



UNIVERSITÀ DEGLI STUDI DI MILANO

FACOLTÀ DI SCIENZE E TECNOLOGIE

Corso di Laurea Triennale in Fisica

# Implementazione dell'Hamiltoniana di Fermi-Hubbard su un Computer Quantistico Circuitale

**Relatore:** Prof. Enrico Prati

Elaborato di:  
**Tommaso Di Luciano**  
Matr. 933841

Anno Accademico 2023-2024







# Indice

<b>Introduzione</b>	<b>1</b>
<b>1 La fisica del modello di Hubbard</b>	<b>3</b>
1.1 Modello di Tight-Binding . . . . .	4
1.2 Costruzione del modello di Hubbard: ipotesi e approssimazioni . . . . .	5
<b>2 Programmazione di un computer quantistico con Qiskit</b>	<b>9</b>
2.1 Circuiti di qubits e porte logiche fondamentali . . . . .	10
2.2 Reticolo cristallino e Hamiltoniana di Hubbard . . . . .	11
<b>3 Implementazione del modello di Hubbard su un computer quantistico</b>	<b>13</b>
3.1 Stati fermionici e base computazionale dei qubits . . . . .	13
3.2 Trasformazione di Jordan-Wigner (JWT): mappare operatori nello spazio dei qubits . . . . .	14
3.2.1 Rappresentazione di osservabili fisiche e località dell'interazione . . .	16
<b>4 Simulazione del modello di Hubbard mediante un circuito quantistico</b>	<b>17</b>
4.1 Algoritmi variazionali . . . . .	17
4.1.1 Teorema variazionale per il calcolo dello stato fondamentale . . . . .	18
4.2 Variational Quantum Eigensolver (VQE) . . . . .	18
4.3 Ottimizzatori . . . . .	19
4.3.1 Ottimizzatori Locali . . . . .	19

## INDICE

---

4.3.2	Ottimizzatori Globali . . . . .	19
4.3.3	Ottimizzatori e metodo di ricerca del minimo . . . . .	20
<b>5</b>	<b>Variational Quantum Eigensolver per reticolo monodimensionale</b>	<b>23</b>
5.1	Preparazione stato iniziale: Reference state . . . . .	24
5.1.1	Implementazione Reference state su circuito di qubit . . . . .	25
5.2	Ansatz: costruzione . . . . .	26
5.2.1	Efficient Hamiltonian Variational Ansatz (EHVA) . . . . .	27
5.3	Efficient Hamiltonian Variational Ansatz per il modello di Hubbard . . . . .	29
5.3.1	Implementazione di EHVA su circuito di qubit . . . . .	31
<b>6</b>	<b>Risultati delle simulazioni del modello di Hubbard su emulatore di un computer quantistico</b>	<b>35</b>
6.1	Necessità di accoppiamento tra settori di spin . . . . .	36
6.2	Confronto dei risultati al variare dell'ottimizzatore classico . . . . .	37
6.2.1	Controlled Random Search (CRS) with local mutation optimizer . . . . .	41
6.2.2	Dividing RECTangles Locally-biased optimizer (DIRECT-L) . . . . .	43
6.2.3	Improved Stochastic Ranking Evolution Strategy optimizer (ISRES) . . . . .	45
6.2.4	ESCH . . . . .	47
6.2.5	Scelta dell'ottimizzatore globale . . . . .	48
6.3	2 siti reticolari . . . . .	49
6.4	3 siti reticolari . . . . .	51
6.5	4 siti reticolari . . . . .	53
6.6	Confronto al variare del numero di siti . . . . .	54
6.7	Implementazione dello stato iniziale attraverso rotazioni di angoli arbitrari . . . . .	54
6.7.1	2 siti reticolari . . . . .	56
6.7.2	3 siti reticolari . . . . .	57
	<b>Conclusioni</b>	<b>59</b>

## INDICE

---

Bibliografia	62
Appendice	63





# Introduzione

I quantum computer si sono rivelati, sin dalla loro formulazione negli anni '80, uno strumento promettente per risolvere problemi di classi computazionali non risolvibili in tempi di ordine polinomiale su computer convenzionali.

Il concetto di computer quantistico nasce dall'idea di sostituire ai bits, caratteristici dei computer classici, dei sistemi quantistici a due stati detti qubits. Questi, potendo trovarsi in una sovrapposizione e in entanglement tra loro, consentono di definire nuove macchine di Turing quantistiche che implicano nuove classi di complessità computazionale che in determinati casi consentono uno speed-up rispetto alle macchine classiche.

Al momento le difficoltà nel produrre qubits fisici che siano stabili, legate allo sviluppo tecnologico delle loro possibili implementazioni, ha portato alla produzione dei cosiddetti Noisy Intermediate-Scale Quantum computer (NISQ computer), sprovvisti di metodi di correzione quantistica degli errori.

In particolare i quantum computer aprono nuove importanti frontiere per quanto riguarda la simulazione di sistemi in cui sono presenti effetti quantistici.

Infatti vi è una forte analogia tra il qubit fisico e lo stato di occupazione di un fermione. Questa analogia può essere sfruttata su un computer NISQ per costruire stati che rappresentino in maniera efficace il comportamento di sistemi, come il modello di Hubbard, a patto di svolgere la computazione compatibilmente con i tempi di decoerenza dei qubits con cui è implementato.

Questo lavoro di Tesi si è concentrato sull'implementazione dell'algoritmo Variational Quantum Eigensolver (VQE) per studiare lo stato fondamentale di un reticolo cristallino monodimensionale descritto attraverso il modello di Hubbard, implementabile su computer circuitale NISQ.

---

Il modello di Hubbard rappresenta il comportamento elettronico caratteristico dei solidi generalizzando il modello di Tight-Binding attraverso un termine a molti-corpi di repulsione tra gli elettroni presenti su uno stesso sito atomico reticolare.

Questo comportamento è descritto da un operatore Hamiltoniano costituito da due termini, ascrivibili rispettivamente all'Hamiltoniana di Tight-Binding e alla repulsione coulombiana nel singolo sito atomico.

L'algoritmo VQE si basa sulla costruzione di un set di stati parametrizzati, detto *ansatz*, attraverso un computer quantistico circuitale. Su questi stati è valutato il valore di aspettazione dell'Hamiltoniana, utilizzato come funzione di costo da minimizzare per ottenere lo stato fondamentale. I parametri che definiscono gli stati sono fatti variare attraverso un algoritmo di ricerca del minimo della funzione di costo, implementato su un computer classico.

La stima dello stato fondamentale consiste nel valore di aspettazione sullo stato caratterizzato dai parametri che identificano il minimo ottenuto.

Il metodo di costruzione dell'*ansatz* implementato è detto Efficient Hamiltonian Variational Ansatz (EHVA) e si basa sulla costruzione degli stati attraverso una serie di step che realizzano una variazione, ispirata a quella dell'evoluzione temporale generata dall'Hamiltoniana, di stati del sistema definiti a partire dal numero di elettroni di valenza considerati.

In questo lavoro di Tesi si è partiti da esempi di implementazione forniti dalla piattaforma Qiskit di IBM Quantum Research per la risoluzione tramite algoritmi classici del modello di Hubbard e si è costruita una nuova implementazione basata sull'algoritmo VQE proposto in letteratura. Per l'implementazione si sono utilizzati fino a 12 qubits simulati attraverso il framework Qiskit. I risultati sono stati elaborati utilizzando diversi tipi di ottimizzatori classici per la ricerca del minimo e analizzando la precisione ottenuta al variare della dimensione dei sistemi considerati e degli step componenti l'implementazione dell'*ansatz*.

Infine si è implementata la possibilità di effettuare rotazioni arbitrarie tra i qubits nella costruzione dello stato iniziale e si è registrata nel caso di un reticolo a 3 siti atomici una precisione del 2.5% rispetto al valore esatto, calcolato dalla diagonalizzazione dell'operatore Hamiltoniano, per l'ottimizzatore Univariate Marginal Distribution Algorithm (UMDA).

# Capitolo 1

## La fisica del modello di Hubbard

In questo Capitolo verrà trattato il modello di Hubbard, strumento utile per studiare le caratteristiche di sistemi di solidi cristallini nel caso in cui l'interazione tra elettroni sul sito atomico non sia trascurabile. Si ricaverà l'Hamiltoniana descrivente il comportamento elettronico che deriva da questo modello facendo alcune considerazioni sulla sua forma particolare.

Nella fisica della materia condensata il modello di Hubbard costituisce il più semplice modello con cui è possibile rappresentare un sistema di fermioni correlati in un solido reticolare. Questo fornisce un'analisi qualitativa di sistemi in cui è considerato il contributo di una o poche bande prossime all'energia di Fermi. Fondamentalmente consiste in una generalizzazione del modello di Tight-Binding, compiuta aggiungendo un termine di repulsione tra elettroni presenti nello stesso sito atomico.

Questa generalizzazione permette di studiare il caso di elettroni interagenti e, nel regime di interazione forte, è possibile utilizzarlo per descrivere qualitativamente i materiali detti isolanti di Mott.

## 1.1 Modello di Tight-Binding

---

### 1.1 Modello di Tight-Binding

Il modello di Tight-Binding è un modello utilizzato per studiare il comportamento dei solidi cristallini nel caso di interazione elettrone-elettrone trascurabile o trattabile perturbativamente. La principale ipotesi alla base del modello consiste nel rappresentare la funzione d'onda che descrive il comportamento elettronico nel solido in termini di orbitali atomici, relativi agli atomi che lo compongono.

Infatti in una regione di spazio in prossimità del nucleo atomico, il potenziale che il reticolo esercita sugli elettroni è molto simile a quello del singolo atomo.

Per questo motivo, inoltre, gli elettroni appartenenti agli stati atomici di core possono essere trascurati negli spostamenti tra siti descritti dal modello perchè appartenenti a stati più fortemente legati, e quindi non in possesso di energie sufficienti per effettuare il salto verso orbitali atomici contigui. Questi partecipano per lo più svolgendo un effetto schermante all'interno del potenziale percepito dagli elettroni appartenenti agli stati più esterni.

Il modello di Tight-Binding descrive quindi gli elettroni meno legati, detti di valenza, come fermioni non interagenti soggetti al potenziale periodico reticolare causato da ioni ed elettroni maggiormente legati. Questi elettroni di valenza possono effettuare salti tra i siti atomici che costituiscono il solido.

La forma dell'operatore Hamiltoniano nel formalismo della seconda quantizzazione, derivata dal modello di Tight-Binding, appare come:

$$H_{el} = - \sum_{i,j} \sum_{\sigma=\uparrow\downarrow} t_{ij} \hat{c}_{i,\sigma}^\dagger \hat{c}_{j,\sigma}$$

Scritta in termini di operatori di creazione e distruzione di elettroni sul singolo sito atomico. A questa si può applicare l'approssimazione di salto tra primi vicini, cioè per ogni elettrone è possibile effettuare uno spostamento solo verso siti atomici adiacenti a quello in cui è localizzato, definita da:

$$\begin{aligned} t_{ij} &= t \quad \text{se } j = \pm 1 \\ t_{ii} &= \varepsilon_i \quad \text{se } j = i \\ t_{ij} &= 0 \quad \text{altrimenti} \end{aligned}$$

con  $\varepsilon_i$  contributo energetico dell'elettrone sul sito atomico.

La forma di questa Hamiltoniana verrà commentata nel paragrafo successivo, riguardante il modello di Hubbard.

## 1.2 Costruzione del modello di Hubbard: ipotesi e approssimazioni

Nel trattare un solido cristallino è interessante studiare il comportamento elettronico del sistema.

In generale l'Hamiltoniana descrivente il sistema nuclei-elettroni è:

$$\sum_{n=1}^{N_n} \frac{\vec{p}_n^2}{2M_n} + \frac{e^2}{2} \sum_{n \neq m=1}^{N_n} \frac{Z_n Z_m}{|\vec{R}_n - \vec{R}_m|} + \sum_{i=1}^{N_e} \frac{\vec{p}_i^2}{2m} + \frac{e^2}{2} \sum_{i \neq j=1}^{N_e} \frac{1}{|\vec{r}_i - \vec{r}_j|} - e^2 \sum_{n=1}^{N_n} \sum_{i=1}^{N_e} \frac{Z_n}{|\vec{R}_n - \vec{r}_i|}$$

In approssimazione di Born-Oppenheimer, ponendo  $N = N_e$ ,  $V_I$  potenziale ionico e  $V_C(\vec{x}) = \frac{e^2}{|\vec{x}|}$  interazione di Coulomb tra elettroni, è possibile ridursi all'Hamiltoniana puramente elettronica:

$$H = \sum_{i=1}^N \left( \frac{\vec{p}_i^2}{2m} + V_I(\vec{x}_i) \right) + \sum_{1 \leq i < j \leq N} V_C(\vec{x}_i - \vec{x}_j)$$

Assumendo che un gran numero di elettroni si trovi nello stato fondamentale  $\psi_0$  di  $H$ , se venisse inserito un altro elettrone localmente nel sistema, questo percepirebbe l'effetto del potenziale  $V_I$  e del potenziale elettrostatico causato dalla densità elettronica  $|\psi_0(\vec{x}_1, \dots, \vec{x}_N)|^2$ , anch'esso periodico come  $V_I$ . Quest'ultimo ha l'effetto di schermare il potenziale ionico e per rappresentare questo fenomeno viene inserito il termine potenziale ausiliario  $V_A(\vec{x})$ , ottenendo:

$$H = \sum_{i=1}^N \left( \frac{\vec{p}_i^2}{2m} + V(\vec{x}_i) \right) + \sum_{1 \leq i < j \leq N} U(\vec{x}_i, \vec{x}_j)$$

Dove si è posto  $V(\vec{x}) = V_I(\vec{x}) + V_A(\vec{x})$  e  $U(\vec{x}, \vec{y}) = V_C(\vec{x} - \vec{y}) - \frac{V_A(\vec{x}) + V_A(\vec{y})}{N-1}$

Il **modello di Tight-Binding** si ottiene mettendosi in approssimazione di mean-field, cioè richiedendo che il termine  $U(\vec{x}, \vec{y})$  sia molto piccolo. Questo significa imporre che esista un potenziale ausiliario  $V_A$  tale che, nella base degli autostati dell'Hamiltoniana di singola particella  $h_1 = \frac{\vec{p}^2}{2m} + V(\vec{x})$ , il valore degli elementi di matrice  $U(\vec{x}, \vec{y})$  risulti trascurabile. Nel caso in cui l'interazione non sia trascurabile, o trattabile perturbativamente, si può ricavare il **modello di Hubbard**.

Per ottenere l'operatore  $H$  nel formalismo della seconda quantizzazione occorre porsi in una base opportuna, come gli autostati dell'Hamiltoniana di singola particella.

## 1.2 Costruzione del modello di Hubbard: ipotesi e approssimazioni

---

Essendo il potenziale  $V(\vec{x})$  periodico, gli autostati dell'Hamiltoniana reticolare saranno funzioni che soddisfano il teorema di Bloch:

$$\varphi_{\alpha\vec{k}}(\vec{x}) = e^{i\vec{k}\vec{x}} u_{\alpha\vec{k}}(\vec{x}) \quad \vec{k} \in 1BZ$$

Dove  $u_{\alpha\vec{k}}(\vec{x})$  è caratterizzato dalla stessa periodicità del reticolo. Vale quindi la relazione per gli autovalori di singola particella:

$$(h_1 \varphi_{\alpha\vec{k}})(\vec{x}) = \varepsilon_{\alpha\vec{k}} \varphi_{\alpha\vec{k}}(\vec{x})$$

È possibile scegliere una base complementare di autostati di singola particella, come le funzioni di Wannier  $\phi_\alpha(\vec{x} - \vec{R}_i)$  definite, chiamando  $L$  il numero di ioni nel reticolo, da:

$$\phi_\alpha(\vec{x}) = \frac{1}{\sqrt{L}} \sum_{\vec{k}} \varphi_{\alpha\vec{k}}(\vec{x})$$

Le funzioni  $\phi_\alpha(\vec{x} - \vec{R}_i)$  sono centrate in  $\vec{R}_i$ , è possibile scegliere un set di funzioni di Wannier caratterizzate da una forte localizzazione nei pressi del sito atomico (cioè che decadano velocemente allontanandosi dalla posizione  $\vec{R}_i$ ). Queste costituiscono quindi l'analogo reticolare delle funzioni d'onda atomiche e sono mutualmente ortogonali per indici di banda  $\alpha$  e di sito  $i$  diversi.

È possibile ricavare la relazione inversa

$$\varphi_{\alpha\vec{k}}(\vec{x}) = \frac{1}{\sqrt{L}} \sum_i e^{i\vec{k}\vec{R}_i} \phi_\alpha(\vec{x} - \vec{R}_i)$$

Introducendo ora gli operatori di creazione  $c_{\alpha\vec{k}a}^\dagger$  e distruzione  $c_{\alpha\vec{k}a}$  che, rispettivamente, creano o distruggono un elettrone con spin  $a$  nello stato di Bloch  $\varphi_{\alpha\vec{k}}(\vec{x})$ , facendone la trasformata di Fourier è possibile ottenere analoghi operatori di creazione e distruzione per le funzioni di Wannier:

$$c_{\alpha ia}^\dagger = \frac{1}{\sqrt{L}} \sum_{\vec{k}} e^{-i\vec{k}\vec{R}_i} c_{\alpha\vec{k}a}^\dagger$$

È possibile quindi definire l'operatore di Campo (Field operator)  $\Psi_a^\dagger$ , che crea un elettrone di spin  $a$  in posizione  $\vec{x}$ , in 2 modi equivalenti:

$$\Psi_a(\vec{x}) = \sum_{\alpha\vec{k}} \varphi_{\alpha\vec{k}}^*(\vec{x}) c_{\alpha\vec{k}a}^\dagger = \sum_{\alpha i} \phi_\alpha(\vec{x} - \vec{R}_i) c_{\alpha ia}^\dagger$$

Riscrivendo in termini del field operator l'Hamiltoniana questa diventa:

$$H = \sum_{a=\uparrow\downarrow} \int d^3\vec{x} \Psi_a^\dagger(\vec{x}) h_1 \Psi_a(\vec{x}) + \frac{1}{2} \sum_{a,b=\uparrow\downarrow} \int d^3\vec{x} d^3\vec{y} \Psi_a^\dagger(\vec{x}) \Psi_b^\dagger(\vec{y}) U(\vec{x}, \vec{y}) \Psi_b(\vec{y}) \Psi_a(\vec{x})$$

In termini di operatori di creazione e distruzione:

$$H = \sum_{\alpha i j a} t_{ij}^{\alpha} c_{\alpha i a}^{\dagger} c_{\alpha j a} + \frac{1}{2} \sum_{\alpha \beta \gamma \delta} \sum_{i j k l} U_{ijkl}^{\alpha \beta \gamma \delta} c_{\alpha i a}^{\dagger} c_{\beta j b}^{\dagger} c_{\gamma k b} c_{\delta l a}$$

Dove sono presenti i termini:

**hopping matrix element**

$$t_{ij}^{\alpha} = \int d^3 \vec{x} d^3 \vec{y} \phi_{\alpha}^*(\vec{x} - \vec{R}_i) (h_1 \phi_{\alpha})(\vec{x} - \vec{R}_j) = \frac{1}{L} \sum_{\vec{k}} e^{i\vec{k}(\vec{R}_i - \vec{R}_j)} \varepsilon_{\alpha \vec{k}}$$

**Overlap integral interaction parameter**

$$U_{ijkl}^{\alpha \beta \gamma \delta} = \int d^3 \vec{x} d^3 \vec{y} \phi_{\alpha}^*(\vec{x} - \vec{R}_i) \phi_{\beta}^*(\vec{x} - \vec{R}_j) U(\vec{x}, \vec{y}) \phi_{\gamma}(\vec{x} - \vec{R}_k) \phi_{\delta}(\vec{x} - \vec{R}_l)$$

Una scelta ottimale di  $\phi_{\alpha}$  e  $V_A$  minimizza l'influenza dell'interazione di Coulomb tra elettroni, in maniera tale che quando i parametri di interazione siano sufficientemente piccoli possano essere trascurati o trattati perturbativamente, portando così al **modello di Tight-Binding**.

Il modello di Hubbard si ottiene quando i parametri di interazione non sono trascurabili ma il loro range di azione è ancora molto piccolo, questo significa considerare interazione coulombiana intra-atomica molto grande rispetto all'interazione coulombiana inter-atomica e al tempo stesso non trascurabile rispetto ai termini di hopping.

Questa situazione è caratteristica di terre rare e transizioni di Mott (da verificare).

Quando la superficie di Fermi giace in una singola banda di conduzione è lecito proiettare l'Hamiltoniana multibanda su una Hamiltoniana efficace monobanda.

Immaginiamo ad esempio una situazione in cui le interazioni tra bande sono deboli e allo stesso tempo le bande eccetto  $\alpha$  sono molto lontane dal livello di Fermi.

Se siamo interessati soltanto alle energie in prossimità del livello di Fermi, l'effetto delle bande ad alta energia consiste nel cambiare i parametri di interazione e di hopping per  $\alpha$ .

## 1.2 Costruzione del modello di Hubbard: ipotesi e approssimazioni

---

Allora è possibile passare da un' Hamiltoniana multibanda:

$$H = \sum_{\alpha i j a} t_{ij}^{\alpha} c_{\alpha i a}^{\dagger} c_{\alpha j a} + \frac{1}{2} \sum_{\alpha \beta \gamma \delta} \sum_{i j k l} U_{ijkl}^{\alpha \beta \gamma \delta} c_{\alpha i a}^{\dagger} c_{\beta j b}^{\dagger} c_{\gamma k b} c_{\delta l a}$$

A un Hamiltoniana monobanda con termini efficaci  $t_{ij}$  e  $U$ :

$$H = \sum_{i j a} t_{ij} c_{i a}^{\dagger} c_{j a} + \frac{U}{2} \sum_i \sum_{a b} c_{i a}^{\dagger} c_{i b}^{\dagger} c_{i b} c_{i a}$$

I parametri efficaci di hopping e di interazione si ricavano rispettivamente dalla Density Functional Theory e dal paragone con i dati sperimentali, ottenendo una descrizione qualitativa.

In generale, il termine repulsivo sarebbe caratterizzato da  $U_i$  caratteristico per ogni sito ma nel caso di reticoli invarianti per traslazioni discrete si ha su ogni sito atomico lo stesso termine di repulsione.

Un' ulteriore semplificazione, compatibile col fatto che le funzioni di Wannier siano fortemente localizzate attorno al sito atomico, si compie nel considerare l'interazione a primi vicini:

$$\begin{aligned} t_{ij} &= -t \delta_{i, i \pm 1} \quad i \neq j \\ t_{ii} &= \varepsilon_i \quad i = j \end{aligned}$$

Inoltre definendo  $n_{ia} = c_{ia}^{\dagger} c_{ia}$  si ottiene la forma:

$$H = -t \sum_{\langle i, j \rangle, a} c_{i a}^{\dagger} c_{j a} + U \sum_i n_{i \uparrow} n_{i \downarrow}$$

**Hamiltoniana di Hubbard**



## Capitolo 2

# Programmazione di un computer quantistico con Qiskit

Nell'implementazione di un algoritmo per computer quantistico, finalizzato allo studio dell'Hamiltoniana di Hubbard, si è fatto uso del pacchetto di python Qiskit per la simulazione e programmazione di computer quantistici.

Questo permette di simulare su computer tradizionali dei qubits, e di comporre circuiti su di essi attraverso porte logiche fondamentali già implementate.

Sono inoltre implementati oggetti che permettono di rappresentare diversi sistemi fisici e ottenere le rispettive Hamiltoniane nel formalismo della seconda quantizzazione. Su queste ultime Qiskit permette di effettuare la trasformazione di Jordan-Wigner per ricavare la rappresentazione degli operatori di seconda quantizzazione in operatori sullo spazio degli stati dei qubits.

Sono infine implementati alcuni algoritmi quantistici, tra cui VQE, e altri algoritmi classici che agiscono sui sistemi fisici rappresentabili in qiskit, questi saranno illustrati nella **Sezione 4.1**.

## 2.1 Circuiti di qubits e porte logiche fondamentali

---

### 2.1 Circuiti di qubits e porte logiche fondamentali

Qiskit permette attraverso l'oggetto `QuantumCircuit` di comporre circuiti di qubits. Definiamo la base computazionale del  $k$ -esimo qubit con gli stati  $\{|0\rangle_k, |1\rangle_k\}$ . Questo in generale si troverà in uno stato  $|q_k\rangle$  sovrapposizione di questi due stati. Lo stato globale di un sistema di  $N$  qubits sarà dato dallo stato prodotto tensore degli stati dei singoli qubits che lo compongono:

$$|q_0\rangle \otimes |q_i\rangle \otimes \dots \otimes |q_{N-1}\rangle$$

Un operatore logico su qubit trasforma lo stato di un qubit  $\alpha|0\rangle + \beta|1\rangle \rightarrow \alpha'|0\rangle + \beta'|1\rangle$ . Perchè questo sia ancora uno stato ammissibile deve valere ancora  $|\alpha'|^2 + |\beta'|^2 = 1$ . Per questo motivo è possibile dimostrare che ogni operatore logico  $\hat{U}$  che agisca su qubits deve essere unitario, cioè  $\hat{U}\hat{U}^\dagger = \hat{U}^\dagger\hat{U} = \mathbf{1}$ . Essendo unitario l'operatore logico è un operazione reversibile.

Un operatore logico può agire su uno o più qubit:

- Gli operatori single-qubit agiscono come rotazioni sullo spazio degli stati definito dalla base  $\{|0\rangle, |1\rangle\}$ :  
 $\hat{U} = e^{i\alpha} R_{\vec{n}}(\theta)$
- Gli operatori multiple-qubit sono rappresentabili come il prodotto tensore di operatori a singolo qubit agenti ognuno su uno spazio degli stati dei singoli qubits, per esempio un operatore a 2 qubits è definito da:  
 $\hat{U} = \hat{U}_1 \otimes \hat{U}_2$  con  $\hat{U}_k$  operatore single-qubits agente sul  $k$ -esimo qubit

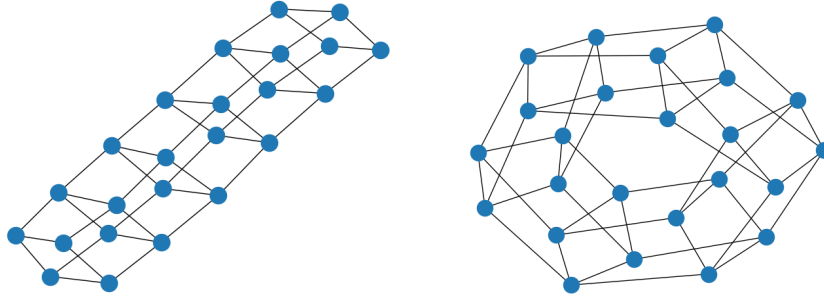
Un circuito è definito dalla composizione (in serie o in parallelo) di operazioni logiche su  $N$  qubits. Questo porta, definito lo stato iniziale come sopra, all'evoluzione del sistema in un nuovo stato all'uscita del circuito.

Possiamo definire delle operazioni logiche fondamentali su qubits con cui comporre circuiti e operazioni più complesse. Queste operazioni sono completamente definite dalla loro azione sui vettori della base computazionale e possono essere rappresentate in forma matriciale assumendo:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

## 2.2 Reticolo cristallino e Hamiltoniana di Hubbard

Qiskit permette di analizzare lo spettro di un sistema reticolare cristallino a partire dal grafo che ne definisce la struttura. A questo vengono assegnati i parametri che ne caratterizzano principalmente il comportamento energetico come i valori del termine di hopping tra siti reticolari e il termine di contributo energetico dato dall'elettrone sul singolo sito atomico. Qiskit permette inoltre di fissare le particolari condizioni al contorno che caratterizzano il reticolo che può essere aperto oppure periodico, come mostrato in Figura 2.1.



**Figura 2.1:** Esempio di reticolo tridimensionale  $2 \times 2 \times 6$  con condizioni al contorno aperte o periodiche

È implementato, inoltre, un oggetto `FermiHubbardHamiltonian` che, dato il grafo descrivente il reticolo che si intende studiare, fornisce l'Hamiltoniana di Hubbard in termini di operatori di creazione e distruzione nel formalismo della seconda quantizzazione (fissati i parametri di hopping, repulsione onsite e il contributo energetico per singolo elettrone).



## Capitolo 3

# Implementazione del modello di Hubbard su un computer quantistico

L'Hamiltoniana del modello di Hubbard in seconda quantizzazione è scritta in termini di operatori di creazione e distruzione che agiscono sugli stati di occupazione degli orbitali di spin presenti nel reticolo.

Si pone quindi il problema di come sia possibile mappare questi stati nello spazio degli stati dei qubits e di come quindi si trasformino gli operatori nel passaggio tra questi due spazi.

### 3.1 Stati fermionici e base computazionale dei qubits

Dal principio di esclusione di Pauli, due fermioni identici non possono trovarsi nello stesso stato quantico. Questo comporta che, considerando uno stato fissato, esso può essere occupato al massimo da un fermione oppure non essere occupato affatto.

Ciò implica che un determinato stato fermionico può essere visto come un autospazio di degenerazione pari a 2 rispetto al proprio numero di occupazione. È possibile quindi definire in questo autospazio degenerare una base, in particolare la base degli autostati del numero di occupazione  $\{|0\rangle, |1\rangle\}$ .

Nel formalismo della seconda quantizzazione, dato un sistema composto da  $N$  fermioni, si caratterizza uno stato generico del sistema come combinazione lineare degli stati definiti dal prodotto tensoriale degli stati fermionici di singola particella con numero di occupazione

### 3.2 Trasformazione di Jordan-Wigner (JWT): mappare operatori nello spazio dei qubits

---

definito.

$$|\psi_1, n_1\rangle |\psi_2, n_2\rangle \dots |\psi_k, n_k\rangle \dots |\psi_N, n_N\rangle$$

Il sistema qubit è definito in modo tale da avere stati che esistono su uno spazio di Hilbert di dimensione 2, rappresentabili quindi in una base  $\{|0\rangle, |1\rangle\}$ , detta base computazionale

Questo parallelismo, tra gli autostati del numero di occupazione e la base computazionale, induce naturalmente a mappare ogni possibile stato fermionico in un sistema qubit in cui la base computazionale rappresenta gli autovettori relativi al numero di occupazione.

### 3.2 Trasformazione di Jordan-Wigner (JWT): mappare operatori nello spazio dei qubits

Un Hamiltoniana nel formalismo della seconda quantizzazione viene convenientemente scritta in termini degli operatori di creazione e distruzione la cui azione, nel caso dei fermioni, su autovettori del numero di occupazione relativi all'orbitale di spin  $k$  è rappresentata come segue:

$$\begin{aligned} a_k^\dagger |0\rangle_k &= |1\rangle_k & a_k^\dagger |1\rangle_k &= 0 \\ a_k |0\rangle_k &= 0 & a_k |1\rangle_k &= |0\rangle_k \end{aligned}$$

Passando a una scrittura matriciale nello spazio definito da un singolo stato fermionico, la cui base è composta dai vettori  $\{|0\rangle_k = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |1\rangle_k = \begin{bmatrix} 0 \\ 1 \end{bmatrix}\}$  otteniamo la forma :

$$a_k^\dagger = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \quad a_k = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

Questi, non essendo operatori unitari, non sono direttamente implementabili come operazioni logiche sullo spazio dei qubit ma possono essere riscritti in termini di matrici di Pauli che invece risultano facilmente rappresentabili.

(Per comodità verrà utilizzata la notazione  $X = \sigma_x, \quad Y = \sigma_y, \quad Z = \sigma_z$ )

$$a_k^\dagger = \frac{X_k - iY_k}{2} \quad a_k = \frac{X_k + iY_k}{2}$$

Questa costruzione funziona bene per descrivere l'azione su un singolo orbitale di spin, ma è fallimentare per quanto riguarda sistemi con 2 o più orbitali di spin: non rispettano le

## Implementazione del modello di Hubbard su un computer quantistico

---

proprietà di antisimmetria legate ai fermioni.

Essendo i fermioni caratterizzati da antisimmetria per scambio, vale la relazione:

$$a_j^\dagger a_k^\dagger = -a_k^\dagger a_j^\dagger$$

Ma la rappresentazione su stati di qubit definita sopra non soddisfa l'antisimmetria:

$$\left(\frac{X_j - iY_j}{2}\right)\left(\frac{X_k - iY_k}{2}\right) = \left(\frac{X_k - iY_k}{2}\right)\left(\frac{X_j - iY_j}{2}\right)$$

La soluzione più semplice a questo problema consiste nella trasformazione di Jordan-Wigner che implementa gli operatori di costruzione/distruzione come:

$$a_1^\dagger = \left(\frac{X - iY}{2}\right) \otimes 1 \otimes 1 \otimes \dots \otimes 1$$

$$a_2^\dagger = Z \otimes \left(\frac{X - iY}{2}\right) \otimes 1 \otimes \dots \otimes 1$$

$$a_3^\dagger = Z \otimes Z \otimes \left(\frac{X - iY}{2}\right) \otimes \dots \otimes 1$$

$$a_3^\dagger = Z \otimes Z \otimes Z \otimes \dots \otimes \left(\frac{X - iY}{2}\right)$$

Operatori di Costruzione e distruzione così implementati nello spazio dei qubits soddisfano le regole di antisimmetria per scambio caratteristiche dei fermioni.

### 3.2 Trasformazione di Jordan-Wigner (JWT): mappare operatori nello spazio dei qubits

---

#### 3.2.1 Rappresentazione di osservabili fisiche e località dell'interazione

Le osservabili fisiche in seconda quantizzazione presentano sempre un numero pari di operatori di costruzione e distruzione a causa delle regole di superselezione.

Le trasformazioni, definite su  $N$  qubits, che ci interessano sono quindi caratterizzate da operatori della forma:

$$\begin{aligned} a_k^\dagger a_k &= [Z^{\otimes(k-1)} \otimes (\frac{X - iY}{2}) \otimes 1^{\otimes(N-k)}] [Z^{\otimes(k-1)} \otimes (\frac{X + iY}{2}) \otimes 1^{\otimes(N-k)}] \\ &= 1^{\otimes(k-1)} \otimes (\frac{1 - Z}{2}) \otimes 1^{\otimes(N-k)} \end{aligned}$$

$$a_k^\dagger a_{k+n} + a_k a_{k+n}^\dagger = \frac{1}{2} [1^{\otimes(k-1)} \otimes X \otimes Z^{\otimes(n-1)} \otimes X \otimes 1^{\otimes(N-k-n)} + 1^{\otimes(k-1)} \otimes Y \otimes Z^{\otimes(n-1)} \otimes Y \otimes 1^{\otimes(N-k-n)}]$$

Finchè vengono trattati sistemi monodimensionali, considerando interazioni tra primi vicini, non si pongono problemi riguardo alla rappresentazione di stati fermionici in termini di qubit: esiste sempre un ordinamento tale che il termine  $Z^{\otimes(n-1)}$  (detto **string operator**) si riduca ad agire come l'identità e quindi l'interazione locale, cioè coinvolgente solo due stati fermionici, venga mappata in una interazione locale tra i rispettivi 2 qubits attraverso JWT, come mostrato in [1].

Per sistemi multidimensionali in generale non esiste un ordinamento tale per cui, in approssimazione di interazione tra primi vicini, lo string operator si riduca ad agire come l'identità. La causa di questo fatto risiede nell'impossibilità di mappare un sistema multidimensionale come quello reticolare in un array di qubits preservando l'adiacenza degli stati mappati. Per questo motivo l'interazione che nello spazio degli stati fermionici agisce localmente tra due stati adiacenti non viene mappata in un'interazione locale tra i rispettivi due qubits, bensì agisce su tutti i qubits che si interpongono tra questi.

Diventa quindi necessario introdurre metodi per aggirare il problema nel caso multidimensionale.



## Capitolo 4

# Simulazione del modello di Hubbard mediante un circuito quantistico

### 4.1 Algoritmi variazionali

Gli algoritmi variazionali sono una classe di algoritmi per quantum computer che mira a risolvere problemi di minimizzazione. Questi sono caratterizzati da 3 componenti principali:

- Una **funzione costo** da minimizzare  $C(\vec{\theta})$
- Un set di stati parametrizzati  $|\psi(\vec{\theta})\rangle$ , detto **ansatz**, su cui valutare la funzione costo per cercarne il minimo, implementati attraverso circuiti di qubit.
- Un **ottimizzatore**, implementato tramite un algoritmo classico, che restituisca il valore dei parametri  $\vec{\theta}$  con cui preparare lo stato alla successiva iterazione.

L'obiettivo di questo tipo di algoritmi è quello di minimizzare la data funzione di costo  $C(\vec{\theta})$  esplorando il set di stati preparati al variare del vettore di parametri  $\vec{\theta}$ . Questi parametri vengono variati, attraverso l'ottimizzatore classico, fino al raggiungimento della convergenza dei valori della funzione di costo.

Il teorema variazionale riportato sotto garantisce la possibilità di mappare il problema della ricerca dello stato fondamentale per un qualsiasi sistema fisico in un problema di minimizzazione.

## 4.2 Variational Quantum Eigensolver (VQE)

---

### 4.1.1 Teorema variazionale per il calcolo dello stato fondamentale

Essendo lo stato fondamentale, per una fissata Hamiltoniana  $H$ , lo stato del sistema considerato corrispondente all'autovalore minore dell'energia, denominato  $E_0$ , vale per ogni generico stato  $|\psi\rangle$  la disuguaglianza:

$$\langle\psi|H|\psi\rangle \geq E_0$$

Quindi è possibile risolvere il problema di trovare lo stato fondamentale di una Hamiltoniana, risolvendo un problema equivalente di minimizzazione di una funzione costo definita come il valore di aspettazione dell'Hamiltoniana.

## 4.2 Variational Quantum Eigensolver (VQE)

L'algoritmo Variational Quantum Eigensolver (VQE) [2] [3] [4] è un algoritmo variazionale, il cui scopo è ottenere l'autovalore e l'autovettore dello stato fondamentale di un dato operatore. Nel nostro caso si tratta dell'operatore Hamiltoniano associato alla descrizione di reticoli cristallini secondo il modello di Hubbard.

L'implementazione di VQE in quanto algoritmo variazionale è rappresentabile nei seguenti step:

- Preparazione, attraverso un operatore unitario  $U_R$ , dello stato iniziale detto Reference State  $|\rho\rangle$ .
- Attraverso un operatore unitario  $U_V(\vec{\theta})$ , detto operatore variazionale, creazione degli stati su cui valutare la funzione costo  
 $|\psi(\vec{\theta})\rangle = U_V(\vec{\theta}) |\rho\rangle = U_A(\vec{\theta}) |0\rangle$
- Valutazione della funzione costo per ogni stato preparato  $\psi(\vec{\theta})$
- Attraverso un ottimizzatore classico scegliere il prossimo set di parametri  $\theta$

Questi step si ripetono fino al raggiungimento della convergenza dei valori di  $C(\vec{\theta})$ . I criteri di convergenza e i metodi di esplorazione dello spazio dei parametri sono determinati dallo specifico ottimizzatore scelto.

## 4.3 Ottimizzatori

Mentre la preparazione dell'ansatz e la funzione di costo dipendono fortemente dal problema analizzato, gli ottimizzatori si basano su metodi di ricerca del minimo pre-esistenti e la scelta del particolare ottimizzatore dipende dalle caratteristiche che definiscono il problema. Un ottimizzatore, fissata la funzione di costo  $C(\vec{\theta})$ , esplora lo spazio dei parametri iterando valutazioni in un nuovo punto dello spazio su cui calcolare la funzione  $C(\vec{\theta})$ . In particolare si può dividere l'insieme degli ottimizzatori in diverse sottoclassi a seconda di come questi esplorano lo spazio:

### 4.3.1 Ottimizzatori Locali

Gli ottimizzatori locali esplorano lo spazio partendo dal punto iniziale che gli viene fornito e spostandosi a seconda dei valori misurati in punti situati nella regione circostante al punto iniziale. Questo implica la possibilità di una convergenza molto veloce ma fortemente condizionata dalla scelta del punto iniziale, rendendo quindi gli ottimizzatori locali particolarmente vulnerabili ai minimi locali.

Tra questi sono stati analizzati in in questo lavoro di Tesi gli ottimizzatori Sequential Least Squares (SLSQP) che, partizionando lo spazio di ricerca del minimo, effettua la ricerca del minimo locale approssimando localmente la funzione di costo come un polinomio di grado 2 [5] e continuous Univariate Marginal Distribution Algorithm (UMDA) [6] che ad ogni iterazione seleziona i punti in cui la valutazione della funzione di costo risulta essere minima, partendo alla prima iterazione da una distribuzione normale univariata.

Altri ottimizzatori locali implementati in Qiskit sono riportati nella Tabella 4.1. Questi sono stati inizialmente utilizzati, ma non esibendo significative differenze, se non nei maggiori tempi di convergenza, non sono stati esposti tra i Risultati nel Capitolo 6

### 4.3.2 Ottimizzatori Globali

Gli ottimizzatori globali effettuano la ricerca del minimo su diverse regioni dello spazio dei parametri, in maniera tale da diminuire il rischio di fermarsi ad un minimo locale ma rallentando inevitabilmente la convergenza.

In Qiskit sono implementati gli ottimizzatori globali Controlled Random Search (CRS), ESCH, DIviding RECTangles Locally-biased RANDomized (DIRECT-L-RAND) ed Improved Stochastic Ranking Evolution (ISRES), per la descrizione di questi ottimizzatori si rimanda al Capitolo 6 e alla documentazione fornita da Qiskit[6].

## 4.3 Ottimizzatori

---

### 4.3.3 Ottimizzatori e metodo di ricerca del minimo

È possibile dividere i metodi di ricerca del minimo (sia locali che globali) in due categorie a seconda di come funziona l'esplorazione dell'intervallo su cui agisce l'ottimizzatore.

Questa può avvenire utilizzando o meno la valutazione del gradiente della funzione costo nello spazio considerato.

#### Gradient-Based

Nel caso in cui fosse possibile ricavare il gradiente della funzione costo, il modo più semplice per ricavarne il minimo partendo da un punto iniziale è quello di variare il vettore dei parametri seguendo la direzione di maggiore discesa del gradiente fino a convergere in un minimo locale.

Ciò garantisce una convergenza molto veloce ma non ci sono garanzie per quanto riguarda l'ottenere una soluzione ottimale.

#### Gradient-Free

Gli ottimizzatori Gradient-Free sono adatti a casi in cui non si possa, o sia computazionalmente troppo costoso, ottenere il gradiente della funzione di costo. Tendenzialmente richiedono un uso maggiore delle risorse computazionali ma presentano un comportamento migliore nell'evitare convergenza a minimi locali.

Optimizers in Qiskit	
Optimizer	Search Type
ADAM	Local
AQGD	Local
CG	Local
COBYLA	Local
L_BFGS_B	Local
GSL	Local
GradientDescent	Local
NELDER_MEAD	Local
NFT	Local
P_BFGS	Local
POWELL	Local
SLSQP	Local
SPSA	Local
QNPSA	Local
TNC	Local
SciPyOptimizer	Local
UMDA	Local
CRS	Global
DIRECT_L	Global
DIRECT_L_RAND	Global
ESCH	Global
ISRES	Global

**Tabella 4.1:** Ottimizzatori implementati in Qiskit



## Capitolo 5

# Variational Quantum Eigensolver per reticolo monodimensionale

Si consideri un reticolo monodimensionale in approssimazione di hopping tra primi vicini. Come accennato nel **paragrafo 4.2.1** non vi sono particolari problemi legati all'ordinamento degli stati fermionici mappati in qubit. Di conseguenza è possibile mappare attraverso la trasformazione di Jordan-Wigner gli operatori di seconda quantizzazione preservando la località dell'interazione.

Come accennato nel **Capitolo 4** ogni orbitale di spin è rappresentabile attraverso un qubit. Avendo come obbiettivo la ricerca del minimo del valore di aspettazione dell'Hamiltoniana, è conveniente considerare, per il passaggio a qubits, solamente gli stati fermionici di minima energia per ogni sito atomico.

Infatti, denominando  $\varepsilon_{0i}$  il minimo contributo energetico legato alla presenza di un elettrone nello sito  $i$ -esimo e  $\varepsilon_{ni}$  i contributi energetici successivi, vale la relazione banale:

$$\varepsilon_{0i} \leq \varepsilon_{ni}$$

Inoltre è possibile riscrivere il contributo di hopping dell'Hamiltoniana come:

$$-t \sum_{\langle i,j \rangle, \sigma} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma} + \sum_{i,\sigma} \varepsilon_i n_{i\sigma}$$

con  $\varepsilon_i$  relativo al particolare contributo considerato per il sito  $i$ -esimo.

È evidente che il contributo  $\varepsilon_{0i}$  minimizzi la seconda parte del termine di hopping ad  $i$  fissato.

Per questo motivo conviene considerare solo gli stati di Wannier che forniscano il minimo contributo per ogni sito e quindi, fissato il numero  $M$  di atomi componenti il reticolo,

## 5.1 Preparazione stato iniziale: Reference state

---

saranno mappati solamente  $2M$  orbitali di spin.

Dunque per rappresentare un reticolo costituito da un numero  $M$  di siti atomici, e quindi  $2M$  orbitali di spin, sono necessari  $2M$  qubit.

Oltre che garantire la convergenza diminuendo la dimensione dello spazio degli stati da esplorare, questo procedimento ha il vantaggio di ridurre il numero di qubit da implementare per risolvere il problema della minimizzazione e garantisce la possibilità di mappare biunivocamente stati fermionici in qubits.

## 5.1 Preparazione stato iniziale: Reference state

Nel processo di minimizzazione di una data funzione di costo, lo stato iniziale da cui si esplora lo spazio dei parametri svolge un ruolo fondamentale, come descritto da IBM Quantum Learning in Variational Algorithm Design [7]. Il processo di ricerca del minimo è infatti fortemente influenzato dalle caratteristiche dello spazio su cui si svolge la ricerca, come la presenza di minimi locali o di Plateau, che potrebbero impedire il raggiungimento del minimo globale (nel caso in cui si utilizzino ottimizzatori locali). Quindi una buona caratterizzazione dello stato iniziale permette di aggirare queste problematiche rendendo più rapida e precisa la convergenza.

Inoltre una buona caratterizzazione dello stato iniziale e della realizzazione dell'ansatz in cui questo viene fatto evolvere possono ridurre notevolmente lo spazio su cui viene effettuata la ricerca rendendo più vantaggioso l'algoritmo.

Caratterizziamo dunque lo spazio dei Reference States su cui agirà VQE.

Per studiare il comportamento elettronico di un sistema reticolare si considera il contributo proveniente solo dagli elettroni di valenza, cioè appartenenti a stati meno legati. Questo fa sì che l'operatore Hamiltoniano sia caratterizzato da un numero fissato di elettroni di valenza  $N_e$ , imponendo così un vincolo ai numeri di occupazione sullo stato del sistema: chiamando  $n_{i,\sigma}$  il numero di occupazione dello stato fermionico il vincolo si esprime con l'equazione

$$\sum_{i,\sigma} n_{i,\sigma} = N_e$$

Nello spazio dei qubits questa condizione si traduce imponendo che il generico vettore di sistema sia dato dal prodotto tensore di vettori di singolo qubit

$$|q_0\rangle \otimes |q_1\rangle \otimes \dots \otimes |q_{2M-1}\rangle \quad q_k = 0, 1$$



tali per cui

$$\sum_{k=0}^{2M-1} q_k = N_e$$

Cioè gli stati per cui  $N_e$  qubits si trovano nello stato  $|1\rangle$ .

Questo vincolo fa sì che diminuisca la dimensione dello spazio da esplorare, ma impone la conservazione del numero di elettroni di valenza  $N_e$  durante l'evoluzione effettuata attraverso il circuito di preparazione dell'ansatz.

### 5.1.1 Implementazione Reference state su circuito di qubit

La condizione che siano presenti un certo numero  $N_e$  di elettroni di valenza viene implementata nella preparazione dello stato iniziale applicando  $N_e$  gate X in una delle configurazioni realizzabili per un sistema di  $2M$  qubits. Il numero di configurazioni ottenibili segue la legge del calcolo combinatorio relativa alle combinazioni semplici: su  $2M$  qubits le configurazioni su cui disporre  $N_e$  gate X sono date da

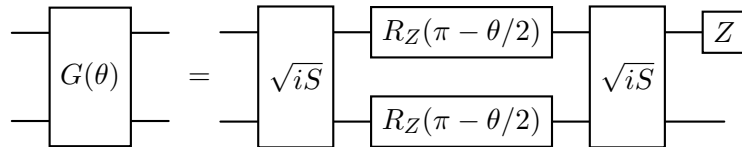
$$\frac{2M!}{(2M - N_e)!N_e!}$$

Successivamente a questi vengono applicate delle rotazioni, come suggerito nell'articolo [3], attraverso dei Gate  $G(\theta)$  definiti dalla matrice:

$$G(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) & 0 \\ 0 & \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In maniera tale da considerare sovrapposizioni di stati a numero di elettroni fissato.

Per implementare queste rotazioni a livello circuitale si utilizza un circuito di profondità pari a 4 layer. Per implementare rotazioni che coinvolgano ogni possibile coppia di qubit contigui è necessario creare almeno due strati di gate di rotazioni portando la profondità totale a 8 layer



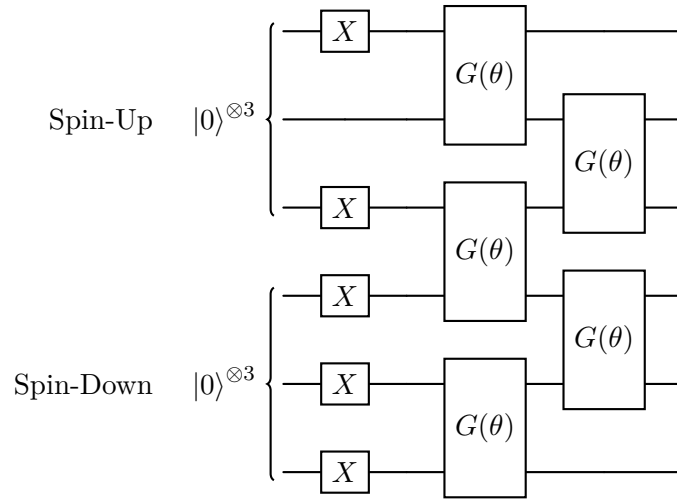
**Figura 5.1:** Circuito per implementare la rotazione  $G(\theta)$ , costruito in [3]

## 5.2 Ansatz: costruzione

---

Nello spazio dei qubits, fissato  $M$  numero di siti del reticolo, si è scelto di rappresentare gli stati di spin up nei primi  $M$  qubit e quelli relativi agli stati spin down nei successivi  $M$ . Così facendo nel termine di hopping, presentando al suo interno prodotti di operatori di costruzione e distruzione su indici di sito diversi, si cancella il termine String Operator dovuto alla trasformazione di Jordan-Wigner.

In un sistema così strutturato quindi si applicano i Gate  $X$  sullo selezionando l'indice di sito e il valore iniziale dello spin. Una volta eseguite le rotazioni, effettuandole anche tra i due settori di spin, lo stato diventa, in generale, una sovrapposizione di tutti i possibili stati a numero di elettroni fissato.



**Figura 5.2:** Circuito Stato iniziale per reticolo monodimensionale a  $N=3$  siti:

Le rotazioni vengono effettuate sia tra qubits contigui appartenenti allo stesso settore di spin che tra qubits contigui appartenenti a settori di spin diversi.

## 5.2 Ansatz: costruzione

La costruzione dell'ansatz, ossia il set di stati su cui viene valutata la funzione di costo da minimizzare, svolge un ruolo chiave nella risoluzione del problema di ricerca dello stato fondamentale.

Infatti una buona caratterizzazione dell'ansatz permette di ridurre notevolmente le dimensioni dello spazio di stati da esplorare, mentre una cattiva caratterizzazione rischia di escludere il minimo ricercato e quindi di ridursi ad esplorare zone in cui non si arriverà

mai alla soluzione ottima.

Inoltre, essendo l'ansatz implementato su Quantum Computer Gate Model, la sua effettiva realizzazione consiste in un circuito su qubits, si pone quindi il problema di creare un circuito che realizzi un certo set di stati nella maniera più efficiente possibile.

Definiamo la profondità di un circuito quantistico come la misura di quanti "layers" di quantum gates, eseguiti in parallelo, servono per completare l'operazione definita dal circuito. Siccome i gate richiedono tempo e potenza di calcolo per essere implementati in una simulazione su computer classico, la profondità del circuito che implementa l'ansatz è uno dei parametri che determinano l'efficienza dell'algoritmo VQE.

L'obiettivo della costruzione dell'ansatz è quindi di ottenere un circuito che consenta l'accesso ad un set di stati contenente lo stato fondamentale o una sua buona approssimazione, riducendo al minimo le operazioni da implementare nel circuito.

### 5.2.1 Efficient Hamiltonian Variational Ansatz (EHVA)

Un Metodo molto utilizzato nello studio dello spettro di operatori Hamiltoniani è il metodo Efficient Hamiltonian Variational Ansatz o EHVA riportato nell'articolo [8], in cui gli stati che compongono l'ansatz sono costruiti attraverso rotazioni generate dai termini che compongono l'Hamiltoniana analizzata.

Consideriamo infatti una famiglia di Hamiltoniane della forma

$$\hat{H} = \sum_a J_a \hat{h}_a$$

in cui i termini  $J_a$  sono scalari e i termini  $\hat{h}_a$  operatori.

Siano  $J_a^0$  e  $J_a^1$  scelte di  $J_a$  che definiscono le Hamiltoniane  $\hat{H}_0$ ,  $\hat{H}_1$  e supponiamo sia facile ottenere il GS di  $\hat{H}_0$  detto  $|\psi_1\rangle$  (per esempio nel caso dell'Hubbard model il modello non interagente).

Supponiamo ora sia possibile evolvere adiabaticamente da  $|\psi_1\rangle$  al GS di  $\hat{H}_1$ , questo processo è detto annealing. Se si effettua una divisione in piccoli step temporali  $dt$  del processo di annealing di evoluzione in un tempo totale  $T$ , allora l'evoluzione è ora rappresentata da una sequenza di rotazioni unitarie generate da Hamiltoniane che interpolano tra  $\hat{H}_0$  e  $\hat{H}_1$  della forma  $\hat{H} = \sum_a J_a \hat{h}_a$ .

Nel caso in cui l'operatore  $\hat{H} = \sum_a J_a \hat{h}_a$  sia composto da termini che commutano tra loro

## 5.2 Ansatz: costruzione

---

è possibile spezzare, senza errore il suo esponenziale come:

$$e^{-i\hat{H}t} = \prod_{j=1}^S \prod_a e^{-itJ_a^j \hat{h}_a}$$

Altrimenti si può ottenere una fattorizzazione approssimata attraverso il metodo di Trotter-Suzuki.

### Decomposizione di Trotter-Suzuki:

Dati  $\hat{A}, \hat{B}$  operatori arbitrari con una relazione di commutazione  $[\hat{A}, \hat{B}] \neq 0$  e un parametro  $x$ , vale la relazione:

$$e^{x(\hat{A}+\hat{B})} \simeq e^{x\hat{A}}e^{x\hat{B}} + O(x^2)$$

L'implementazione effettiva di questo metodo per un dato operatore Hamiltoniano avviene quindi applicando in sequenza gli operatori di evoluzione temporale generati dai singoli termini componenti l'Hamiltoniana  $\hat{H}_1$ , risultando in una scomposizione esatta nel caso in cui tra loro commutino ed utilizzando una scomposizione di Trotter-Suzuki altrimenti.

I coefficienti che compongono questa sequenza sono i parametri con cui si esplora lo spazio degli stati. Questi non vengono scelti attraverso la fattorizzazione esatta o il metodo di Trotter-Suzuki, bensì vengono lasciati liberi di variare.

Dividendo quindi in  $S$  step e troncando l'espansione di Trotter-Suzuki al primo ordine, si ottiene:

$$e^{-i\hat{H}_1 T/\hbar} = \prod_{j=0}^S e^{-i\hat{H}_1^j dt/\hbar} = \prod_{j=0}^S e^{-i \sum_a J_a^j \hat{h}_a dt/\hbar} \simeq \prod_{j=0}^S \prod_a e^{-i\theta_a^j \hat{h}_a}$$

Il metodo EHVA consiste quindi nel considerare un ansatz costituito da stati della forma riportata sopra in cui, dato uno stato che sia soluzione dell'Hamiltoniana  $\hat{H}_0$ , si compie un'evoluzione attraverso le rotazioni  $\hat{U}_a(\theta_a^j) = e^{-i\theta_a^j \hat{h}_a}$  generate dai termini che compongono l'Hamiltoniana che si intende analizzare.

Data quindi un'Hamiltoniana  $\hat{H}_1$  composta da termini  $\{\hat{h}_k\}_{k=1}^n$  e lo stato fondamentale dell'Hamiltoniana  $\hat{H}_0$  chiamato  $|\psi_1\rangle$ , otteniamo al termine di uno step uno stato della forma:

$$|\psi_T\rangle = e^{i\theta_n \hat{h}_n} \dots e^{i\theta_1 \hat{h}_1} |\psi_1\rangle$$

Se i termini che costituiscono l'Hamiltoniana  $\hat{H}_1$  sono caratterizzati da una qualche simmetria, allora gli stati  $|\psi_1\rangle$  e  $|\psi_T\rangle$  trasformano allo stesso modo sotto questa simmetria. Questo permette di ottenere la conservazione dei numeri quantici legati ad essa collegati durante l'evoluzione.

### 5.3 Efficient Hamiltonian Variational Ansatz per il modello di Hubbard

È possibile riscrivere l'operatore Hamiltoniano descrivente il modello di Hubbard in termini di tre operatori  $\hat{h}_h$ ,  $\hat{h}_v$ ,  $\hat{h}_U$  descrittivi rispettivamente i termini di hopping orizzontale, verticale e di interazione On-Site, l'idea alla base di questa scomposizione è stata originariamente presentata in [8].

$$\hat{H} = \sum_{\langle i,j \rangle \sigma} t_{ij} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma} + U \sum_i \hat{n}_{i\uparrow} \hat{n}_{i\downarrow} = \hat{h}_h + \hat{h}_v + \hat{h}_u$$

Costruendo quindi l'ansatz attraverso il metodo EHVA utilizzando un numero  $S$  di step, si ottiene il set di stati parametrizzati:

$$|\psi_T(\vec{\theta})\rangle = \prod_{b=1}^S [\hat{U}_h(\frac{\theta_h^b}{2}) \hat{U}_v(\frac{\theta_v^b}{2}) \hat{U}_U(\frac{\theta_U^b}{2})] |\psi_1\rangle$$

Nel caso del reticolo monodimensionale il termine di hopping verticale non è presente, si considera quindi:

$$|\psi_T(\vec{\theta})\rangle = \prod_{b=1}^S [\hat{U}_h(\frac{\theta_h^b}{2}) \hat{U}_U(\frac{\theta_U^b}{2})] |\psi_1\rangle$$

Riscrivendo la prima sommatoria dell'Hamiltoniana isolando esplicitamente la parte di contributo energetico data dal singolo elettrone sul sito atomico rispetto ai termini di salto tra siti otteniamo:

$$\sum_{\sigma=\uparrow\downarrow} \sum_{\langle i,j \rangle} t_{ij} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma} = \sum_{\sigma=\uparrow\downarrow} [\sum_i \varepsilon_i \hat{c}_{i\sigma}^\dagger \hat{c}_{i\sigma} - \sum_i \sum_{j=i\pm 1} t_{ij} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma}] = \hat{h}_\varepsilon + \hat{h}_h$$

Dove  $\varepsilon_i = -t_{ii}$ . Possiamo quindi scomporre l'Hamiltoniana del modello di Hubbard nei termini:

$$\hat{H} = \hat{h}_\varepsilon + \hat{h}_h + \hat{h}_U$$

con, ponendosi in approssimazione di primi vicini e in ipotesi di isotropia dello spazio (valida per reticolo invarianti per traslazioni discrete), i termini:

$$\begin{aligned} \hat{h}_h &= -t \sum_{\sigma=\uparrow\downarrow} \sum_i \sum_{j=i\pm 1} \hat{c}_{i\sigma}^\dagger \hat{c}_{j\sigma} \\ \hat{h}_\varepsilon &= \sum_{\sigma=\uparrow\downarrow} \sum_i \varepsilon_i \hat{c}_{i\sigma}^\dagger \hat{c}_{i\sigma} = \sum_{\sigma=\uparrow\downarrow} \sum_i \varepsilon_i \hat{n}_{i\sigma} \end{aligned}$$

### 5.3 Efficient Hamiltonian Variational Ansatz per il modello di Hubbard

---

$$\hat{h}_U = U \sum_i \hat{n}_{i\uparrow} \hat{n}_{i\downarrow}$$

Attraverso la trasformazione di Jordan-Wigner è possibile riscrivere la somma di operatori di costruzione e distruzione in termini delle matrici di Pauli, come visto nella **Sezione 4.2.1**, ottenendo:

$$\hat{h}_h = -\frac{t}{2} \sum_{\sigma=\uparrow\downarrow} \sum_i [X_{i,\sigma} X_{i+1,\sigma} + Y_{i,\sigma} Y_{i+1,\sigma}]$$

$$\hat{h}_\varepsilon = \sum_{\sigma=\uparrow\downarrow} \sum_i \varepsilon_i \frac{(1-Z)_{i,\sigma}}{2}$$

$$\hat{h}_U = \frac{U}{4} \sum_i [1_{i,\uparrow} 1_{i,\downarrow} - Z_{i\uparrow} - Z_{i\downarrow} - Z_{i\uparrow} Z_{i\downarrow}]$$

Si è utilizzata la notazione  $O_i O_j$  come forma compatta degli operatori agenti sul sistema di N qubit che in generale avrebbero la forma:  $1^{\otimes(i-1)} \otimes O \otimes 1^{\otimes(j-i-1)} \otimes O \otimes 1^{\otimes(N-j)}$

Gli operatori unitari che caratterizzano la costruzione dell'ansatz sono quindi riscrivibili come rotazioni generate dalle matrici di Pauli.

La rotazione relativa al termine di hopping:

$$\begin{aligned} e^{-i\theta_h \hat{h}_h} &= e^{it\frac{\theta}{2} \sum_{j,\sigma} [X_{j\sigma} X_{j+1,\sigma} + Y_{j,\sigma} Y_{j+1,\sigma}]} = e^{it\frac{\theta}{2} \sum_{j,\sigma} [X_{j,\sigma} X_{j+1,\sigma}]} e^{it\frac{\theta}{2} \sum_{j,\sigma} [Y_{j,\sigma} Y_{j+1,\sigma}]} \\ &= \prod_{j,\sigma} e^{it\frac{\theta}{2} X_{j,\sigma} X_{j+1,\sigma}} e^{it\frac{\theta}{2} Y_{j,\sigma} Y_{j+1,\sigma}} \end{aligned}$$

Dove ciascun membro della produttoria è rappresentabile con la matrice agente sullo spazio degli stati dei qubits  $j, \sigma$  e  $j+1, \sigma$ :

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(t\theta) & i\sin(t\theta) & 0 \\ 0 & i\sin(t\theta) & \cos(t\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Siccome vale la relazione

$$e^{iX_{j,\sigma} X_{k,\sigma}\theta} = 1\cos(\theta) + i(X_{j,\sigma} X_{k,\sigma})\sin(\theta)$$

La rotazione relativa al termine energetico On-Site:

$$e^{-i\theta \hat{h}_\varepsilon} = e^{-i\frac{\theta}{2} \sum_{j,\sigma} \varepsilon_j (1-Z)_{j,\sigma}} = \prod_{j,\sigma} e^{-i\varepsilon_j \frac{\theta}{2} (1-Z)_{j,\sigma}} = \prod_j e^{-i\varepsilon_j \frac{\theta}{2}} e^{i\varepsilon_j \frac{\theta}{2} Z_{j,\sigma}}$$

Implementabile come rotazione  $Rz(-\varepsilon_j \frac{\theta}{2})$  a meno di una fase globale.

La rotazione relativa al termine di repulsione On-Site:

$$e^{-i\theta\hat{h}_U} = e^{-i\frac{U}{4}\theta\sum_j[1_{j\uparrow}1_{j\downarrow}-Z_{j\uparrow}-Z_{j\downarrow}-Z_{j\uparrow}Z_{j\downarrow}]}$$

Implementabile con gate Controlled Rotation lungo z  $CRz(U\theta)$

È quindi possibile implementare semplicemente a livello circuitale gli stati parametrizzati  $|\psi(\vec{\theta})\rangle$

In figura 5.7 è possibile vedere una versione schematizzata che riassume il funzionamento dell'algoritmo VQE per il modello di Hubbard per cristalli monodimensionali.

### 5.3.1 Implementazione di EHVA su circuito di qubit

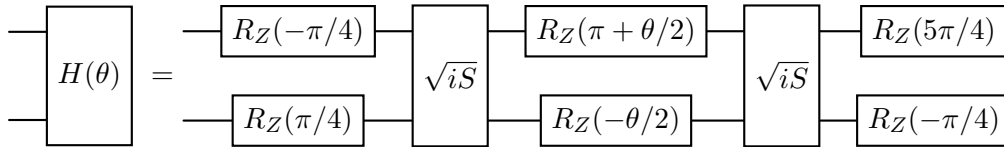
Utilizzando, per descrivere un reticolo monodimensionale ad M siti, il circuito descritto nella **Sezione 6.1.1**, i gate saranno indicati con la notazione a singolo indice che varia da 0 a 2M ad indicare i qubits su cui agiscono (da 0 a M stati di spin up, da M a 2M stati di spin down).

Nell'implementazione su circuito dell'ansatz si è utilizzata la costruzione degli operatori di hopping in termini solo dell'operatore di pauli  $\hat{Z}$  e delle rotazioni da esso generate, presentata nell'articolo [3].

L'operatore unitario relativo al termine di hopping, rappresentato sulla coppia di qubits (k, k+1) dalla matrice:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(t\theta) & i\sin(t\theta) & 0 \\ 0 & i\sin(t\theta) & \cos(t\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

è implementabile attraverso il circuito rappresentato in Figura 5.3:



**Figura 5.3:** Implementazione gate di hopping costruita in [3]

Che definisce un Gate a 2 qubits  $H(\theta)$  di profondità pari a 5 gates

Essendo la componente di hopping agente solo tra stati caratterizzati dallo stesso valore di

### 5.3 Efficient Hamiltonian Variational Ansatz per il modello di Hubbard

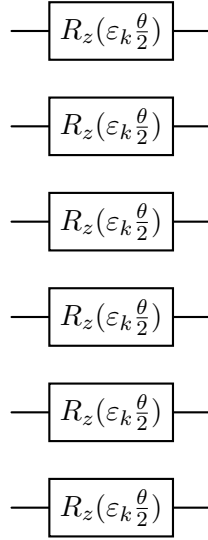
---

spin, i gate a 2 qubit che rappresentano la loro azione nel circuito non agiranno su qubit appartenenti a settori di spin diversi.

Per quanto riguarda l'operatore relativo alla componente di energia On-Site:

$$e^{-i\theta h_\varepsilon} = \prod_k e^{-i\varepsilon_k \frac{\theta}{2}} e^{i\varepsilon_k \frac{\theta}{2} Z_k}$$

Questo è rappresentabile come un layer di rotazioni single-qubit rispetto all'asse z di angolo pari a  $\varepsilon_k \frac{\theta}{2}$ .



**Figura 5.4:** Implementazione circuitale del termine di energia On-Site come layer di rotazioni generate dall'operatore di Pauli Z nel caso di reticolo a 3 siti atomici

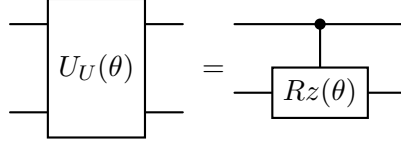
Questo crea una fase globale davanti allo stato ed è quindi possibile ometterlo nella costruzione dell'ansatz.

Infine l'operatore dovuto alla componente di repulsione On-Site:

$$e^{-i\theta h_U} = e^{-i\frac{U}{4}\theta \sum_{k=1}^M [1_k 1_{k+M} - Z_k - Z_{k+M} - Z_k Z_{k+M}]}$$

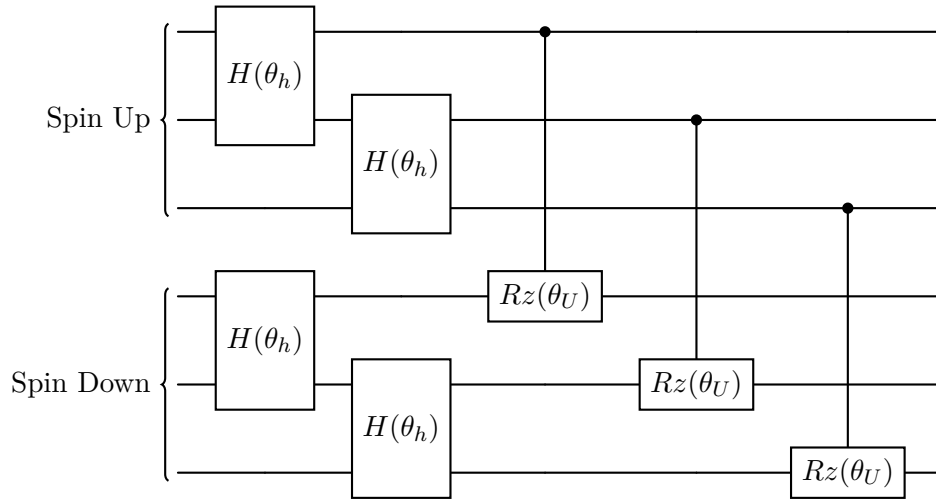
È implementato attraverso il gate controlled-rotation  $\text{CRz} = 1 \otimes |0\rangle\langle 0| + R_Z(\theta) \otimes |1\rangle\langle 1|$ :





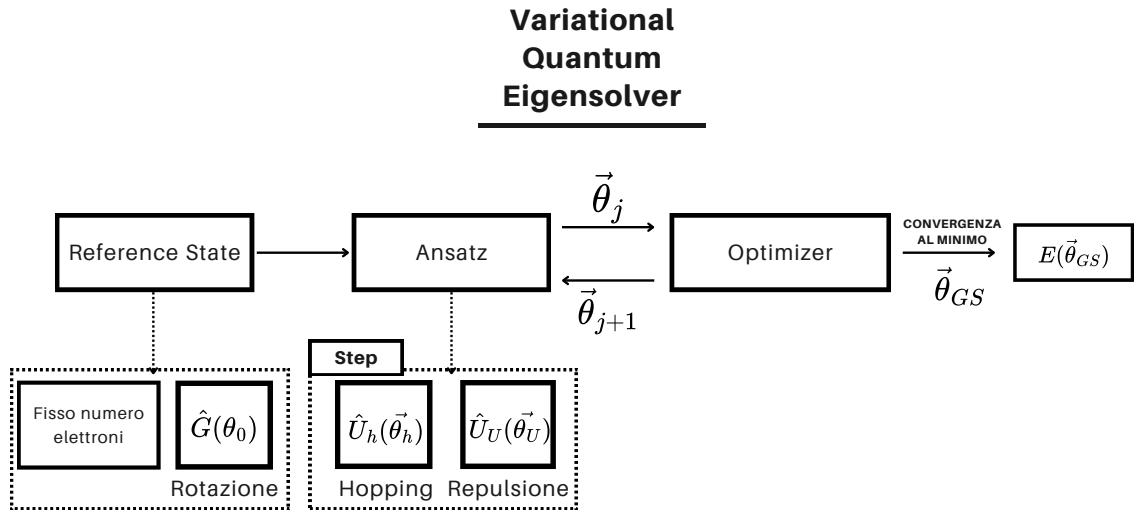
**Figura 5.5:** Implementazione gate di Repulsione On-Site

Seguendo questa implementazione per ogni step componente il circuito, otteniamo una profondità, ripetuta  $S$  volte, di 14 layer. Al crescere del numero di step aumenta la dimensione dello spazio degli stati considerati, e di conseguenza aumentano le probabilità di ottenere un'approssimazione migliore ma anche i costi per implementare il circuito di costruzione dell'ansatz.



**Figura 5.6:** Esempio di circuito per la costruzione dell'ansatz ad uno step nel caso di un reticolo a 3 siti

### 5.3 Efficient Hamiltonian Variational Ansatz per il modello di Hubbard



**Figura 5.7:** Diagramma che descrive il funzionamento dell'algoritmo VQE per un reticolo monodimensionale descritto dal modello di Hubbard

## Capitolo 6

# Risultati delle simulazioni del modello di Hubbard su emulatore di un computer quantistico

In questo Capitolo sono esposti i risultati ottenuti implementando l'algoritmo VQE attraverso diversi ottimizzatori e il metodo di costruzione dell'ansatz EHVA[8]. I risultati sono confrontati con valori ottenuti attraverso metodi analitici.

Qiskit offre la possibilità di ottenere, per sistemi sufficientemente piccoli, lo spettro esatto di un operatore di seconda quantizzazione: ciò avviene mediante un algoritmo di tipo classico, chiamato NumPyEigenSolver, che diagonalizza l'operatore e ne restituisce autovalori e autovettori, al variare del numero di elettroni di valenza.

Questo strumento è stato utilizzato per avere un confronto rispetto ai risultati ottenuti da VQE, sia per quanto riguarda gli autovalori che per gli autostati.

Nel seguito si mostrano i risultati ottenuti per l'energia dello stato fondamentale stimando la sua precisione rispetto ai risultati classici ricorrendo a diversi ottimizzatori.

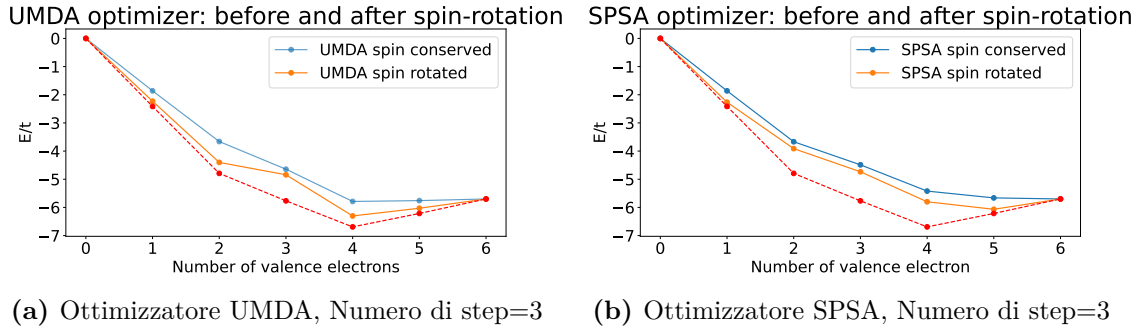
Le descrizioni degli ottimizzatori utilizzati e delle loro specifiche di implementazioni sono state rielaborate dalla documentazione relativa al framework Qiskit [6]

I valori sono riportati in termini del rapporto tra il valore di aspettazione dell'energia e il termine di hopping  $E/t$ , in quanto quest'ultimo nel modello di Hubbard determina l'ordine di grandezza degli autovalori.

## 6.1 Necessità di accoppiamento tra settori di spin

### 6.1 Necessità di accoppiamento tra settori di spin

Inizialmente si è analizzato il sistema senza considerare la possibilità di effettuare rotazioni tra i diversi settori di spin – da non confondersi con le rotazioni di singolo qubit – nella creazione del reference state. Questo vincolo ha ristretto notevolmente lo spazio degli stati su cui valutare l'aspettazione dell'Hamiltoniana. In Figura 6.1 si può vedere un confronto tra i risultati ottenuti da diversi ottimizzatori locali prima e dopo aver implementato le rotazioni tra settori di spin nel caso del reticolo a 3 siti come mostrato nel circuito in Figura 5.2.



**Figura 6.1:** Confronto dei risultati ottenuti da due ottimizzatori locali attraverso ansatz con o senza rotazioni tra i qubits di spin up e quelli di spin down con ottimizzatori UMDA e SLSQP

Distance from exact ground state's eigenvalue				
	UMDA		SLSQP	
Ne	spin conserved	spin rotated	spin conserved	spin rotated
1	0.552098708	0.137097801	0.552097602	0.134746143
2	1.130500291	0.391489739	1.123480627	0.378394278
3	1.123323060	0.925591047	1.111636409	0.924158682
4	0.903211739	0.384848105	0.898793512	0.378394310
5	0.455899998	0.184660534	0.455867585	0.134762735

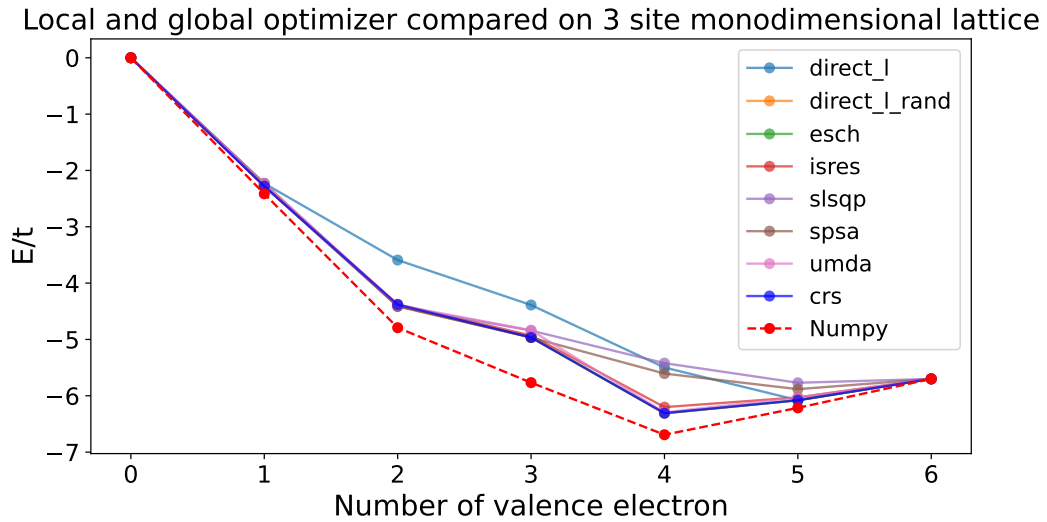
**Tabella 6.1:** Distanze tra la stima dello stato fondamentale e il valore esatto calcolato tramite NumPyEigenSolver prima e dopo l'implementazione della rotazione tra settori di spin

Nella Tabella 6.1 sono riportati i valori ottenuti per le stime dell'energia nello stato fondamentale al variare del numero di elettroni di valenza, prima e dopo l'implementazione della rotazione tra settori di spin. Il valore relativo al numero di elettroni di valenza nullo rappresenta il caso di reticolo con banda vuota.

## 6.2 Confronto dei risultati al variare dell'ottimizzatore classico

Dopo aver implementato la possibilità di effettuare rotazioni tra i settori di spin, sono state effettuate simulazioni utilizzando sia ottimizzatori locali che globali.

In questa sezione sono presentati i risultati ottenuti implementando, nella preparazione dello stato iniziale, un layer di rotazioni tra ogni coppia di qubits caratterizzate tutte dallo stesso angolo. Così facendo lo spazio naturale del vettore dei parametri coincide con lo spazio  $\mathbf{R}^{2S+1}$  dove S è il numero di step utilizzati per la costruzione dell'ansatz.



**Figura 6.2:** Confronto tra diversi ottimizzatori

In particolare tra gli ottimizzatori locali sono stati utilizzati Sequential Least Squares Programming optimizer (SLSQP) e Continuous Univariate Marginal Distribution Algorithm (UMDA) al variare del passo con cui esploravano lo spazio dei parametri ma senza registrare sostanziali miglioramenti rispetto a quello definito di default.

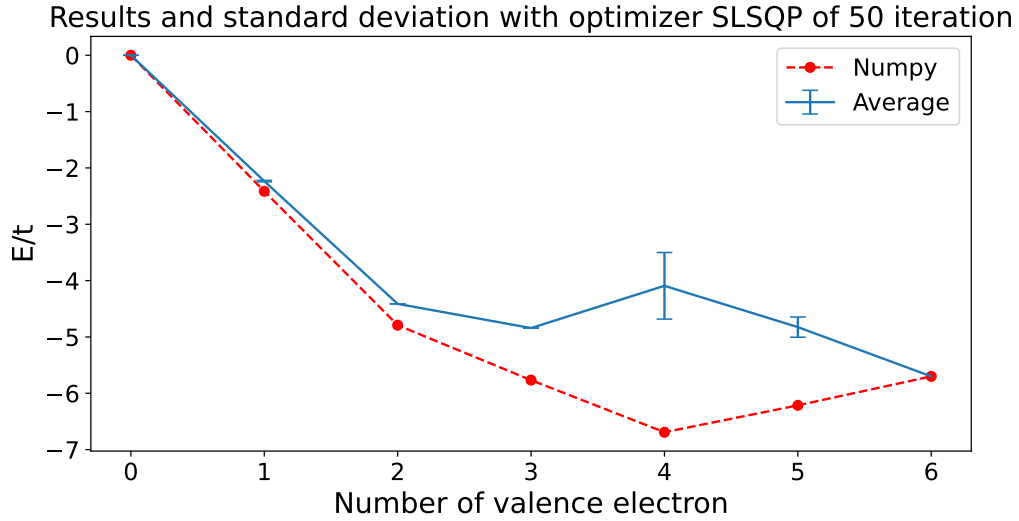
## 6.2 Confronto dei risultati al variare dell'ottimizzatore classico

---

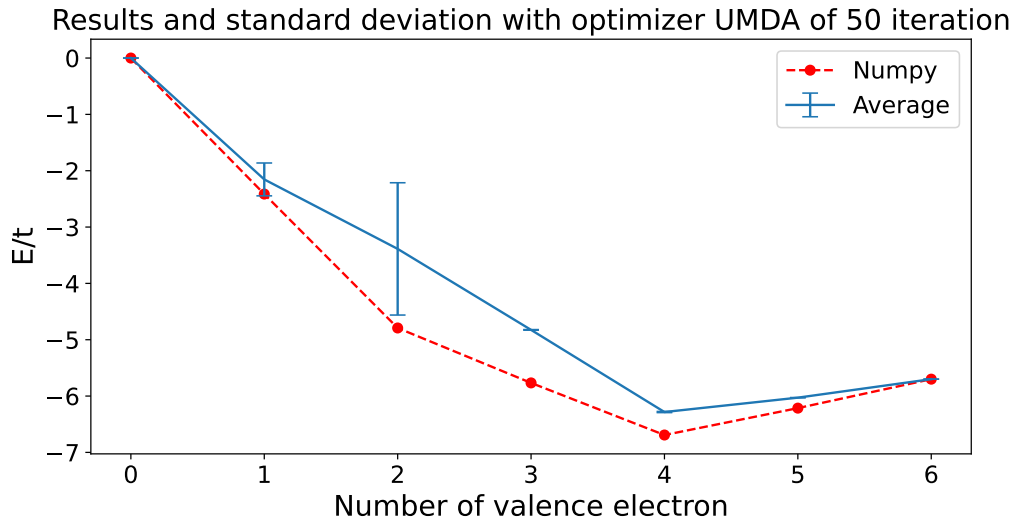
SLSQP è un algoritmo che ad ogni iterazione risolve sottoproblemi di ottimizzazione su un modello quadratico che localmente rappresenta il problema di ottimizzazione originale [5].

UMDA è un algoritmo di ricerca del minimo basato su una distribuzione normale univariata che effettua ad ogni iterazione una selezione dei punti più performanti da utilizzare all'iterazione successiva [6].

La scelta di questi due ottimizzatori è stata dettata dal fatto che presentassero le migliori performance, sia in termini di tempo di esecuzione che di risultati numerici, tra gli ottimizzatori locali.



**Figura 6.3:** Stime dell'energia dello stato fondamentale ottenute con ottimizzatore SLSQP in 50 valutazioni per reticolo a 3 siti e ansatz costituito da 3 steps



**Figura 6.4:** Stime dell'energia dello stato fondamentale ottenute con ottimizzatore UMDA in 50 valutazioni per reticolo a 3 siti e ansatz costituito da 3 steps

## 6.2 Confronto dei risultati al variare dell'ottimizzatore classico

Eigenvalues SLSQP optimizer			
Ne	eigenvalue	standard dev	distance real value
0	$-3.418932 \times 10^{-15}$	$2.691388 \times 10^{-15}$	$-3.467504 \times 10^{-15}$
1	-2.234343	$1.327546 \times 10^{-2}$	$1.798704 \times 10^{-1}$
2	-4.413104	$2.057690 \times 10^{-5}$	$3.784002 \times 10^{-1}$
3	-4.842329	$4.631066 \times 10^{-5}$	$9.241807 \times 10^{-1}$
4	-4.092406	$5.908681 \times 10^{-1}$	2.599098
5	-4.825589	$1.791231 \times 10^{-1}$	1.388625
6	-5.700000	N.A.	$7.993606 \times 10^{-15}$

**Tabella 6.2:** Stime dello stato fondamentale ottenute per il reticolo a 3 siti con ottimizzatore SLSQP

Eigenvalues UMDA optimizer			
Ne	eigenvalue	standard dev	distance real value
0	$-1.078831 \times 10^{-14}$	$6.030590 \times 10^{-16}$	$-1.083689 \times 10^{-14}$
1	-2.154467	$2.916718 \times 10^{-1}$	$2.597470 \times 10^{-1}$
2	-3.387714	1.174236	1.403790
3	-4.825358	$5.494042 \times 10^{-3}$	$9.411522 \times 10^{-1}$
4	-6.283493	$1.011352 \times 10^{-2}$	$4.080114 \times 10^{-1}$
5	-6.028354	$5.584493 \times 10^{-4}$	$1.858592 \times 10^{-1}$
6	-5.700000	$1.685874 \times 10^{-7}$	$-8.881784 \times 10^{-15}$

**Tabella 6.3:** Stime dell'energia dello stato fondamentale ottenute per il reticolo a 3 siti con ottimizzatore UMDA

Come mostrato in Figura 6.3 e 6.4, gli ottimizzatori locali, anche se caratterizzati per la maggior parte da tempi di convergenza più veloci, sono fortemente influenzati dalla presenza di minimi locali.

Nelle Tabelle 6.2 e 6.3 sono riportati i valori delle stime ottenute per l'energia dello stato fondamentale al variare del numero di elettroni di valenza, rispettivamente per l'ottimizzatore SLSQP e UMDA. A queste sono associati i valori ottenuti per la deviazione standard e la distanza dai valori esatti delle energie per lo stato fondamentale ottenuti tramite l'algoritmo per computer classico NumPyEigenSolver.

In particolare SLSQP fornisce stime molto distanti dai valori reali ed entrambi sono inficiati da una grande deviazione standard su alcuni degli autovalori ottenuti, non garantendo quindi al termine di un'esecuzione singola il raggiungimento di una stima realistica dello stato fondamentale.

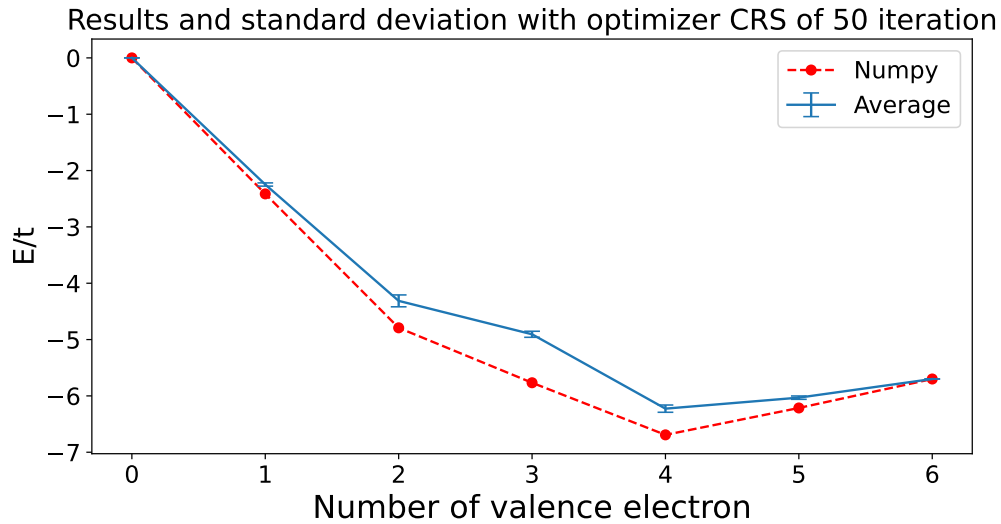


Si è deciso quindi di utilizzare ottimizzatori globali.

Nella Sezioni successive vengono confrontate le performance ottenute per diversi ottimizzatori globali, applicati all'algoritmo VQE con ansatz composto da 3 step, nel caso del reticolo monodimensionale a 3 siti atomici.

### 6.2.1 Controlled Random Search (CRS) with local mutation optimizer

CRS è un ottimizzatore globale che partendo da una set randomico di punti li fa evolvere randomicamente nello spazio per valutarne la funzione di costo.



**Figura 6.5:** Stime dell'energia dello stato fondamentale ottenute con ottimizzatore CRS in 50 valutazioni su reticolo a 3 siti con ansatz di 3 steps

## 6.2 Confronto dei risultati al variare dell'ottimizzatore classico

---

Eigenvalues CRS Optimizer			
Ne	eigenvalue	standard dev	distance real value
0	$-1.122727 \times 10^{-14}$	$4.076433 \times 10^{-16}$	$-1.127584 \times 10^{-14}$
1	-2.247673	$2.866536 \times 10^{-2}$	$1.665402 \times 10^{-1}$
2	-4.313029	$1.054271 \times 10^{-1}$	$4.784749 \times 10^{-1}$
3	-4.906121	$5.370022 \times 10^{-2}$	$8.603892 \times 10^{-1}$
4	-6.226259	$6.441822 \times 10^{-2}$	$4.652450 \times 10^{-1}$
5	-6.031013	$2.966050 \times 10^{-2}$	$1.832003 \times 10^{-1}$
6	-5.700000	$2.064765 \times 10^{-7}$	$-1.154632 \times 10^{-14}$

**Tabella 6.4:** Stime dello stato fondamentale ottenute per il reticolo a 3 siti con ottimizzatore CRS in 50 valutazioni

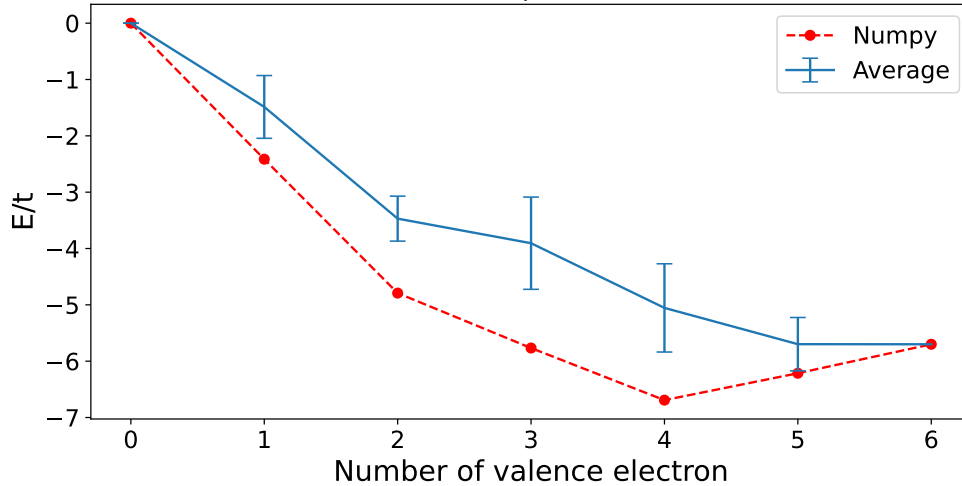
Dalla Tabella 6.8 si può osservare che la stima ottenuta attraverso CRS per lo stato fondamentale è caratterizzata da un errore statistico al di sotto dei 3.6 punti percentuali rispetto al valore medio, e da una distanza dal valore esatto inferiore ai 14.9 punti percentuali rispetto al valore esatto.

Il metodo CRS garantisce una convergenza pressochè costante alla stessa stima dello stato fondamentale, la cui distanza dai valori reali è quindi dovuta solo al metodo con cui sono stati caratterizzati gli stati dell'ansatz.

### 6.2.2 DIviding RECTangles Locally-biased optimizer (DIRECT-L)

DIviding RECTangles (DIRECT) è un ottimizzatore basato sulla divisione del dominio di ricerca in rettangoli progressivamente più piccoli ad ogni iterazione. La versione DIRECT-L è la versione 'localmente pesata' di DIRECT che rende l'algoritmo più influenzato rispetto alla ricerca locale, è quindi più efficiente per funzioni con pochi minimi locali. DIRECT-L-RAND è la variante basata su alcune randomizzazioni nella decisione vicino ad eventuali vincoli.

Results and standard deviation with optimizer DIRECT-L-RAND of 50 iteration



**Figura 6.6:** Stime dell'energia dello stato fondamentale ottenute con ottimizzatore DIRECT-L-RAND in 50 valutazioni su reticolo a 3 siti con ansatz di 3 steps

## 6.2 Confronto dei risultati al variare dell'ottimizzatore classico

---

Eigenvalues DIRECT-L-RAND Optimizer			
Ne	eigenvalue	standard dev	distance real value
0	$-1.008041 \times 10^{-14}$	$2.381269 \times 10^{-15}$	$-1.012898 \times 10^{-14}$
1	-1.486692	$5.569917 \times 10^{-1}$	$9.275211 \times 10^{-1}$
2	-3.469886	$3.990338 \times 10^{-1}$	1.321618
3	-3.905823	$8.189204 \times 10^{-1}$	1.860687
4	-5.054249	$7.832478 \times 10^{-1}$	1.637255
5	-5.698162	$4.735486 \times 10^{-1}$	$5.160511 \times 10^{-1}$
6	-5.700000	N.A.	$-3.019807 \times 10^{-14}$

**Tabella 6.5:** Stime dello stato fondamentale ottenute per il reticolo a 3 siti con ottimizzatore DIRECT-L-RAND in 50 valutazioni

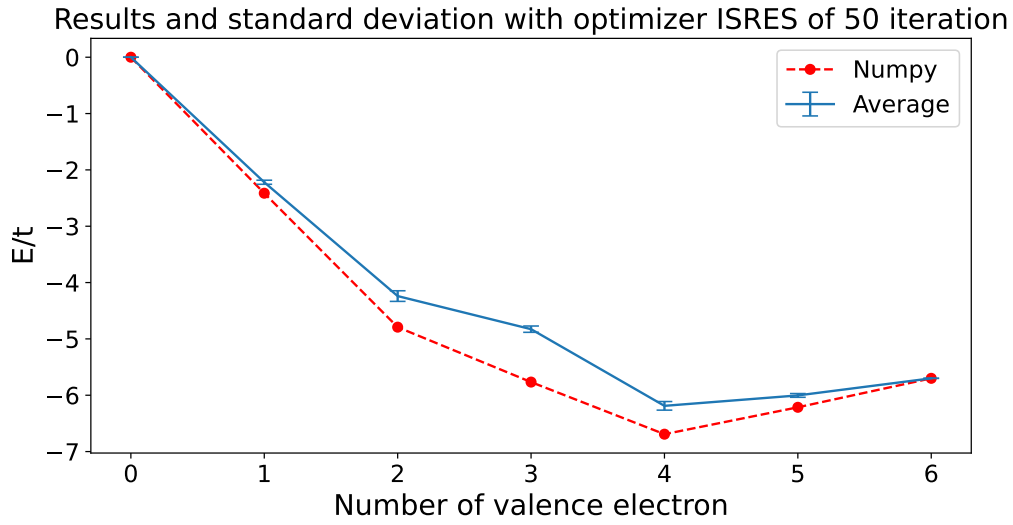
Le stime ottenute attraverso l'ottimizzatore DIRECT-L-RAND sono riportate in Tabella 6.5, sono presenti i valori della deviazione standard misurata su 50 iterazioni dell'algoritmo e la distanza tra i valori ottenuti e quelli esatti calcolati con l'algoritmo per computer classici NumPyEigenSolver.

La stima ottenuta attraverso DIRECT-L-RAND per lo stato fondamentale è caratterizzata da un errore statistico compreso tra gli 8 e i 37 punti percentuali rispetto al valore medio e una distanza dal valore esatto tra gli 8 e i 38 punti percentuali rispetto al valore reale.

Si può notare come l'esecuzione di DIR-L-RAND sia legata ad un errore particolarmente pesante sulla stima dello stato fondamentale, probabilmente a causa della presenza di minimi locali della funzione valore di aspettazione dell'energia nello spazio dei parametri. Questa sua caratteristica rende l'ottimizzatore DIRECT-L-RAND inadatto per essere utilizzato nell'algoritmo VQE.

### 6.2.3 Improved Stochastic Ranking Evolution Strategy optimizer (ISRES)

Improved Stochastic Ranking Evolution Strategy (ISRES) è un ottimizzatore che utilizza metodi euristici per evitare minimi locali. Utilizza una strategia evolutiva basata sulla combinazione di regole di variazione rispetto al punto e variazioni del differenziale.



**Figura 6.7:** Stime dell'energia dello stato fondamentale ottenute con ottimizzatore ISRES in 50 valutazioni su reticolo a 3 siti con ansatz di 3 steps

Eigenvalue ISRES Optimizer			
Ne	eigenvalue	standard dev	distance real value
0	$-1.147152 \times 10^{-14}$	$9.658689 \times 10^{-16}$	$-1.152009 \times 10^{-14}$
1	-2.219316	$3.603360 \times 10^{-2}$	$1.948980 \times 10^{-1}$
2	-4.239827	$9.423324 \times 10^{-2}$	$5.516773 \times 10^{-1}$
3	-4.827210	$5.607362 \times 10^{-2}$	$9.393005 \times 10^{-1}$
4	-6.188287	$7.636872 \times 10^{-2}$	$5.032176 \times 10^{-1}$
5	-6.003406	$3.325434 \times 10^{-2}$	$2.108073 \times 10^{-1}$
6	-5.700000	$1.192093 \times 10^{-7}$	$-1.154632 \times 10^{-14}$

**Tabella 6.6:** Stime dello stato fondamentale ottenute per il reticolo a 3 siti con ottimizzatore ISRES in 50 valutazioni

## 6.2 Confronto dei risultati al variare dell'ottimizzatore classico

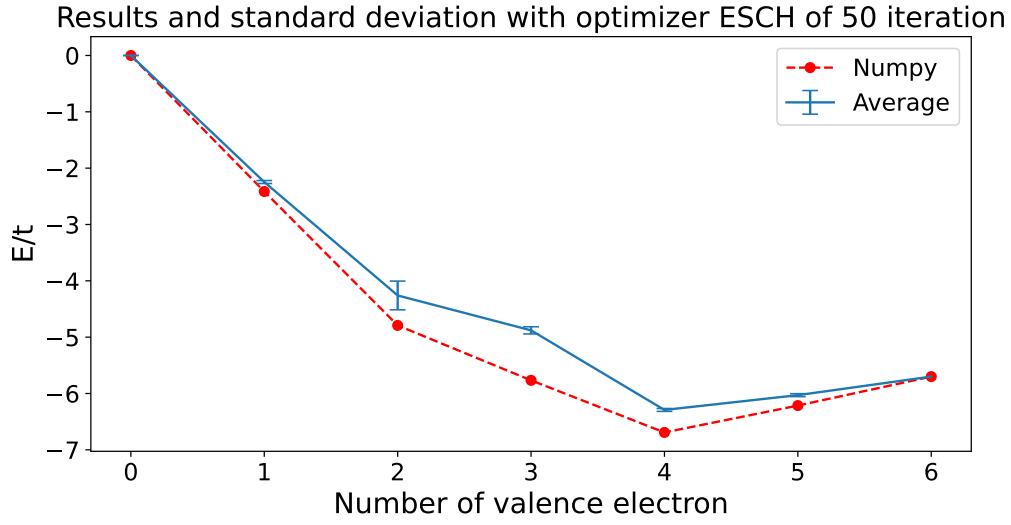
---

Le stime ottenute attraverso l'ottimizzatore ISRES sono riportate in Tabella 6.6, sono presenti i valori della deviazione standard misurata su 50 iterazioni dell'algoritmo e la distanza tra i valori ottenuti e quelli esatti calcolati con l'algoritmo per computer classici NumPyEigenSolver.

L'ottimizzatore ISRES fornisce delle stime dello stato fondamentale al variare del numero di elettroni di valenza con deviazione standard pari al più a 8.4 punti percentuali rispetto al valore medio ottenuto ed una distanza dal valore esatto pari al più a 16 punti percentuali rispetto al valore esatto.

#### 6.2.4 ESCH

ESCH è un algoritmo evolutivo per ottimizzazione globale implementato da Carlos Henrique da Silva Santos [9]



**Figura 6.8:** Stime dell'energia dello stato fondamentale ottenute con ottimizzatore ESCH in 50 valutazioni su reticolo a 3 siti con ansatz di 3 steps

Eigenvalue ESCH Optimizer			
Ne	eigenvalue	standard dev	distance real value
0	$-1.490572 \times 10^{-14}$	$1.362403 \times 10^{-15}$	$-1.495429 \times 10^{-14}$
1	-2.246027	$2.650217 \times 10^{-2}$	$1.681861 \times 10^{-1}$
2	-4.259462	$2.546750 \times 10^{-1}$	$5.320417 \times 10^{-1}$
3	-4.879136	$6.354282 \times 10^{-2}$	$8.873736 \times 10^{-1}$
4	-6.291909	$2.547958 \times 10^{-2}$	$3.995948 \times 10^{-1}$
5	-6.029440	$2.636530 \times 10^{-2}$	$1.847733 \times 10^{-1}$
6	-5.700000	N.A.	$-2.930989 \times 10^{-14}$

**Tabella 6.7:** Stime dello stato fondamentale ottenute per il reticolo a 3 siti con ottimizzatore ESCH in 50 valutazioni

Le stime ottenute attraverso l'ottimizzatore ESCH sono riportate in Tabella 6.7, sono

## 6.2 Confronto dei risultati al variare dell'ottimizzatore classico

---

presenti i valori della deviazione standard misurata su 50 iterazioni dell'algoritmo e la distanza tra i valori ottenuti e quelli esatti calcolati con l'algoritmo per computer classici NumPyEigenSolver.

L'ottimizzatore ESCH fornisce delle stime dello stato fondamentale al variare del numero di elettroni di valenza con deviazione standard pari al più a 9 punti percentuali rispetto al valore medio ottenuto ed una distanza dal valore esatto pari al più a 15.4 punti percentuali rispetto al valore esatto.

### 6.2.5 Scelta dell'ottimizzatore globale

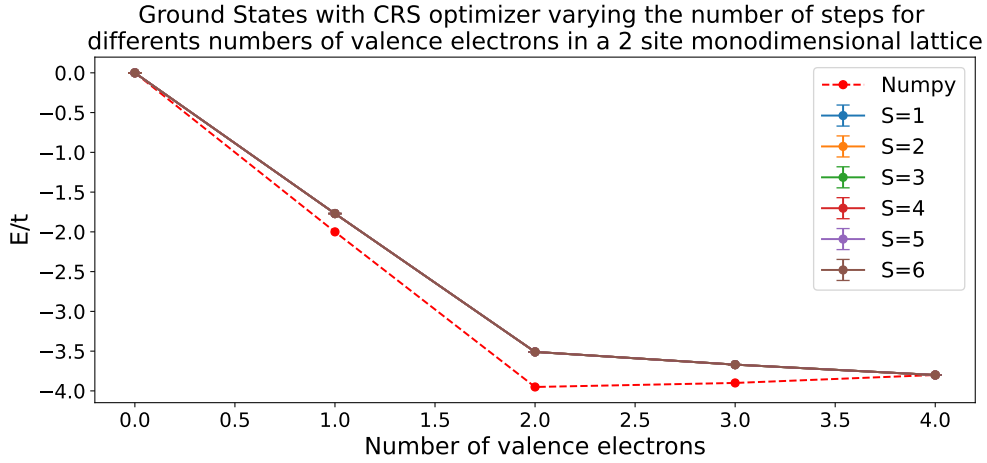
Fatta eccezione di DIRECT-L-RAND la distanza dal valore reale ottenuta con i diversi ottimizzatori rimane pressochè invariata, lasciando intendere che in generale questa sia dovuta alla costruzione degli stati su cui avviene poi la valutazione, piuttosto che al particolare ottimizzatore. Questo fatto vale sia per quanto riguarda i valori medi che per quelli ottimali. Non esiste quindi un ottimizzatore che ottenga un significativo successo per quanto concerne l'avvicinarsi al valore esatto dello stato fondamentale.

Tutti gli ottimizzatori presentano risultati particolarmente precisi in corrispondenza dei valori corrispondenti al caso di banda piena e banda vuota. Questo è dovuto al fatto che in corrispondenza di tali configurazione gli stati del sistema di  $N$  qubits sono rappresentati dai vettori  $|0\rangle^{\otimes N}$  nel caso di banda vuota e da  $|1\rangle^{\otimes N}$  nel caso di banda piena. Per questo motivo tutte le misure effettuate sui qubits che costituiscono il circuito restituiscono lo stato  $|0\rangle$  o  $|1\rangle$  con probabilità pari a 1 portando sempre agli stessi risultati della misura di energia.

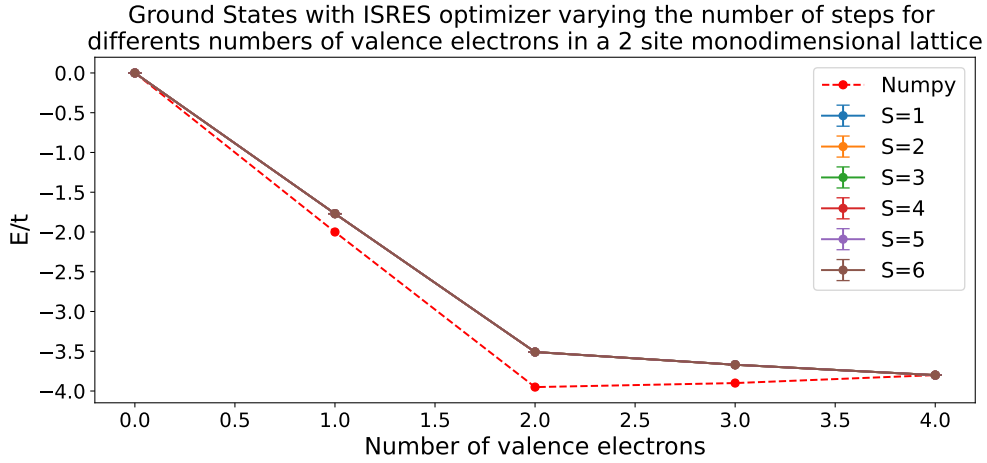
Si è deciso quindi di effettuare un'analisi al variare dei siti reticolari utilizzando gli ottimizzatori CRS e ISRES in quanto presentano una convergenza più precisa (con minore deviazione standard) alla stima dello stato fondamentale.



### 6.3 2 siti reticolari



**Figura 6.9:** Stime dell'energia dello stato fondamentale al variare del numero di steps ottenute con ottimizzatore CRS in 50 valutazioni nel caso del reticolo a 2 siti atomici



**Figura 6.10:** Stime dell'energia dello stato fondamentale al variare del numero di steps ottenute con ottimizzatore 50 valutazioni nel caso del reticolo a 2 siti atomici

Utilizzando entrambi gli ottimizzatori le valutazioni dello stato fondamentale per il reticolo monodimensionale a 2 siti presentano valori molto simili senza registrare sostanziali cambiamenti al crescere del numero di steps.

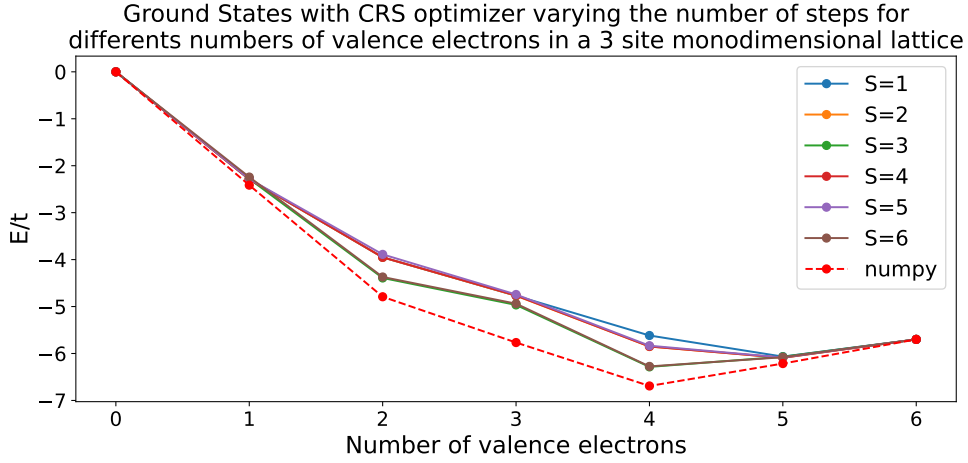
### 6.3 2 siti reticolari

---

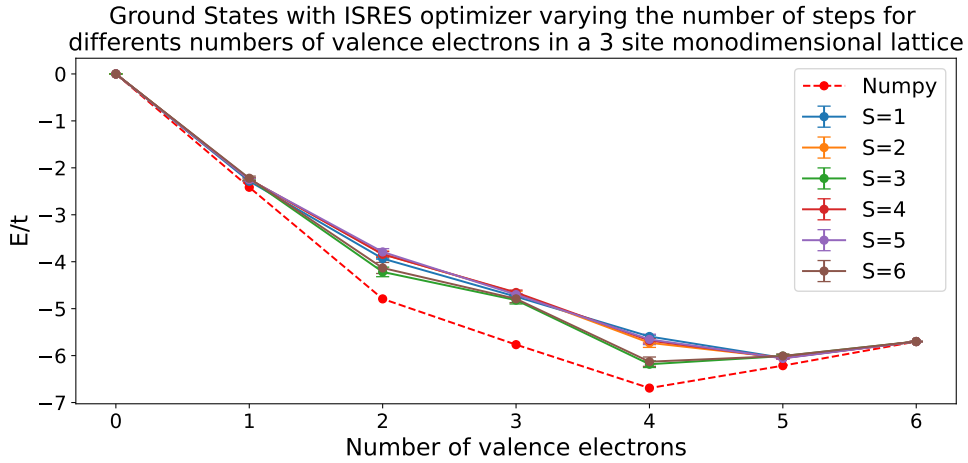
Average distance from real value ISRES		
Number of steps	distance	standard dev
1	0.30063098935004606	0.0993372328883049
2	0.300625994122421	0.09936729674473137
3	0.3007127884864471	0.09936455991431739
4	0.3008117645118699	0.09938287557869421
5	0.3007128482361428	0.0992724821452743
6	0.30074146115658945	0.09927430083084329
Average distance from real value CRS		
Number of steps	distance	standard dev
1	0.30035528653469984	0.09921077338664272
2	0.3003990297768047	0.09922044739613897
3	0.30055326261200666	0.09929665889632627
4	0.3006103542656366	0.09937995711159099
5	0.3005907896699728	0.09921817520507518
6	0.30060967442448466	0.09930971558640722

**Tabella 6.8:** Distanza media dai valori reali al variare del numero di step senza considerare i casi a banda vuota o piena per 50 valutazioni nel caso del reticolo a 2 siti atomici

## 6.4 3 siti reticolari



**Figura 6.11:** Stime dell'energia dello stato fondamentale al variare del numero di steps ottenute con ottimizzatore CRS in 50 valutazioni nel caso del reticolo a 3 siti atomici



**Figura 6.12:** Stime dell'energia dello stato fondamentale al variare del numero di steps ottenute con ottimizzatore ISRES in 50 valutazioni nel caso del reticolo a 3 siti atomici

Nel caso del reticolo monodimensionale a 3 siti atomici i risultati migliori risultano essere caratterizzati dall'implementazione dell'ansatz a 3 e 6 step mentre prestazioni peggiori caratterizzano le implementazioni effettuate con 2, 4, e 5 steps.

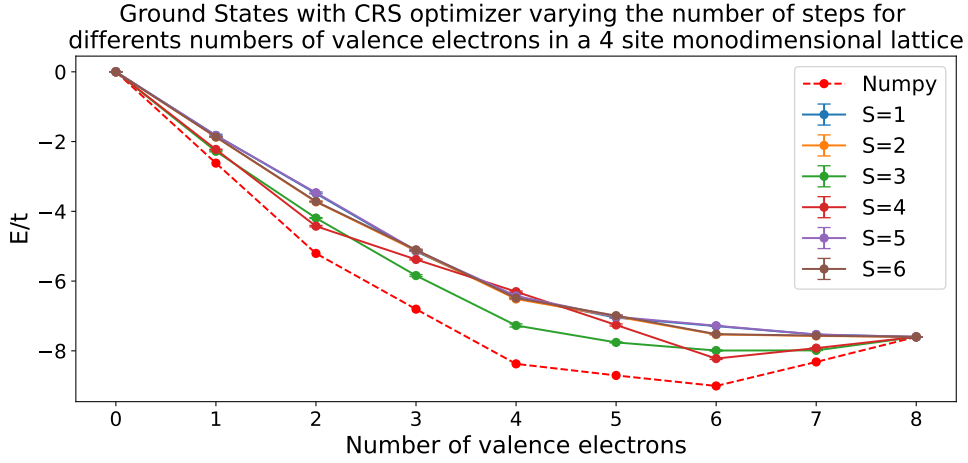
## 6.4 3 siti reticolari

---

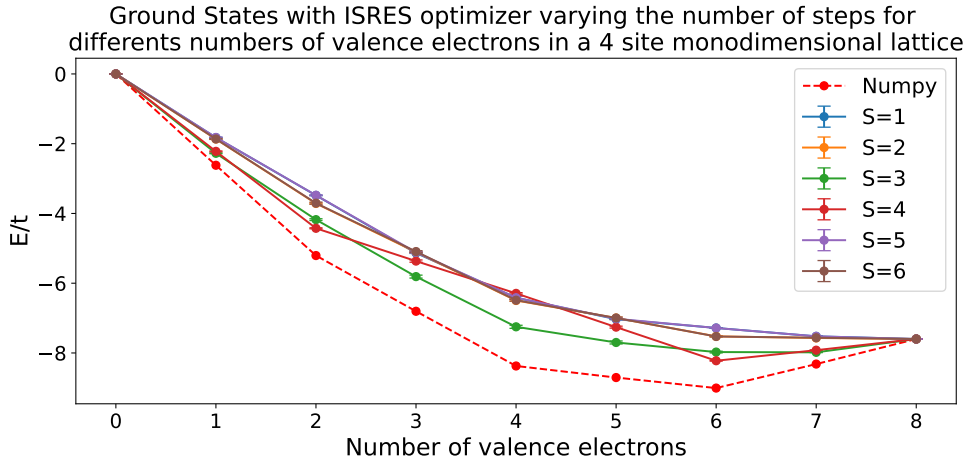
Average distance from real value ISRES		
Number of steps	distance	standard dev
1	0.6551147638254572	0.42161893193373345
2	0.6712495017072669	0.4134943350124549
3	0.4824692109482084	0.2817310221344912
4	0.6803722966471546	0.42454480819117985
5	0.6843838650673907	0.42919912202597854
6	0.5189904440318925	0.2948279283819353
Average distance from real value CRS		
Number of steps	distance	standard dev
1	0.6365215963577004	0.41623910983436124
2	0.5844856659118725	0.3814460137957513
3	0.37740798820505894	0.2423957995590219
4	0.5865990300791388	0.3818030343922081
5	0.608809909147255	0.394629838366105
6	0.39454285172637943	0.24715565838696466

**Tabella 6.9:** distanza media dai valori reali al variare del numero di step senza considerare i casi a banda vuota o piena per 10 valutazioni nel caso del reticolo a 3 siti atomici

## 6.5 4 siti reticolari



**Figura 6.13:** Stime dell'energia dello stato fondamentale al variare del numero di steps ottenute con ottimizzatore CRS in 10 valutazioni



**Figura 6.14:** Stime dell'energia dello stato fondamentale al variare del numero di steps ottenute con ottimizzatore ISRES in 10 valutazioni nel caso del reticolo a 4 siti atomici

In Figura 6.13 e 6.14 si può vedere come globalmente le approssimazioni migliori siano fornite dalla costruzione dell'ansatz a 3 step , ad eccezione del caso a 4 elettroni di valenza per cui i valori ottenuti con ansatz a 4 step forniscono una maggiore precisione.

## 6.6 Confronto al variare del numero di siti

---

Average distance from real value ISRES		
Number of steps	distance	standard dev
1	1.4768287940230092	0.4412522817615956
2	1.3938524259314735	0.42630732936120097
3	0.8376367697832517	0.3174302219081865
4	1.0466551076035093	0.5812136387852521
5	1.4823418348817712	0.44245345670351494
6	1.3993507522199806	0.42782607843518566
Average distance from real value CRS		
Number of steps	distance	standard dev
1	1.460987776471137	0.43114307732038987
2	1.3811782660974115	0.4199170027112885
3	0.8159834972354718	0.3074459014717544
4	1.0423981367872495	0.5779839174542317
5	1.4803833718028492	0.4445666090001751
6	1.397862058960549	0.42769872228898914

**Tabella 6.10:** distanza media dai valori reali al variare del numero di step senza considerare i casi a banda vuota o piena per 10 valutazioni nel caso del reticolo a 4 siti atomici

## 6.6 Confronto al variare del numero di siti

In conclusione, implementando nella costruzione dello stato iniziale il layer di rotazioni tra qubits caratterizzato da un unico valore angolare, le stime migliori per lo stato fondamentale dell'Hamiltoniana di Hubbard nel caso di reticoli monodimensionali sono state ottenute utilizzando l'ottimizzatore CRS. Questo, in quanto caratterizzato da una bassa deviazione standard e convergenza alle stime migliori rispetto ai valori esatti dell'energia, garantisce una maggiore precisione nel raggiungimento di una stima affidabile dello stato fondamentale.

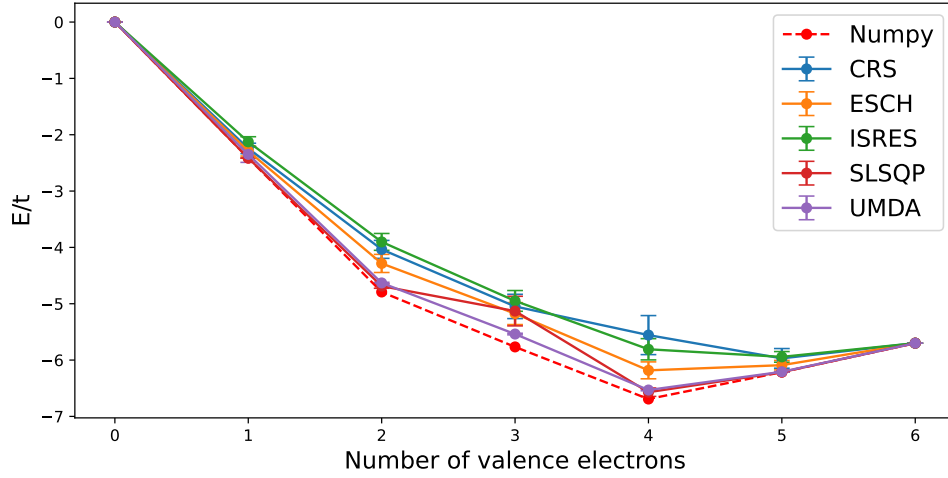
## 6.7 Implementazione dello stato iniziale attraverso rotazioni di angoli arbitrari

Nel caso in cui il layer di rotazioni con cui si costruisce lo stato iniziale venga composto permettendo ad ogni rotazione tra coppia di qubits di ruotare di un angolo indipendente-

mente dalle altre, si osserva un comportamento in antitesi con quanto osservato nel caso di singolo angolo di rotazione.

In Figura 6.15 è mostrato il comportamento di diversi ottimizzatori implementati nel caso appena descritto per il reticolo a 3 siti atomici ed ansatz composto da 3 steps.

**Local and global optimizer compared on 3 site monodimensional lattice with reference state composed of arbitrary rotations**



**Figura 6.15:** Ottimizzatori locali e globali a confronto per 50 valutazioni nel caso di reticolo monodimensionale a 3 siti con ansatz composto da 3 steps

Gli ottimizzatori locali SLSQP e UMDA restituiscono adesso stime più precise mentre gli ottimizzatori globali sono caratterizzati da una maggiore distanza rispetto al valore esatto.

Average distance from exact values		
Optimizer	Average distance	standard dev
CRS	0.4321053512687542	0.40671771500766757
ESCH	0.2650490648818115	0.24050969890792687
ISRES	0.449763513678607	0.3738319401678906
SLSQP	0.12319625323828513	0.21521765189769523
UMDA	0.08871181215633953	0.08686534325008291

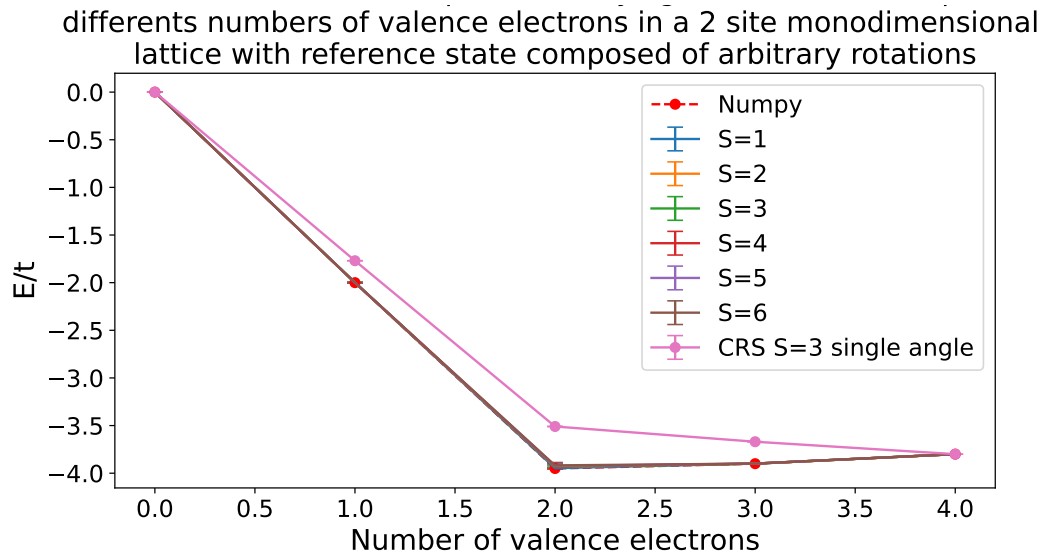
**Tabella 6.11:** Distanza media calcolata su 50 valutazioni dei diversi ottimizzatori per ansatz costruito con 3 steps

Nelle sezioni successive viene analizzato il comportamento dell'ottimizzatore CRS nei casi

## 6.7 Implementazione dello stato iniziale attraverso rotazioni di angoli arbitrari

di reticolo monodimensionale a 2 e 3 siti atomici, confrontandolo con il caso analizzato in precedenza per singolo angolo di rotazione iniziale.

### 6.7.1 2 siti reticolari



**Figura 6.16:** Stime dell'energia dello stato fondamentale al variare del numero di steps ottenute con ottimizzatore CRS in 10 valutazioni per reticolo a 2 siti atomici e stato iniziale costruito con rotazioni arbitrarie

Si può notare da Figura 6.16 che le stime per il caso di reticolo a 2 siti atomici sono coincidenti con i valori calcolati esattamente da NumPyEigensolver. I risultati ottenuti nel caso di singolo angolo di rotazione risultano essere più distanti per qualsiasi valore del numero di elettroni di valenza, esclusi i casi di banda piena o vuota.

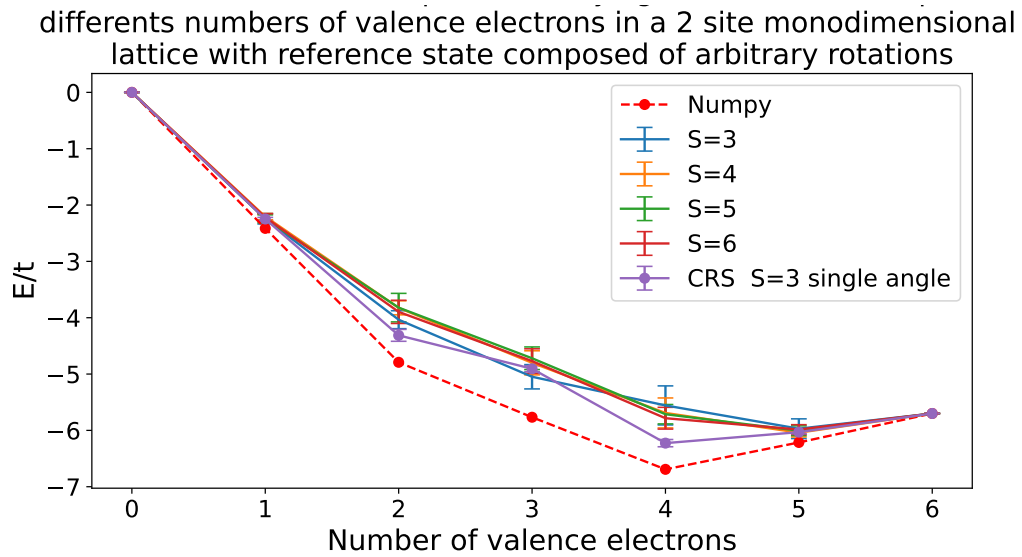
In Tabella 6.12 sono riportati, come esempio, i valori delle stime dello stato fondamentale al variare del numero di elettroni di valenza per l'ansatz composto da 1 step con le relative deviazioni standard e le distanze dal valore reale.



Eigenvalues CRS optimizer multi-rotation			
Ne	eigenvalue	standard dev	distance real value
0	$-2.445266 \times 10^{-15}$	$2.162264 \times 10^{-16}$	$-2.403633 \times 10^{-15}$
1	-1.998954	$2.036492 \times 10^{-3}$	$1.045859 \times 10^{-3}$
2	-3.946942	$6.327902 \times 10^{-3}$	$3.682415 \times 10^{-3}$
3	-3.899279	$2.071139 \times 10^{-3}$	$7.205429 \times 10^{-4}$
4	-3.800000	N.A	$-3.996803 \times 10^{-15}$

**Tabella 6.12:** Valori delle energie nello stato fondamentale per il reticolo a 2 siti con stato iniziale composto da rotazioni arbitrarie

### 6.7.2 3 siti reticolari



**Figura 6.17:** Stime dell'energia dello stato fondamentale al variare del numero di steps ottenute con ottimizzatore CRS in 10 valutazioni per reticolo a 3 siti atomici e stato iniziale costruito con rotazioni arbitrarie

Per quanto riguarda il caso a 3 siti reticolari l'ottimizzatore CRS è più performante nell'implementazione ad angolo singolo piuttosto che in quella a rotazioni arbitrarie. Non sono disponibili i valori relativi alla costruzione con numero di steps 1 e 2.

## 6.7 Implementazione dello stato iniziale attraverso rotazioni di angoli arbitrari

In Tabella 6.13 sono riportati, come esempio, i valori delle stime dello stato fondamentale al variare del numero di elettroni di valenza per l'ansatz composto da 3 steps con le relative deviazioni standard e le distanze dal valore reale. In Tabella 6.14 sono riportati i valori della distanza media per le diverse implementazioni dell'ansatz.

Eigenvalues CRS optimizer multi-rotation			
Ne	eigenvalue	standard dev	distance real value
0	$-1.003711 \times 10^{-14}$	$4.699967 \times 10^{-16}$	$-1.008568 \times 10^{-14}$
1	-2.243402	$9.265096 \times 10^{-2}$	$1.708117 \times 10^{-1}$
2	-4.035204	$1.592683 \times 10^{-1}$	$7.563005 \times 10^{-1}$
3	-5.048027	$2.158435 \times 10^{-1}$	$7.184826 \times 10^{-1}$
4	-5.556658	$3.465674 \times 10^{-1}$	1.134846
5	-5.969917	$1.746388 \times 10^{-1}$	$2.442968 \times 10^{-1}$
6	-5.700000	0.000000	$-1.243450 \times 10^{-14}$

**Tabella 6.13:** Valori delle energie nello stato fondamentale per il reticolo a 3 siti con stato iniziale composto da rotazioni arbitrari

Average distance from real value CRS		
S	average distance	standard dev
3	0.4321053512687542	0.40671771500766757
4	0.47374875923634285	0.44285513405007587
5	0.48206363911223143	0.4536971852388223
6	0.460766610118279	0.4182921556665032

**Tabella 6.14:** valori della distanza media al variare del numero di steps costituenti l'ansatz per reticolo a 3 siti

# Conclusioni

In questo lavoro di Tesi è stato implementato, su un simulatore fornito dal framework Qiskit di IBM Quantum Research, l'algoritmo VQE per computer quantistici di tipo NISQ con l'obiettivo di stimare lo stato fondamentale di un reticolo cristallino monodimensionale, descritto attraverso il modello di Hubbard. Tramite la trasformazione di Jordan-Wigner sono stati mappati gli stati degli elettroni di valenza del sistema in stati di qubits. Attraverso il metodo di evoluzione degli stati detto Efficient Hamiltonian Variational Ansatz, ispirato all'evoluzione temporale generata dall'Hamiltoniana di Hubbard, è stato costruito un set di stati parametrizzati su cui si è valutato il valore di aspettazione dell'energia.

I parametri caratteristici degli stati esplorati sono stati fatti variare attraverso l'utilizzo di diversi algoritmi di ottimizzazione per computer tradizionali ricercando il minimo dell'energia.

Le stime dello stato fondamentale, per diversi valori del numero di occupazione della banda di valenza, ottenute attraverso questi ottimizzatori sono state confrontate al fine di identificare quale ottimizzatore restituisse risultati migliori.

Come termine di paragone sono stati utilizzati valori esatti ottenuti diagonalizzando l'operatore Hamiltoniano tramite l'algoritmo NumPyEigensolver.

In conclusione si è osservato che tra gli algoritmi di ottimizzazione implementati da Qiskit, quello che ha restituito stime migliori con maggiore precisione nella convergenza, nel caso di stato iniziale implementato con rotazioni definite da un singolo angolo, è stato l'algoritmo Controlled Random Search con una distanza dal valore reale degli stati fondamentali di pochi punti percentuali rispetto al valore reale osservato.

In particolare le stime più precise sono state realizzate utilizzando un ansatz composto da 3 step con una distanza media dai valori reali al variare del numero di elettroni di valenza pari all'9.5 % nel reticolo a 2 siti, 8.3 % nel reticolo a 3 siti e 12.3% nel reticolo a 4 siti. Infine si è osservato che nel caso in cui nella costruzione dello stato iniziale si lasciassero gli angoli delle rotazioni liberi di variare i risultati migliori, nel caso di reticolo a 3 siti

---

atomici, venivano raggiunti dagli ottimizzatori locali. In particolare l'ottimizzatore UMDA ha raggiunto una distanza del 2.5 % rispetto al valore reale.

# Bibliografia

- [1] F Verstraete e J I Cirac. “Mapping local Hamiltonians of fermions to local Hamiltonians of spins”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2005.09 (set. 2005), P09012–P09012. ISSN: 1742-5468. DOI: 10.1088/1742-5468/2005/09/p09012. URL: <http://dx.doi.org/10.1088/1742-5468/2005/09/P09012>.
- [2] Jules Tilly et al. “The Variational Quantum Eigensolver: A review of methods and best practices”. In: *Physics Reports* 986 (nov. 2022), 1–128. ISSN: 0370-1573. DOI: 10.1016/j.physrep.2022.08.003. URL: <http://dx.doi.org/10.1016/j.physrep.2022.08.003>.
- [3] Stasja Stanisic et al. “Observing ground-state properties of the Fermi-Hubbard model using a scalable algorithm on a quantum computer”. In: *Nature Communications* 13.1 (ott. 2022). ISSN: 2041-1723. DOI: 10.1038/s41467-022-33335-4. URL: <http://dx.doi.org/10.1038/s41467-022-33335-4>.
- [4] Rongxin Xia e Sabre Kais. *Qubit coupled cluster singles and doubles variational quantum eigensolver ansatz for electronic structure calculations*. 2020. arXiv: 2005.08451 [quant-ph].
- [5] *Sequential quadratic programming*, Wikipedia. [https://en.wikipedia.org/wiki/Sequential\\_quadratic\\_programming](https://en.wikipedia.org/wiki/Sequential_quadratic_programming).
- [6] *qiskit.algorithms.optimizers Optimizer available in qiskit framework*. <https://docs.quantum.ibm.com/api/qiskit/qiskit.algorithms.optimizers>.
- [7] *Variational algorithm design, IBM Quantum Learning*. <https://learning.quantum.ibm.com/course/variational-algorithm-design/reference-states>.
- [8] Dave Wecker, Matthew B. Hastings e Matthias Troyer. “Progress towards practical quantum variational algorithms”. In: *Physical Review A* 92.4 (ott. 2015). ISSN: 1094-1622. DOI: 10.1103/physreva.92.042303. URL: <http://dx.doi.org/10.1103/PhysRevA.92.042303>.

## BIBLIOGRAFIA

---

- [9] Carlos Santos, Marcos Gonçalves e Hugo Enrique Hernández Figueroa. “Designing Novel Photonic Devices by Bio-Inspired Computing”. In: *Photonics Technology Letters, IEEE* 22 (set. 2010), pp. 1177 –1179. doi: 10.1109/LPT.2010.2051222.

# Appendice

```
#!/bin/env python3
from qiskit.circuit import Parameter
import numpy as np
from qiskit import *
from tqdm import tqdm
from qiskit_nature.second_q.hamiltonians.lattices import LineLattice, Boundary
from qiskit_nature.second_q.hamiltonians import FermiHubbardModel
import scipy as sp
from qiskit.algorithms.minimum_eigensolvers import VQE
from qiskit.algorithms.minimum_eigensolvers import NumPyMinimumEigensolver
from qiskit.algorithms.eigensolvers import NumPyEigensolver
from qiskit.algorithms.optimizers import CRS
from qiskit.primitives import Estimator
from qiskit_nature.second_q.algorithms import GroundStateEigensolver
from qiskit_nature.second_q.mappers import JordanWignerMapper
from qiskit_nature.second_q.problems import LatticeModelProblem
from qiskit.quantum_info import Statevector
from qiskit_nature.second_q.operators import FermionicOp
from qiskit.quantum_info.operators import Operator
import math as m
import pandas as pd
from itertools import combinations

from qiskit.circuit.library import iSwapGate
sqrt_iSWAP = iSwapGate().power(1/2)
```

---

*# Build G*

```
def G(_gamma):
    G_q = QuantumRegister(2)
    G_circ = QuantumCircuit(G_q, name='G')
    G_circ.append(sqrt_iSWAP, [G_q[0], G_q[1]])
    G_circ.rz(m.pi-_gamma/2, G_q[0])
    G_circ.rz(_gamma/2, G_q[1])
    G_circ.append(sqrt_iSWAP, [G_q[0], G_q[1]])
    G_circ.z(G_q[0])
    G_circ.barrier()
    G = G_circ.to_instruction()
    return G
```

```
def H(_gamma):
    h_q = QuantumRegister(2)
    h_circ = QuantumCircuit(h_q, name='H')
    h_circ.rz(-m.pi/4, h_q[0])
    h_circ.rz(-m.pi/4, h_q[1])
    h_circ.append(sqrt_iSWAP, [h_q[0], h_q[1]])
    h_circ.rz(m.pi+_gamma/2, h_q[0])
    h_circ.rz(m.pi+_gamma/2, h_q[1])
    h_circ.append(sqrt_iSWAP, [h_q[0], h_q[1]])
    h_circ.rz(5*m.pi/4, h_q[0])
    h_circ.rz(5*m.pi/4, h_q[1])
    h_circ.barrier()
    H = h_circ.to_instruction()
    return H
```



```
#!/bin/env python3
```

```
def ansatz(_M, _S, _alpha, _theta, _phi):

    circ=QuantumCircuit(2*_M)
    circ.barrier()
    for i in range(1, 2*_M-2, 2):
        circ.append(G(_alpha[0]), [i, i+1])
    for i in range(0, 2*_M-1, 2):
        circ.append(G(_alpha[0]), [i, i+1])
    for i in range(1, 2*_M-2, 2):
        circ.append(G(_alpha[0]), [i, i+1])
    circ.barrier()
    # parte pari
    for j in range(2, _S+2):
        #spin up
        for i in range(0, _M, 2):
            if i==_M-1:
                continue
            circ.append(H(_theta[j]), [i, i+1])
        for i in range(1, _M, 2):
            if i==_M-1:
                continue
            circ.append(H(_theta[j]), [i, i+1])

        #spin down
        for i in range(_M, 2*_M, 2):
            if i==2*_M-1:
                continue
            circ.append(H(_theta[j]), [i, i+1])

        for i in range(_M+1, 2*_M, 2):
            if i==2*_M-1:
                continue
            circ.append(H(_theta[j]), [i, i+1])
    #end
    circ.barrier()
```

---

```

    for i in range(0, _M):
        circ.crz(U*0.25*_phi[j], i, i+_M) # Ha una fase diversa
    circ.barrier()
    for i in range(0, _M, 2):
        if i==_M-1:
            continue

    circ.barrier()

return circ

```

```

M=3 #numero di siti del reticolo
t=1.0 # energia di hopping
U=0.1 # energia di coulomb
S=3 # numero di steps di trotter-suzuki

```

```

mapper=JordanWignerMapper()
lattice = LineLattice(M, t, onsite_parameter=-t)
fhm=FermiHubbardModel(lattice, onsite_interaction=U)
# operatore in 2 quantiz
op=mapper.map(fhm.second_q_op())
opt=CRS()

```

```

##### VQE #####
# creiamo tutte le possibili configurazioni di stati iniziali

```

```

theta = [Parameter(f' [{j}] ') for j in range(S+2) ]
phi = [Parameter(f'phi[{j}] ') for j in range(S+2) ]
alpha = [Parameter(f'alpha[{j}] ') for j in range(S+2) ]

```

```

Nu_max=M
Nd_max=M
N_iter=int(sys.argv[1])

```

```

pos_up=[i for i in range(M)]          #qubit spin up
pos_down=[M+i for i in range(M)]      #qubit spin down

# operatore numero
op_list = [FermionicOp({f"+_{i}_"-_{i}": 1.0},
                      num_spin_orbitals=M*2).to_matrix().toarray()
           for i in range(M*2)]
up_op_list = [FermionicOp({f"+_{2*i}_"-_{2*i}": 1.0},
                          num_spin_orbitals=M*2).to_matrix().toarray()
              for i in range(M)]
down_op_list = [FermionicOp({f"+_{2*i+1}_"-_{2*i+1}": 1.0},
                           num_spin_orbitals=M*2).to_matrix().toarray()
                for i in range(M-1)]
df_crs=pd.DataFrame(columns=['Nu', 'Nd', 'Nu_fin', 'Nd_fin',
                             'eigen', 'N_init', 'N_fin', 'avg', 'max', 'var'])

# esploriamo tutte le possibili combinazioni
for Nu in range(Nu_max+1): # spin up
    for Nd in range(Nd_max+1): #spin down
        print()
        print(f"({Nu}_spin_up, {Nd}_spin_down)")
        # possibili realizzazioni a Nu fix
        comb_up = combinations(pos_up, Nu)

        for k in list(comb_up):
            # possibili realizzazioni a Nd fix
            comb_down = combinations(pos_down, Nd)

            for j in list(comb_down):
                cir=QuantumCircuit(2*M)
                if(Nu!=0):
                    # fissiamo numero di occupazione 1 sul qubit k
                    cir.x(k)
                if(Nd!=0):
                    # fissiamo numero di occupazione 1 sul qubit j
                    cir.x(j)

```

---

```

# costruiamo ansatz
ansatztest=ansatz(M,S, alpha , theta , phi)
# componiamo reference-state e ansatz
ansatztest=cir.compose(ansatztest)

# inizializziamo a 0 la varianza
var=0
# da cambiare eventualmente
init=np.random.rand (3*S+1)
vqe_solver=VQE(Estimator(), ansatztest , opt ,
               initial_point=init)
#inizializzazione di VQE
# risultati
GSRES=vqe_solver.compute_minimum_eigenvalue(op)

avg=GSRES.eigenvalue/N_iter
maxeigen=GSRES.eigenvalue
var+=pow(GSRES.eigenvalue , 2)/N_iter
for i in tqdm(range(N_iter-1)):
    init=np.random.rand (3*S+1) # da cambiare eventualmente
    # re-inizializzazione di VQE
    vqe_solver=VQE(Estimator(), ansatztest , opt ,
                   initial_point=init)

    # risultati
    result=vqe_solver.compute_minimum_eigenvalue(op)
    #calcola media
    avg+=result.eigenvalue/N_iter
    # calcola varianza
    var+=pow(result.eigenvalue , 2)/N_iter
    #salva valore migliore
    if (result.eigenvalue<=GSRES.eigenvalue):
        GSRES=result
    # salva valore peggiore
    if(result.eigenvalue>maxeigen):
        maxeigen=result.eigenvalue
# fine calcolo varianza

```

```

var=var-pow(avg, 2)
# salva parametri ottimali
optparam=GSRES.optimal_parameters
#circuito stato migliore
c=GSRES.optimal_circuit.assign_parameters(optparam)
# stato migliore
state = Statevector.from_int(0, 2**(2*M))
state = state.evolve(c)

n_fermions = 0 # calcola il numero di elettroni
for oper in op_list:
    n_fermions += state.expectation_value(
        Operator(oper)).real

n_up=0 # calcola il numero di elettroni spin up
for oper in up_op_list:
    n_up += state.expectation_value(Operator(oper)).real

n_down=0 # calcola il numero di elettroni spin down
for oper in down_op_list:
    n_down += state.expectation_value(Operator(oper)).real

# salva in dataframe
df_crs.loc[len(df_crs)]=[Nu, Nd, n_up, n_down,
    GSRES.eigenvalue, Nu+Nd, n_fermions,
    avg, maxeigen, var]

df_crs.to_csv(f'CRS_stat_{S}Step_{M}siti_{N_iter}iter.txt', index=False)

```

