operatori unitari che agiscono solo su due o meno componenti di un vettore e dunque in computazione quantistica sono spesso utilizzati per indicare trasformazioni che interessano un singolo qubit.

Esempio 1.4.1. Si veda il caso in cui U ha dimensione 3×3 e dunque un forma del tipo

$$U = \begin{bmatrix} a & d & g \\ b & e & h \\ c & f & j \end{bmatrix}. \tag{1.4.1}$$

Si cercano i two-level operators unitari U_1, U_2, U_3 tale che $U_1 U_2 U_3 U = I$ da cui segue

$$U = U_1^\dagger U_2^\dagger U_3^\dagger. \tag{1.4.2}$$

Essendo U_1, U_2, U_3 two-level operators unitari, lo saranno anche $U_1^\dagger U_2^\dagger U_3^\dagger$ e dunque l'eq.1.4.2 individua una scomposizione di U in two-level operators

unitari. Si veda ora come costruire U_1, U_2, U_3 nel caso particolare citato (1.4.1). Se $b = 0$ si imposterà $U_1 \equiv I$, nel caso contrario

$$U_1 \equiv \begin{bmatrix} \frac{a^*}{\sqrt{|a|^2+|b|^2}} & \frac{b^*}{\sqrt{|a|^2+|b|^2}} & 0 \\ \frac{b}{\sqrt{|a|^2+|b|^2}} & \frac{-a}{\sqrt{|a|^2+|b|^2}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.4.3}$$

ottenendo in entrambi casi

$$U_1 U = \begin{bmatrix} a' & d' & g' \\ 0 & e' & h' \\ c' & f' & j' \end{bmatrix}. \tag{1.4.4}$$

La procedura per la costruzione di U_2 è molto simile: se $c' = 0$ si imposta

$$U_2 \equiv \begin{bmatrix} a'^* & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.4.5}$$

altrimenti

$$U_2 \equiv \begin{bmatrix} \frac{a'^*}{\sqrt{|a'|^2+|c'|^2}} & 0 & \frac{c'^*}{\sqrt{|a'|^2+|c'|^2}} \\ 0 & 1 & 0 \\ \frac{c'}{\sqrt{|a'|^2+|c'|^2}} & 0 & \frac{-a'}{\sqrt{|a'|^2+|c'|^2}} \end{bmatrix}, \tag{1.4.6}$$

ottenendo

$$U_2 U_1 U = \begin{bmatrix} 1 & d'' & g'' \\ 0 & e'' & h'' \\ 0 & f'' & j'' \end{bmatrix}. \tag{1.4.7}$$

In quanto sia U, U_1 e U_2 sono operatori unitari, segue che $d'' = g'' = 0$. Infine si pone

$$U_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & e''^* & f''^* \\ 0 & h''^* & j''^* \end{bmatrix} \tag{1.4.8}$$

rendendo immediata la verifica $U_3U_2U_1U = I$ e la scomposizione di U in two-level operators unitari.

Generalizzando il processo al caso n-dimensionale, si esegue un procedimento simile a quello descritto nell'Esempio 1.4.1 per scrivere

$$U = V_1 \dots V_k \tag{1.4.9}$$

con V_i two-level operators unitari e $k \leq n(n-1)/2$. Un corollario del precedente risultato è che che un operatore unitario arbitrario agente su n qubit può essere decomposto come prodotto di al massimo $2^{n-1}(2^n - 1)$ two-level operators unitari. Quello appena citato è un limite superiore, tanto che nella maggior parte dei casi si ottengono scomposizioni molto più efficienti. Quanto visto permette di scomporre una generica trasformazione come prodotto di trasformazioni su singoli qubit.

1.4.2 Universalità del gate CNOT e dei gate agenti su singolo qubit

Ora si mostrerà come una generica trasformazione unitaria U possa essere scomposta attraverso gate CNOT e gate agenti su singoli qubit. Siano s e t due numeri binari distinti, si dice *Grey code* una sequenza di numeri binari che inizia con s ed è conclusa da t i cui componenti differiscono per un solo bit.

Esempio 1.4.2. Sia $s = 0001$ e $t = 0100$, allora il Grey code che congiunge s con t è

$$\begin{gathered} 0001 \\ 0010 \\ 0011 \\ 0100 \end{gathered} \tag{1.4.10}$$

Sia U un two-level operator unitario agente su n qubits. Si suppone che U agisca non trivialmente solo sullo spazio generato dagli stati $|s\rangle$ e

$|t\rangle$, con $s = s_1 \dots s_n$ e $t = t_1 \dots t_n$ le espansioni binarie di s e t. Sia $\tilde{U}$ la rappresentazione 2×2 unitaria di U nello spazio generato dagli stati $|s\rangle$ e $|t\rangle$. Siano $g_1 \dots g_m$ gli elementi del Grey code con $g_1 = s$ e $g_m = t$. I seguenti passaggi permettono di scomporre U in un gate agente su qubit singolo e gates CNOT.

Si inizia con lo scambiare gli stati $|g_1\rangle$ e $|g_2\rangle$. Supponendo che g_1 e g_2 differiscano per un bit in posizione i, allora lo scambio verrà raggiunto effettuando un flip controllato dell' i-esimo qubit, con la condizione che tutti gli altri bit di g_1 e g_2 coincidano. Si ripete il procedimento fino ad effettuare lo scambio tra $|g_{m-2}\rangle$ e $|g_{m-1}\rangle$. Come risultato effettivo si avrà una sequenza di $m-2$ operazioni che spostano lo stato $|g_{k+1}\rangle$ in $|g_k\rangle$ $(k = 1, \dots, m-2)$ e $|g_1\rangle$ in $|g_{m-1}\rangle$. Si suppone ora che $|g_{m-1}\rangle$ e $|g_m\rangle$ differiscano per il j-esimo bit. Si applica dunque il gate controlled-$\tilde{U}$ sul target qubit j, con la condizione che tutti gli altri qubit di $|g_{m-1}\rangle$ e $|g_m\rangle$ coincidano. Per concludere si riportano gli stati all'ordine iniziale effettuando il procedimento inverso rispetto a quello descritto in precedenza. In questo modo si è scomposto il gate U in un gate agente su singolo qubit e a gate del tipo CNOT.

Esempio 1.4.3. Per un operatore agente solo sullo spazio generato da $\{|000\rangle, |111\rangle\}$ la scomposizione viene rappresentata in Fig.1.4.1.

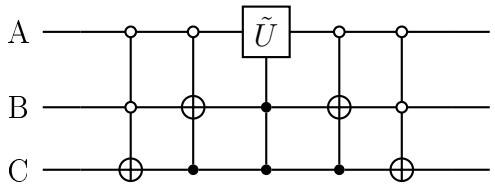


Figura 1.4.1: Scomposizione di un operatore agente sullo spazio generato da $\{|000\rangle, |111\rangle\}$ in un gate su qubit singolo e gates CNOT

Ognuna delle controlled-oprations citate necessitano $O(n)$ gates, e lo stesso vale per l'operazione controlled-U. Questo implica che l'implementazione di U necessita $O(n^2)$ gates. Ricordando le conclusioni della sottosezio-

ne 1.4.1, una generica trasformazione può essere scritta come prodotto di $O(2^{2n}) = O(4^n)$ two-level operators, portando il numero di gates complessivi necessari a $O(4^n n^2)$.

1.4.3 Set discreti di gates universali

Non è possibile che un set discreto di gates universali possa riprodurre un gate generico *esattamente* in quanto gli operatori unitari sono un gruppo continuo [7]. Si parla quindi di *approssimare* efficacemente qualsiasi gate attraverso un set discreto di operatori universali.

Siano U, V due operatori unitari agenti sullo stesso spazio di Hilbert. U è il gate che si intende introdurre nel circuito quantistico e V la sua approssimazione, ciò che verrà effettivamente implementato in pratica. Si dice *errore* la quantità

$$E(U,V) \equiv \max_{|\psi\rangle} \|(U-V)|\psi\rangle\|, \tag{1.4.11}$$

dove si prende il massimo su tutti gli stati dello spazio.

Sia M un elemento POVM (1.3.4) e p_U e p_V le probabilità di ottenere come risultato M nel caso fossero applicati rispettivamente U o V ad un qubit nello stato iniziale $|\psi\rangle$. Si definisce $|\Delta\rangle \equiv (U-V)|\psi\rangle$. Allora

$$\begin{aligned}
|p_u - p_V| &= |\langle\psi| U^\dagger M U |\psi\rangle - \langle\psi| V^\dagger M V |\psi\rangle| && (1.4.12)\\
&= |\langle\psi| U^\dagger M |\Delta\rangle + \langle\Delta| M V |\psi\rangle| && (1.4.13)\\
&\leq |\langle\psi| U^\dagger M |\Delta\rangle| + |\langle\Delta| M V |\psi\rangle| && (1.4.14)\\
&\leq \||\Delta\rangle\| + \||\Delta\rangle\| && (1.4.15)\\
&\leq 2E(U,V), && (1.4.16)
\end{aligned}$$

dove nel passaggio 1.4.14 si è utilizzata la disuguaglianza di Cauchy-Schwarz. Dunque la probabilità di ottenere lo stesso risultato è maggiore se l'errore $E(U,V)$ diminuisce.

Si supponga ora di applicare una serie di gates $V_1, \ldots, V_m$ nel tentativo di approssimare la serie di gates $U_1, \ldots, U_m$. Per un generico stato $|\psi\rangle$ vale

$$\begin{aligned} E(U_2U_1, V_2V_1) &= \|(U_2U_1 - V_2V_1)|\psi\rangle\| && (1.4.17)\\ &= \|(U_2U_1 - V_2U_1)|\psi\rangle + (V_2U_1 - V_2V_1)|\psi\rangle\| && (1.4.18)\\ &\leq \|(U_2U_1 - V_2U_1)|\psi\rangle\| + \|(V_2U_1 - V_2V_1)|\psi\rangle\| && (1.4.19)\\ &\leq E(U_2, V_2) + E(U_1, V_1), && (1.4.20) \end{aligned}$$

dove nel passaggio 1.4.19 si è utilizzata la disuguaglianza triangolare. Procedendo per induzione si verifica

$$E(U_m \ldots U_1, V_m \ldots V_1) \leq \sum_{j=1}^{m} E(U_j, V_j). \qquad (1.4.21)$$

Secondo le disuguaglianze (1.4.16) e (1.4.21), in un circuito in cui si approssimano m gates U_j con m gates V_j è sufficiente che $E(U_j, V_j) \leq \delta/(2m)$ perché le probabilità dei risultati in seguito alla misura differiscano al massimo $\delta > 0$ da quelle corrette[1].

1.4.4 Set standard

Si definisce il gate $\pi/8$, anche detto gate T

$$\pi/8 \equiv \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} = e^{i\pi/8} \begin{bmatrix} e^{-i\pi/8} & 0 \\ 0 & e^{i\pi/8} \end{bmatrix}. \qquad (1.4.22)$$

Il set discreto costituito dal gate di Hadamard, gate di fase, gate CNOT e gate $\pi/8$ è detto *set standard* e si mostrerà ora come attraverso esso sia possibile approssimare qualsiasi tipo di gate agente su singolo qubit.

Ricordando le definizioni (1.1.6), (1.1.8), si può verificare $T = R_z(\pi/4)$, $HTH = R_x(\pi/4)$ a meno di un fattore di fase globale. Componendo le due trasfor-

[1] Con probabilità corrette si intendono le probabilità nel caso in cui nel circuito quantistico si fossero utilizzati i gate non approssimati U_j.

mazioni

$$\begin{aligned} THTH &= R_z(\pi/4)R_x(\pi/4) && (1.4.23)\\ &= \exp(-i\frac{\pi}{8}Z)\exp(-i\frac{\pi}{8}X) && (1.4.24)\\ &= (\cos(\frac{\pi}{8}I) - i\sin(\frac{\pi}{8}Z))(\cos(\frac{\pi}{8}I) - i\sin(\frac{\pi}{8}X)) && (1.4.25)\\ &= \cos^2(\frac{\pi}{8}I) - i(\cos(\frac{\pi}{8}(X+Z)) + \sin(\frac{\pi}{8}Y))\sin\frac{\pi}{8} && (1.4.26)\\ &= R_{\hat{n}}(\theta), && (1.4.27) \end{aligned}$$

dove $\hat{n} = (\cos(\pi/8), \sin(\pi/8), \cos(\pi/8))$ e $\cos(\theta/2) = \cos^2(\pi/8)$. Dunque è possibile costruire $R_{\hat{n}}(\theta)$ attraverso i gate H e $\pi/8$, inoltre si può dimostrare che θ è un multiplo irrazionale di 2π.

In Appendice A si dimostra come attraverso $R_{\hat{n}}(\theta)$ si possa approssimare con precisione arbitraria $R_{\hat{n}}(\alpha)$. Per ogni $\epsilon > 0$ esiste n per cui

$$E(R_{\hat{n}}(\alpha), R_{\hat{n}}(\theta)^n) < \frac{\epsilon}{3}. \tag{1.4.28}$$

Inoltre

$$HR_{\hat{n}}(\alpha)H = R_{\hat{m}}(\alpha), \tag{1.4.29}$$

dove $m = (\cos(\pi/8), -\sin(\pi/8), \cos(\pi/8))$, da cui segue

$$E(R_{\hat{m}}(\alpha), R_{\hat{m}}(\theta)^n) < \frac{\epsilon}{3}. \tag{1.4.30}$$

Inerpretando il risultato del Teorema (1.1.1) possiamo scrivere un operatore unitario arbitrario come

$$U = R_{\hat{n}}(\beta)R_{\hat{m}}(\gamma)R_{\hat{n}}(\zeta) \tag{1.4.31}$$

trascurando un fattore di fase globale. Dati tre interi positivi n_1, n_2, n_3, ricordando la disuguaglianza (1.4.21), possiamo finalmente affermare

$$E(U, R_{\hat{n}}(\theta)^{n_1}HR_{\hat{n}}(\theta)^{n_2}HR_{\hat{n}}(\theta)^{n_3}) < \epsilon. \tag{1.4.32}$$

Ciò significa che qualsiasi operatore unitario agente su un singolo qubit può essere approssimato entro ϵ usando un circuito composto solamente da Hadamard gates e gates $\pi/8$.

1.4.5 Teorema di Solovay-Kitaev

Secondo le conclusioni tratte nelle Sezioni (1.4.2) e (1.4.4) si può finalmente approssimare un circuito quantistico composto da m gates. Secondo la disuguaglianza (1.4.21), per ottenere un'accurateza ϵ sull'intero circuito è necessario approssimare ogni gate con un accuratezza entro ϵ/m.

Può sorgere ora il dubbio di quanto possa essere efficiente una procedura di questo genere. Si enuncia il seguente teorema:

Teorema 1.4.4 (Teorema di Solovay-Kitaev)**.** Qualsiasi circuito quantistico composto da m CNOT gates o gates agenti su singolo qubit può essere approssimato usando $O(m \log^c(m/\epsilon))$ gates del set standard.

Per quanto riguarda il fine di questo testo, la dimostrazione è omessa e c viene considerata una costante circa pari a 2 [3]. Il precedente teorema indica dunque una crescita logaritmica del numero di gate necessari per approssimare un qualsiasi circuito quantistico, un andamento che permette di "scalare" il procedimento a circuiti complessi.

Capitolo 2

Quantum Errors e Stabilizer codes

In questo capitolo verrà introdotto il concetto di errori nella computazione, presentandone il formalismo matematico necessario per descriverli e per implementare codici in grado di correggerli. Inizialmente vengono descritte le proprietà della matrice densità di un sistema, uno strumento particolarmente adatto al descrivere sistemi aperti. Dunque vengono descritte le quantum operations, utilizzate per descrivere i processi che introducono errori nella computazione, e presentati alcuni esempi. Finalmente si introducono i concetti generali della quantum-error correction e il caso particolare degli stabilizer codes. Per una trattazione più esaustiva si rimanda a [15], [4].

2.1 Matrice densità

Sia $|\psi_k\rangle$ un set di stati possibili di un dato sistema e p_k la loro rispettiva probabilità. L'insieme $\{p_k, |\psi_k\rangle\}$ è detto *ensamble di stati puri* e si definisce *matrice densità* l'operatore

$$\rho = \sum_k p_k \left|\psi_k\right\rangle \left\langle\psi_k\right| . \tag{2.1.1}$$

Il sistema è in uno stato puro se $\rho = |\psi\rangle \langle\psi|$ per un determinato stato $|\psi\rangle$, e

dunque $\mathrm{Tr}\,(\rho^2) = 1$. Altrimenti ρ è detto uno stato *misto* e vale

$$\mathrm{Tr}\,(\rho^2) \leq 1. \tag{2.1.2}$$

Gli stati misti bidimensionali (un singolo qubit) possono essere rappresentati secondo una generalizzazione della rappresentazione di Bloch descritta nella Sezione 1.1:

$$\rho = \frac{I + \vec{r} \cdot \vec{\sigma}}{2}, \tag{2.1.3}$$

dove $\vec{r} = (r_x, r_y, r_z)$ è un vettore reale detto *vettore di Bloch*, mentre $\vec{\sigma} = (X, Y, Z)$ individua le matrici di Pauli descritte nella Sezione 1.1.

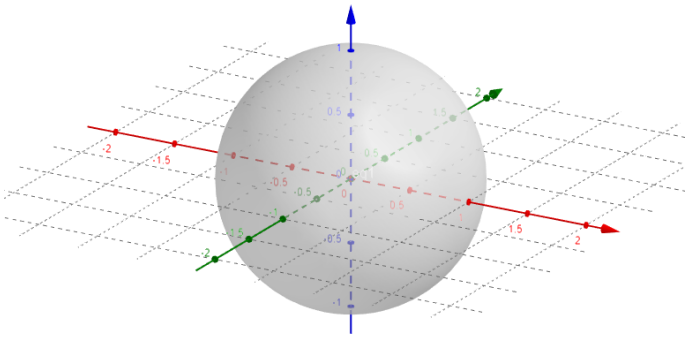


Figura 2.1.1: [16] Rappresentazione di Bloch di una matrice densità. Gli assi cartesiani rappresentano il valore delle tre componenti del vettore di Bloch. Gli stati puri stanno sulla superficie della sfera mentre gli stati misti stanno all'interno.

Dalla (2.1.2) si verifica che deve valere $||\vec{r}|| \leq 1$, in cui l'uguaglianza vale nel caso di stati puri mentre gli altri casi rappresentano gli stati misti. Questo significa che gli stati puri si trovano sulla superficie della sfera di Bloch mentre gli stati misti stanno all'interno.

Uno stato $|\psi_k\rangle$ si trasforma secondo una trasformazione U in $U\,|\psi_k\rangle$, analogamente una matrice densità ρ si trasforma secondo una trasformazione U

$$\rho \quad \xrightarrow{U} \quad \rho' = U\rho U^\dagger. \tag{2.1.4}$$

Siano R, S due sistemi distinti il cui stato complessivo è descritto dalla matrice densità ρ_{RS}. Si definisce *matrice densità ridotta* per il sistema R la matrice densità

$$\rho_R = \mathrm{tr}_S\left(\rho_{RS}\right), \tag{2.1.5}$$

dove tr_S individua l'operazione di *traccia parziale* sul sistema S. La traccia parziale è definita

$$\mathrm{tr}_S\left(|r_i\rangle\langle r_j| \otimes |s_i\rangle\langle s_j|\right) \equiv |r_i\rangle\langle r_j| \,\mathrm{Tr}(|s_i\rangle\langle s_j|), \tag{2.1.6}$$

con $|r_i\rangle, |r_j\rangle$ vettori nello spazio degli stati si R e $|s_i\rangle, |s_j\rangle$ vettori nello spazio degli stati di R.

2.2 Quantum operations

Finora si è parlato di sistemi quantistici chiusi, in cui si sono trascurate le interazioni con l'esterno e il sistema risultava completamente isolato. Questa è ovviamente un'astrazione e verranno ora introdotti gli strumenti per poter descrivere un sistema *aperto*, non più isolato ma descrivibile come un sistema principale S interagente con l'ambiente esterno E.

Si suppone che S ed E siano inizialmente non correlati e quindi descritti da una matrice densità

$$\rho = \rho_S \otimes \rho_E \tag{2.2.1}$$

e poi interagiscano attraverso una trasformazione unitaria U. Allora la matrice densità di S in seguito all'interazione risulta

$$\rho'_S = \mathrm{tr}_E\left[U(\rho_S \otimes \rho_E)U^\dagger\right]. \tag{2.2.2}$$

Sia $|e_k\rangle$ una base ortonormale dello spazio degli stati di E, allora non è restrittivo presupporre

$$\rho_E = |e_0\rangle\langle e_0|. \tag{2.2.3}$$

La (2.2.2) diventa

$$\rho'_S = \sum_k K_k \rho_S K_k^\dagger, \tag{2.2.4}$$

con $K_k \equiv \langle e_k | U | e_0 \rangle$. Questi operatori soddisfano una relazione di completezza

$$\sum_k K_k K_k^\dagger = I_S, \tag{2.2.5}$$

dove I_S è l'operatore identico per il sistema S.

In generale è detta *quantum operation* una mappa $\mathcal{E}$ che presenta le seguenti proprietà:

- $\mathrm{Tr}\,[\mathcal{E}\rho] \leq 1$ per ogni ρ operatore di densità.
- $\mathcal{E}$ è lineare convessa, ovvero $\mathcal{E}(\sum_i q_i \rho_i) = \sum_i q_i \mathcal{E}\rho_i$
- $\mathcal{E}$ è completamente positiva, ovvero se R è un ulteriore sistema deve valere $(\mathcal{I}_R \otimes \mathcal{E})(A) \geq 0$ per ogni operatore positivo agente sul sistema composito RS, con $\mathcal{I}_R$ mappa identità su R.

Una quantum operation può essere sempre esplicitata nella forma

$$\mathcal{E}\rho = \sum_k K_k \rho_S K_k^\dagger, \tag{2.2.6}$$

dove gli operatori $\{K_k\}$ sono detti *operation elements* oppure operatori di Kraus [12]. Una quantum operation è detta CPTP map (completely positive trace preserving) se vale $\mathrm{Tr}\,[\mathcal{E}\rho] = 1 \ \forall \ \rho$.

Si possono definire

$$\rho_{S_k} \equiv \frac{K_k \rho_S K_k^\dagger}{\mathrm{Tr}\,(K_k \rho_S K_k^\dagger)} \tag{2.2.7}$$

e

$$p(k) \equiv \mathrm{Tr}\,(K_k \rho_S K_k^\dagger). \tag{2.2.8}$$

e dunque riscrivere la (2.2.6) nella forma

$$\mathcal{E}\rho = \sum_k K_k \rho_S K_k^\dagger = \sum_k p(k) \rho_{S_k}. \tag{2.2.9}$$

Ora risulta intuitivo interpretare l'azione di una quantum operation come un'operazione che sostituisce allo stato ρ_S lo stato $K_k \rho_S K_k^\dagger / \mathrm{Tr}\,(K_k \rho_S K_k^\dagger)$ con probabilità $\mathrm{Tr}\,(K_k \rho_S K_k^\dagger)$. Questo formalismo ci permette di rappresentare

canali quantistici disturbati, in cui il sistema può subire trasformazioni secondo processi che vengono considerati stocastici markoviani [5] che possono risultare in errori nella computazione.

2.3 Esempi di quantum operations

Ricordando la rappresentazione di Bloch (2.1.3) si vede ora come una quantum operation agisce su un singolo qubit:

$$\rho \xrightarrow{\mathcal{E}} \rho' = \mathcal{E}\rho, \tag{2.3.1}$$

$$\vec{r} \xrightarrow{\mathcal{E}} \vec{r'} = M\vec{r} + \vec{c}, \tag{2.3.2}$$

dove M è una matrice reale 3×3 e $\vec{c}$ un vettore costante. Questa è una mappa affine, che mappa la sfera di Bloch su sè stessa, rappresentando una deformazione, rotazione e traslazione della sfera stessa.

2.3.1 Canali bit-flip e phase-flip

Il primo esempio che viene riportato è il canale *bit flip* che scambia gli stati $|0\rangle$ e $|1\rangle$ con probabilità $1 - p$. Gli operation elements sono

$$E_0 = \sqrt{p}I, \quad E_1 = \sqrt{1-p}X. \tag{2.3.3}$$

L'effetto di questo bit flip channel sulla sfera di Bloch (Fig. 2.1) è mostrato in Figura 2.3.1.

Analoghi sono i canali *phase flip* i cui operation elements sono

$$E_0 = \sqrt{p}I, \quad E_1 = \sqrt{1-p}Z, \tag{2.3.4}$$

e il canale *bit-phase flip* i cui operation elements sono

$$E_0 = \sqrt{p}I, \quad E_1 = \sqrt{1-p}Y. \tag{2.3.5}$$

L'effetto di entrambi i canali sono mostrati in Figura 2.3.2.

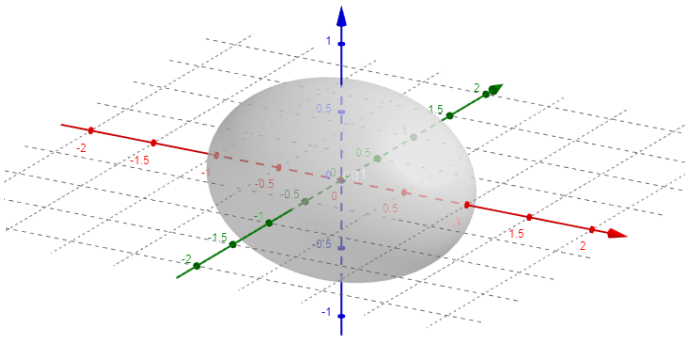


Figura 2.3.1: Rappresentazione grafica dello stato del qubit in seguito all'attraversamento del flip channel, nell'esempio $p = 0.3$. Si noti come avvenga una contrazione lungo il piano $\hat{y} - \hat{z}$.

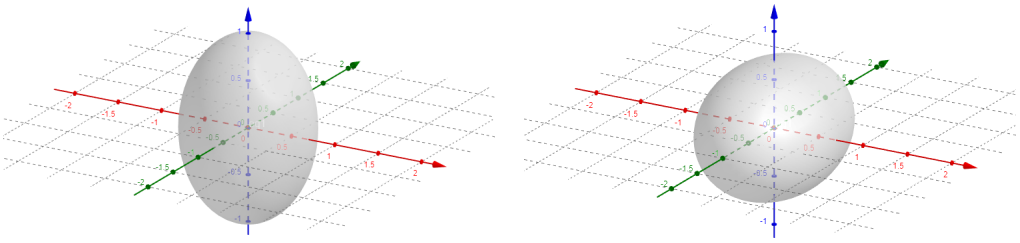


Figura 2.3.2: Rappresentazione grafica dello stato del qubit in seguito all'attraversamento rispettivamente del phase channel e del bit-phase channel, nell'esempio $p = 0.3$. Si noti come in un caso avvenga una contrazione lungo il piano $\hat{x} - \hat{y}$ e nell'altro una contrazione lungo il piano $\hat{x} - \hat{z}$.

2.3.2 Canale depolarizzante

Il canale *depolarizzante* (depolarizing channel) ha come effetto appunto quello di depolarizzare un qubit con probabilità p: il qubit rimane invariato con probabilità $1 - p$, altrimenti viene sostituito con lo stato misto $I/2$. La quantum operation corrispondente è

$$\mathcal{E}\rho = p\frac{I}{2} + (1 - p)\rho, \tag{2.3.6}$$

e il suo effetto è mostrato in Figura 2.3.3. Si noti come alternativamente la

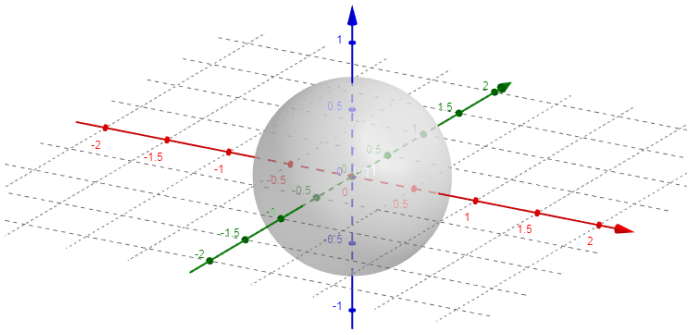


Figura 2.3.3: Rappresentazione grafica dello stato del qubit in seguito all'attraversamento del depolarizing channel, nell'esempio $p = 0.3$. Si noti come in un caso avvenga una contrazione uniforme dell'intera sfera.

(2.3.6) possa essere riscritta nella forma

$$\mathcal{E}\rho = (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z), \tag{2.3.7}$$

e dunque si può considerare il depolarizing channel come un canale in cui lo stato rimane invariato con probabilità p mentre ciascuno degli operatori di Pauli può essere applicato con probabilità $p/3$. Infine si può generalizzare il tutto per un sistema con dimensione n, in cui la quantum operation corrispondente è

$$\mathcal{E}\rho = p\frac{I}{n} + (1-p)\rho. \tag{2.3.8}$$

2.4 Quantum Error Correction

Per evitare che le interazioni dell'ambiente con il sistema risultino in errori di computazione si implementano codici particolari detti *quantum error correcting codes*. Nel farlo è necessario tenere conto delle caratteristiche peculiari dei sistemi quantistici, tra cui l'impossibilità di copiare stati quantistici arbitrari [13], la continuità degli errori[1] e l'interferenza dell'operazione di mi-

[1]Un qubit può essere interessato da un numero arbitrario di errori, rendendo la deformazione dello stato un processo continuo.

sura sul sistema (1.3.2). Un generico error correcting code si articola in due stadi differenti:

- *Error-detection*, fase in cui un eventuale errore viene individuato.
- *Recovery*, in cui in base alle informazioni ottenute nella fase di error-detection si agisce per far fronte all'errore.

Questi due processi possono essere racchiusi in una CPTP map $\mathcal{R}$, che verrà chiamata *error-correction* operation o *recovery map*. Perché questa operazione sia eseguita con successo, si richiede

$$(\mathcal{R} \circ \mathcal{E})\rho \propto \rho. \tag{2.4.1}$$

Teorema 2.4.1. Sia C un quantum code, P un proiettore su C e $\mathcal{E}$ una quantum operation con operation elements $\{E_i\}$. Allora esiste una error-correction operation $\mathcal{R}$ per $\mathcal{E}$ su C se e solo se

$$PE_i^\dagger E_j P = \alpha_{ij} P,$$

per una qualsiasi matrice hermitiana complessa α.

Gli operation elements $\{E_i\}$ sono detti *errori*[2]. Nel caso esista $\mathcal{R}$, si dice che $\{E_i\}$ costituisce un set di errori *correggibile*.

Dimostrazione. Siano $\{E_i\}$ operation elements che soddisfano le condizioni del Teorema 2.4.1. Siccome α è una matrice hermitiana, la si può esprimere nella forma diagonale $d = u^\dagger \alpha u$, con u una matrice unitaria. Si definiscono dunque gli operatori $F_k \equiv \sum_i u_{ik} E_i$, che risultano essere loro stessi operation elements di $\mathcal{E}$ [15]. Allora vale

$$PF_k^\dagger F_l P = \sum_{ij} u_{ki}^\dagger u_{jl} PE_i^\dagger EjP. \tag{2.4.2}$$

[2]Questa definizione è evidentemente indipendente rispetto a quella dell'Eq. 1.4.11, nel proseguire dal testo sarà chiaro a ciò a cui ci si sta referendo.

Utilizzando la condizione del Teorema si ricava

$$PF_k^\dagger F_l P = \sum_{ij} u_{ki}^\dagger \alpha_{ij} u_{jl} P = d_{kl} P \tag{2.4.3}$$

per definizione di d.

Utilizzando una scomposizione polare [15] si scrive $F_k P = U_k \sqrt{P F_k^\dagger F_k P} = \sqrt{d_{kk}} U_k P$ per una trasformazione unitaria U_k. Si può interpretare l'effetto di F_k come una rotazione dello spazio del codice all'interno dello spazio definito dal proiettore $P_k \equiv U_k P U_k^\dagger = F_k P U_k^\dagger / \sqrt{d_{kk}}$. Dalla (2.4.3) si ottiene

$$P_l P_k = P_l^\dagger P_k = \frac{U_l P F_l^\dagger F_k P U_k^\dagger}{\sqrt{d_{ll} d_{kk}}} = 0. \tag{2.4.4}$$

P_k è una projective measurement che permette di identificare gli errori. La recovery è effettuata applicando $U_k^\dagger$. Infatti per uno stato ρ del codice vale

$$U_k^\dagger P_k F_l \sqrt{\rho} = U_k^\dagger P_k^\dagger F_l P \sqrt{\rho} \tag{2.4.5}$$

$$= \frac{U_k^\dagger U_k P F_k^\dagger F_l P \sqrt{\rho}}{\sqrt{d_{kk}}} \tag{2.4.6}$$

$$= \delta_{kl} \sqrt{d_{kk}} P \sqrt{\rho} \tag{2.4.7}$$

$$= \delta_{kl} \sqrt{d_{kk}} \sqrt{\rho}. \tag{2.4.8}$$

La quantum operation composta dalla detection-recovery corrisponde a $\mathcal{R}(\sigma) = \sum_k U_k^\dagger P_k \sigma P_k U_k$, per cui vale

$$\mathcal{R}(\mathcal{E}(\rho)) = \sum_k U_k^\dagger P_k F_l \rho F_l^\dagger P_k U_k \tag{2.4.9}$$

$$= \sum_{kl} \delta_{kl} d_{kk} \rho \tag{2.4.10}$$

$$\propto \rho, \tag{2.4.11}$$

come richiesto dalla (2.4.1).

Si vuole verificare ora che la condizione del teorema sia necessaria. Si supponga che $\{E_i\}$ sia un set di errori che è perfettamente correggibile da una quantum operation $\mathcal{R}$ con operation elements $\{R_j\}$. Si definisce $\mathcal{E}_C(\rho) = \mathcal{E}(P\rho P)$ da cui segue

$$\mathcal{R}(\mathcal{E}_C(\rho)) \propto P\rho P \quad \forall\, \rho. \tag{2.4.12}$$

Riscrivendo esplicitamente la dipendenza utilizzando gli operation elements si ha

$$\sum_{ij} R_j E_i P \rho P E_i^\dagger R_j^\dagger = cP\rho P \quad \forall\, \rho. \tag{2.4.13}$$

Allora la quantum operation con operation elements $\{R_j E_i\}$ è identica alla quantum operation il cui singolo operation element è $\sqrt{c}P$. Esistono dei numeri complessi c_{ki} per cui [15]

$$R_k E_i P = c_{ki} P. \tag{2.4.14}$$

Prendendo l'equazione complessa coniugata si verifica $PE_i^\dagger R_k^\dagger = c_{ki}^* P$ e allora $PE_i^\dagger R_k^\dagger R_k E_j P = c_{ki}^* c_{kj} P$. Essendo $\mathcal{R}$ una CPTP map si deduce

$$PE_i^\dagger E_j P = \alpha_{ij} P, \tag{2.4.15}$$

con $\alpha_{ij} \equiv \sum_k c_{ki}^* c_{kj}$ una matrice hermitiana. □

2.5 Stabilizer Codes

Un caso particolare di quantum error correcting code sono gli *stabilizer codes*. Un esempio permetterà di rendere più chiara l'introduzione di questo particolare tipo di codici: è facile verificare che per uno stato

$$|\psi\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \tag{2.5.1}$$

vale $X_1 X_2 |\psi\rangle = |\psi\rangle$ e $Z_1 Z_2 |\psi\rangle = |\psi\rangle$. Si dice dunque che lo stato $|\psi\rangle$ è *stabilizzato* da $X_1 X_2$ e $Z_1 Z_2$.

Si definisce un *gruppo di Pauli* su n qubit

$$\mathcal{P}_n \equiv \{\pm 1, \pm i\} \times \{I, X, Y, Z\}^{\otimes n}, \tag{2.5.2}$$

che costituisce un gruppo di matrici per l'operazione di moltiplicazione tra matrici. Gli elementi di $\mathcal{P}_n$ sono detti *stabilizer operators*. Sia S un sottogruppo di $\mathcal{P}_n$ e V_S lo spazio degli stati stabilizzati dagli elementi di S. Allora S è detto *stabilizzatore* di V_S. Spesso si preferisce esprimere S tramite i suoi

generatori $S = \langle g_1, \ldots, g_n \rangle$, ovvero quegli elementi che possono generare tutti gli altri attraverso il loro prodotto. I generatori di S sono detti indipendenti se rimuovendone uno qualsiasi il gruppo generato è diverso da quello generato in precedenza. Gli elementi di V_S sono autostati simultanei associati all'autovalore $+1$ di tutti gli elementi di S:

$$\forall\, S_i \in S, \quad S_i \left|\psi\right\rangle = \left|\psi\right\rangle . \tag{2.5.3}$$

Ciò è equivalente a richiedere che $\left|\psi\right\rangle$ sia autostato simultaneo associato all'autovalore $+1$ dei generatori di S: sia S_g l'insieme dei generatori $\{g_1, \ldots, g_n\}$ di S, allora

$$\forall\, g_i \in S_g, \quad g_i \left|\psi\right\rangle = \left|\psi\right\rangle . \tag{2.5.4}$$

Si supponga di applicare una trasformazione U allo spazio V_S stabilizzato da S. Per ogni elemento $\left|\psi\right\rangle$ di V_S e s di S si ha

$$U \left|\psi\right\rangle = Us \left|\psi\right\rangle = UsU^\dagger U \left|\psi\right\rangle , \tag{2.5.5}$$

dunque lo stato $U \left|\psi\right\rangle$ è stabilizzato da $UsU^\dagger$. Si può dedurre che lo spazio $UV_S \equiv \{U \left|\psi\right\rangle \,|\, \left|\psi\right\rangle \in V_S\}$ è stabilizzato dallo spazio $USU^\dagger \equiv \left\{UsU^\dagger | s \in S\right\}$ e che se $S = \langle g_1, \ldots, g_n \rangle$, allora $USU^\dagger = \langle Ug_1U^\dagger, \ldots, Ug_nU^\dagger \rangle$. L'insieme degli U per cui $U\mathcal{P}_nU^\dagger = \mathcal{P}_n$ è detto *normalizer* di $\mathcal{P}_n$ e indicato con $N(\mathcal{P}_n)$.

Esempio 2.5.1. Si prenda in considerazione un sistema di n-qubit nello stato stabilizzato da $\langle Z_1, Z_2, \ldots, Z_n \rangle$, che risulta essere $\left|0\right\rangle^{\otimes n}$. Se ora l'Hadamard gate H viene applicato ad ogni qubit, lo stato risultante $\left|+\right\rangle^{\otimes n}$ è stabilizzato da $\langle X_1, X_2, \ldots, X_n \rangle$, infatti $HZH^\dagger = X$.

Perché lo spazio V_S non coincida con lo spazio nullo è necessario che:

- gli elementi del gruppo stabilizzatore S commutino tra loro

$$\forall\, S_i, S_j \in S, \quad [S_i, S_j] = 0; \tag{2.5.6}$$

-

$$-I \notin S. \tag{2.5.7}$$

Lo spazio vettoriale V_S stabilizzato da $S \in \mathcal{P}_n$ che soddisfa queste condizioni è spesso detto *stabilizer code* e indicato con $C(S)$.

Il seguente teorema enuncia le condizioni per cui si stabilisce se un set di errori è correggibile descritte dal Teorema (2.4.1) nel formalismo dei stabilizer codes.

Teorema 2.5.2. Sia S lo stabilizzatore dello stabilizer code $C(S)$. Sia $\{E_i\}$ è un insieme di operatori $\in \mathcal{P}_n$ per cui $E_i^\dagger E_j \notin N(S) - S \; \forall \; i, j$. Allora $\{E_i\}$ è un insieme di errori correggibili per $C(S)$.

Dimostrazione. Sia P un proiettore su $C(S)$. Per i, j fissati si distinguono il caso in cui $E_i^\dagger E_j \in S$ oppure $E_i^\dagger E_j \in \mathcal{P}_n - N(S)$ Nel primo caso vale $PE_i^\dagger E_j P = P$ in quanto P è invariante per moltiplicazione di elementi di S. Nel caso in cui $E_i^\dagger E_j \in \mathcal{P}_n - N(S)$ si suppone che $\left\{E_i^\dagger E_j, g_1\right\} = 0$ per un arbitrario $g_1 \in S$. Sia $S = \langle g_1, \dots g_{n-k} \rangle$, allora

$$P = \frac{\prod_{l=1}^{n-k}(I + g_l)}{2^{n-k}} \tag{2.5.8}$$

da cui dall'anti-commutatività di $E_i^\dagger E_j$ e g_1 deriva

$$E_i^\dagger E_j P = (I + g_1) E_i^\dagger E_j \frac{\prod_{l=1}^{n-k}(I + g_l)}{2^{n-k}}. \tag{2.5.9}$$

Ricordando $P(I - g_1) = 0$ in quanto $(I - g_1)(I + g_1) = 0$, si verifica che $PE_i^\dagger E_j P = 0 \; \forall \; \mathcal{P}_n - N(S)$ □

.

Esempio 2.5.3. *bit flip code con 3 qubit.* Il codice di tre qubit stabilizzato da $\langle Z_1 Z_2, Z_2 Z_3 \rangle$ corrisponde agli stati $|000\rangle$ e $|111\rangle$. Ricordando che un'operazione di bit flip corrisponde all'azione dell'operatore X su un qubit, identifichiamo il gruppo di trasformazioni generato dai prodotti tensoriali degli operatori $\{I, X_1, X_2, X_3\}$ come i possibili bit flip possibili. Secondo il Teorema 2.5.2 questo gruppo di errori risulta correggibile dallo stabilizzatore $\langle Z_1 Z_2, Z_2 Z_3 \rangle$. Nel caso avvenissero gli errori X_1 o X_2 lo stato risultante sarebbe stabilizzato

da $\langle -Z_1Z_2, Z_2Z_3\rangle$. Nel caso l'errore avvenuto fosse X_3 allora lo stato risultante in questo caso sarebbe stabilizzato da $\langle Z_1Z_2, -Z_2Z_3\rangle$. Queste informazioni vengono utilizzate per l'error-detection (Sez. 2.4). Una volta identificato l'errore è sufficiente applicare al qubit interessato la trasformazione inversa per portare a termine la correzione.

Esempio 2.5.4. *Shor code.* Il *Shor Code* è un codice che protegge da qualsiasi errore su un singolo qubit codificato in 9 qubit. Si procede codificando il qubit nel modo seguente:

$$|0\rangle \to |+++\rangle\,, \qquad |1\rangle \to |---\rangle\,,$$

e dunque codificando ognuno dei tre qubit per cui

$$|+\rangle \to \frac{|000\rangle + |111\rangle}{\sqrt{2}}, \qquad |-\rangle \to \frac{|000\rangle - |111\rangle}{\sqrt{2}}.$$

La codifica risultante è dunque

$$|0\rangle \to |0_L\rangle \equiv \frac{(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)}{2\sqrt{2}}, \tag{2.5.10}$$

$$|1\rangle \to |1_L\rangle \equiv \frac{(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)}{2\sqrt{2}}. \tag{2.5.11}$$

Per semplicità si studia la situazione in cui un errore arbitrario interessa il primo qubit. Si suppone che lo stato del qubit codificato sia $|\psi\rangle = \alpha\,|0_L\rangle + \beta\,|1_L\rangle$ e che poi venga interessato da rumore per cui per una determinata quantum operation $\mathcal{E}$ lo stato finale risulta essere $\mathcal{E}(|\psi\rangle\langle\psi|) = \sum_i E_i\,|\psi\rangle\langle\psi|\,E_i^\dagger$. Si esplicita l'operatore un operatore generico E_i agente sul primo qubit nella forma

$$E_i = e_{i0}I + e_{i1}X_1 + e_{i2}Z_1 + e_{i3}X_1Z_1, \tag{2.5.12}$$

e dunque lo stato $E_i\,|\psi\rangle$ può essere descritto dalla sovrapposizione di quattro termini, $|\psi\rangle$, $X_1\,|\psi\rangle$, $Z_1\,|\psi\rangle$ e $X_1Z_1\,|\psi\rangle$. Misurando lo stato la sovrapposizione collassa e viene rilevato uno dei quattro stati, per cui per correggere l'errore è sufficiente applicare la trasformazione inversa per ottenere lo stato originale $|\psi\rangle$.

Si riporta lo stabilizer code associato al codice di Shor, i generatori dello stabilizzatore sono

$$
\begin{aligned}
g_1 &= ZZIIIIII, \\
g_2 &= IZZIIIII, \\
g_3 &= IIIZZIII, \\
g_4 &= IIIIZZII, \\
g_5 &= IIIIIIZZI, \\
g_6 &= IIIIIIIZZ, \\
g_7 &= XXXXXXIII, \\
g_8 &= IIIXXXXXX.
\end{aligned}
$$

Questa notazione indica per esempio $ZZIIIIII = Z \otimes Z \otimes I \otimes I \otimes I \otimes I \otimes I \otimes I \otimes I = Z_1 Z_2$.

Gli errori su singolo qubit risultano tutti correggibili attraverso questo stabilizzatore secondo il Teorema 2.5.2. Infatti il prodotto di qualsiasi due errori risultano $\in S$ o alternativamente non appartengono a $N(S)$, dunque il codice di Shor può essere impiegato per correggere qualsiasi errore su qubit singolo.

In realtà si può dimostrare che è possibile correggere qualsiasi con un codice che codifica un singolo qubit in cinque qubit [15]. Lo stabilizer corrispondente a questo particolare caso è quello generato da

$$
\begin{aligned}
g_1 &= XZZXI, \\
g_2 &= IXZZX, \\
g_3 &= XIXZZ, \\
g_4 &= ZXIXZ.
\end{aligned}
$$

Capitolo 3

Toric Code

In questo capitolo verrà introdotto lo stabilizer code topologico Toric Code, la cui trattazione esaustiva è effettuata in [1]. Per poter comprenderne la struttura viene descritto innanzitutto il formalismo dei complessi di catene Z_2 su reticolo, attraverso cui si possono descrivere tutte le caratteristiche e la struttura del codice. Si procede descrivendo lo spazio stabilizzante e i suoi generatori, oltre all'operatore hamiltoniano che permette di individuare le eccitazioni. Dunque si approfondiscono le caratteristiche di particolari stati eccitati per poi studiare lo stato fondamentale e la sua degenerazione in correlazione con la topologia del toro. Infine vengono illustrate le idee fondamentali per la correzione pratica degli errori in computazione.

3.1 Z_2 Chain Complex

Prima di descrivere il Toric Code, si introducono i complessi di catene Z_2 [4] che permetteranno un'adeguata descrizione del codice.

Si consideri un reticolo [2] $G = (V, E, F)$ immerso in una superficie e costituito dai vertici $V = \{v_k\}$, lati $E = \{e_k\}$ e facce $F = \{f_k\}$. Si definisce lo spazio vettoriale C_0 generato dalla base $B(C_0) = \{v_k\}$ composta dai vertici

$v_k \in V$. Un qualsiasi vettore in C_0

$$c_0 = \sum_k z_k v_k \tag{3.1.1}$$

con $z_k = \{0, 1\}$ è detto 0-chain. Lo spazio vettoriale C_0 è un gruppo abeliano per l'addizione modulo 2, ovvero $c_0 \oplus c_0' = c_0' \oplus c_0 \in C_0 \ \forall \ c_0, c_0' \in C_0$, in cui l'addizione a modulo 2 è fatta su ogni componente. Analogamente si definiscono gli spazi vettoriali C_1 e C_2 le cui rispettive basi sono $B(C_1) = \{e_l\}$ $B(C_2) = \{f_m\}$. Gli elementi

$$c_1 = \sum_l z_l e_l, \tag{3.1.2}$$

$$c_2 = \sum_m z_m f_m, \tag{3.1.3}$$

con $z_l, z_m = \{0, 1\}$ sono detti 1-chain e 2-chain. In generale un vettore costruito con con vettori di dimensione i è detta una i-chain $c_i \in C_i$.

Si definisce un omomorfismo $\partial_i : C_i \to C_{i-1}$ tale che

$$\partial_i \circ \partial_{i-1} = 0. \tag{3.1.4}$$

In particolare $\partial_i c_i$ consiste nella $(i-1)$-chain che individua il bordo di c_1. Se ne mostra un esempio in Figura 3.1.1. Molto spesso il pedice i in ∂_i viene omesso.

Una catena c_i è detta *ciclo* se $\partial c_i = 0$, inoltre il ciclo è detto triviale se $c_i \in \operatorname{Img} \partial_{i+1}$. I gruppi abeliani C_i, connessi dagli operatori di bordo ∂_i per cui $\partial_i \circ \partial_{i-1} = 0$ danno luogo a un *complesso di catene* Z_2 (Z_2 chain complex) in quanto $\operatorname{Img} \partial_{i-1} \in \operatorname{Ker} \partial_i$. Si definisce l'omologia [8] H_i come gruppo quoziente formato dai cicli e dai cicli triviali:

$$H_i = \operatorname{Ker} \partial_i \,/\, \operatorname{Img} \partial_{i-1}. \tag{3.1.5}$$

Un elemento $h \in H_i$ è detto homology class: se due catene c_i e c_i' appartengono alla stessa homology class allora esiste una catena di ordine superiore c_{i+1} tale che $c_i = c_i' + \partial c_{i+1}$. In questo caso c_i e c_i' sono detti omologicamente equivalenti.

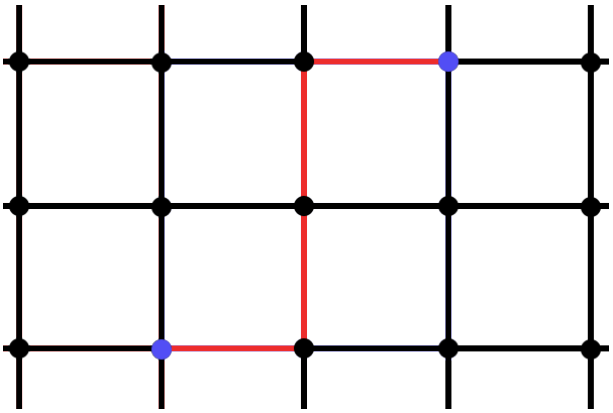


Figura 3.1.1: In rosso una catena c_1, in blu la catena $c_0 = \partial c_1$ individua il bordo di c_1. Il reticolo mostrato in figura è il caso particolare di reticolo quadrato.

Si definisce inoltre il reticolo *duale* $\bar{G} = (\bar{V}, \bar{E}, \bar{F})$ immerso in una superficie e costituito dai vertici $\bar{V} = F$, lati $\bar{E} = E$ e facce $\bar{F} = V$. In particolare il reticolo duale è costruito così che due vertici $\bar{v}, \bar{v}' \in \bar{V}$ sono connessi dal lato $\bar{e}$ se le corrispondenti due facce f e f' condividono lo stesso lato e. Si può dunque costruire una Z_2 chain complex su $\bar{G}$ attraverso le basi duali $\bar{B}(C_i)$, le catene duali $\bar{c}_i$ e gli operatori di bordo $\bar{\partial}_i : \bar{C}_i \to \bar{C}_{i-1}$.

Queste strutture si utilizzano per costruire stabilizer codes: su ogni lato $e_l \in E$ di G viene definito un qubit. Allora si definisce

$$W(c_1) = \prod_l W_l^{z_l}, \tag{3.1.6}$$

con $c_1 = \sum_l z_l e_l$ una generica 1-chain e $W_l \in \{X_l, Y_l, Z_l\}$ un operatore di Pauli agente sul qubit sul lato e_l secondo la notazione 1.2.2. Segue che $W(c_1 + c_1') = W(c_1)W(c_1')$. Si definisce inoltre il prodotto scalare $c_1 \cdot c_1' = \bigoplus_l z_l z_l'$ dove $\bigoplus_l$ indica la somma con modulo 2 in serie su l.

Dati $X(c_1)$ e $Z(c_1')$ per due 1-chain c_1 e c_1' è facile verificare:

$$c_1 \cdot c_1' = 0 \;\Leftrightarrow\; [X(c_1), Z(c_1')] = 0, \tag{3.1.7}$$

$$c_1 \cdot c_1' = 1 \;\Leftrightarrow\; \{X(c_1), Z(c_1')\} = 0. \tag{3.1.8}$$

Sia $M(\partial_i)$ la rappresentazione matriciale di ∂_i rispetto alla basi $B(C_i)$ e $B(C_{i-1})$. Si ha

$$(M(\partial_i)c_i) \cdot c_{i-1} = c_i \cdot (M(\partial_i)^T c_{i-1}). \tag{3.1.9}$$

Per definizione $B(C_0) = \bar{B}(C_2), B(C_1) = \bar{B}(C_1)$ e dunque

$$M(\partial_1) = M(\bar{\partial}_2)^T, \quad M(\partial_2) = M(\bar{\partial}_1)^T. \tag{3.1.10}$$

Dalla Eq.(3.1.4) si ricava $M(\bar{\partial}_2)^T M(\partial_2) = 0$. Allora

$$\bar{\partial}\bar{c}_2 \cdot \partial c_2 = 0 \ \ \forall\ \bar{c}_2 \in \bar{C}_2, c_2 \in C_2. \tag{3.1.11}$$

3.2 Struttura del Toric Code

Sia $G = (V, E, F)$ un reticolo quadrato $n \times n$ e $\bar{G} = (\bar{V}, \bar{E}, \bar{F})$ il reticolo duale corrispondente, entrambi immersi in una superficie toroidale $\mathcal{T}$. Ad ogni lato $e \in E$ è associato un qubit, dunque se si indica la cardinalità di un insieme A con $|A|$, la dimensione dello spazio di Hilbert che descrive lo stato del sistema associato a G è $2^{|E|}$.

Si indica con $\delta v \equiv \partial \bar{f}$ l'insieme di lati e congiunti da v. Si definiscono i generatori dello stabilizzatore S (Sez. 2.5) per ogni vertice v e faccia f

$$A_{v_k} = X(\delta v_k) = X(\partial \bar{f}_k), \tag{3.2.1}$$

$$B_{f_m} = Z(\partial f_m) \tag{3.2.2}$$

secondo l'Eq. 3.1.6. Valgono dunque $\prod_k A_{v_k} = I$ e $\prod_m B_{f_m} = I$, allora il numero totale di generatori dello stabilizzatore indipendenti è pari a

$$|F| + |V| - 2 = 2n^2 - 2. \tag{3.2.3}$$

Il numero di qubit risulta

$$|E| = 2n^2, \tag{3.2.4}$$

indicando che la dimensione dello spazio stabilizzato V_S associato a S è [15]

$$2^{|E|-(|F|+|V|-2)} = 2^2. \tag{3.2.5}$$

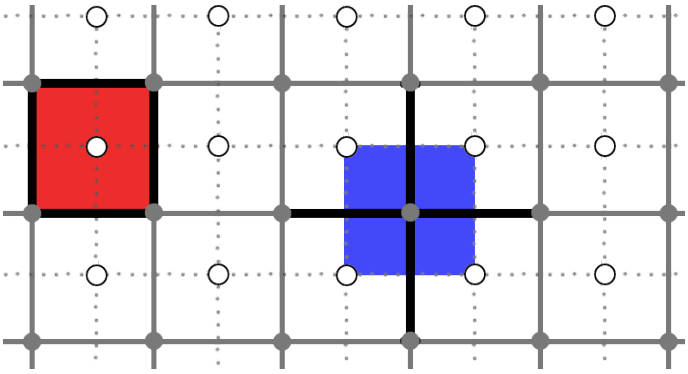


Figura 3.2.1: Reticolo quadrato G e corrispondente reticolo duale $\bar{G}$ tratteggiato. In rosso è evidenziata una faccia f_m circoscritta all'interno del bordo in grassetto ∂f_m. In blu è evidenziata una faccia $\bar{f}_k$ e in grassetto è evidenziato il δv_k corrispondente.

Dalla (3.1.7) ricordando (3.1.11) si può affermare che $[A_{v_k}, B_{f_m}] = 0$. Per definizione il codice stabilizzato $|\psi\rangle$ soddisfa

$$A_{v_k} |\psi\rangle = |\psi\rangle\,, \quad B_{f_m} |\psi\rangle = |\psi\rangle \quad \forall\, f_m \in F, v_k \in V. \tag{3.2.6}$$

Inoltre dalle proprietà degli operatori di Pauli segue

$$A_{v_k}^2 = B_{f_m}^2 = I, \tag{3.2.7}$$

dunque gli autovalori di A_{v_k} e B_{f_m} possono essere solo ± 1.

Si definisce l'operatore

$$H_{TC} = -\sum_k A_{v_k} - \sum_m B_{f_m} \tag{3.2.8}$$

che misura l'energia del sistema. Dalle considerazioni appena fatte si attesta che l'energia più bassa associata al sistema è $E_0 = -(|V|+|F|)$, che si verifica quando lo stato del sistema soddisfa

$$A_{v_k} |\psi_0\rangle = |\psi_0\rangle\,, \quad B_{f_m} |\psi_0\rangle = |\psi_0\rangle$$

per ogni k ed m. Questi stati sono detti *vacuum states*. Uno di questi risulta essere

$$\left|\psi_0^{(1)}\right\rangle = \frac{1}{\sqrt{2}} \prod_k (I + A_{v_k}) \, |0\rangle^{\otimes |E|}\,. \tag{3.2.9}$$

3.3 Stati eccitati e quasi-particelle

Si consideri

$$|\psi'\rangle = X_k \left|\psi_0^{(1)}\right\rangle, \tag{3.3.1}$$

che può essere interpretato come il vacuum state dell'Eq. 3.2.9 che ha subito un bit flip (Sez. 2.3.1) sul qubit posizionato sul lato e_k. Vale

$$W_p \circ W'_q = W'_q \circ W_p \qquad \text{se } p = q \text{ e } W = W' \tag{3.3.2}$$

$$W_p \circ W'_q = -W'_q \circ W_p \quad \text{altrimenti,} \tag{3.3.3}$$

se $W_l \in \{X_l, Y_l, Z_l\}$. Allora

$$B_{f_m} |\psi'\rangle = \left(\prod_{e_l \in \partial f_m} Z_l\right) \circ X_k \left|\psi_0^{(1)}\right\rangle \tag{3.3.4}$$

$$= -X_k \circ \left(\prod_{e_l \in \partial f_m} Z_l \left|\psi_0^{(1)}\right\rangle\right) \tag{3.3.5}$$

$$= -X_k \left|\psi_0^{(1)}\right\rangle = -|\psi'\rangle \tag{3.3.6}$$

se $e_k \in \partial f_m$, e dunque $|\psi'\rangle$ non risulta essere un vacuum state. Lo stesso vale per $B_{f_{m'}} |\psi'\rangle = -|\psi'\rangle$ se $e_k = \partial f_m \cap \partial f_{m'}$, mentre per le altre $|F| - 2$ facce e $|V|$ vertici vale

$$A_v |\psi'\rangle = |\psi'\rangle, \tag{3.3.7}$$

$$B_f |\psi'\rangle = |\psi'\rangle. \tag{3.3.8}$$

Dunque l'energia dello stato $|\psi'\rangle$ corrisponde a

$$E_1 = E_0 + 4. \tag{3.3.9}$$

Sia

$$|\psi''\rangle = Z_j \left|\psi_0^{(1)}\right\rangle \tag{3.3.10}$$

con $e_j = \delta v_a \cap \delta v_b$, che può essere interpretato come il vacuum state dell'Eq. 3.2.9 che ha subito un phase flip (Sez. 2.3.1) sul qubit posizionato sul lato e_j. Allora

$$A_{v_a} |\psi''\rangle = A_{v_b} |\psi''\rangle = -|\psi''\rangle, \tag{3.3.11}$$

mentre per tutti gli altri $|V| - 2$ vertici e $|F|$ facce vale

$$A_v \left|\psi''\right\rangle = \left|\psi''\right\rangle , \tag{3.3.12}$$

$$B_f \left|\psi''\right\rangle = \left|\psi''\right\rangle . \tag{3.3.13}$$

L'energia dello stato $|\psi''\rangle$ risulta dunque pari a E_1 (3.3.9), ed è facile intuire che un generico stato eccitato ha energia $E = E_0 + 4m$, con $m \in \mathbb{N}$. Dunque gli operatori di vertice e faccia A_v e B_f sono in grado di individuare eccitazioni sul reticolo G, riuscendo a identificare vari stati come vacuum states oppure stati eccitati svolgendo l'azione di Error Detection (Sez. 2.4).

Le eccitazioni locali del reticolo possono essere interpretate come *quasi-particelle* (Fig. 3.3.1):

- Nel caso dell'autostato 3.3.1 si può associare l'eccitazione delle facce f_m e $f_{m'}$ all'esistenza di due quasi-particelle, una per ogni faccia.

- Nel caso dell'autostato 3.3.10 si può associare l'eccitazione dei vertici v_a e v_b all'esistenza di due quasi-particelle, una per ogni vertice.

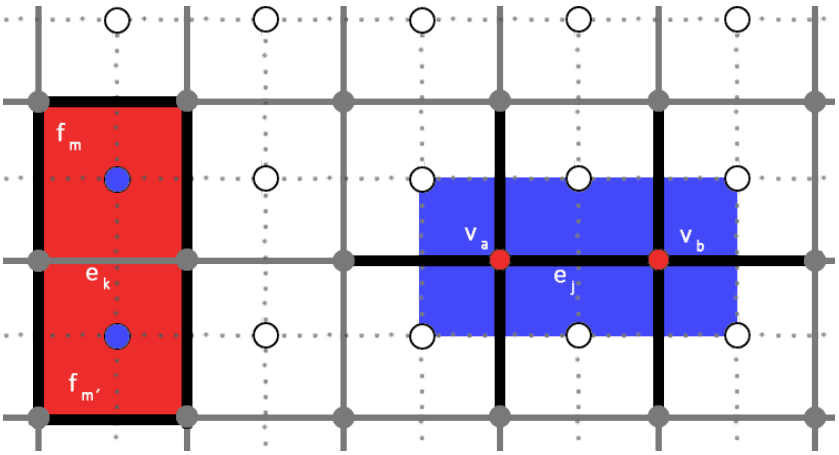


Figura 3.3.1: Rappresentazione di due quasi-particelle (in blu) create sulle facce f_m e $f_{m'}$ dall'operatore X_k agente su e_k e due quasi-particelle (in rosso) create sui vertici v_a e v_b dall'operatore Z_j agente su e_j.

Entrambi i tipi di quasi-particelle risultano avere la medesima energia $E_{qp} = 2$, ma sono identificate in maniera differente: quelle individuate dagli operatori di faccia B_f saranno chiamate di *tipo* m, mentre quelle individuate dagli operatori di vertice A_v saranno chiamate di *tipo* e. Le particelle m ed e risultano essere *anioni* [14].

Si studia ora il caso generale

$$|\psi\rangle = W_k \left|\psi_0^{(1)}\right\rangle, \tag{3.3.14}$$

osservando che una seconda applicazione di W_k su $|\psi\rangle$ riporta il sistema allo stato di vacuum (3.2.9) in quanto $W_k^2 = I_k$. Questo ci porta a considerare ogni quasi-particella come la sua stessa *anti* quasi-particella e alla costruzione di un operatore del tipo

$$O_{jk}^{\alpha} = W_j W_k, \quad \alpha = x, z, \tag{3.3.15}$$

dove α indica la natura delle trasformazioni eseguite dai W. L'azione di questo operatore sullo stato $|\psi\rangle$ è quella di eliminare la coppia di quasi-particelle nella vicinanza del lato k e creane un nuovo paio nelle vicinanze del lato j, per questo viene detto un operatore di *trasporto*. E' importante notare che dopo l'azione dell'operatore di trasporto il numero di quasi-particelle presenti rimane invariato, cambia solamente la loro disposizione nel reticolo.

In Appendice B si mostra come gli operatori di Pauli possono essere descritti secondo il formalismo di Fock [9] e come questo permetta di descrivere l'intera struttura del codice attraverso gli operatori di creazione e distruzione.

3.4 Degenerazione dello stato fondamentale

Sia γ un *cammino*, ovvero una 1-chain composta da lati di G congiunti due a due, si può costruire

$$O_{\gamma}^{z} = \prod_{j \,:\, e_j \in \gamma} Z_j \tag{3.4.1}$$

l'operatore che trasporta una quasi-particella lungo i vertici che congiungono i lati che costruiscono γ. Se il cammino è chiuso, nessuna quasi-particella è coinvolta nel processo e dunque

$$\left|\psi_0^{(1)}\right\rangle' = O_\gamma^z \left|\psi_0^{(1)}\right\rangle \tag{3.4.2}$$

risulta essere un vacuum state. Allo stesso modo sia γ^* una 1-chain composta da lati del reticolo duale $\bar{G}$ congiunti due a due, e

$$O_{\gamma^*}^x = \prod_{j\,:\,\bar{e}_j \in \gamma^*} X_j \tag{3.4.3}$$

l'operatore di trasporto costruito su γ^*. Allora se γ^* è chiuso

$$\left|\psi_0^{(1)}\right\rangle'' = O_{\gamma^*}^x \left|\psi_0^{(1)}\right\rangle \tag{3.4.4}$$

è un vacuum state, così come

$$\left|\psi_0^{(1)}\right\rangle''' = O_{\gamma^*}^x \circ O_\gamma^z \left|\psi_0^{(1)}\right\rangle . \tag{3.4.5}$$

Si vedano in Figura 3.4.1 esempi di operatori di trasporto definiti su G e $\bar{G}$.

Siano $\{v_k\} \subseteq V$ e $\{f_m\} \subseteq F$ due sottoinsiemi di V e F, si definiscono

$$F_A = \prod_k A_{v_k}, \quad F_B = \prod_m B_{f_m} \tag{3.4.6}$$

composizioni di un numero finito di operatori di vertice e di faccia rispettivamente. Si noti come questi operatori non trasportano quasi-particelle, al massimo deformano i cammini su cui agiscono (Fig. 3.4.2).

Valgono

$$F_A \circ O_{\gamma^*}^x = O_{\gamma^*}^x \circ F_A = I \tag{3.4.7}$$

$$F_B \circ O_\gamma^z = O_\gamma^z \circ F_B = I \tag{3.4.8}$$

se e solo se γ e γ^* sono cammini chiusi *contraibili* [8] che contengono le facce e i vertici su cui agiscono F_A e F_B. Dunque γ e γ^* possono essere considerate discretizzazione di curve omologicamente equivalenti (Sez. 3.1) ad un punto sul toro se la (3.4.7) e la (3.4.8) sono soddisfatte. Questo ci

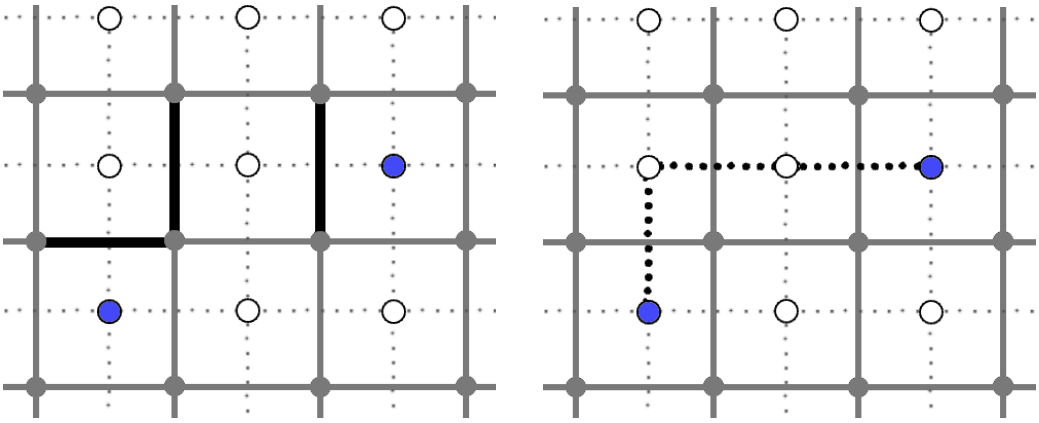


Figura 3.4.1: Nella prima figura una coppia di quasi-particelle è creata a causa dell'azione degli operatori X_j sulla 1-chain di G in grassetto. Nella seconda figura una coppia di quasi-particelle è creata a causa dell'azione degli operatori X_j sulla 1-chain duale di $\bar{G}$ in grassetto tratteggiato.

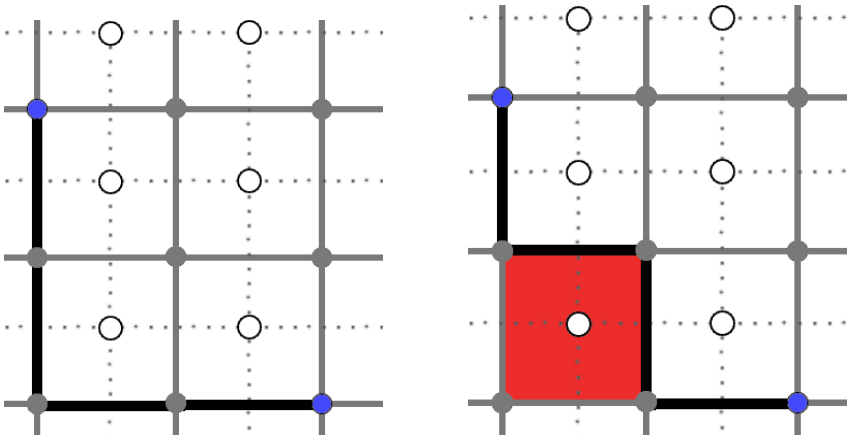


Figura 3.4.2: Sulla sinistra sono mostrate un paio di quasi-particelle generate dall'azione dell'operatore di trasporto O^z_γ lungo il cammino γ sottolineato in grassetto. Sulla destra la deformazione del cammino in seguito all'azione dell'operatore di faccia B_f su una delle facce per cui per un lato e vale $e \in \gamma, e \in \partial f$.

porta a considerare gli stati (3.4.2), (3.4.4) e (3.4.5) riconducibili allo stato fondamentale (3.2.9).

Due curve appartenenti al toro $\mathcal{T}$ e non contraibili sono $\bar{\gamma}_1$ e $\bar{\gamma}_2$ mostrate in Figura 3.4.3 [17]. Gli stati

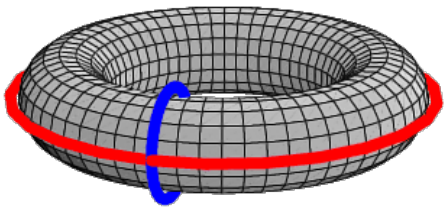


Figura 3.4.3: Sul toro sono evidenziate le discretizzazioni delle curve che generano l'omologia del toro $\mathcal{T}$ [8]. In blu la curva $\bar{\gamma}_1$ che circonda l'anello del toro e in blu la rosso $\bar{\gamma}_2$ che ne circonda il foro.

$$\left|\psi_0^{(2)}\right\rangle = O^{\alpha}_{\bar{\mathcal{C}}_1} \left|\psi_0^{(1)}\right\rangle, \tag{3.4.9}$$

$$\left|\psi_0^{(3)}\right\rangle = O^{\beta}_{\bar{\mathcal{C}}_2} \left|\psi_0^{(1)}\right\rangle, \tag{3.4.10}$$

$$\left|\psi_0^{(4)}\right\rangle = O^{\alpha}_{\bar{\mathcal{C}}_1} \circ O^{\beta}_{\bar{\mathcal{C}}_2} \left|\psi_0^{(1)}\right\rangle, \tag{3.4.11}$$

risultano essere vacuum states. $\bar{\mathcal{C}}$ indica un cammino non contraibile di G se $\alpha, \beta = z$ oppure di $\bar{G}$ nel caso in cui $\alpha, \beta = x$.

Si procederà ora a spiegare come di questi stati siano di particolare interesse. Così come il gate Z non effettua la transizione $|0\rangle \leftrightarrow |1\rangle$, allo stesso modo un operatore del tipo $O^{z}_{\bar{\gamma}}$ che agisce sullo stato (3.2.9) non genera uno stato completamente indipendente da quello iniziale. Dunque gli unici vacuum states descritti finora indipendenti dallo stato (3.2.9) sono

$$\left|\psi_0^{(2)}\right\rangle = O^{x}_{\bar{\gamma}_1} \left|\psi_0^{(1)}\right\rangle, \tag{3.4.12}$$

$$\left|\psi_0^{(3)}\right\rangle = O^{x}_{\bar{\gamma}_2} \left|\psi_0^{(1)}\right\rangle, \tag{3.4.13}$$

$$\left|\psi_0^{(4)}\right\rangle = O^{x}_{\bar{\gamma}_1} \circ O^{x}_{\bar{\gamma}_2} \left|\psi_0^{(1)}\right\rangle. \tag{3.4.14}$$

Questo può essere compreso meglio osservando che gli operatori B_f e Z commutano, dunque Z non è in grado di deformare gli stati del reticolo. Si dice che lo stato è *quadruplamente degenere*, come ci si si aspettava dall'Eq 3.2.5. Questo può essere interpretato come l'esistenza di quattro distinte fasi corrispondenti alla medesima energia, che esistono a causa della topologia della superficie in cui è immerso il reticolo. Si procederà indicando un generico vacuum state con $|\psi_0\rangle$.

3.5 Error Correction nel Toric Code

Si delineano ora le idee fondamentali per la correzione pratica degli errori in computazione. Si consideri la situazione in cui viene misurato uno stato $|\psi\rangle$ eccitato che possiamo immaginare essere lo stato fondamentale (3.2.9) con la presenza con due quasi-particelle di tipo m, come per esempio quello mostrato in Fig. 3.4.1. Possiamo considerare questo stato come generato dall'azione di un operatore di trasporto $O^x_{\gamma^\star}$ (3.4.3) sul vacuum state, con $\gamma^\star$ un cammino aperto. Si noti come quest'ultimo non è unico. Ricordando che lo spazio V_S stabilizzato dal codice contiene tutti gli stati eccitati su cammini chiusi contraibili (3.4.2) (3.4.4), è sufficiente correggere lo stato $|\psi\rangle = O^x_{\gamma^\star}\left|\psi_0^{(1)}\right\rangle$ completando il cammino $\gamma^\star$ in modo da renderlo chiuso e contraibile attraverso una recovery map $\mathcal{R}$ per cui

$$\mathcal{R}\,|\psi\rangle = \left|\psi_0^{(1)}\right\rangle. \tag{3.5.1}$$

E' importante che $\mathcal{R}$ completi $\gamma^\star$ rendendolo chiuso contraibile altrimenti

$$\mathcal{R}\,|\psi\rangle = \left|\psi_0^{(i)}\right\rangle \neq \left|\psi_0^{(1)}\right\rangle, \qquad i = 2,3,4, \tag{3.5.2}$$

incorrendo in quello che viene detto *errore logico*.

I modi con cui completare $\gamma^\star$ non è unico e per convenzione si sceglie il cammino chiuso più corto. Si consideri la situazione in cui si verificano numerosi errori su qubit su lati allineati, ovvero la situazione più scomoda da risolvere. Se si si completa $\gamma^\star$ con il cammino più corto allora il Toric

Code può risolvere al massimo $\frac{N-1}{2}$ errori di questo tipo. Infatti se ci sono più di $\frac{N-1}{2}$ adiacenti completare $\gamma^\star$ porta alla creazione di un cammino non contraibile, incorrendo in un errore logico. Asintoticamente il numero di errori è $N \cdot p$ con $p < \frac{1}{2}$ e dunque si avranno sempre un numero di errori maggiore di $\frac{N-1}{2}$. Per questo motivo nella pratica è necessario individuare tutte le eccitazioni la cui correzione può portare ad un errore logico [10] per evitare di incorrere in errori di computazione.

Conclusioni

All'inizio dell'elaborato sono state introdotte le nozioni generali per poter descrivere i computer quantistici, i codici quantistici e gli algoritmi quantistici. Poi si è proceduto nel descrivere i gate su qubit singolo introducendo i più importanti e le loro correlazioni. Dunque sono stati introdotti i gate su qubit multipli, ponendo particolare attenzione sui controlled gates tra cui il CNOT gate e il Toffoli gate, di fondamentale importanza per mostrare la relazione tra computazione classica e quantistica. In seguito si è spiegato il processo di misura e come questo interagisca con il sistema. Quindi si è studiato come poter approssimare qualsiasi trasformazione attraverso un insieme finito di gates. A questo fine sono stati riportati i two-level operators e studiato il concetto di universalità di un set di gate. Infine è stato riportato il teorema di Solovay-Kitaev.

Il formalismo per la descrizioni degli errori e per la loro correzione è realizzato partendo dalle matrici di densità del sistema, di particolare utilità nella descrizione di sistemi aperti. Sono state definite le quantum operations, attraverso cui si descrivono gli errori, di cui si è riportato qualche esempio. Quindi sono state poste le basi della Quantum Error Correction e il caso particolare degli Stabilizer Codes, con qualche esempio significativo.

Infine nell'ultimo capitolo si è studiato il Toric Code. Per poter capire come questo codice sia strutturato e definito è stato necessario prima porre le basi dei complessi di catene Z_2 su un reticolo. In seguito è stata descritta la struttura e le caratteristiche del codice, in particolare i generatori dello spazio stabilizzatore. Sono state dunque studiate alcune delle possibili eccitazioni

possibili del sistema e come individuarle, associandole a particelle fittizie dette quasi-particelle. Successivamente si è studiato lo stato fondamentale, approfondendone le caratteristiche e analizzandone la degenerazione, collegando i risultati ottenuti alla topologia del toro. Infine sono state poste le idee fondamentali per la correzione pratica in seguito alla rilevazione di errori in computazione.

Appendice A

Approssimazione di $R_{\hat{n}}(\alpha)$ attraverso $R_{\hat{n}}(\theta)$

Sia $\delta > 0$ la precisione desiderata del circuito e $N > 2\pi/\delta$ un numero intero. Si definisce $\theta_k \in [0, 2\pi]$ tale che $\theta_k = (k\theta) mod(2\pi)$. Per il principio dei cassetti[1] di Dirichlet esistono j, k distinti compresi tra 1 e N per cui vale

$$|\theta_{k-j}| = |\theta_k - \theta_j| < \frac{2\pi}{N} < \delta. \tag{A.0.1}$$

Siccome $j \neq k$ e θ è un multiplo irrazionale di 2π deve essere $\theta_{k-j} \neq 0$. Segue che la sequenza $\theta_{l(k-j)}$ ricopre tutto $[0, 2\pi]$ al variare di l. Allora per ogni $\epsilon > 0$ esiste n per cui

$$E(R_{\hat{n}}(\alpha), R_{\hat{n}}(\theta)^n) < \frac{\epsilon}{3}. \tag{A.0.2}$$

[1] Il principio dei cassetti afferma che se A e B sono due insiemi finiti e B ha cardinalità strettamente minore di A, allora non esiste alcuna funzione iniettiva da A a B.

Appendice B

Rappresentazione di Fock

Si definiscono gli operatori di creazione e distruzione rispettivamente

$$a^\dagger = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad a = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \tag{B.0.1}$$

tale che

$$a^\dagger a + a a^\dagger = I. \tag{B.0.2}$$

Allora

$$a^\dagger |0\rangle = |1\rangle \;\text{ e }\; a|1\rangle = |0\rangle, \tag{B.0.3}$$

$$a^\dagger |1\rangle = a|0\rangle = 0, \tag{B.0.4}$$

da cui si può intuire il motivo del nome di questi operatori. Gli operatori di Pauli possono essere costruiti attraverso gli operatori di costruzione e distruzione, infatti

$$X = a^\dagger a \quad Z = a a^\dagger - a^\dagger a. \tag{B.0.5}$$

Esprimendo gli operatori di Pauli secondo la (B.0.5), i generatori dello stabilizzatore (3.2.2) che costruiscono l'Hamiltoniana (3.2.8), le eccitazioni (3.3.1) (3.3.10) così come gli operatori di trasporto (3.3.15) sono tutti esprimibili attraverso gli operatori di creazione e distruzione di Fock. Lo spettro energetico del Toric Code può essere descritto conoscendo le caratteristiche dello stato fondamentale e tenendo conto delle eccitazioni rappresentate dalla creazione e distruzione di quasi-particelle.

Bibliografia

[1] M. F. Araujo de Resende, A pedagogical overview on 2D and 3D Toric Codes and the origin of their topological orders, Reviews in Mathematical Physics, 2019. arXiv:1712.01258v2

[2] G. Birkhoff, Lattice Theory, American Mathematical Society, 1967.

[3] C. M. Dawson, M. A. Nielsen, The Solovay-Kitaev algorithm, 2005, arXiv:quant-ph/0505030v2.

[4] K. Fujii, Quantum computation with topological codes: from qubit to topological fault-tolerance (Vol. 8), Springer, 2015.

[5] C. Gardiner, P. Zoller, Quantum Noise, Springer Verlag, 2004.

[6] D. J. Griffiths, S. F. Darrell, Introduction to quantum mechanics, Cambridge University Press, 2018.

[7] B. C. Hall, Lie Groups, Lie Algebras, and Representations: An Elementary Introduction, Graduate Texts in Mathematics, Springer Verlag, 2015.

[8] A. Hatcher, Algebraic Topology, Cambridge University Press, 2002.

[9] C. Itzykson, J. B. Zuber, Quantum Field Theory, McGraw-Hill Inc., 1980.

[10] MJ. Kastoryano, Lecture notes of the Quantum Error Correction at University of Cologne, Wintersemester 2018/2019. http://www.thp.uni-koeln.de/kastoryano/ExSheets/Notes_v7.pdf

[11] A. Kay, Tutorial on the Quantikz Package, Royal Holloway University of London, 2019, arXiv:1809.03842v5.

[12] K. Kraus, States, Effects and Operations: Fundamental Notions of Quantum Theory, Springer Verlag, 1983.

[13] G. Lindblad, G. A General, No-Cloning Theorem. Letters in Mathematical Physics 47, 1999.

[14] C. Nayak, S. H. Simon, A. Stern, M. Freedman, S. D. Sarma, Non-Abelian anyons and topological quantum computation. Reviews of Modern Physics, 80(3), 1083, 2018.

[15] M. A. Nielsen, I. L. Chuang, Quantum Computation and Information Theory, Cambridge University Press, 2010.

[16] www.geogebra.org/3d

[17] mathworld.wolfram.com/images/eps-gif/IntersectionHomologyTorus_800.gif