

Università degli Studi di Napoli “Federico II”

Scuola Politecnica e delle Scienze di Base
Area Didattica di Scienze Matematiche Fisiche e Naturali

Dipartimento di Fisica “Ettore Pancini”



Laurea triennale in Fisica

Simulazione di Quantum Annealing per un problema di Number Partitioning

Relatori:

Dott. Procolo Lucignano
Prof. Vittorio Cataudella

Candidato:

Sara Giordano
N85000598

A.A. 2016/2017

Indice

1	Limiti dei computer classici e Quantum Computation	5
1.1	Classi di complessità computazionale	5
1.2	Quantum Computation: alternativa alla computazione classica	7
2	Quantum Annealing	9
2.1	Problemi di ottimizzazione	9
2.2	L'idea del Quantum Annealing (o Adiabatic Quantum Computation)	10
2.3	Modello di Ising: legame con il Quantum Annealing	11
2.4	Teorema Adiabatico e problema di Landau-Zener	12
3	Problema di Number Partitioning	19
3.1	Classificazione di complessità del problema	19
3.2	Hamiltoniana del problema	20
3.3	Simulazione di Quantum Annealing in ambiente Mathematica	22
3.3.1	Matrice Densità ed equazione di Liouville	23
3.3.2	Implementazione dell'algoritmo	25
3.3.3	Risultati ottenuti	30
4	Conclusioni	37

Capitolo 1

Limiti dei computer classici e Quantum Computation

In ambito scientifico esiste un'ampia gamma di problemi (di natura matematica, fisica o informatica) per i quali, allo stato attuale, non si riescono ad elaborare algoritmi di risoluzione che una macchina deterministica possa eseguire in un tempo polinomiale in funzione dei dati in entrata. Più chiaramente: il tempo di esecuzione dell'algoritmo (e quindi di risoluzione del problema) è una funzione non polinomiale del numero di input del problema stesso. Ciò vuol dire che al crescere dei dati iniziali il tempo di risoluzione aumenta, ad esempio, esponenzialmente.

Come "macchina deterministica", si intende un macchinario che esegua un algoritmo deterministico, ad esempio un computer. Il tempo impiegato è proporzionale al numero di operazioni necessarie a risolvere il problema, se si ritiene fissato il tempo necessario a svolgere una singola operazione.

1.1 Classi di complessità computazionale

La teoria della complessità computazionale [10] classifica molti problemi computazionali attraverso delle classi di complessità basate sulle risorse necessarie a risolvere questi problemi. Se consideriamo come risorsa il tempo di risoluzione, allora alcune classi di complessità sono: classe NP (ovvero Non-deterministic Polynomial), $NP - complete$, $NP - hard$. La sigla NP sta a indicare che, teoricamente, una macchina non-deterministica può risolvere il problema in un tempo polinomiale. NP è la classe di problemi per i quali, data una soluzione, la validità di questa può essere verificata in un tempo polinomiale su un normale computer. La classe analoga è quella dei problemi P (polynomial), che comprende problemi risolvibili in tempo polinomiale da una macchina deterministica. Per comprendere meglio la differenza fra le classi P e NP consideriamo ad esempio il problema di trovare i fattori primi di un numero intero n . Per questo tipo di problema non si conoscono

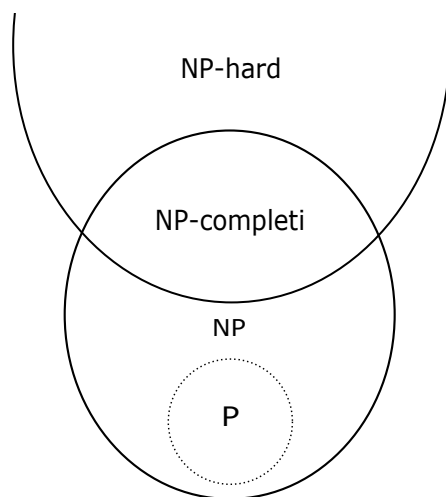


Figura 1.1: Relazioni fra alcune classi di complessità con $P \neq NP$

ancora delle soluzioni "veloci" su computer classici, questo suggerisce che il problema non appartenga alla classe P . Allo stesso tempo però se viene dato un numero x che si dice sia fattore di n , possiamo verificare la veridicità di questa affermazione molto velocemente, in un tempo polinomiale. La fattorizzazione quindi è un problema che appartiene alla classe NP .

Ecco un quadro più schematico delle relazioni fra alcune delle varie classi di complessità: la classe di problemi NP include la classe P ($P \subseteq NP$). Alcuni problemi NP , detti NP -*completi*, formano una sottoclasse di NP . Sono problemi NP "più difficili" degli altri; in altre parole trovare un algoritmo che risolva in tempo polinomiale un problema NP -*completo* permette di poter ricavare da esso risoluzioni di tutti gli altri problemi della classe NP in un tempo polinomiale. Si conoscono centinaia di problemi di particolare rilevanza che appartengono alla classe degli NP -*completi* [6]. I problemi NP -*completi* fanno parte, inoltre, della classe NP -*hard*. Quest'ultima è composta da problemi almeno difficili quanto i problemi NP -*completi* e non si riduce solo a problemi decisionali ma anche a problemi di ottimizzazione e di ricerca. Un problema è considerato NP -*hard* se un qualunque problema NP può essere ridotto ad esso polinomialmente.

In questa classificazione abbiamo considerato solo il tempo come risorsa per definire la complessità di un algoritmo, considerando la risorsa spazio come illimitata. Ciò non è sicuramente realistico e andrebbe analizza-

ta anche l'ottimizzazione dell'algoritmo rispetto allo spazio di archiviazione disponibile. A questo livello di approfondimento però possiamo trascurare la componente spaziale nell'ottimizzazione concentrandoci sulla parte temporale.

1.2 Quantum Computation: alternativa alla computazione classica

Si evidenzia quindi l'esistenza di una vasta gamma di problemi di forte interesse scientifico che risultano intrattabili con computer classici. Alcuni esempi importanti sono: Soddisfacibilità (o 3SAT) e Exact Cover nel gruppo dei "covering problems", Number Partitioning nei problemi di partizione, il problema del commesso viaggiatore, la fattorizzazione di grandi interi e molti altri ancora.

Nella trattazione di problemi del genere entra, durante gli anni '90, la meccanica quantistica con l'ideazione della "Quantum Computation" alternativa alla computazione classica. Un risultato importante che spinge ad approfondire questo nuovo approccio venne nel 1994 da Peter Schor [14]: la prova che il problema della fattorizzazione di grandi interi (codificati da N bit), ritenuto un problema classicamente appartenente alla classe NP , può essere risolto con un numero di operazioni quantistiche che cresce polinomialmente in funzione di N , cioè in una classificazione quantistica della complessità risulta appartenere a una classe analoga a P .

Un computer quantistico è basato essenzialmente su circuiti analoghi ai circuiti informatici ma, appunto, *quantistici*. Si tratta di un modello elaborato nel 1989 da Deutsch [17] che prevede una serie di operazioni elementari sui q-bit, analoghi ai bit della computazione classica. Un q-bit, a differenza di un normale bit che può prendere uno solo fra i valori 0 e 1, si troverà sempre in uno stato di sovrapposizione di $|0\rangle$ e $|1\rangle$, cioè lo *stato* del q-bit sarà una combinazione lineare dei due. Nel caso si tratti un sistema di spin quantistici si può facilmente identificare un q-bit con lo spin di una particella: lo spin-up $|+\frac{1}{2}\rangle$ sarà analogo a $|1\rangle$ e lo spin-down $|-\frac{1}{2}\rangle$ analogo al $|0\rangle$. Si utilizzano quindi, analogamente all'informatica in senso classico, delle porte logiche quantistiche; degli operatori unitari che agiscono sul prodotto tensore fra spin di molte particelle come delle funzioni logiche booleane. Questo tipo di approccio è universale: ogni tipo di problema di computazione semplice può essere implementato attraverso una combinazione appropriata di *quantum logic gates* [2]. Il problema principale nell'elaborare algoritmi attraverso circuiti quantistici si trova nelle scale temporali dell'elaborazione di questi che incontrano il problema, intrinseco alla meccanica quantistica, della *decoerenza* e del *rilassamento*, derivanti dall'interazione di un sistema quantistico con l'ambiente esterno. La decoerenza ha un effetto distruttivo sull'informazione trasportata nel modello circuitale quantistico poiché parte

di questa informazione viene immagazzinata proprio nelle fasi relative degli stati di base e, di conseguenza, quando il tempo di elaborazione di un algoritmo supera il tempo di coerenza degli stati si ha perdita di informazione. Essendo stato dimostrato che la Quantum Adiabatic Computation è polinomialmente equivalente al modello circuitale quantistico [16] si può pensare ad essa come un'alternativa alla Quantum Computation classica essendo, inoltre, meno inficiata da problemi di decoerenza.

Capitolo 2

Quantum Annealing

L'elaborazione di algoritmi quantistici, che utilizzino la Quantum Information tradizionale basata sulla *Circuit Theory* (Universal Quantum Computation), può richiedere tempi comparabili con quelli che intervengono in fenomeni di decoerenza [16]. Un altro metodo di computazione quantistica, che sembra meno affetto da decoerenza, ha radice nello sfruttare i risultati del Teorema Adiabatico della Meccanica Quantistica: la Adiabatic Quantum Computation basata sul concetto di annealing quantistico.

2.1 Problemi di ottimizzazione

Alcuni dei problemi più trattati con questo metodo sono problemi di ottimizzazione [5]. Si tratta di problemi modellizzabili attraverso la ricerca del minimo di una funzione, detta funzione costo, che caratterizza il sistema in esame. In genere la funzione costo da minimizzare dipende da un numero molto grande di variabili e ciò rende difficile e non immediata la ricerca della soluzione ottimale. Questo è dovuto a due fattori principali: lo spazio delle configurazioni dove va ricercato il minimo è di per sé molto grande, inoltre è possibile che ci siano vincoli nel problema grazie ai quali la funzione costo potrebbe possedere molti minimi relativi. Prendiamo, ad esempio, il problema del "Commesso Viaggiatore". Consiste nel trovare il percorso ottimale che un fattorino deve percorrere per raggiungere N punti su una mappa minimizzando il consumo di risorse; in questo problema lo spazio delle configurazioni, nel quale va ricercato un minimo, cresce di dimensione esponenzialmente con N e non si conosce una regola standard per trovare la soluzione ottimale in questo spazio [1].

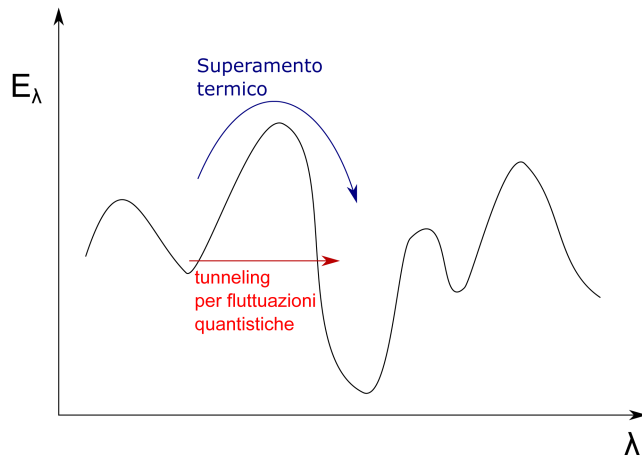


Figura 2.1: Superamento delle barriere fra un minimo relativo e il minimo assoluto "termicamente" o per tunneling quantistico

2.2 L'idea del Quantum Annealing (o Adiabatic Quantum Computation)

L'idea principale ha radici nel Classic Thermal Annealing (CA) [5]. Analogamente a quest'ultimo, per raggiungere in modo più efficiente il minimo assoluto dell'energia, si introduce la natura quantistica del sistema attraverso un termine inizialmente dominante che va spegnendosi mentre il sistema evolve. Ciò ci permette di non rimanere "bloccati" in minimi locali dell'energia, ma di spostarci da questi attraversando per effetto tunnel i massimi che si frappongono fra un minimo e l'altro. Nell'annealing classico ciò è possibile grazie a fluttuazioni termiche che permettono invece di "superare" tali massimi locali. [Figura 2.2].

Supponiamo ad esempio di avere un'Hamiltoniana H_1 che rappresenta l'energia del sistema in esame e quindi la funzione da ottimizzare. Il suo stato fondamentale codifica la soluzione del nostro problema. Di questa Hamiltoniana conosciamo solo la forma; la determinazione dello stato fondamentale non è banale e lo spazio delle configurazioni del sistema, in cui va ricercato il minimo dell'energia, è molto grande. Prendiamo ora una seconda Hamiltoniana H_0 di cui conosciamo lo spettro; la loro somma darà l'Hamiltoniana H totale del nostro sistema. Introduciamo la dipendenza temporale come segue:

$$H(t) = \left(1 - \frac{t}{t_f}\right)H_0 + \frac{t}{t_f}H_1$$

Inizialmente si prepara il sistema nello stato fondamentale dell'Hamiltoniana H_0 che chiamiamo $|\psi(0)\rangle = |\psi_0\rangle$. A $t = 0$ si ha infatti $H(0) = H_0$. All'au-

mentare di t verso un tempo finale t_f il termine H_0 si riduce sempre di più, mentre si accende il termine H_1 . Al tempo finale si avrà $H(t_f) = H_1$ e se l'evoluzione del sistema, governata dall'equazione di Schrödinger

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = H(t) |\psi(t)\rangle$$

è stata sufficientemente "lenta", grazie al Teorema Adiabatico [8] il nostro sistema si troverà nello stato fondamentale di $H(t)$: $|\psi(t_f)\rangle = |\psi_1\rangle$ che al tempo $t = t_f$ coincide con lo stato fondamentale di H_1 e quindi con la soluzione del nostro problema iniziale. Successivamente vedremo come stimare un tempo di evoluzione t_f sufficiente a permettere un'evoluzione adiabatica del sistema.

Alla fine del processo di annealing un possibile stimatore per valutare la bontà di questo approccio è l'energia residua ϵ_{res} : si tratta della differenza fra l'energia dello stato finale $|\psi(t_f)\rangle$, raggiunto attraverso l'evoluzione adiabatica del sistema, e l'energia attesa dello stato fondamentale di H_1 , ϵ_0 , che possiamo calcolare analiticamente per casi semplici.

2.3 Modello di Ising: legame con il Quantum Annealing

Ideato da Ernst Ising nel 1920 [4] questo modello nacque per studiare le proprietà magnetiche di alcuni materiali; in particolare le transizioni di fase fra stati di magnetizzazione differenti. Il modello descrive l'interazione fra particelle poste nei vertici di un reticolo in uno spazio R^d . Indichiamo con: Z^d la partizione di R^d occupata dal reticolo, con $n_i = 0, 1$ il numero di occupazione del sito del reticolo e con $n = \{n_i\}_{i \in Z^d}$ la generica configurazione delle particelle. L'Hamiltoniana del modello, nel caso più generale è la seguente:

$$H(n) = - \sum_{i \neq j} \phi_{ij} n_i n_j$$

Dove ϕ_{ij} rappresenta l'interazione fra la particella nel sito i e la particella nel sito j . Un'applicazione meno generale è quella usata per descrivere un sistema di spin classici di Ising. Nel sito i -esimo la configurazione della particella è data dal suo spin classico: $s_i = \pm 1$. Questo permette di definire il sistema completamente assegnando una configurazione di spin $s = \{s_i\}_{i \in \Lambda}$ dove Λ è l'insieme dei punti di Z^d che costituisce il reticolo. Lo spazio delle configurazioni avrà dimensione $2^{\#(\Lambda)}$ (con $\#(\Lambda)$ cardinalità dell'insieme Λ). L'Hamiltoniana del modello così definito sarà:

$$H_\Lambda(s) = - \sum_{i \in \Lambda} h_i s_i - \sum_{\substack{i, j \in \Lambda \\ i \neq j}} J_{ij} s_i s_j + W_\Lambda(s)$$

Dove il primo termine descrive l'interazione con un campo magnetico h , il secondo descrive l'interazione fra diversi spin del sistema attraverso la matrice J_{ij} e il terzo termine descrive eventuali condizioni al bordo.

La scelta di un'Hamiltoniana di Ising è guidata dal fatto che la ricerca del ground state di questo modello è esso stesso un problema NP-hard da risolvere per un computer classico e appartiene anche alla classe di problemi NP-completi [7], di conseguenza è possibile ricondursi polinomialmente alla soluzione di tutti gli altri problemi NP-completi una volta trovata per questo modello una soluzione efficiente.

In particolare possiamo vedere un vetro di spin di Ising come un problema NP-completo quando il problema da risolvere è scoprire l'energia dello stato fondamentale del sistema. Se invece ci chiediamo *quale sia* lo stato fondamentale, stiamo cercando di risolvere un problema NP-hard. Date le proprietà della classe di problemi NP-completi possiamo, da qualunque problema NP-completo, ricondurci all'Hamiltoniana del modello di Ising e, risolvendo il problema decisionale sull'energia dello stato fondamentale, stiamo risolvendo il problema NP-completo di interesse.

2.4 Teorema Adiabatico e problema di Landau-Zener

Prendiamo in considerazione prima in generale un sistema descritto da un'Hamiltoniana $H(t)$ che per il tempo $t = 0$ si trova in un autostato $\psi(t = 0)$ di $H(t = 0)$. Cambiando le condizioni del sistema con una perturbazione che agisca in un tempo $t_1 - t_0 = \Delta t \rightarrow \infty$ il sistema potrà adattarsi ai cambiamenti che avvengono con la perturbazione e istante per istante il sistema si troverà nell'autostato dell'Hamiltoniana $H(t)$ *corrispondente* all'autostato iniziale $\psi(t = 0)$. Questo processo è detto "processo adiabatico". Un "processo diabatico" al contrario avviene quando una perturbazione agisce molto rapidamente $\Delta t \rightarrow 0$. In questo caso il sistema non ha tempo di adattarsi alle condizioni nuove e lo stato finale risulta essere una sovrapposizione di autostati dell'Hamiltoniana $H(t_1)$. Il teorema adiabatico venne dimostrato nel 1928 da M. Born e V. A. Fock [11] e, nell'ambito del quantum annealing, ci permette di arrivare allo stato fondamentale dell'Hamiltoniana che codifica il nostro problema da uno stato fondamentale di un'Hamiltoniana iniziale diversa. La perturbazione che applichiamo al sistema deve quindi essere di tipo adiabatico, cioè svolgersi in un tempo lungo al punto tale da permettere l'adiabaticità. Due principali assunzioni vennero fatte da M. Born e V. A. Fock per provare il teorema adiabatico:

- lo spettro dell'Hamiltoniana $H(t)$ deve essere un puro spettro discreto istante per istante. In altre parole deve essere possibile quantificare il gap energetico fra un autostato e l'altro.

- gli autovalori non devono essere degeneri, devo poter quindi identificare univocamente lo stato in cui si trova il sistema dalla sua energia.

Successivamente il teorema venne provato anche in assenza di queste condizioni [12].

Determinare in quali condizioni una transizione, che avviene in un tempo Δt , è o meno una transizione adiabatica richiede la risoluzione del Landau-Zener problem [13]. È un modello molto semplice di un sistema a due stati, che può essere poi generalizzato a sistemi più complessi. Consideriamo questo sistema che, da $t = -\infty$ fino all'istante di tempo in cui inizia la perturbazione, è rappresentato da un'Hamiltoniana con due autostati $|\psi_1\rangle$ e $|\psi_2\rangle$ detti "autostati diabatici". Viene poi accesa una perturbazione che agisce sul sistema per un intervallo di tempo Δt . Durante la perturbazione gli autostati del sistema sono una combinazione lineare di $|\psi_1\rangle$ e $|\psi_2\rangle$:

$$|\Phi^{1,2}\rangle = c_1(t) |\psi_1\rangle + c_2(t) |\psi_2\rangle$$

dove $c_1(t)$ e $c_2(t)$ sono le ampiezze di probabilità degli autostati diabatici. L'Hamiltoniana del problema è:

$$H = at\sigma_z + b\sigma^+ + b^*\sigma^-$$

e per $b \in \mathbb{R}$ si ha:

$$H = at\sigma_z + b\sigma_x$$

con:

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

matrici di Pauli, e

$$\sigma^+ = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad \sigma^- = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$$

con la relazione

$$\sigma_x = \frac{\sigma^+ + \sigma^-}{2}.$$

Questa Hamiltoniana applicata allo stato $|\Phi^{1,2}\rangle$ dà la seguente equazione di Schrödinger:

$$i\hbar \frac{\delta}{\delta t} \begin{pmatrix} c_1(t) \\ c_2(t) \end{pmatrix} = \begin{bmatrix} at & b \\ b^* & -at \end{bmatrix} \begin{pmatrix} c_1(t) \\ c_2(t) \end{pmatrix}.$$

Questo modello ha in sé alcune semplificazioni:

- La perturbazione al sistema è una funzione lineare del tempo con "velocità" data da a . Questa variabile a indica il rate di variazione di energia, infatti dimensionalmente si ha $[a] = \left[\frac{E}{t} \right]$

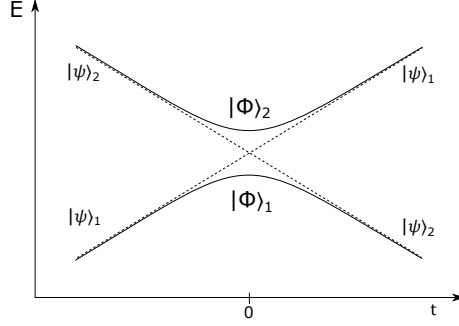


Figura 2.2: Sistema a due stati di Landau-Zener

- La differenza di energia fra gli autostati diabatici del sistema varia anch'essa linearmente nel tempo
- Il termine di accoppiamento b è costante nel tempo

Per il grafico dello spettro dell'Hamiltoniana vedere la Fig.[2.2] Calcolando gli autovalori corrispondenti ai due autovettori adiabatici otteniamo:

$$E^{1,2} = \pm \sqrt{(at)^2 + b^2}$$

da cui si ricava che il minimo gap energetico esistente fra i due autostati è al tempo $t = 0$:

$$\Delta_{min} = E^2 - E^1 = 2|b|.$$

Per risolvere l'equazione di Schrödinger bisogna quindi risolvere un sistema di due equazioni differenziali accoppiate, per determinare la dipendenza temporale dei coefficienti di $|\psi_1\rangle$ e $|\psi_2\rangle$. Da essi si ricava la probabilità che, partendo dallo stato adiabatico $|\Phi^1\rangle$ coincidente a $t \rightarrow -\infty$ con lo stato diabatico $|\psi_1\rangle$, alla fine dell'evoluzione temporale del sistema ci si trovi nello stesso stato adiabatico $|\Phi^1\rangle$ e si abbia così una *transizione di Landau Zener*. Il sistema di equazioni differenziali accoppiate è il seguente:

$$\begin{cases} i\hbar\dot{c}_1(t) = atc_1(t) + bc_2(t) \\ i\hbar\dot{c}_2(t) = b^*c_1(t) - atc_2(t) \end{cases}$$

Disaccoppiando le equazioni si arriva a

$$\ddot{c}_1(t) + \left(\frac{ia}{\hbar} + \frac{a^2 t^2}{\hbar^2} + \frac{|b|^2}{\hbar^2} \right) c_1(t) = 0 \quad (2.1)$$

(analogamente per $c_2(t)$). L. Landau, C. Zener et al. nel 1932 [13] studiarono questo sistema risolvendo questo tipo di equazione differenziale di secondo grado, ottenendo la *Formula di Landau-Zener* che si riferisce alla probabilità di una transizione *adiabatica*

$$P_{|\psi_1\rangle}(t \rightarrow +\infty) = e^{-\pi|b|^2/\hbar a}$$

per un'evoluzione adiabatica questa probabilità deve essere quindi $P_{|\psi_1\rangle} \approx 0$, vediamo che questo accade quanto più il fattore a risulta piccolo rispetto a $|b|^2/\hbar$ che è definita dalla gap istantanea a $t = 0$, cioè la minima gap, e quindi quanto più la perturbazione avviene "lentamente". Mostriamo come arrivare alla *Formula di Landau-Zener* senza risolvere direttamente le equazioni differenziali di secondo grado che si presentano. Prima di tutto vediamo che per $b = 0$ si avrebbero due equazioni differenziali del secondo ordine disaccoppiate

$$i\hbar \frac{\delta}{\delta t} \begin{pmatrix} c_1(t) \\ c_2(t) \end{pmatrix} = \begin{bmatrix} at & 0 \\ 0 & -at \end{bmatrix} \begin{pmatrix} c_1(t) \\ c_2(t) \end{pmatrix} \rightarrow \begin{cases} i\hbar \frac{\delta c_1(t)}{\delta t} = at c_1(t) \\ i\hbar \frac{\delta c_2(t)}{\delta t} = -at c_2(t) \end{cases}$$

con soluzioni:

$$c_1(t) = c_1(t_0) e^{-\frac{i}{\hbar} \int_{t_0}^t at' dt'} \quad c_2(t) = c_2(t_0) e^{\frac{i}{\hbar} \int_{t_0}^t at' dt'}$$

consideriamo quindi di esprimere i coefficienti $c_1(t)$ e $c_2(t)$ separando il contributo della parte fuori diagonale (parte di interazione) dalla parte in diagonale in questo modo:

$$c_1(t) = g_1(t) e^{-\frac{i}{\hbar} \int_{t_0}^t at' dt'} \quad c_2(t) = g_2(t) e^{\frac{i}{\hbar} \int_{t_0}^t at' dt'}$$

Ciò che ci interessa valutare è la probabilità $|c_1(t)|^2$ al tempo finale t_f , notando che $|c_1(t)|^2 = |g_1(t)|^2$ possiamo sostituire $c_1(t)$ così espressa nell'equazione differenziale di secondo grado (2.1) e ottenere

$$\ddot{g}_1(t) - \frac{2iat}{\hbar} \dot{g}_1(t) + \frac{|b|^2}{\hbar^2} g_1(t) = 0 \quad (2.2)$$

Da risolvere per trovare la soluzione asintotica $g_1^f = g_1(t \rightarrow \infty)$ dalla quale ricaviamo la probabilità $P_{|\psi_1\rangle}(t \rightarrow +\infty) = |g_1^f|^2$. Per risolvere questa equazione differenziale facciamo prima delle considerazioni sulle condizioni iniziali e finali del sistema.

- Sappiamo che i due livelli energetici del sistema, per $t \rightarrow +\infty$, si separano indefinitamente. L'interazione fra i due livelli quindi cessa e $g_1(t)$ tende a un valore costante indipendente dal tempo: g_1^f . Ciò si evince dalla forma delle soluzioni del sistema di equazioni differenziali per $b = 0$.
- Per $t \rightarrow +\infty$ nella matrice

$$\begin{bmatrix} at & b \\ b^* & -at \end{bmatrix}$$

i termini fuori diagonale diventano trascurabili rispetto ai termini in diagonale che dipendono linearmente da t . Ciò ci permette di trascurare nell'equazione (2.2) per $t \rightarrow +\infty$ i termini $\ddot{g}$ e $\dot{g}$ che, con $g_1(t \rightarrow +\infty) = g_1^f$ costante, tendono entrambi a 0.

Nell'equazione (2.2) trascuriamo solamente il termine $\ddot{g}(t)$. L'andamento di $\dot{g}_1(t)$ dovrà essere: $\dot{g}_1(t) \approx 1/t$ per avere $g_1(t \rightarrow +\infty) = g_1^f$ costante (d'ora in poi omettiamo il pedice 1 per comodità):

$$2iat\dot{g}(t) = \frac{|b|^2}{\hbar}g(t)$$

Integrando questa equazione otteniamo

$$g(t) = g(t_0)e^{-\frac{i|b|^2}{2a\hbar}\ln\frac{t}{t_0}}$$

dove t_0 è un istante iniziale arbitrario ma fissato in modo tale che il range di tempo di evoluzione di $g(t)$ sia ampio. Differenziamo ora due volte $g(t)$ per ottenere $\dot{g}(t)$ e $\ddot{g}(t)$

$$\begin{aligned}\dot{g}(t) &= -\frac{i|b|^2g(t_0)}{2a\hbar t}e^{-\frac{i|b|^2}{2a\hbar}\ln\frac{t}{t_0}} \\ \ddot{g}(t) &= \left(i - \frac{|b|^2}{2a\hbar}\right)\frac{|b|^2g(t_0)}{2a\hbar t^2}e^{-\frac{i|b|^2}{2a\hbar}\ln\frac{t}{t_0}}\end{aligned}$$

calcoliamo il rapporto $\frac{\ddot{g}(t)}{g(t)}$, che sarà successivamente utile nel calcolo di g^f

$$\frac{\ddot{g}(t)}{g(t)} = \left(i - \frac{|b|^2}{2a\hbar}\right)\frac{|b|^2}{2a\hbar t^2}$$

questo rapporto può essere valutato in $t = 0$, dalla (2.2) ottenendo

$$\frac{\ddot{g}(0)}{g(0)} = -\frac{|b|^2}{\hbar^2} \quad (2.3)$$

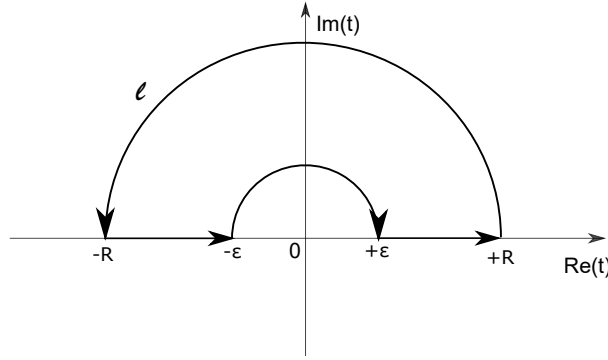


Figura 2.3: Integrale nel piano complesso

Moltiplichiamo ora l'equazione differenziale (2.2) per $\frac{dt}{g(t)t}$ e integriamo da $t = -\infty$ a $t = +\infty$:

$$\int_{-\infty}^{+\infty} dt \frac{\ddot{g}(t)}{tg(t)} - \frac{2ia}{\hbar} \int_{-\infty}^{+\infty} dt \frac{\dot{g}(t)}{g(t)} + \frac{|b|^2}{\hbar^2} \int_{-\infty}^{+\infty} \frac{dt}{t} = 0$$

Vediamo che

$$\int_{-\infty}^{+\infty} dt \frac{\dot{g}(t)}{g(t)} = \ln(g_f)$$

$$\int_{-\infty}^{+\infty} \frac{dt}{t} = 0$$

e, con la condizione iniziale $g(t \rightarrow -\infty = 1)$, troviamo

$$\ln(g_f) = -\frac{i\hbar}{2a} \int_{-\infty}^{+\infty} dt \frac{\ddot{g}(t)}{tg(t)} \quad (2.4)$$

Per risolvere questo integrale possiamo applicare il teorema dei residui per risolvere la divergenza in $t = 0$ svolgendo nel piano complesso un integrale sulla curva nella [Figura 2.3]. La curva l dell'integrale è così definita:

- da $-R$ a $-\epsilon$ sull'asse reale
- semicerchio superiore di raggio $|\epsilon|$ in senso orario
- da ϵ a R sull'asse reale
- semicerchio superiore di raggio $|R|$ in senso antiorario.

La funzione $g(t)$ è in generale una funzione complessa che non tende a 0 nel piano complesso in funzione di t . Ciò ci permette di assumere che $\ddot{g}(t)/g(t)$ sia una funzione analitica nel piano complesso e di conseguenza sono valide le ipotesi necessarie per applicare il Teorema dei Residui. Inoltre la funzione

$\ddot{g}(t)/g(t)$ non ha una dipendenza esponenziale da t . Per t complesso, quindi, non avrà crescite esponenziali quando $|t| \rightarrow +\infty$. Applicando poi il limite $\epsilon \rightarrow 0^+$ e $R \rightarrow +\infty$ l'integrale sulla curva sarà:

$$\oint_l dt \frac{\ddot{g}(t)}{tg(t)} = \int_{-\infty}^{+\infty} dt \frac{\ddot{g}(t)}{tg(t)} + \int_{\epsilon \rightarrow 0^+} dt \frac{\ddot{g}(t)}{tg(t)} + \int_{R \rightarrow +\infty} dt \frac{\ddot{g}(t)}{tg(t)}$$

La curva non contiene singolarità di $\ddot{g}(t)/tg(t)$, quindi il primo membro è nullo. L'ultimo integrale per $R \rightarrow +\infty$ va a zero anch'esso poiché la funzione integranda

$$\frac{\ddot{g}(t)}{tg(t)} \approx \frac{1}{t} \frac{\ddot{g}(t)}{g(t)} \approx \frac{1}{t^3}$$

dai calcoli fatti precedentemente. Infine l'integrale per $\epsilon \rightarrow 0^+$ si può calcolare con il teorema dei residui per $t = 0$:

$$\int_{\epsilon \rightarrow 0^+} dt \frac{\ddot{g}(t)}{tg(t)} = -i\pi \text{Res} \left(\frac{\ddot{g}(t)}{tg(t)} \Big|_{t=0} \right)$$

utilizzando l'equazione (2.3) per valutare $\frac{\ddot{g}(t)}{g(t)}$ in $t = 0$ ottengo:

$$\int_{\epsilon \rightarrow 0^+} dt \frac{\ddot{g}(t)}{tg(t)} = -i\pi \text{Res} \left(-\frac{|b|^2}{\hbar^2 t}, t = 0 \right) = \lim_{t \rightarrow 0} -\frac{i\pi |b|^2}{\hbar^2 t} t = \frac{i\pi |b|^2}{\hbar^2}$$

Quindi l'equazione (2.4) sarà:

$$\ln(g_f) = -\frac{i\hbar}{2a} \int_{-\infty}^{+\infty} dt \frac{\ddot{g}(t)}{tg(t)} = -\frac{\pi |b|^2}{2a\hbar}$$

e g_f sarà

$$g_f = e^{-\frac{\pi |b|^2}{2\hbar a}}$$

Possiamo quindi ricavare $P_{|\psi_1\rangle}(t \rightarrow +\infty)$:

$$P_{|\psi_1\rangle}(t \rightarrow +\infty) = |g_f|^2 = e^{-\frac{\pi |b|^2}{\hbar a}} \quad (2.5)$$

e di conseguenza la probabilità di rimanere nello stato fondamentale:

$$P_{|\Phi^1\rangle}(t \rightarrow \infty) = 1 - P_{|\psi_1\rangle} = 1 - e^{-\frac{\pi |b|^2}{\hbar a}} \quad (2.6)$$

Vediamo quindi che più il fattore $|b|$ è grande, più è piccola la probabilità di restare nello stato $|\psi_1\rangle$ e quindi la probabilità di restare nell'autostato $|\Phi^1\rangle$ è sempre più grande al crescere del gap energetico fra i due livelli che, ricordiamo, è legato a $|b|$ da $|b| = \Delta/2$. Ricordiamo poi che il fattore a rappresenta invece la "velocità" di evoluzione del sistema, di conseguenza più il sistema evolve velocemente meno riesce ad adattarsi ai cambiamenti repentini e la probabilità $P_{|\psi_1\rangle}$ cresce al crescere di a .

Capitolo 3

Problema di Number Partitioning

In questa tesi ci occuperemo di un problema di partizione numerica o "Number Partitioning". Questo è un caso particolare del "Subset sum problem" un problema che si incontra, ad esempio, in crittografia [1].

3.1 Classificazione di complessità del problema

Il Number Partitioning è il "più semplice" dei problemi NP-hard poiché esistono algoritmi classici pseudo-polinomiali che permettono di ottenere delle soluzioni sia ottimali che approssimate per molte istanze del problema, appartiene quindi anche alla classe dei problemi NP-completi.

Dato un insieme N di numeri interi positivi bisogna determinare se è possibile dividerlo in due sottoinsieme N_1 e N_2 tali che la somma degli elementi del primo sottoinsieme sia uguale alla somma degli elementi del secondo.

Si può vedere che per un numero limitato di interi N la verifica di una tale possibilità è molto semplice, ad esempio:

$$N = \{2, 3, 5\} \Rightarrow N_1 = \{2, 3\}, \quad N_2 = \{5\} \quad (3.1)$$

In questo caso è stato possibile determinare una partizione che risolvesse il problema.

Quando però il numero di interi dell'insieme N cresce, o cresce il valore degli interi, il numero di operazioni che servono per effettuare tutte le partizioni possibili e verificare l'uguaglianza delle somme cresce anch'esso notevolmente. Il tempo impiegato da uno degli algoritmi classici usati per risolvere il problema è una funzione $f(n) \propto O(Kn)$ dove n è il numero di interi dell'insieme N e K è la somma di tutti gli interi.

3.2 Hamiltoniana del problema

Per codificare questo problema in una Hamiltoniana quantistica possiamo utilizzare un modello di Heisenberg 1D, cioè un'implementazione quantistica del modello di Ising 1D [3]. Procediamo come segue: data l'Hamiltoniana tipica di un modello di Ising classico per n spin $s_i = \pm 1$

$$H(s_1, \dots, s_n) = - \sum_{\substack{i,j \\ i \neq j}} J_{ij} s_i s_j - \sum_{i=1}^n h_i s_i$$

- prendiamo un sistema di n particelle dotate di spin $\chi = \pm \frac{1}{2}$ lungo l'asse z e utilizziamo la rappresentazione:

$$\left| +\frac{1}{2} \right\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}; \quad \left| -\frac{1}{2} \right\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix};$$

per i vettori degli stati spin-up e spin-down, per gli aggiunti:

$$\left\langle +\frac{1}{2} \right| = (1, 0); \quad \left\langle -\frac{1}{2} \right| = (0, 1);$$

- una configurazione di spin delle n particelle quindi sarà

$$|\psi\rangle = \left| \pm \frac{1}{2} \right\rangle_1 \otimes \dots \otimes \left| \pm \frac{1}{2} \right\rangle_n$$

e il suo aggiunto

$$\langle \psi| = \left\langle \pm \frac{1}{2} \right|_1 \otimes \dots \otimes \left\langle \pm \frac{1}{2} \right|_n$$

- σ_i^z è la matrice di Pauli $\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ scritta nella rappresentazione precedente agente sullo spin i -esimo. σ_i^z è associata allo spin classico s_i dell'Hamiltoniana di Ising attraverso il principio di corrispondenza.
- σ_i^x è la matrice di Pauli $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ scritta nella rappresentazione precedente agente sullo spin i -esimo. Queste matrici ci serviranno proprio per il processo di annealing quantistico entrando nella parte di "fluttuazioni quantistiche" dell'Hamiltoniana del problema.
- ad ogni particella associamo uno ed un solo intero del nostro insieme $N = \{n_1, \dots, n_n\}$, n_i associato alla particella i -esima.
- lo stato del nostro sistema sarà composto da n spin associati agli n interi.

avremo la rappresentazione operatoriale seguente:

$$H^*(\sigma_1^z, \dots, \sigma_n^z, \sigma_1^x, \dots, \sigma_n^x) = -\left(\sum_i^n n_i \sigma_i^z\right)^2$$

dove $H^* = \frac{H}{\Gamma_x}$ con Γ_x è una costante definita come "ampiezza di hopping" con dimensioni di un'energia $[\Gamma_x] = [E]$ e H viene espressa in unità di Γ_x . Questa Hamiltoniana codifica nel suo stato fondamentale la soluzione al problema di Number Partitioning.

Se prendiamo la sommatoria in H^* vediamo infatti che, agendo su una configurazione di spin fra le 2^n possibili e proiettando il risultato sulla stessa, si ottiene una somma di tutti gli interi della n-pla ciascuno con segno positivo o negativo a seconda dello spin della particella associata all'intero (rispettivamente $|+\frac{1}{2}\rangle$ e $|-\frac{1}{2}\rangle$):

$$\langle \chi_1 \dots \chi_n | \sum_i^n n_i \sigma_i^z | \chi_1 \dots \chi_n \rangle = \sum_i^n n_i \langle \chi_i | \sigma_i^z | \chi_i \rangle$$

sapendo che

$$\left\langle \frac{1}{2} \left| \sigma^z \right| \frac{1}{2} \right\rangle = +1 \quad \left\langle -\frac{1}{2} \left| \sigma^z \right| -\frac{1}{2} \right\rangle = -1$$

otteniamo proprio la sommatoria di cui sopra.

Stiamo infatti dividendo gli interi di N in due gruppi, l'uno associato a particelle con spin up e l'altro a particelle con spin down. La sommatoria darà 0 come risultato quando la somma degli interi del primo gruppo sarà uguale alla somma degli interi del secondo gruppo. Elevando al quadrato la sommatoria otteniamo che il minimo dell'energia sulle configurazioni corrisponda proprio alla n-pla di spin che azzerava la sommatoria.

Introducendo la notazione $|1\rangle = |+\frac{1}{2}\rangle$, $|0\rangle = |-\frac{1}{2}\rangle$ le configurazioni del mio sistema di spin corrispondono a tutte le n-ple possibili di 0 e 1.

La matrice dell'Hamiltoniana H^* avrà solo elementi diagonali poiché

$$\left\langle \frac{1}{2} \left| \sigma^z \right| -\frac{1}{2} \right\rangle = 0 \quad \left\langle -\frac{1}{2} \left| \sigma^z \right| \frac{1}{2} \right\rangle = 0$$

e sulla diagonale si troveranno quindi tutti i quadrati delle possibili sommatorie degli interi dell'insieme N . Lo stato fondamentale avrà l'elemento di diagonale corrispondente pari a 0.

Introduciamo la dipendenza temporale, con i fattori tipici di un'Hamiltoniana di QA, e aggiungiamo un termine che introduce "fluttuazioni quantistiche":

$$H(t) = \left(\sum_i^n n_i \sigma_i^z\right)^2 \left(\frac{t}{t_f}\right) - \sum_{i=1}^n \frac{1}{2} \sigma_i^x \left(1 - \frac{t}{t_f}\right)$$

Si possono quindi indicare i due termini rispettivamente con H_f e H_i .

La matrice H_i ha il ruolo di al far "oscillare" gli spin delle particelle da spin-up a spin-down e viceversa, facendo spostare il sistema nelle configurazioni possibili. Infatti lo spin classico del modello di Ising non è equivalente ad uno spin quantistico, poiché può assumere solo due valori. Stiamo infatti passando da variabili in uno spazio Z_2 a variabili in uno spazio $SU(2)$. Introducendo l'Hamiltoniana H_i stiamo proprio sfruttando invece la natura quantistica degli spin trattati per arrivare al minimo dell'Hamiltoniana H_f .

La matrice relativa all'Hamiltoniana H_i , grazie a

$$\sigma^x \left| +\frac{1}{2} \right\rangle = \left| -\frac{1}{2} \right\rangle \quad \sigma^x \left| -\frac{1}{2} \right\rangle = \left| +\frac{1}{2} \right\rangle$$

avrà sempre gli elementi di diagonale pari a 0, dato che $\langle +\frac{1}{2} | -\frac{1}{2} \rangle = 0$. Vediamo che, come detto in precedenza, quando $t \rightarrow t_f$ il termine H_i si "spegne" e si "accende" il termine che codifica il nostro problema numerico. Partiamo quindi dal porre il sistema nello stato fondamentale dell'Hamiltoniana H_i per trovarci, alla fine dell'evoluzione adiabatica, nello stato *corrispondente* ad esso, cioè lo stato fondamentale di H_f che codifica la partizione cercata. Questo processo porterà ad una soluzione soddisfacente se l'evoluzione del sistema risulta effettivamente un'evoluzione adiabatica. Per dare una stima del tempo di evoluzione necessario ad un processo di annealing si può rivedere il problema come un sistema di Landau-Zener. Ricordando che l'Hamiltoniana di Landau-Zener (H_{LZ}) è:

$$H_{LZ} = at\sigma_z + b\sigma_x$$

con H_i nel ruolo dell'Hamiltoniana a $t \rightarrow -\infty$ e H_f per $t \rightarrow +\infty$. Nel nostro caso il rate di variazione dell'energia (la "velocità" a) è $a \approx \frac{\Delta E_{min}}{t_f}$. Data $\Delta \approx 2|b|$, minimo gap di energia esistente fra autostati differenti dell'Hamiltoniana calcolati istantaneamente in tutto l'intervallo di tempo di evoluzione, data (2.6) si dovrà avere

$$\frac{|b|^2 \pi}{2\hbar a} = \frac{\Delta^2 \pi t_f}{\hbar \Delta} \gg 1 \implies t_f \gg \frac{\hbar}{\Delta}$$

3.3 Simulazione di Quantum Annealing in ambiente Mathematica

Fino ad ora è stata fatta una descrizione del procedimento dal punto di vista teorico; la sua implementazione, e quindi la nostra *simulazione* di un processo di annealing quantistico, avviene su un computer che lavora classicamente e quindi il tempo effettivo di risoluzione del problema risulta diverso dal tempo adiabatico t_f che è, invece, un parametro dell'algoritmo in sé. Attraverso un algoritmo che simuli l'evoluzione adiabatica di un sistema quantistico, si

riesce a stimare, variando il tempo di annealing t_f , quanto il metodo sia buono, in base ad un parametro chiamato *energia residua*:

$$\epsilon_{res} = \epsilon_{sim} - \epsilon_0.$$

Questa non è altro che la differenza fra l'energia del sistema ottenuta tramite l'evoluzione adiabatica simulata e l'energia dello stato fondamentale di H_f che, per un numero limitato di spin, siamo in grado di valutare esattamente mediante diagonalizzazione. Viene così stimato il grado di "bontà" del metodo di QA e il parametro t_f ci permette di sapere anche quanto impiegherebbe un computer quantistico a risolvere il nostro problema di Number Partitioning. Ovviamente l'energia residua risulterà sempre ≥ 0 ma più è vicina allo 0 più il metodo sarà stato efficace.

3.3.1 Matrice Densità ed equazione di Liouville

Per lavorare sull'evoluzione temporale del sistema utilizziamo la matrice densità $\rho(t)$. Essa è particolarmente comoda per descrivere sistemi multipartiti e/o in interazione con un environment classico. In questo lavoro pur ignorando fenomeni di decoerenza e dissipazione utilizzeremo ugualmente il formalismo della matrice densità. Definiamone brevemente alcune proprietà [9].

L'operatore densità è definito come:

$$\rho = \sum_n W_n |\psi_n\rangle \langle \psi_n|$$

dove $|\psi_n\rangle$ sono gli stati presenti in una generica mistura e i coefficienti W_n sono i loro pesi statistici nella mistura stessa.

Esprimiamo gli stati della mistura in una base da noi scelta $\{|\phi_n\rangle\} = |\phi_1\rangle, |\phi_2\rangle, \dots$:

$$|\psi_n\rangle = \sum_{m'} a_{m'}^{(n)} |\phi_{m'}\rangle \quad \langle \psi_n| = \sum_m a_m^{(n)*} \langle \phi_m|$$

l'operatore ρ risulta essere:

$$\rho = \sum_{nm} W_n a_{m'}^{(n)} a_m^{(n)*} |\phi_{m'}\rangle \langle \phi_m|$$

da cui si ottiene la matrice densità rappresentata nella base $\{|\phi_n\rangle\}$, i cui elementi sono:

$$\rho_{i,j} = \langle \phi_i | \rho | \phi_j \rangle = \sum_n W_n a_{m'}^{(n)} a_m^{(n)*}$$

se gli elementi di base corrispondono agli stati della mistura, allora gli elementi di diagonale della matrice saranno le probabilità di trovare la mistura in quel particolare stato di base (che in genere corrisponde alla base di

autovettori di un operatore)

$$\rho_{mm} = \left| a_m^{(n)} \right|^2.$$

Sapendo, quindi, che il valor medio di un operatore O sulla mistura è dato da

$$\langle O \rangle = \sum_n W_n \langle \psi_n | O | \psi_n \rangle$$

esprimendo gli stati nella base scelta otteniamo

$$\langle O \rangle = \sum_{mm'} \sum_n W_n a_{m'}^{(n)} a_m^{(n)*} \langle \phi_m | O | \phi_{m'} \rangle = \sum_{mm'} \langle \phi_{m'} | \rho | \phi_m \rangle \langle \phi_m | O | \phi_{m'} \rangle = \text{tr}(\rho O).$$

Tramite la matrice densità quindi possiamo ottenere anche il valor medio di qualsiasi operatore.

Definiamo ora l'operatore di evoluzione temporale $U(t)$ partendo dall'equazione di Schrödinger

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = H(t) |\psi(t)\rangle.$$

l'operatore $U(t)$ si definisce tramite l'evoluzione dello stato del sistema $|\psi_n(0)\rangle$

$$|\psi(t)\rangle = U(t) |\psi(0)\rangle \quad , \quad \langle \psi(t) | = \langle \psi(0) | U(t)^\dagger.$$

Nel caso in cui $H(t)$ non dipenda esplicitamente dal tempo avremo:

$$U(t) = e^{-\frac{i}{\hbar} H t} \quad , \quad U(t)^\dagger = e^{+\frac{i}{\hbar} H t}$$

inoltre, se si fissa lo stato iniziale del sistema a $|\psi(0)\rangle$, si deve imporre $U(0) = 1$ e si ha $U(t)U(t)^\dagger = 1$. L'equazione di Schrödinger con l'introduzione dell'operatore evoluzione temporale diventa:

$$i\hbar \frac{\delta U(t)}{\delta t} |\psi(0)\rangle = H(t) U(t) |\psi(0)\rangle,$$

che può essere vista come un'equazione operatoriale

$$i\hbar \frac{\delta U(t)}{\delta t} = H(t) U(t),$$

e l'equazione aggiunta di questa sarà

$$-i\hbar \frac{\delta U(t)^\dagger}{\delta t} = U(t)^\dagger H(t).$$

Applichiamo adesso le proprietà dell'operatore $U(t)$ per ottenere un'equazione di evoluzione temporale della matrice densità: l'equazione di Liouville. Per $t = 0$ definiamo la matrice densità $\rho(0) = \sum_n W_n |\psi_n(0)\rangle \langle \psi_n(0)|$

con $|\psi_n(0)\rangle$ che si evolve temporalmente come dato dall'operatore $U(t)$. Applicando quanto appena detto abbiamo:

$$\rho(t) = \sum_n W_n |\psi_n(t)\rangle \langle \psi_n(t)| = \sum_n W_n U(t) |\psi_n(0)\rangle \langle \psi_n(0)| U(t)^\dagger = U(t) \rho(0) U(t)^\dagger$$

derivando rispetto al tempo e moltiplicando per $i\hbar$ otteniamo l'equazione cercata:

$$i\hbar \frac{\delta \rho(t)}{\delta t} = i\hbar \frac{\delta U(t)}{\delta t} \rho(0) U(t)^\dagger + i\hbar U(t) \rho(0) \frac{\delta U(t)^\dagger}{\delta t}$$

che, per le proprietà dell'operatore evoluzione temporale, sarà uguale a:

$$i\hbar \frac{\delta \rho(t)}{\delta t} = H(t) U(t) \rho(0) U(t)^\dagger - U(t) \rho(0) U(t)^\dagger H(t)$$

da cui otteniamo infine l'equazione che verrà poi utilizzata nell'algoritmo per sviluppare numericamente l'evoluzione temporale adiabatica del sistema (equazione di Liouville):

$$i\hbar \frac{\delta \rho(t)}{\delta t} = [H(t), \rho(t)]. \quad (3.2)$$

3.3.2 Implementazione dell'algoritmo

Facciamo una descrizione dell'algoritmo utilizzato:

- Ad una particolare N-pla di interi facciamo corrispondere una N-pla di spin. Rappresentiamo tutte le possibili configurazioni di spin tramite tutte le possibili N-ple di 0 e 1, dove $|+\frac{1}{2}\rangle = 1$ e $|-\frac{1}{2}\rangle = 0$. Otteniamo quindi la nostra base nello spazio delle configurazioni che sarà di dimensione 2^N .

Ad esempio con tre interi avremo uno spazio di dimensione $2^3 = 8$. Indicando con $|\uparrow\rangle$ lo spin up $|+\frac{1}{2}\rangle$ e con $|\downarrow\rangle$ lo spin down $|-\frac{1}{2}\rangle$ il mio spazio delle configurazioni a tre spin sarà:

$$\begin{aligned} |0\rangle &= (0, 0, 0) = |\downarrow, \downarrow, \downarrow\rangle \\ |1\rangle &= (0, 0, 1) = |\downarrow, \downarrow, \uparrow\rangle \\ |2\rangle &= (0, 1, 0) = |\downarrow, \uparrow, \downarrow\rangle \\ |3\rangle &= (0, 1, 1) = |\downarrow, \uparrow, \uparrow\rangle \\ |4\rangle &= (1, 0, 0) = |\uparrow, \downarrow, \downarrow\rangle \\ |5\rangle &= (1, 0, 1) = |\uparrow, \downarrow, \uparrow\rangle \\ |6\rangle &= (1, 1, 0) = |\uparrow, \uparrow, \downarrow\rangle \\ |7\rangle &= (1, 1, 1) = |\uparrow, \uparrow, \uparrow\rangle \end{aligned} \quad (3.3)$$

Data l'Hamiltoniana del nostro problema $H_f = -\left(\sum_i^N n_i \sigma_i^z\right)^2$ osserviamo che esiste una degenerazione intrinseca del problema, quando una configurazione partiziona la N-pla lo farà anche la configurazione *complementare* ad essa, ad esempio: per l'N-pla

$$N = \{1, 2, 3\}$$

la configurazione di stato fondamentale che partiziona l'insieme (con la notazione binaria prima introdotta) è $|\psi\rangle_0 = |\downarrow, \downarrow, \uparrow\rangle = (0, 0, 1)$, ma lo è allo stesso modo la configurazione $|\psi'\rangle_0 = |\uparrow, \uparrow, \downarrow\rangle = (1, 1, 0)$. Per eliminare questa degenerazione possiamo fissare uno degli spin corrispondente, per esempio, all'ultimo intero dell'N-pla. Così facendo si riduce il problema che prima trattava N spin a un problema che ne tratta $N - 1$, riducendo le dimensioni dello spazio delle fasi di un fattore 2 e riducendo quindi di molto anche le operazioni che l'algoritmo stesso svolgerà. Chiamiamo questo numero di spin "spin efficaci".

Andiamo quindi a considerare non più una base di N-ple di 0 e 1 ma, fissato l'N-simo spin a $|\uparrow\rangle$, una base di (N-1)-ple di 0 e 1.

- Creiamo un indice per gli elementi di questo insieme di n-ple ($n = N - 1$) facendo corrispondere a ciascun elemento il suo corrispondente in decimale. Questo ci permetterà di scorrere più facilmente fra gli elementi della base.

Per ottenere la matrice Hamiltoniana del problema è necessario tradurre l'azione delle matrici di Pauli sugli stati di base in operazioni facilmente esprimibili iterativamente. Mentre le matrici σ_i^z , dando contributi solo diagonali, sono facilmente trattabili, la parte di Hamiltoniana non diagonale è più complessa da implementare. La creazione dell'indice di cui sopra permette di esprimere in forma molto semplice l'azione delle matrici σ_i^x , in seguito riportiamo la parte di codice relativa all'azione di $H_i = \sum_i \sigma_i^x$.

Matrici di Pauli in ambiente Mathematica

```
Do[i,
[...]
Do[k,
  If[
    StatiBase[[i]][[k]] == 0
    temp = IndiceDecimale[[i]] + 2ne-k;
    index = Position[IndiceDecimale, temp][[1, 1]];
    H[[index, i]] = H[[index, i]] - L (1 - s),

    temp = IndiceDecimale[[i]] - 2ne-k;
```

```

index = Position[IndiceDecimale, temp][[1, 1]];
H[[index, i]] = H[[index, i]] - L (1 - s);
]
, {k, 1, ne}]
, {i, 1, DimBase}];

```

dove:

StatiBase è l'array composto dalle n-ple della base del nostro spazio delle configurazioni. Ad esempio, per $n=3$, si tratta delle triple:

$$\text{StatiBase} = \{(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), (1, 0, 0), (1, 0, 1), (1, 1, 0), (1, 1, 1)\}$$

H è la matrice Hamiltoniana

IndiceDecimale è l'indice che ad ogni elemento della base (visto come un numero binario) attribuisce il corrispondente decimale

ne è il numero di spin efficaci ($N-1$ spin, con N numero di spin totali ciascuno associato ad un intero)

s è il parametro che esprime l'evoluzione temporale; è stato normalizzato in modo tale che $s \in [0, 1]$ e successivamente verrà moltiplicato per il fattore t_f

DimBase è il numero di n-ple contenute nella base $\text{DimBase} = 2^{ne} = 2^{N-1}$

Negli **StatiBase** gli indici i e k permettono di posizionarsi sul k -simo spin dell' i -simo elemento della base, ad esempio posizionarsi sul primo spin del primo elemento di base (considerando il primo elemento come $(0, 0, \dots, 0)$) vuol dire prendere il primo spin da destra, che in questo caso sarà:

StatiBase[[1]][[1]] = 0

quindi: nel ciclo più esterno l'indice i scorrerà lungo gli elementi della base mentre nel ciclo più interno l'indice k scorrerà all'interno di ogni elemento di base su tutti gli $N - 1$ spin efficaci (ne).

La condizione verificata dall' "If" distingue gli spin $|\frac{1}{2}\rangle$ dagli spin $|\frac{1}{2}\rangle$ distinguendo così l'azione di σ_k^x che, ricordiamo, effettua uno "switch" degli spin up in spin down e viceversa. Descriviamo l'azione delle matrici di Pauli sulla base precedentemente scritta facendo un breve esempio:

- fissiamo, senza perdere di generalità, un indice $k = 2$ per indicare lo spin k - *simo* del nostro sistema. Ciò vuol dire che la matrice di Pauli σ_k^x agirà sullo spin k - *simo*, nel nostro caso abbiamo:

$$\sigma_k^x = \sigma_k^+ + \sigma_k^- = \sigma_2^+ + \sigma_2^-$$

- prendiamo adesso uno degli elementi della nostra base, ad esempio $|0\rangle = |\downarrow, \downarrow, \downarrow\rangle$ e facendo agire la matrice di Pauli σ_2^x abbiamo:

$$\sigma_2^x |0\rangle = (\sigma_2^+ + \sigma_2^-) |0\rangle = (\sigma_2^+ + \sigma_2^-) |\downarrow, \downarrow, \downarrow\rangle$$

avendo

$$\sigma_2^+ |\downarrow, \downarrow, \downarrow\rangle = |\downarrow, \uparrow, \downarrow\rangle \quad , \quad \sigma_2^- |\downarrow, \downarrow, \downarrow\rangle = 0$$

otteniamo

$$\sigma_2^x |0\rangle = \sigma_2^x |\downarrow, \downarrow, \downarrow\rangle = |\downarrow, \uparrow, \downarrow\rangle = |2\rangle$$

quindi l'azione di σ_2^x ci restituisce lo stato $|2\rangle$, di conseguenza l'elemento

$$\langle 2 | \sigma_2^x | 0 \rangle \neq 0$$

- determinando quindi l'azione di σ_k^x su tutti gli elementi di base e variando l'indice $k = 0, \dots, N - 1$ ottengo la matrice dell'Hamiltoniana H_i .

Vediamo infatti che si utilizza un indice temporaneo *temp* che ha la funzione appunto di immagazzinare l'informazione su quale indice decimale corrisponda all'elemento ottenuto dall'azione di σ_k^x : sfruttando le semplici proprietà del codice binario si nota che fare uno switch di uno spin down (o di uno spin up) k -esimo di un elemento della base corrisponde a sommare(o sottrarre) una potenza 2^{ne-k} al decimale relativo al numero binario che rappresenta quell'elemento di base. Viene così localizzato dall'indice temporaneo *temp* il numero decimale corrispondente all'elemento della base ottenuto facendo agire σ_k^x e la funzione *Position* permette di creare un altro indice *index* che individui l'elemento di base stesso. Così facendo si generano tutti gli elementi di matrice fuori diagonale che l'Hamiltoniana di fluttuazioni quantistiche crea.

Per preparare il sistema nello stato fondamentale all'istante $t = 0$, dopo aver ottenuto la matrice Hamiltoniana sia per l'istante iniziale, si è ordinata la base delle configurazioni in modo tale da avere come primo elemento di base proprio lo stato fondamentale dell'Hamiltoniana iniziale e si è posta la condizione iniziale sulla matrice densità:

$$\rho = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & & \\ \vdots & & \ddots & \\ 0 & & & 0 \end{pmatrix} \rightarrow \rho_{11} = |a_1|^2 = 1$$

dove $|a_1|^2$ è la probabilità del sistema di trovarsi nello stato corrispondente (secondo un'evoluzione adiabatica) allo stato fondamentale di H_1 , o anche "popolazione" dello stato in esame.

Nella seguente parte di codice vediamo appunto l'inizializzazione della matrice densità:

Inizializzazione Matrice Densità

```
{Energy, V} = Eigensystem[H /. s -> s0];
{Energy, V} = {Energy[[#]], V[[#]]} &@ Ordering[Energy];

ρ = ConstantArray[0, Dimensions[H]];
ρ[[1, 1]] = 1;
ρσz = Transpose[V].ρ.V;
```

Energy è la lista degli autovalori di H all'istante iniziale.

V è la lista di autovettori di H all'istante iniziale.

Nella prima parte del codice vengono calcolati e ordinati autovettori e autovalori di H, nella seconda parte invece viene inizializzata la matrice densità ρ e successivamente viene portata nella base di autovettori ordinata di H, ottenendo $\rho_{\sigma z}$.

L'approssimazione numerica della soluzione dell'equazione di Liouville (3.2) è stata ottenuta con il metodo di Runge-Kutta per il quarto ordine, il modulo seguente richiama il calcolo del membro di destra dell'equazione di Liouville (commutatore fra ρ e H^*):

Commutatore

```
rightHandSide[t_, ρσz] := Module[{Hs, Comm},
  Hs = H /. s -> t;
  Comm = Hs.ρσz - ρσz.Hs;
  -i tf Comm
]
```

Pur non spiegando in dettaglio il metodo di Runge-Kutta [15], è riportato in seguito il codice utilizzato:

Metodo di Runge-Kutta del quarto ordine

```
Do[
  step = s0 + (i - 1)δs;
  nextStep = step + δs;
  {Energy, V} = Eigensystem[H /. s -> nextStep];
  {Energy, V} = {Energy[[#]], V[[#]]} &@ Ordering[Energy];
  δ1 = δs rightHandSide[step, ρσz];
  δ2 = δs rightHandSide[step + δs/2, ρσz + δ1/2];
  δ3 = δs rightHandSide[step + δs/2, ρσz + δ2/2];
  δ4 = δs rightHandSide[nextStep, ρσz + δ3];

  ρσz = ρσz + 1/6 (δ1 + 2 δ2 + 2 δ3 + δ4);
  ρ = V.ρσz.Transpose[V];
  pop1[[i + 1]] = {step, Re[ρ[[1, 1]]]};
  dataE[[i + 1]] = {nextStep, Re[Tr[ρσz.H /. s -> nextStep]]};
```

$, \{i, 1, N_{\max} - 1\}]$

alcune specificazioni:

$s0$ è il valore iniziale del parametro temporale, fissato a $s0 = 0$

step e nextStep sono indici del parametro temporale in avanzamento durante l'evoluzione adiabatica

δ_i sono parametri del metodo numerico di Runge-Kutta che permettono un'approssimazione della soluzione dell'equazione differenziale di Liouville.

Ad ogni ciclo del metodo Runge-Kutta viene ricalcolato lo spettro istantaneo dell'Hamiltoniana e la matrice viene ordinata secondo i valori degli autovalori in ordine crescente.

3.3.3 Risultati ottenuti

In questa sezione descriviamo un'applicazione dell'algoritmo descrivendo il risultato dell'annealing quantistico per partizionare in due insiemi di egual somma la seguente N-pla:

$$N = \{1, 3, 4, 5, 7, 8\}$$

Tale N-pla di sei interi è partizionabile nel modo seguente:

$$N_1 = \{1, 8, 5\} \quad , \quad N_2 = \{3, 4, 7\}.$$

Lo stato fondamentale che partiziona questa N-pla è $|1, 0, 0, 1, 0, 1\rangle$ [Figura 3.1] ricordandoci che la degenerazione intrinseca è stata eliminata fissando lo spin dell'ultimo intero a spin-up.

Fissando l'ultimo spin dell' N-pla, gli spin efficaci con cui opererà effettivamente l'algoritmo sono $N - 1$. Risulta infatti che l'Hamiltoniana al tempo finale t_f è una matrice $2^5 \times 2^5$ in cui l'autovalore relativo alla configurazione che partiziona la nostra N-pla di interi è appunto l'autovalore 0, in particolare corrisponde alla configurazione di $N - 1$ spin $|1, 0, 0, 1, 0\rangle = |\uparrow, \downarrow, \downarrow, \uparrow, \downarrow\rangle$.

Valutando l'energia residua per diversi valori del tempo finale di annealing osserviamo che, come previsto, diminuisce progressivamente [Figura 3.2] raggiungendo valori prossimi allo 0 al crescere di t_f . In particolare per $t_f \Gamma_x / \hbar = 320$ abbiamo $\epsilon_{res} / \Gamma_x \approx 0.000794$. Possiamo anche valutare la popolazione dello stato fondamentale alla fine dell'evoluzione: $\rho_{11}(t_f)$ dovrà essere ≈ 1 [Figura 3.3].

Si può vedere inoltre, dal grafico in [Figura 3.4], l'andamento della popolazione dello stato fondamentale nell'avanzamento dell'evoluzione adiabatica (qui $t_f \Gamma_x / \hbar$ è stato fissato a 320).

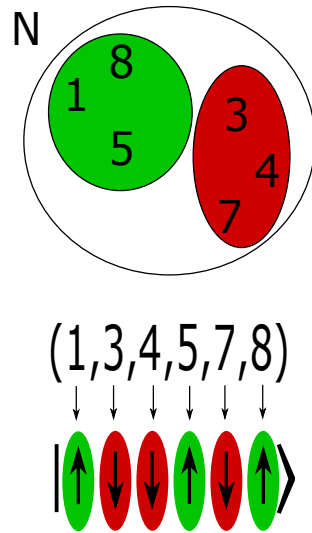


Figura 3.1: Stato fondamentale che partiziona l'N-pla

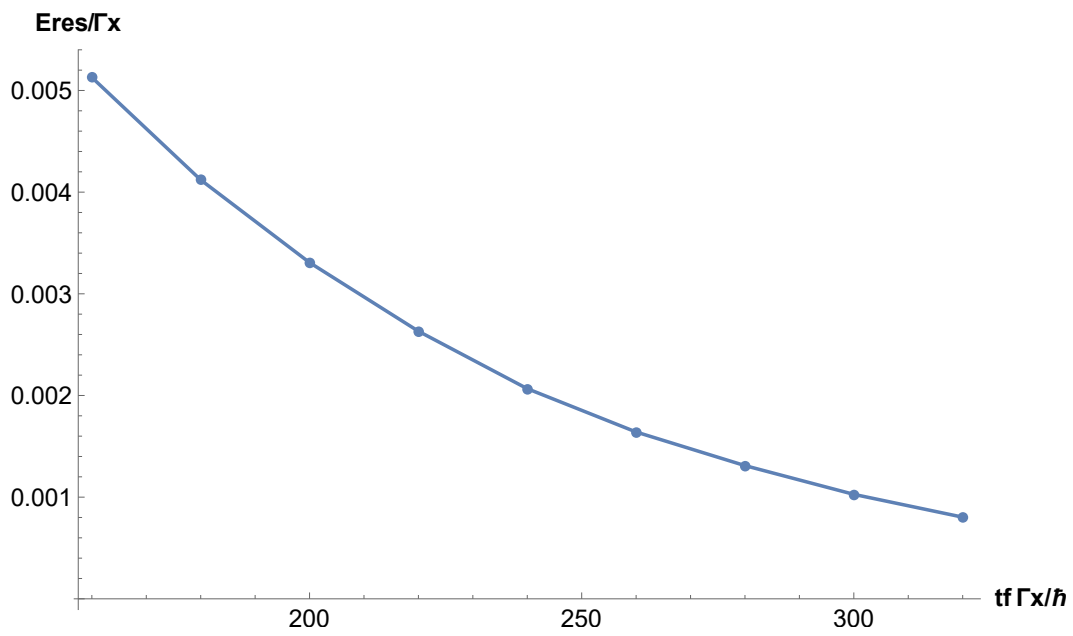


Figura 3.2: Energia residua in funzione di t_f

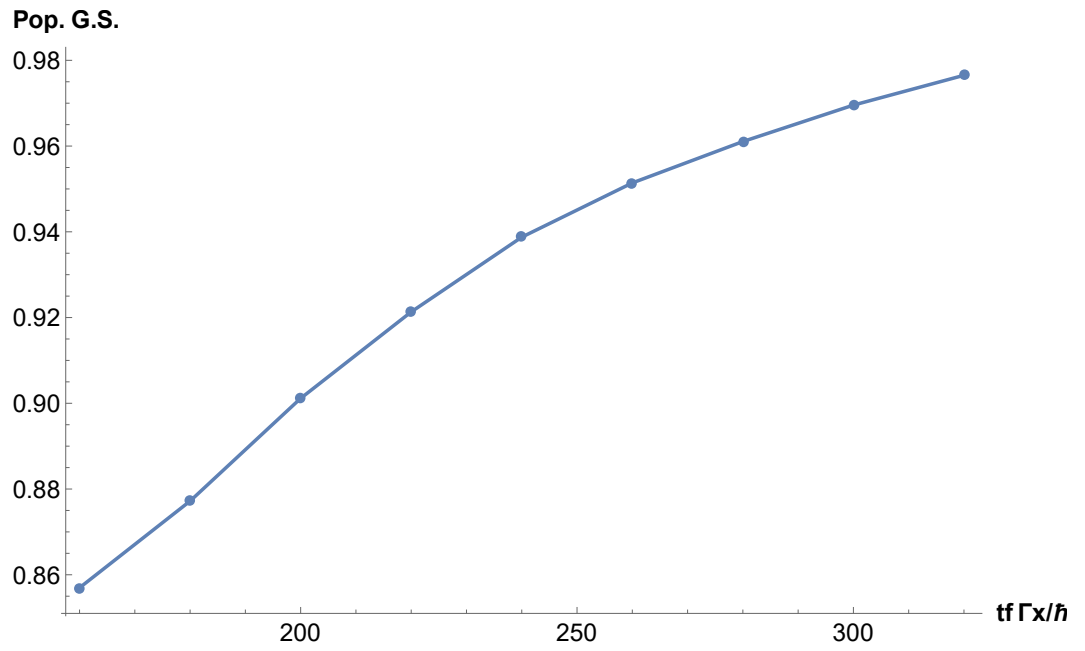


Figura 3.3: Popolazione dello stato fondamentale in funzione di t_f

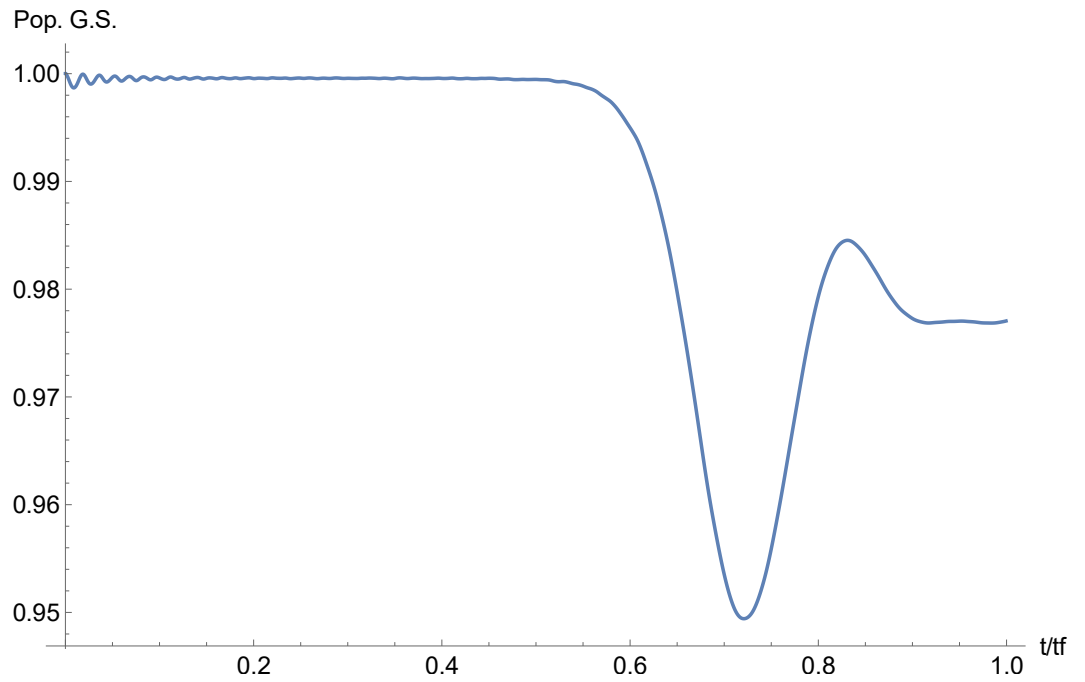


Figura 3.4: Popolazione dello stato fondamentale durante l'evoluzione adiabatica

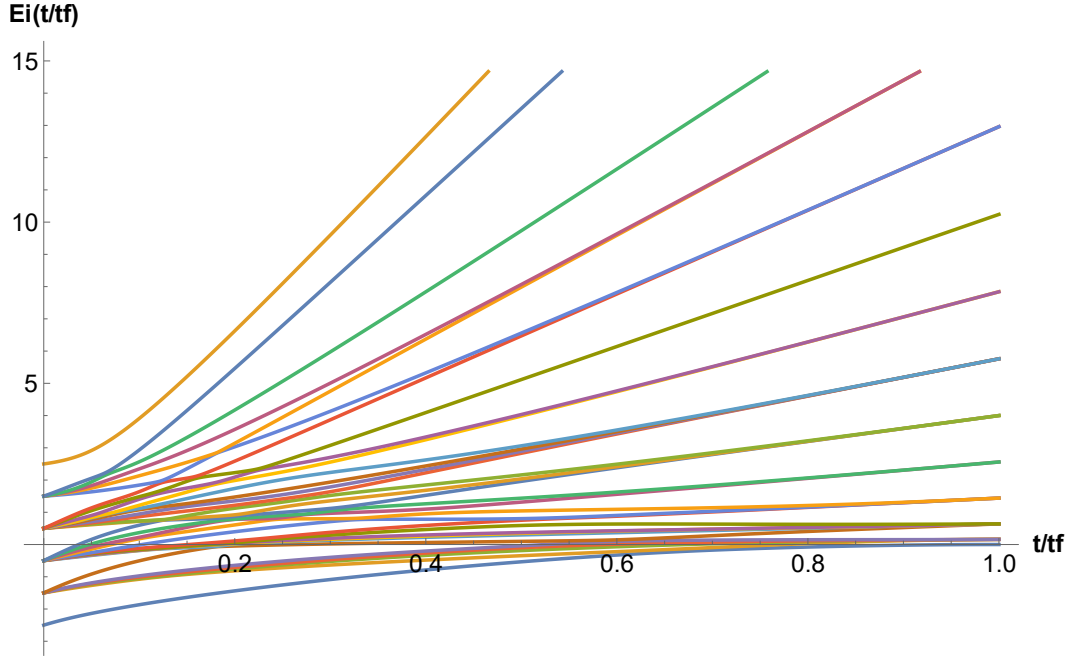


Figura 3.5: Spettro istantaneo dell'Hamiltoniana durante l'evoluzione adiabatica

Si vede come, partendo dal valore 1 da noi fissato all'istante iniziale, la popolazione dello stato fondamentale istantaneo ha un calo improvviso verso la fine dell'evoluzione, assestandosi poi attorno ad un valore $\approx 0,98$. Resta quindi in un range accettabile attorno al valore 1. Il minimo trovato fra $t/t_f \in [0.6, 0.8]$ può essere attribuito ad un momento dell'evoluzione adiabatica dove lo spettro istantaneo dell'Hamiltoniana presenta i primi livelli eccitati particolarmente vicini allo stato fondamentale. Lo spettro istantaneo è particolarmente infittito in quel tratto dell'evoluzione, ciò provoca una maggiore probabilità di salti verso altri livelli per transizioni di Landau-Zener che quindi provocano un abbassamento improvviso della popolazione dello stato fondamentale. Per cercare conferma di questa affermazione possiamo confrontare il grafico dello spettro istantaneo dell'Hamiltoniana con il grafico della popolazione dello stato fondamentale [Figura: 3.6].

Andando a valutare lo spettro istantaneo dell'Hamiltoniana nell'istante iniziale e nell'istante finale dell'evoluzione temporale possiamo notare che mentre per $t = 0$ il gap energetico fra lo stato fondamentale e il primo stato eccitato è (in dimensioni di Γ_x) $\Delta = 1$ per $t = t_f$ abbiamo $\Delta \approx 0.1$ quindi di circa un ordine di grandezza inferiore. Questo infittimento dei livelli energetici, come visto dalle transizioni Landau-Zener, fa aumentare la probabilità di transizione verso un livello energetico superiore allo stato fondamentale

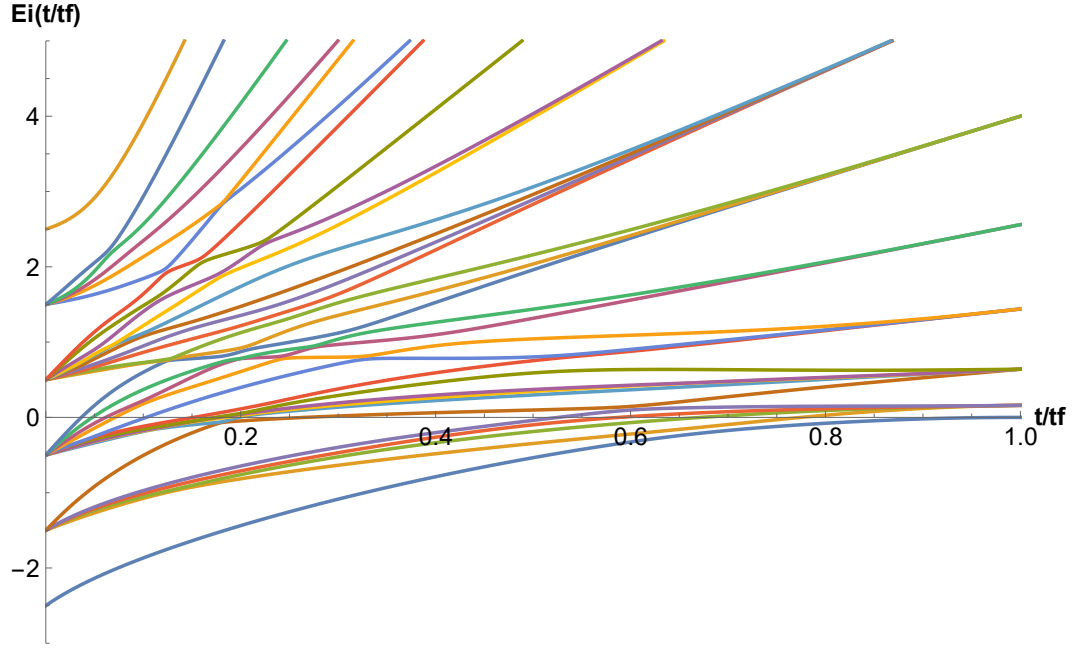


Figura 3.6: Spettro istantaneo dell'Hamiltoniana durante l'evoluzione adiabatica; ingrandimento

	E_0	E_1	Δ
t_0	$-2.5(deg.0)$	$-1.5(deg.5)$	1
t_f	$0(deg.0)$	$0.16(deg.4)$	≈ 0.1

(vedere equazione (2.5)) che, ricordiamo, dipende dal gap energetico fra gli autostati dello spettro dell'Hamiltoniana.

Si può quindi concludere che il QA sia un buon metodo per ottenere rapide elaborazioni di soluzioni per il problema di Number Partitioning.

Possiamo analizzare inoltre un caso di una N-pla che non sia partizionabile esattamente in due insiemi. Di quest'ultima possiamo esaminare sempre la configurazione corrispondente all'autovalore più piccolo dell'Hamiltoniana H_f cioè la configurazione "più vicina" ad una partizione esatta. Prendiamo l'N-pla

$$N = \{2, 3, 7, 13\}$$

Lo stato fondamentale della sua Hamiltoniana non avrà energia $\epsilon_0 = 0$ ma sarà comunque il minimo fra le energie dello spettro del sistema al tempo t_f :

$$N_1 = \{2, 3, 7\} \quad N_2 = \{13\}$$

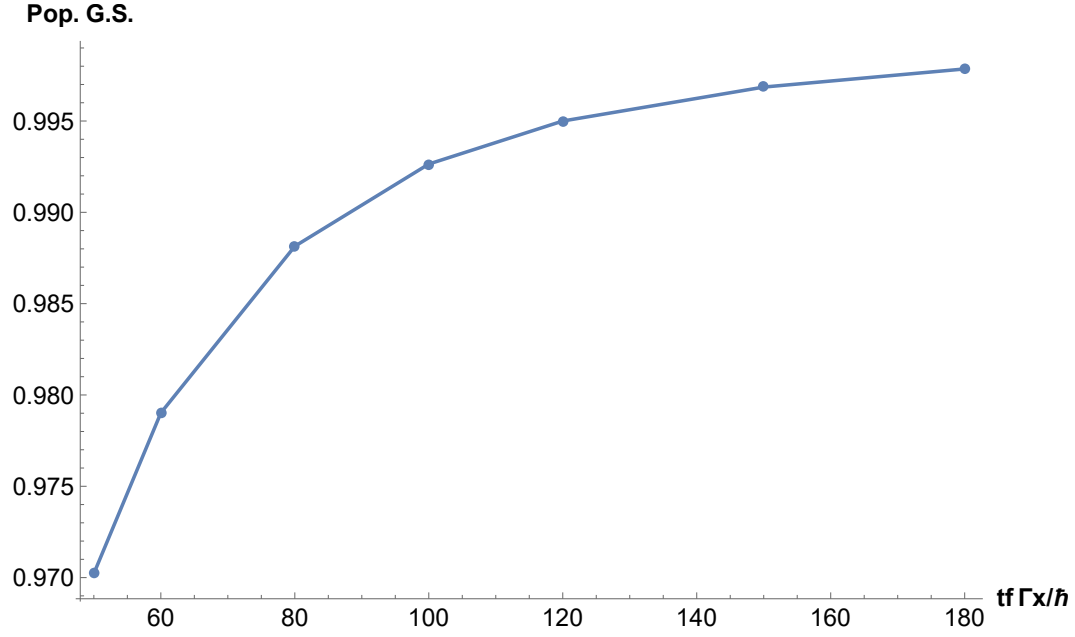


Figura 3.7: Popolazione dello stato fondamentale per N-pla non partizionabile

corrispondente allo stato $|0, 0, 0, 1\rangle = |\downarrow, \downarrow, \downarrow, \uparrow\rangle$.

Vediamo che, anche in questo caso, gli andamenti della popolazione e dell'energia residua sono soddisfacenti [Figura 3.8] e [Figura 3.7] quindi la procedura di annealing quantistico restituisce con successo lo stato fondamentale dell'Hamiltoniana al tempo finale, ma questo sarà sempre ad energia $\neq 0$ perché l'N-pla scelta non è partizionabile in due insiemi di egual somma.

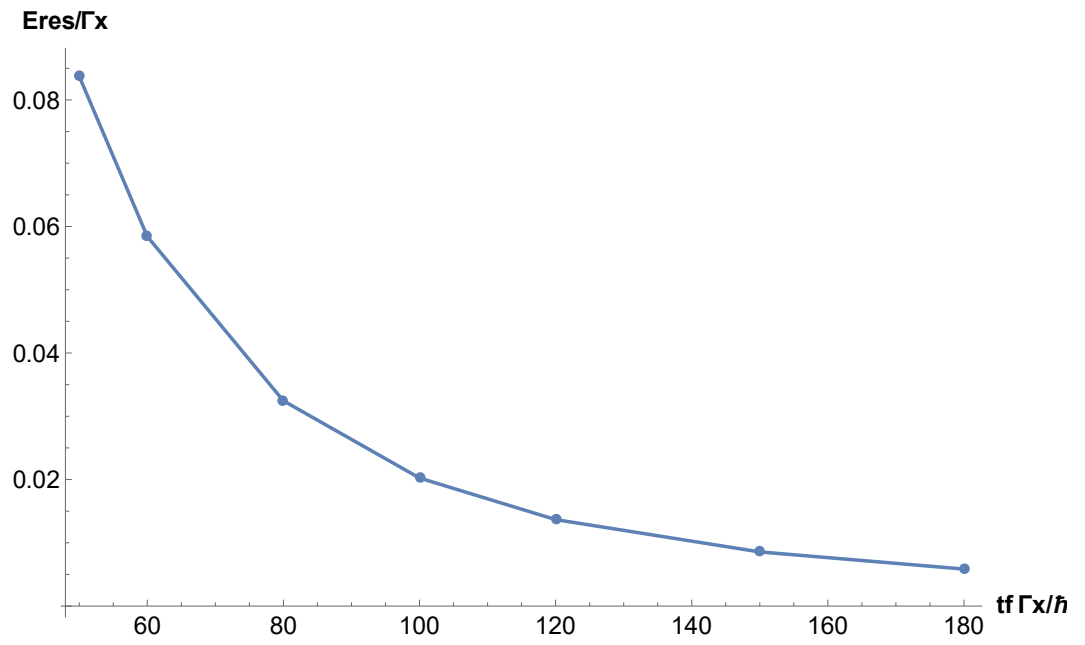


Figura 3.8: Energia Residua in funzione di t_f per N-pla non partizionabile

Capitolo 4

Conclusioni

In teoria dell'informazione ci sono problemi per i quali, ad oggi, non esistono algoritmi che abbiano un tempo di risoluzione polinomiale in funzione del numero di input N : i cosiddetti problemi appartenenti alle classi NP, NP-complete ed NP-hard. Esempi di problemi appartenenti a queste categorie sono la fattorizzazione di grandi interi o il problema del commesso viaggiatore (Travelling Salesman). Questi problemi risultano intrattabili classicamente dal punto di vista informatico per grandi N , si sono quindi cercate nuove strade per implementare algoritmi che funzionino non più con processi classici ma con processi quantistici.

Nel 1994 venne dimostrato da Peter Schor [14] che il problema della fattorizzazione di grandi interi (codificati da N bit), ritenuto un problema classicamente appartenente alla classe NP , poteva essere risolto con un numero di operazioni quantistiche che cresce polinomialmente in funzione di N . Questo fu il primo passo verso la Quantum Computation basata su circuiti quantistici, essenzialmente analoghi a circuiti classici ma nei quali i bit classici sono rimpiazzati dai q-bit e le porte logiche tradizionali da porte logiche quantistiche.

Purtroppo i dispositivi quantistici presentano fenomeni di decoerenza e problemi relativi all'interazione con l'ambiente. Tempi troppo lunghi di elaborazione di tali algoritmi causano quindi perdita di informazione proprio per la natura quantistica del sistema e questo porta al fallimento di tali algoritmi. Nel gli ultimi decenni è stato dimostrato che la Adiabatic Quantum Computation è polinomialmente equivalente al modello circuitale quantistico [16]. Si può quindi pensare ad essa come un'alternativa alla Quantum Computation classica che sia meno inficiata da problemi di decoerenza.

Il Quantum Annealing si basa sui principi del Thermal Annealing classico [5] e permette di raggiungere lo stato fondamentale di un'Hamiltoniana appositamente scelta per codificare la soluzione al nostro problema numerico. Ciò viene fatto sfruttando i risultati del Teorema Adiabatico della Meccanica Quantistica: un sistema che si trovi in un autostato di una certa Hamilto-

niana, se evolve nel tempo abbastanza lentamente alla fine dell'evoluzione si troverà nell'autostato corrispondente a quello iniziale ma dello spettro dell'Hamiltoniana finale, in generale diversa da quella iniziale. L'Hamiltoniana totale del sistema è del tipo

$$H(t) = H_i \left(1 - \frac{t}{t_f}\right) + H_f \left(\frac{t}{t_f}\right)$$

dove H_f è l'Hamiltoniana che, nello stato fondamentale, codifica la soluzione al problema numerico da risolvere, mentre H_i è un'Hamiltoniana di "semplice" nel cui stato fondamentale prepariamo il sistema all'istante iniziale $t = 0$. Il termine inizialmente presente è solo H_i , questo andrà spegnendosi all'avanzare dell'evoluzione mentre H_f sarà l'unico termine a sopravvivere a $t = t_f$. Se t_f è abbastanza "grande" da permettere un'evoluzione adiabatica allora il sistema si troverà nello stato finale voluto: lo stato fondamentale di H_f che codifica appunto la soluzione al problema numerico. La probabilità P_{gs} che il sistema rimanga nello stato fondamentale durante un'evoluzione adiabatica è calcolabile analiticamente; nel semplice modello di Landau-Zener essa è:

$$P_{gs}(t \rightarrow \infty) = 1 - e^{-\frac{\pi|b|^2}{\hbar a}}$$

dove $a \approx \frac{\Delta E_{min}}{t_f}$ è il rate di variazione dell'energia dello stato fondamentale del nostro sistema e il minimo gap energetico nel sistema è $\Delta E_{min} = \Delta \approx 2|b|$. Questa probabilità è calcolata per un tempo finale $t \rightarrow +\infty$ ma per un'approssimazione a tempi finiti si può legittimamente stimare

$$t_f \gg \frac{\hbar}{\Delta}$$

Nel limite di tempi t_f lunghi possiamo ragionevolmente assumere che tale probabilità sia ≈ 1 anche per Hamiltoniane più complesse nel regime adiabatico.

Quindi in questa tesi applichiamo questa idea al problema di Number Partitioning; si tratta di determinare una partizione di un insieme di interi in due sottoinsiemi di egual somma. Ad esempio prendendo l'insieme

$$N = \{1, 2, 3\}$$

è possibile suddividerlo in

$$N_1 = \{1, 2\} \quad , \quad N_2 = \{3\}$$

in questo modo la somma degli elementi presenti nel primo insieme sarà uguale alla somma degli elementi presenti nel secondo insieme. La soluzione a un problema numerico del genere è facilmente codificata dallo stato fondamentale della versione quantistica di un'Hamiltoniana di Ising del tipo:

$$H_f = \left(\sum_i^N n_i \sigma_i^z \right)^2$$

Ad ogni intero dell'N-pla viene associata una particella che avrà quindi uno spin classico $s_i = \pm 1$ al quale si associa secondo il principio di corrispondenza $s_i \rightarrow \sigma_i^z$. Lo spazio delle configurazioni di N spin sarà quindi di dimensione 2^N e fra queste va trovato il minimo dell'energia del sistema, lo stato fondamentale $|\psi\rangle_0$ dell'Hamiltoniana H_f che ha energia $E_0 = 0$. Per l'esempio precedente lo stato fondamentale è

$$|\psi\rangle_0 = |+\frac{1}{2}, +\frac{1}{2}, -\frac{1}{2}\rangle = |\uparrow, \uparrow, \downarrow\rangle$$

ma lo è allo stesso modo

$$|\psi\rangle_0 = |-\frac{1}{2}, -\frac{1}{2}, +\frac{1}{2}\rangle = |\downarrow, \downarrow, \uparrow\rangle.$$

Questa degenerazione può essere eliminata fissando l'ultimo spin del sistema a spin-up e riducendo così lo spazio delle configurazioni da 2^N a 2^{N-1} .

Introducendo l'Hamiltoniana iniziale, nella quale gli spin sono allineati lungo l'asse x, l'Hamiltoniana di annealing da studiare è:

$$H(t) = H_f\left(\frac{t}{t_f}\right) + H_i\left(1 - \frac{t}{t_f}\right) = \left(\sum_i^n n_i \sigma_i^z\right)^2 \left(\frac{t}{t_f}\right) - \sum_{i=1}^n \frac{1}{2} \sigma_i^x \left(1 - \frac{t}{t_f}\right).$$

Per la simulazione del processo adiabatico prepariamo il sistema nello stato fondamentale di H_i e, attraverso il metodo numerico di Runge-Kutta del quarto ordine, dividiamo l'intervallo di tempo $\Delta t = t_f - t_0$ in un numero fisso di intervalli e, istante per istante, calcoliamo lo spettro dell'Hamiltoniana, la popolazione dello stato fondamentale e l'energia residua. Quest'ultimo parametro ci serve per valutare la "bontà" del metodo utilizzato

$$\epsilon_{res} = \epsilon_{sim} - \epsilon_0.$$

Questa non è altro che la differenza fra l'energia del sistema ottenuta tramite l'evoluzione adiabatica simulata e l'energia dello stato fondamentale di H_f che, per un numero limitato di spin, siamo in grado di valutare esattamente mediante diagonalizzazione.

Il codice utilizzato è stato implementato in linguaggio Mathematica. Dalla simulazione su un'N-pla partizionabile di 6 interi $N = \{1, 3, 4, 5, 7, 8\}$ si ottiene che al crescere di t_f l'energia residua diminuisce progressivamente come si vede in figura 3.2. Ad esempio per $t_f \Gamma_x / \hbar = 320$ abbiamo $\epsilon_{res} / \Gamma_x \approx 0.000794$. Analizzando inoltre l'andamento, sempre al crescere di t_f , della popolazione dello stato fondamentale si vede che essa resta ≈ 1 [figura 3.3] e per $t_f \Gamma_x / \hbar = 320$ si ha $\rho_{|\psi\rangle_0} \approx 0.976505 \approx 1$.

Effettuando una simulazione anche per un'N-pla non partizionabile di 4 interi $N = \{2, 3, 7, 13\}$, pur non avendo uno stato fondamentale ad energia $E_0 = 0$, con l'evoluzione adiabatica ci si avvicina allo ad energia più bassa al tempo t_f . Cioè l'annealing ripartisce gli interi nei due sottoinsiemi con

somme più simili possibile fra loro. Gli andamenti dell'energia residua e della popolazione dello stato fondamentale sono analoghi agli andamenti trovati per un N -pla partizionabile: [Figura 3.8] e [Figura 3.7].

Da queste simulazioni di Quantum Annealing abbiamo quindi tratto dei risultati soddifacenti nella risoluzione di un problema NP-hard piuttosto "semplice" e per un numero piccolo di input da elaborare. Potendo conoscere la soluzione del problema per via analitica, in questo caso, abbiamo potuto valutare la bontà del metodo risolutivo. Si può quindi concludere che il Quantum Annealing può essere utilizzato anche in casi più complessi portando comunque risultati soddifacenti. Essendo (come detto in precedenza) un metodo che presenta meno problemi di decoerenza, rispetto ai circuiti quantistici, può essere una valida alternativa a questi ultimi. In questa tesi non sono stati trattati problemi relativi all'interazione con l'ambiente esterno che possono alterare il processo di annealing, ma lo studio teorico del quantum annealing in interazione con un ambiente esterno è uno dei passi successivi per una futura realizzazione concreta di processi di annealing in calcolatori quantistici.

Bibliografia

- [1] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, D. Preda *A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem*, Science, 1 (2001).
- [2] M. A. Nielsen, I. L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, 10th Anniversary edition published, 1 (2010).
- [3] W. Ruml, J. T. Ngo, J. Marks, S. Shieber, *Easily Searched Encodings For Number Partitioning*, Journal of Optimization Theory and Applications, Springer, 1 (1996).
- [4] A. Coniglio, *Elementary Course In Statistical Mechanics*, vol. 1 (2008).
- [5] G. E. Santoro, E. Tosatti, *Optimization using quantum mechanics: quantum annealing through adiabatic evolution*, Journal of PhysicsA: Mathematical and General, 1 (2006).
- [6] John Kleinberg, Éva Tardos, *Algorithm Design*, vol. 1 (2006).
- [7] Andrew Lucas, *Ising formulations of many NP problems*, Frontiers in Physics 2, 5 (2014)
- [8] T. Kato, *On the Adiabatic Theorem of Quantum Mechanics*, Journal of the Physical Society of Japan (1950)
- [9] K. Blum, *Density Matrix Theory and Application, 3rd Edition*, Springer Series on Atomic, Optical and Plasma Physics 64 (2012)
- [10] E. Benjamin, K. Huang, A. Kamil e J. Kittiyachavalit, *Quantum Computability and Complexity and the Limits of Quantum Computation* (2003).
- [11] M. Born e V. Fock, *Beweis des Adiabatenatzes*, Zeitschrift für Physik, Volume 51, Number 3-4, Page 165 (1928).
- [12] J. E. Avron e A. Elgart, *Adiabatic Theorem without a Gap Condition*, Communications in Mathematical Physics (1999).

- [13] L. Landau *Zur Theorie der Energieübertragung. II*, Physikalische Zeitschrift der Sowjetunion, Vol. 2, pp. 46-51. (1932).
- [14] P. Shor, *Algorithms for quantum computation: Discrete log and factoring*, Proceedings of the 35th Annual Symposium on the Foundations of Computer Science, IEEE Computer Society Press pp. 124-134 (1994)
- [15] C. Runge, H. König, *Vorlesungen über numerisches Rechnen*, Springer (1924)
- [16] D. Aharonov et al. "Adiabatic Quantum Computation is equivalent to standard Quantum Computation", SIAM Jour. Comp. 37.1 (2007)
- [17] D. Deutsch , *Quantum theory, the Church-Turing principle and the universal quantum computer*, Proceedings of the Royal Society of London A 400, pp. 97-117 (1985)