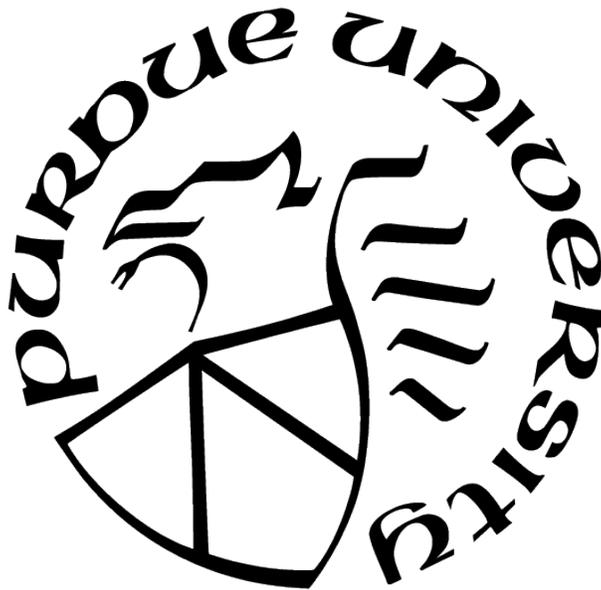# QUANTUM COMPUTATION FOR ELECTRONIC STRUCTURE CALCULATIONS

by

Rongxin Xia

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

Department of Physics and Astronomy

West Lafayette, Indiana

December 2020

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Sabre Kais, Chair

Department of Chemistry

Dr. Yong P. Chen

Department of Physics and Astronomy

Dr. Luis M. Kruczenski

Department of Physics and Astronomy

Dr. Yexiang Xue

Department of Computer Science

Approved by:

Dr. John P. Finley

ACKNOWLEDGMENTS

TABLE OF CONTENTS

5

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

This dissertation contains four projects: transforming electronic structure Hamiltonian to approximating Ising-type Hamiltonian to enable electronic structure calculations by quantum annealing, quantum-assisted restricted Boltzmann machine for electronic structure calculations, hybrid quantum classical neural network for calculating ground state energies of molecules and qubit coupled cluster single and double excitations variational quantum eigensolver for electronic structure. In chapter 1 we present a general introduction of quantum computer, including a brief introduction of two quantum computing model: gate model and quantum annealing model. We also give a general review about electronic structure calculations on quantum computer. In chapter 2, we show an approximating mapping between the electronic structure Hamiltonian and the Ising Hamiltonian. The whole mapping is enabled by first enlarging the qubits space to transform the electronic structure Hamiltonian to a diagonal Hamiltonian. Then introduce ancilla qubits to transform the diagonal Hamiltonian to an Ising-type Hamiltonian. We also design an algorithm to use the transformed Hamiltonian to obtain the approximating ground energy of the original Hamiltonian. The numerical simulation results of the transformed Hamiltonian for H_2, He_2, HeH^+, and LiH molecules match the exact numerical calculations of the original Hamiltonian. This demonstrates that one can map the molecular Hamiltonian to an Ising-type Hamiltonian which could easily be implemented on currently available quantum hardware. In chapter 3, we report a hybrid quantum algorithm employing a restricted Boltzmann machine to obtain accurate molecular potential energy surfaces. By exploiting a quantum algorithm to help optimize the underlying objective function, we obtained an efficient procedure for the calculation of the electronic ground state energy for a small molecule system. Our approach achieves high accuracy for the ground state energy for H_2, LiH, H_2O at a specific location on its potential energy surface with a finite basis set. With the future availability of larger-scale quantum computers, quantum machine learning techniques are set to become powerful tools to obtain accurate values for electronic structures. In chapter 4, we present a hybrid quantum classical neural network that can be trained to perform electronic structure calculation and generate potential energy curves of simple molecules. The method is based on the combina-

tion of parameterized quantum circuit and measurements. With unsupervised training, the neural network can generate electronic potential energy curves based on training at certain bond lengths. To demonstrate the power of the proposed new method, we present results of using the quantum-classical hybrid neural network to calculate ground state potential energy curves of simple molecules such as H_2, LiH and BeH_2. The results are very accurate and the approach could potentially be used to generate complex molecular potential energy surfaces. In chapter 5, we introduce a new variational quantum eigensolver (VQE) ansatz based on the particle preserving exchange gate to achieve qubit excitations. The proposed VQE ansatz has gate complexity up-bounded to $O(n^4)$ where n is the number of qubits of the Hamiltonian. Numerical results of simple molecular systems such as BeH_2, H_2O, N_2, H_4 and H_6 using the proposed VQE ansatz gives very accurate results within errors about 10^{-3} Hartree.

1. INTRODUCTION

Quantum computation has been first proposed by Feyman in 1982 [1]. Unlike classical computer, quantum computer utilizes properties of quantum mechanics, for example, superposition and entanglement, to achieve possible speed up compared to classical computer. Researches on quantum computation mainly focus on developing new feasible quantum computers as well as designing quantum algorithms. In this chapter, we first give a brief review on quantum computation, including quantum bit and two most common quantum computing model – gate model and adiabatic model. Furthermore, one popular research direction of quantum computation is solving quantum physics and quantum chemistry problems, especially electronic structure calculations, on quantum computer. In this chapter, we also give a brief review about approaches to enable electronic structure calculations on quantum computer, including transformations to map the electronic structure problems on quantum computer as well as quantum algorithms to solve electronic structure problems.

1.1 Quantum Computation

1.1.1 Quantum Bit

The elementary computation unit for quantum computation is the quantum bit or the qubit, which is analogous to the bit of classical computation. Unlike the classical bit, which can solely be in one of two states, qubit can be in the superposition of two states represented by a complex vector $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with the two complex-valued coefficients α and β and $|\alpha|^2 + |\beta|^2 = 1$. The state of the qubit can also be written as a 1×2 vector as below.

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

When do measurements on the qubit above, the probability of getting final measured state of the qubit as $|0\rangle$ is $|\alpha|^2$ and getting final state of the qubit as $|1\rangle$ is $|\beta|^2$. Furthermore, the multiple qubits state can be represented by the weighted summation of tensor products of single qubit states as $|\Psi\rangle = \sum_i c_i |\Psi_i\rangle$ where $|\Psi_i\rangle$ can be written as $|\psi_1^i \psi_2^i ... \psi_n^i\rangle$ and $|\psi_1^i \psi_2^i ... \psi_n^i\rangle = |\psi_1^i\rangle \otimes |\psi_2^i\rangle ... \otimes |\psi_n^i\rangle$ for $|\psi_j^i\rangle \in \{|0\rangle, |1\rangle\}$ is the tensor product of single qubit

states. Because of the superposition property of the qubit, quantum computation can save the resource of qubits when representing large data. For a vector with dimension N we only need $\log_2 N$ qubits to represent this vector. Reduced resources to represent data also show the potential of speedup in quantum computing.

Furthermore, if the state of the the multiple qubits state can not be decomposed into a single tensor product of single qubit states, the multiple qubits state is an entangled state. For example, $|01\rangle = |0\rangle \otimes |1\rangle$ is a untangled state while $\frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle$ is an entangled state.

1.1.2 Gate Model

One of the most common quantum computing model is the gate model, which is based on elementary single and multiple qubits gates. In this section we will give a brief review on elementary one and two qubits gates and how these quantum gates can be used to construct universal quantum computation.

Single Qubit Gate

Analogous to classical computer, which uses wires and logic gates to perform computation, gate model quantum computer also uses qubits and quantum gates to perform computation. In classical computation, the wire holds the information and logic gates manipulate the information on the wire. In quantum computation, qubit hold information and quantum gates manipulate the information on the qubit. For example, consider a qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ and a X gate defined as:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

The state of qubit after X gate would be $X|\psi\rangle = \alpha|1\rangle + \beta|0\rangle$. As the normalization rule of the single qubit state in 1.1, we have $|\alpha|^2 + |\beta|^2 = 1$ or the gate should be unitary. In another word, for a single qubit gate U it must satisfy $U^\dagger U = I$ so that we have $\langle\psi|U^\dagger U|\psi\rangle = \langle\psi|\psi\rangle = 1$ where $U^\dagger$ is the adjoint matrix of U [2].

The elementary one qubit gates consisting of Pauli matrices (X, Y, Z), the Hardmard gate (H), the phase gate (S) and $\pi/8$ gate (T) [2]:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \quad T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{bmatrix}$$

Multiple Qubits Gate

Quantum gate is called multiple qubits gates when it manipulates multiple qubits at the same time. However, all qubits gates manipulating more than two qubits can be decomposed to products of single qubit and two qubits controlled operations gates [2]. Thus here we just talk about two qubits controlled operations gate. The most common qubits controlled operation gate is controlled-NOT gate or CNOT gate. Considering two qubits $|c\rangle$ and $|t\rangle$ with qubit $|c\rangle$ being the control qubit and qubit $|t\rangle$ being the target qubit, the CNOT gate works as $\text{CNOT}|c\rangle|t\rangle = |c\rangle|c+t\rangle$ in term of computation basis [2]. For example $\text{CNOT}|1\rangle|1\rangle = |1\rangle|0\rangle$ while $\text{CNOT}|0\rangle|1\rangle = |0\rangle|1\rangle$. The CNOT gate can be written as:

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Moreover, there are advanced controlled gates called controlled-U gate or CU gate where U is a arbitrary unitary single qubit gate. Considering two qubits $|c\rangle$ and $|t\rangle$ with qubit $|c\rangle$ being the control qubit and qubit $|t\rangle$ being the target qubit, CU gate works as $\text{CU}|c\rangle|t\rangle = |c\rangle U^c|t\rangle$ in term of computation basis [2]. For example $\text{CU}|1\rangle|t\rangle = |1\rangle U|t\rangle$ while $\text{CU}|0\rangle|1\rangle = |0\rangle|1\rangle$.

Measurement

The final part of the quantum circuit is the measurement. Measurements will make the superposition state collapse to measured basis. For a final state $|\phi\rangle$ which can be written in the superposition of measured basis $\{|\psi_i\rangle\}$ as $|\phi\rangle = \sum_i a_i|\psi_i\rangle$ where $\{a_i\}$ are complex number and $\sum_i |a_i|^2 = 1$, the probability of final measurement result being $|\psi_i\rangle$ is $|a_i|^2$.

Furthermore, there are two important principles of measurement in quantum gate model. Quantum measurements are performed intermediately sometimes during quantum circuit. For example, the measurement result is used to do classically conditional controlled operation on other qubits. The first principle is the principle of deferred measurement [2]. As stated in the reference [2], measurements can be done at the end of the quantum circuit instead of an intermediate stage [2]. As the principle states, if the measurement results are used to do conditional controlled operations, this can be replaced by a quantum controlled-U gate.

Another principle of measurement is called principle of implicit measurement. [2]. As stated in the reference [2], any unmeasured qubits at the end of the quantum circuits can be viewed as measured [2]. Considering we have a quantum circuit of two qubits. At the end of the quantum circuit, we only do measurements on the qubit 1. The statistics of measurements of qubit 1 will only be determined by the reduced density matrix of qubit 1. However, if we also do measurements on qubit 2, the measurement results will not affect the statistics of measurements of qubit 1.

Universal Quantum Computation

To show the gate model with only elementary single qubit quantum gates as well as CNOT gate is universal, here we show arbitrary quantum gates can be decomposed into the product of elementary single qubit quantum gates and CNOT gate. Consider a universal quantum gate on multiple qubits N, which can always be written as an unitary matrix with dimension 2^N. The first step of the decomposition is to decompose the $2^N \times 2^N$ unitary matrix into the product of the two-level unitary matrix [2]. The next step is to decompose the two-level unitary matrix into the product of elementary single qubit quantum gates and CNOT gate. Gray code [2] shows that any two-level unitary matrix can be decomposed into

the product of elementary single qubit gates and CNOT gates. Thus for any quantum gate it can be decomposed into elementary quantum gate set or the the gate model with elementary single qubit quantum gates as well as CNOT gate is universal.

1.1.3   Adiabatic Model

Another common quantum computation model is the adiabatic model proposed by Edward Farhi [3] which is based on adiabatic theorem [4]. Considering a Hamiltonian $H(s)$ where $s = t/T$. t is the current time and T is the total evolution time. The eigenstates and eigenvalues of $H(s)$ can be written as $|l, s\rangle$ and $E(l, s)$ [3]. l is the level of the eigenstate and $l = 0$ represents the lowest eigenstate. If $E(1, s) > E(0, s)$ holds for all $0 \le s \le 1$ and the system starts from $|\phi(0)\rangle = |l = 0, s = 0\rangle$ which is the ground state of $H(0)$ at $t = 0$, we have [3]:

$$\lim_{T \to \infty} |\langle \phi(T)|l = 0, s = 1\rangle| = 1 \tag{1.1}$$

where $|\phi(T)\rangle$ is the state of the system at $t = T$. The evolution time T has to satisfy:

$$T \gg \frac{\epsilon}{g_{min}^2} \tag{1.2}$$

where $g_{min} = min_{0 \le s \le 1}(E(1, s) - E(0, s))$ and $\epsilon = max_{0 \le s \le 1}|\langle l = 1, s|\frac{dH(s)}{ds}|l = 0, s\rangle|$ [3].

1.2   Electronic Structure Calculations On Quantum Computer

To enable electronic structure calculation on the quantum computer, the first step is to mapping current electronic structure problems to quantum computing model. The whole procedure is enabled by second quantization to get a Hamiltonian in the format of creation and annihilation operators. Then use the mapping between the creation and annihilation operators and Pauli matrices to transform the Hamiltonian into the matrix format, which can be utilized by quantum computation.

17

1.2.1 Second Quantization

The first step of preparing the electronic structure Hamiltonian is to write down the second quantization Hamiltonian. The state of the system can be written as the tensor product of the state of each spin orbital, which can be occupied or not occupied. If the state is $|0\rangle$, the corresponding spin orbital is not occupied. If the state is $|1\rangle$, the corresponding spin orbital is occupied. Thus, the state of the system can be written as $|f_n...f_1\rangle$ where $f_i \in \{0,1\}$. The electronic structure Hamiltonian with Born-Oppenheimer approximation can be expressed in the weighted summation of products of creation and annihilation operators $a_i^\dagger$ and a_i as:

$$\hat{H} = h_0 + \sum_{i,j} h_{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{i,j,k,l} h_{ijkl} a_i^\dagger a_j^\dagger a_k a_l. \tag{1.3}$$

h_0 is the nuclear repulsion energy. The above coefficients h_{ij} and h_{ijkl} are one- and two-electron integrals which can be computed by orbital integrals.

The next step is to use certain transformations as mapping between the operator representation and Pauli matrices, $\{\sigma_x, \sigma_y, \sigma_z\}$ as well as the identity matrix I. The transformed molecular Hamiltonian takes the general form $\sum_i c_i P_i$ where c_i is the coefficient and P_i is the tensor product of Pauli matrices, $\{\sigma_x, \sigma_y, \sigma_z\}$ and the 2×2 identity matrix I.

1.2.2 Transform To Pauli Matrices Format

After the second quantization, the electronic structure Hamiltonian is written in the format consisting of creation and annihilation operators. The next step is to map the creation and annihilation operators into Pauli matrices so that the Hamiltonian can be unitized by quantum computation. In the discussion throughout this dissertation, if not specifically noted, the qubits are indexed starting from 1.

Jordan-Wigner Transformation

The most common transformation used in the quantum computation is Jordan-Wigner transformation [5], [6]. The idea of Jordan-Wigenr transformation is intuitive: Qubit j with state $|0\rangle$ represents the corresponding spin orbital j is not occupied. Qubit j with state $|1\rangle$

represents the corresponding spin orbital j is occupied. Jordan-Wigner transformation stores the occupation information locally on each qubits and thus extra terms are needed for the parity information. For a n-spin system, the Jordan-Wigner transformation for the operator a_j and $a_j^\dagger$ are in the format as below [7]:

$$
\begin{aligned}
a_j^\dagger &\to \frac{1}{2}(\sigma_x^j - i\sigma_y^j) \otimes_{i=1}^{j-1} \sigma_z^i \\
a_j &\to \frac{1}{2}(\sigma_x^j + i\sigma_y^j) \otimes_{i=1}^{j-1} \sigma_z^i
\end{aligned}
\tag{1.4}
$$

The problem is due to the non-locality of storing the parity information, the cost of parity operator, $\otimes_{i=1}^{j-1}\sigma_z^i$ is $O(n)$.

The Parity Transformation

Another common transformation used in the quantum computation is the parity transformation. The qubit j is in the state of the summation of the occupations of spin orbital from 1 to j module 2. In another word, the parity transformation stores the parity information of the spin orbital with index less than or equal to j to qubit j. For a n spin system the transformation of the operators a and $a^\dagger$ are in the format as below [7]:

$$
\begin{aligned}
a_j^\dagger &\to \frac{1}{2}(\otimes_{i=j+1}^n \sigma_x^i \otimes \sigma_x^j \otimes \sigma_z^{j-1} - i \otimes_{i=j+1}^n \sigma_x^i \otimes \sigma_y^j) \\
a_j &\to \frac{1}{2}(\otimes_{i=j+1}^n \sigma_x^i \otimes \sigma_x^j \otimes \sigma_z^{j-1} + i \otimes_{i=j+1}^n \sigma_x^i \otimes \sigma_y^j)
\end{aligned}
\tag{1.5}
$$

The problem is due to the non-locality of storing the occupation information, the cost of occupation operator, $\otimes_{i=j+1}^n \sigma_x^i$ is $O(n)$.

Bravyi-Kitaev Transformation

Bravyi-Kitaev transformation is a combination of the Jordan-Wigner transformation and the parity transformation. The storage of the occupation information and the parity information is balanced for better simulation efficiency [7], [8]. For the qubit with odd index, the state of qubit represents the occupation the current spin orbital. For the qubit with even index, the state of the qubit represents the parity of certain adjacent spin orbitals. For a even index j, when $\log_2 j$ is an integer, the qubit stores all parity of the spin orbitals with index

19

less or equal to j, otherwise the qubit stores the spin orbital with index in the subdividing binary set [9]. the For a n spin system the transformation of the operators a and $a^\dagger$ are in the format as below:

$$a_j^\dagger \to \frac{1}{2}(\sigma_x^{U(j)} \otimes \sigma_x^j \otimes \sigma_z^{P(j)} - i\sigma_x^{U(j)} \otimes \sigma_y^j \otimes \sigma_z^{P(j)})$$
$$a_j \to \frac{1}{2}(\sigma_x^{U(j)} \otimes \sigma_x^j \otimes \sigma_z^{P(j)} + i\sigma_x^{U(j)} \otimes \sigma_y^j \otimes \sigma_z^{P(j)})$$

(1.6)

$U(j)$ and $P(j)$ represent the update set and parity set of the spin orbital j for the Bravyi-Kiteav transformation. $U(j)$ is the updated set of qubits to be changed when a_j or $a_j^\dagger$ applied on the spin orbital j. $P(j)$ is the parity set of qubits to decide phase change when a_j or $a_j^\dagger$ applied on the spin orbital j [7]. Because size of $U(j)$ and $P(j)$ scale $O(\log n)$ [8] the cost of the Brevyi-Kiteav transformation is $O(\log n)$.

Binary Code Transformation

Recently, a new transformation is proposed by Mark Steudtner et al [10], which uses classical code to map the creation and annihilation operators into Pauli matrices. The general format of the mapping can be written as:

$$a_j^\dagger \to \frac{1}{2}\sigma_x^{U_j} \otimes (1 + \sigma_z^{F(j)}) \otimes \sigma_z^{P(j)}$$
$$a_j \to \frac{1}{2}\sigma_x^{U(j)} \otimes (1 - \sigma_z^{F(j)}) \otimes \sigma_z^{p(j)}$$

(1.7)

$U(j)$ is the update set of qubits to be changed when $a_j^\dagger$ or a_j is applied on the spin orbital j. $F(j)$ is the flip set of qubits to check whether applying $a_j^\dagger$ or a_j will give 0. $P(j)$ is the parity set of the qubits to check the phase change when $a_j^\dagger$ or a_j is applied on the spin orbital j [10]. An simple example is the Jordan-Wigner transformation. For Jordan-Wigner transformation, $U(j) = \{j\}$, $F(j) = \{j\}$ and $P(j) = \{1...j-1\}$. Thus Jordan-Wigner transformation using binary code transformation can be written as:

$$a_j^\dagger \to \frac{1}{2}(\sigma_x^j - i\sigma_y^j) \otimes_{i=1}^{j-1} \sigma_z^i$$
$$a_j \to \frac{1}{2}(\sigma_x^j + i\sigma_y^j) \otimes_{i=1}^{j-1} \sigma_z^i$$

(1.8)

which is same as the original Jordan-Wigner transformation.

Furthermore, binary code transformation can also use other classical code like checksum code, to utilize the particle preservation or spin symmetry to reduce the number of qubits [10]. For example, we can utilize the spin symmetry which requires the number of spin-up and spin-down electrons to be same, which can be encoded by the checksum code and used by the binary code transformation to save 2 qubits of the final Hamiltonian [10].

1.3   Quantum Computation for Electronic Structure Calculations

Current applications of quantum computation for electronic structure calculations focuses on solving the ground state and excited states of chosen system. Two main approaches for electronic structure calculations on quantum computation are Phase Estimation Algorithm(PEA) and Variational Quantum Eigensolver(VQE).

1.3.1   Phase Estimation Algorithm

The Phase Estimation Algorithm(PEA) has been proposed to calculate the eigenvalue of the corresponding eigenstate of a Hamiltonian by using quantum Fourier transform. The PEA can be divided into the phase calculation and the inverse quantum Fourier transformation. Phase calculation is enabled by controlled-U gate as in Figure 1.1.
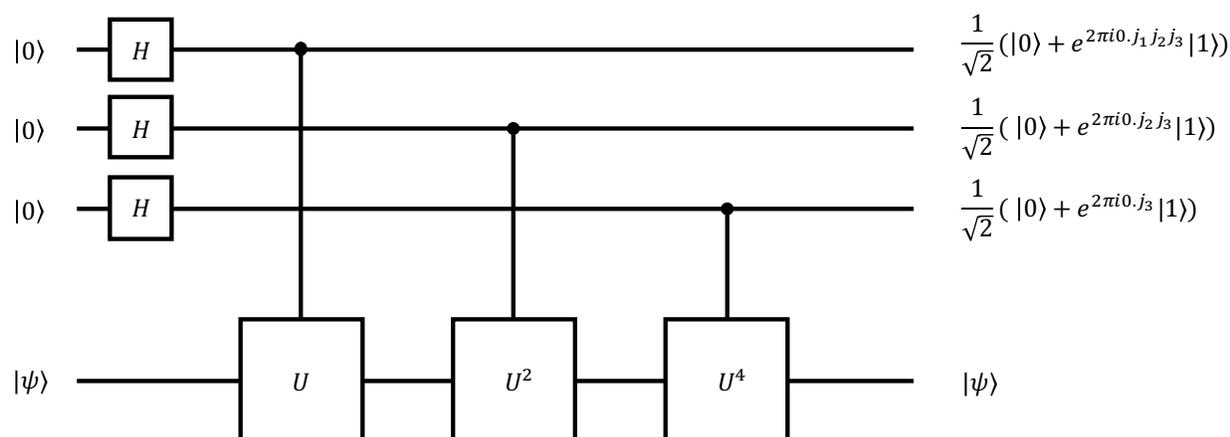
Figure 1.1. First part of PEA is enabled by controlled-U gate

21

$U = \mathrm{e}^{2\pi\mathrm{i}H}$ and H is a Hamiltonian. The first register is initialized as $\frac{1}{2^{n/2}} \sum_{k=0}^{2^n-1} |k\rangle$ by using Hardmard gates and the second register takes the input of a eigenstate of the Hamiltonian H. As in the Figure 1.1, suppose the eigenstate is $|\psi\rangle$ and corresponding eigenvalue is $E_\psi = 0.\mathrm{j}_1\mathrm{j}_2\mathrm{j}_3$. We have the state after the first part should be:

$$\frac{(|0\rangle + \mathrm{e}^{2\pi\mathrm{i}0.\mathrm{j}_1\mathrm{j}_2\mathrm{j}_3}|1\rangle)(|0\rangle + \mathrm{e}^{2\pi\mathrm{i}0.\mathrm{j}_2\mathrm{j}_3}|1\rangle)(|0\rangle + \mathrm{e}^{2\pi\mathrm{i}0.\mathrm{j}_3}|1\rangle)}{2\sqrt{2}}|\psi\rangle \tag{1.9}$$

The second part is inverse quantum Fourier transform as in Figure 1.2. The gate R_i denotes the unitary transformation:

$$R_\mathrm{i} = \begin{bmatrix} 1 & 0 \\ 0 & \mathrm{e}^{2\pi/2^\mathrm{i}} \end{bmatrix}$$
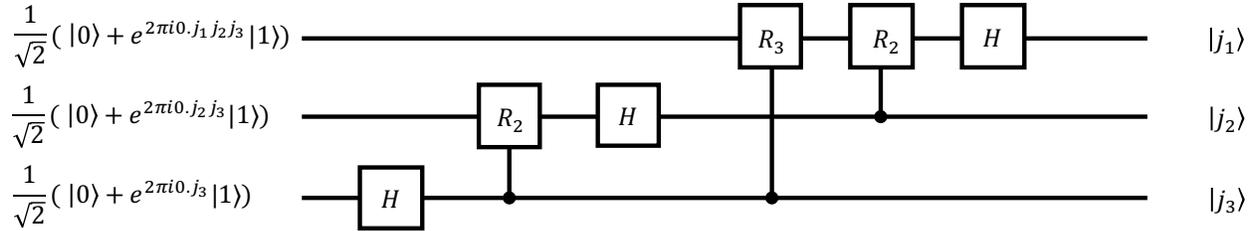
Figure 1.2. Second part of PEA is enabled by inverse Quantum Fourier Transformation

As Figure 1.2 illustrates that $\frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{2\pi\mathrm{i}0.\mathrm{j}_3}|1\rangle)$ would become $|\mathrm{j}_3\rangle$ after Hardmard gate. $\frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{2\pi\mathrm{i}0.\mathrm{j}_2\mathrm{j}_3}|1\rangle)$ would become $|\mathrm{j}_2\rangle$ after controlled rotation gate and Hardmard gate. $\frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{2\pi\mathrm{i}0.\mathrm{j}_1\mathrm{j}_2\mathrm{j}_3}|1\rangle)$ would become $|\mathrm{j}_1\rangle$ after controlled rotation gates and Hardmard gate. Thus, by measuring states of three qubits we would be able to get $E_\psi = 0.\mathrm{j}_1\mathrm{j}_2\mathrm{j}_3$

However, PEA has several problems. The probability of the getting the correct eigenvalue equals to the overlap between the input and target eigenstate, which will require a highly approximating precomputed eigenstate as the input. Furthermore, PEA requires controlled-U gates base on the Hamiltonian, which will require too many number of elementary quantum gates to be implemented on current Noisy Intermediate-Scale Quantum (NISQ) [11] device.

1.3.2 Variational Quantum Eigensolver

Another approach to solve electronic structure problems on the quantum computer is variational quantum eigensolver (VQE), which is based on parameterized quantum circuit, variationally calculate ground state energies. In the following sections, we give brief introductions on parameterized quantum circuit, variational quantum eigenslover ansatz and evaluation of Hamiltonian.

Parameterized Quantum Circuit

Parameterized quantum circuit (PQC), also called variational quantum circuit, is a quantum circuit consisting of quantum gates. The general parameterized quantum circuit can be written as:

$$U(\vec{\theta}) = \prod_i U_i \tag{1.10}$$

where $U(\vec{\theta})$ represents the PQC, $\vec{\theta}$ are adjustable parameters and U_i can be quantum gates not taking parameters and quantum gates taking parameters. The most common quantum gates taking parameters are rotation gates:

$$R_x(\theta) = \mathrm{e}^{-\mathrm{i}\frac{\theta}{2}\sigma_x} \quad R_y(\theta) = \mathrm{e}^{-\mathrm{i}\frac{\theta}{2}\sigma_y} \quad = \mathrm{e}^{-\mathrm{i}\frac{\theta}{2}\sigma_z} \tag{1.11}$$

The output state by the PQC $U(\vec{\theta})$ and input state $|\phi\rangle$ can be written as $U(\vec{\theta})|\phi\rangle$. By adjusting $\vec{\theta}$, we can adjust the output state to approximate arbitrary states.

Variational Quantum Eigensolver Ansatz

A new quantum algorithm based on variational algorithm has been proposed to enable electronic structure calculation on current NISQ device. Variational Quantum Eigensolver (VQE) is a hybrid quantum and classical algorithm [12] based on PQC. By optimizing the parameters of PQC, VQE tries to make the output state approximating the ground state. Thus, the ground state energy can be estimated by evaluating the corresponding Hamiltonian.
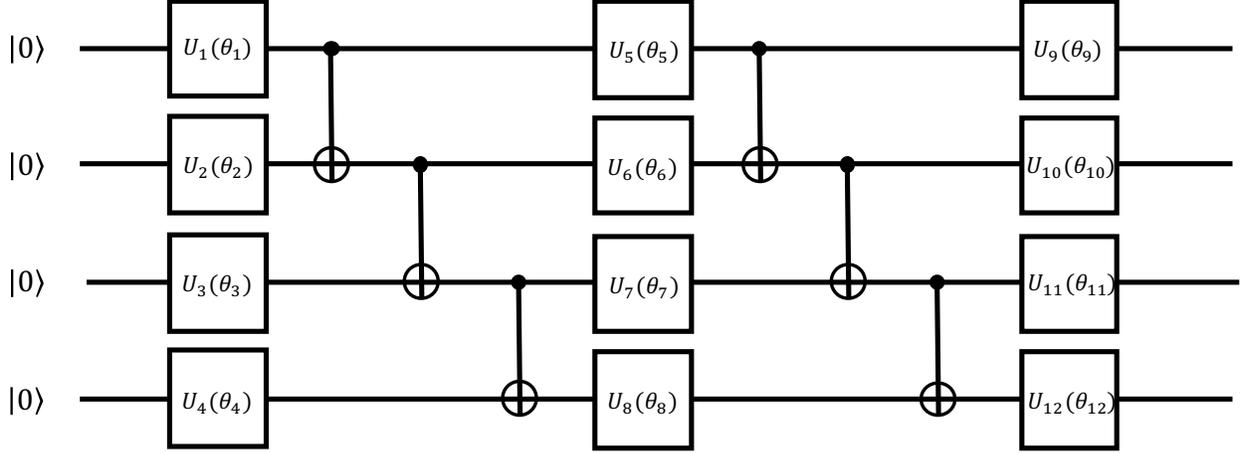
Figure 1.3. Example of VQE quantum circuit. $U_i(\theta_i)$ is an unitary quantum gate which can be adjusted by the parameter θ_i.

$|\Theta\rangle$ is the final state of the quantum circuit. The expectation value of a Hamiltonian H of the final state can be calculated by measuring the final state:

$$\langle\Theta|H|\Theta\rangle = \langle 0|U^\dagger(\vec{\theta})HU(\vec{\theta})|0\rangle \tag{1.12}$$

The expectation $\langle\Theta|H|\Theta\rangle$ can be optimized by adjusting parameters $\vec{\theta}$. By using classical optimizing methods, the expectation value can be optimized to minimal value which can be used as approximating ground energy of H. One example of VQE is shown as Figure 1.3. VQE is more favorable for NISQ quantum computers for that it has low circuit depth [13], [14].

Evaluation of Hamiltonian

After certain transformations, for example, Jordan-Wigner transformation, parity transformation or Bravyi-Kitaev transformation etc, the Hamiltonian will be written as weighted summation of tensor products of Pauli matrices and 2×2 identity matrix:

$$H = \sum_i c_i P_i \tag{1.13}$$

To evaluate H, the evaluation is done by evaluating each tensor product term so that $\langle H \rangle = \sum_i c_i \langle P_i \rangle$. $\langle P_i \rangle$ can be obtained by measuring the qubits after I, $R_x(\frac{\pi}{2})$ or $R_y(\frac{-\pi}{2})$. Let us choose a two qubits system as an example. If P_i is the tensor product of σ_z and I, $\langle P_i \rangle$ can be obtained by the probabilities of measuring qubits in different bases. For example, $P_i = I^1 \sigma_z^2$, we can write $\langle P_i \rangle$ as:

$$\langle P_i \rangle = \langle \phi | I^1 \sigma_z^2 | \phi \rangle = a_{00} - a_{01} + a_{10} - a_{11} \tag{1.14}$$

where a_{00}, a_{01}, a_{10} and a_{11} are the probabilities of measuring the final state $|\phi\rangle$ in bases $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$.

If P_i is the tensor product including σ_x and σ_y, $R_x(\frac{\pi}{2})$ and $R_y(\frac{-\pi}{2})$ can be appended after corresponding qubits and $\langle P_i \rangle$ can be obtained by the probabilities of measuring qubits in different bases. For example, $P_i = \sigma_x^1 \sigma_y^2$, we can write $\langle P_i \rangle$ as:

$$\begin{aligned} \langle P_i \rangle &= \langle \phi | R_y^1(\frac{\pi}{2}) R_x^2(\frac{-\pi}{2}) \sigma_x^1 \sigma_y^2 R_x^2(\frac{\pi}{2}) R_y^1(\frac{-\pi}{2}) | \phi \rangle \\ &= \langle \phi | \sigma_z^1 \sigma_z^2 | \phi \rangle = a_{00} - a_{01} - a_{10} + a_{11} \end{aligned} \tag{1.15}$$

where a_{00}, a_{01}, a_{10} and a_{11} are the probabilities of measuring the final state $|\phi\rangle = R_x^2(\frac{\pi}{2}) R_y^1(\frac{-\pi}{2}) |\phi\rangle$ in bases $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$ and R_x^2 R_y^1 are R_x gate on qubit 2 and R_y gate on qubit 1.

2. TRANSFORM ELECTRONIC STRUCTURE PROBLEMS TO APPROXIMATING ISING-TYPE HAMILTONIAN

Solving the Schrödinger equation is hard due to the dimension of the Hilbert space increasing exponentially with the number of the particles in the system. Exact solution requires large computing resource. Thus, quantum chemistry tries to use approximate methods to solve the Schrödinger equation to chemical accuracy (around 1 kacl/mole). A few contributions from past decades includes *ab initio*, Density Functional, Density Matrix, Algebraic, Quantum Monte Carlo and Dimensional Scaling [15]–[17]. However, all these methods need large computing resource when the size of the system increases.[1]

One important problem in quantum chemistry, electronic structure problem, as [18]–[20] suggested, would be exponentially hard for classical methods. In the meantime, people developed a lot quantum algorithm to simulate electronic structure on the quantum computer, which gives a new promising route to solve these problems [14], [21]. Recently, there has been an attempt at using an adiabatic quantum computing model to perform electronic structure calculations [22]. The fundamental concept behind the adiabatic quantum computing (AQC) method is to define a problem Hamiltonian, H_P, engineered to have its ground state encode the solution of a corresponding computational problem. The system is initialized in the ground state of a beginning Hamiltonian, H_B, which is easily solved classically. The system is then allowed to evolve adiabatically as: $H(s) = (1 - s)H_B + sH_P$ (where s is a time parameter, $s \in [0, 1]$). The adiabatic evolution is governed by the Schrödinger equation for the time-dependent Hamiltonian $H(s(t))$.

The largest scale implementation of AQC to date is by D-Wave Systems [23], [24]. In the case of the D-Wave device, the physical process undertaken which acts as an adiabatic evolution is more broadly called *quantum annealing* (QA). The quantum processors manufactured by D-Wave are essentially a transverse Ising model with tunable local fields and coupling coefficients. The governing Hamiltonian is given as: $H = \sum_i \Delta_i \sigma_x^i + \sum_i h_i \sigma_z^i + \sum_{i,j} J_{ij} \sigma_z^i \sigma_z^j$; where the parameters Δ_i, h_i and J_{ij} are the physically tunable field, self-interaction and

---

[1]Contents in this chapter are reprinted and adapted with permission from Xia, R., Bian, T., & Kais, S. (2017). Electronic structure calculations and the Ising Hamiltonian. The Journal of Physical Chemistry B, 122(13), 3384-3395. Copyright (2017) American Chemical Society

site-site interaction. The qubits are connected in a specified graph geometry, permitting the embedding of arbitrary graphs. The adiabatic evolution is initialized at $H_B = -h\sum_i \sigma_x^i$ and evolves into the problem Hamiltonian: $H_P = \sum_i h_i \sigma_z^i + \sum_{i,j} J_{ij}\sigma_z^i \sigma_z^j$. Therefore any combinatorial optimization NP-hard problem may be encoded into the parameter assignments, $\{h_i, J_{ij}\}$, of H_P and may exploit the adiabatic evolution under $H(s) = (1-s)H_B + sH_P$ as a method for reaching the ground state of H_P.

In this chapter, we present a detailed approximating mapping of electronic structure problems to Ising-type Hamiltonian, which can be implemented on the D-Wave system. We also presented numerical results of the mapping to show the correctness of our mapping. The structure of the chapter consists of mapping electronic structures to Pauli matrices format, approximately mapping Pauli matrices format to k-local diagonal Hamiltonian, decrease the locality of k-local diagonal Hamiltonian to Ising-type Hamiltonian and numerical results of the approximating mapping.

2.1   Mapping Electronic Structure Hamiltonian to Pauli Matrices Format

Explicitly, this general procedure begins with a second quantization description of a fermionic system in which N single-particle states can be either empty or occupied by a fermionic particle [17], [25] as described in section 1.4.2. Thus, the molecular electronic Hamiltonian with Born-Oppenheimer approximation can be written as:

$$\hat{H} = h_0 + \sum_{i,j} h_{ij}a_i^\dagger a_j + \frac{1}{2}\sum_{i,j,k,l} h_{ijkl}a_i^\dagger a_j^\dagger a_k a_l. \tag{2.1}$$

h_0 is the nuclear repulsion energy. The above coefficients h_{ij} and h_{ijkl} are one and two-electron integrals can be computed by quantum chemistry methods. The next step is to convert to a Pauli matrix representation of the creation and annihilation operators. We can then use transformations as described in section 1.4.2 to map between the second quantization operators and Pauli matrices $\{\sigma_x, \sigma_y, \sigma_z\}$ as well as 2×2 identity matrices. The molecular Hamiltonian takes the general form:

$$H = \sum_i c_i P_i \tag{2.2}$$

Within the above, the c_i is the coefficient and P_i is the tensor product of Pauli matrices $\{\sigma_x, \sigma_y, \sigma_z\}$ as well as 2×2 identity matrix.

2.2 Approximate Mapping Pauli Matrices Format Hamiltonian to K-local Diagonal Hamiltonian

2.2.1 Hilbert Space Mapping

The mapping is enabled by enlarging the n qubits space to $r \times n$ qubits space where r is the number of repeating n qubits. Consider a quantum state in the n qubit space:

$$|\psi\rangle = \sum_i a_i |\phi_i\rangle \tag{2.3}$$

With $\{|\phi_i\rangle\}$ as the basis set consisting of bases representing combinations of spin-up and spin-down qubits. Here we just consider $\{a_i\}$ are real values for that the eigenstates of Pauli matrices format electronic structure Hamiltonian can always only have real coefficients if using real-valued basis set as well as real-valued transformation. The new mapping state in the mapped $r \times n$ qubits space can be written as:

$$|\Psi\rangle = \otimes_{j=1}^{r} |\Phi_j\rangle \tag{2.4}$$

With $|\Phi_j\rangle$ representing the the state of the j_{th} n qubits in the $r \times n$ qubits space. For the n qubits system, the coefficient of basis a_i may be a positive or a negative real value while $|\Psi\rangle$ does not contain sign information, we introduce an extra information $\{S(j)\}$ for the $r \times n$ qubits to store the sign information of each n qubits. $S(j)$ represents the sign value of the j_{th} n qubits in the $r \times n$ qubits system, which can be $\{1, -1\}$ and has the relationship $r = \sum_{j=1}^{r} |S(j)|$. We introduce the new variable $b_i = \sum_{|\Phi_j\rangle = |\phi_i\rangle} S(j)$ to count for the repetition of basis $|\phi_i\rangle$ with sign information in the $r \times n$ qubits space. The relationship between a_i and b_i can be written as:

$$a_i \approx \frac{b_i}{\sqrt{\sum_m b_m^2}} \tag{2.5}$$

To illustrate our mapping, we give several mappings examples as in Figure 2.1. The first example is to map 2 qubits space to 4 qubits space. For a state $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ the

mapping result is $|\Psi\rangle = |0011\rangle$ with $S(1) = S(2) = 1$. It has the relationship $a_i = \frac{b_i}{\sqrt{\sum_m b_m^2}}$ with $a_1 = a_2 = \frac{1}{\sqrt{2}}$ and $b_1 = b_2 = 1$. The second example is to map 2 qubits space to 4 qubits space. For a state $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$ the mapping result is $|\Psi\rangle = |0011\rangle$ with $S(1) = -S(2) = 1$. It has the relationship $a_i = \frac{b_i}{\sqrt{\sum_m b_m^2}}$ with $a_1 = -a_2 = \frac{1}{\sqrt{2}}$ and $b_1 = -b_2 = 1$. The third example is to map 2 qubits space to 8 qubits space. For a state $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ the mapping result is $|\Psi\rangle = |00111111\rangle$ with $S(1) = S(2) = S(3) = -S(4) = 1$. It has the relationship $a_i = \frac{b_i}{\sqrt{\sum_m b_m^2}}$ with $a_1 = -a_2 = \frac{1}{\sqrt{2}}$ and $b_1 = b_2 = 1$. Note here b_2 is calculated by $b_2 = S(2) + S(3) + S(4) = 1$
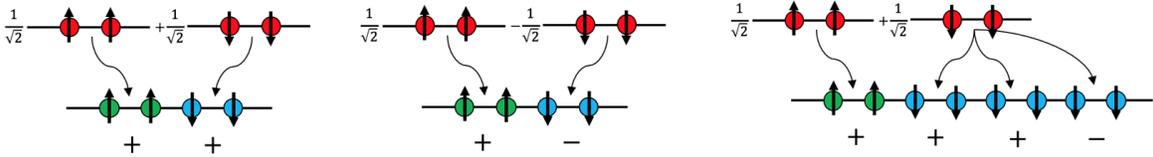
Figure 2.1. Left: A mapping between a two qubits state with two positive coefficients and a four qubits state. **Middle**: A mapping between a two qubits state with one positive coefficient and one negative coefficient and a four qubits state. **Right** A mapping between a two qubits state and a eight qubits state.

2.2.2 Operator Mapping

After the mapping of the Hilbert space, the mapping of the operator should follow the same way. First let us introduce several notations we will use in the following description of the operator mapping. i_k qubit in the $r \times n$ qubits space represents the i_{th} qubit in the k_{th} n qubits of total $r \times n$ qubits. We also mark the k_{th} n qubits of total $r \times n$ qubits are in the n qubits basis $|\phi_{b(k)}\rangle$ or $|\Phi_k\rangle = |\phi_{b(k)}\rangle$.

The mapping of the Pauli matrices and the Identity matrices can be written as:

$$\sigma_x^i \to \frac{1 - \sigma_z^{i_j}\sigma_z^{i_k}}{2} \qquad \sigma_y^i \to i\frac{\sigma_z^{i_k} - \sigma_z^{i_j}}{2}$$
$$\sigma_z^i \to \frac{\sigma_z^{i_j} + \sigma_z^{i_k}}{2} \qquad I^i \to \frac{1 + \sigma_z^{i_j}\sigma_z^{i_k}}{2} \tag{2.6}$$
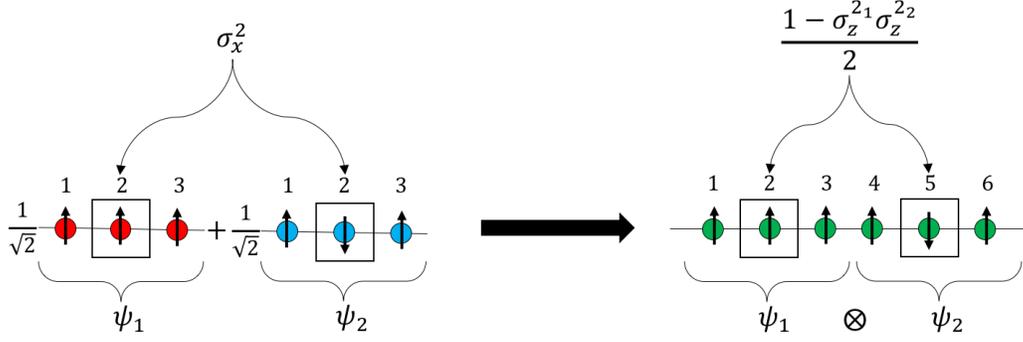
Figure 2.2. Example of mapping σ_x^2 between different basis from a 3 qubits state to a 6 qubits state. The spin operators act on the second qubit in the original Hamiltonian basis and on the second and fifth qubits in the mapped Hamiltonian basis.

Note here σ_x^i represents the Pauli matrix σ_x on the i_{th} qubits in the n qubits space and $\sigma_z^{i_j}$ represents the Pauli matrix σ_z on the i_{th} qubit of the j_{th} n qubits in the $r \times n$ qubits space.

Lemma 1:

$$\langle \phi_{b(j)} | \otimes_i I_i | \phi_{b(k)} \rangle = \langle \Psi | \prod_i \frac{1 + \sigma_z^{i_j} \sigma_z^{i_k}}{2} | \Psi \rangle \tag{2.7}$$

Proof: I_i in the space of $|\psi\rangle$ is to check if i_{th} digits of $|\phi_{b(j)}\rangle$ and $|\phi_{b(k)}\rangle$ are the same or not. If they are the same, it yields 1, otherwise 0. On the other hand, $\frac{1 + \sigma_z^{i_j} \sigma_z^{i_k}}{2}$ in the space of $|\Psi\rangle$ is to check the i_{th} digits of the j_{th} n-qubits subspace ($|\phi_{b(j)}\rangle$) and the k_{th} n-qubit subspace ($|\phi_{b(k)}\rangle$) are the same or not. If they are the same it yields 1, otherwise 0. (For $\frac{1 + sigma_z^{i_j} \sigma_z^{i_k}}{2}$ we omit the operators for other digits which are the identity I.)

Lemma 2:

$$\langle \phi_{b(j)} | \otimes_{i<m} I_i \otimes \sigma_x^m \otimes_{i>m} I_i | \phi_{b(k)} \rangle$$
$$= \langle \Psi | \prod_{i<m} \frac{1 + \sigma_z^{i_j} \sigma_z^{i_k}}{2} \times \frac{1 - \sigma_z^{m_j} \sigma_z^{m_k}}{2} \times \prod_{i>m} \frac{1 + \sigma_z^{i_j} \sigma_z^{i_k}}{2} | \Psi \rangle \tag{2.8}$$

Proof: σ_x^i in the space of $|\psi\rangle$ is to check if the i_{th} digit of $|\phi_{b(j)}\rangle$ and $|\phi_{b(k)}\rangle$ are the same or not. If they are the same it yields 0, otherwise 1. Similarly, $\frac{1 - \sigma_z^{i_j} \sigma_z^{i_k}}{2}$ in the space $|\Psi\rangle$ is to

verify the i_{th} digits of the j_{th} n-qubit subspace ($|\phi_{b(j)}\rangle$) and the k_{th} n-qubit subspace ($|\phi_{b(k)}\rangle$) are identical. If they are the same it gives 0, otherwise 1. (For $\frac{1-\sigma_z^{i_j}\sigma_z^{i_k}}{2}$ we omit operators for other digits which are the identity I.)

Also, σ_y^i and $\mathbf{i}\frac{\sigma_z^{i_k}-\sigma_z^{i_j}}{2}$ have the same function in different spaces. These operators are used to check the i_{th} digits of $|\phi_{b(j)}\rangle$ and $|\phi_{b(j)}\rangle$. Also, σ_z^i and $\frac{\sigma_z^{i_j}+\sigma_z^{i_k}}{2}$ have the same function in different spaces to check the i_{th} digits of $|\phi_{b(j)}\rangle$ and $|\phi_{b(j)}\rangle$. This can be easily verified by the above discussion.

Theorem 1: Any real valued Hamiltonian in the space of n qubits space in the format of Pauli and Identity Matrices can be mapped approximately to the $r \times n$ as described above.

Proof: First we introduce a new notation, we mark the mapping between j_{th} n qubits and k_{th} n qubits in $|\Psi\rangle$, $\frac{1-\sigma_z^{i_j}\sigma_z^{i_k}}{2}$ as $X_i^{(j,k)}$, $\frac{\sigma_z^{i_j}+\sigma_z^{i_k}}{2}$ as $Z_i^{(j,k)}$, $\mathbf{i}\frac{\sigma_z^{i_k}-\sigma_z^{i_j}}{2}$ as $Y_i^{(j,k)}$ and $\frac{1+\sigma_z^{i_j}\sigma_z^{i_k}}{2}$ as $I_i^{(j,k)}$.

If the Hamiltonian H in the n qubits space can be written as:

$$H = \sum_i h_i \otimes_{a_i} \sigma_x^{a_i} \otimes_{b_i} \sigma_y^{b_i} \otimes_{c_i} \sigma_z^{c_i} \otimes_{d_i} I^{d_i} \tag{2.9}$$

where h_i is the coefficient and a_i, b_i, c_i d_i are indexes of qubits.

We can write mapped $H'_{(j,k)}$ in the $r \times n$ qubits as:

$$H'_{(j,k)} = \sum_i h_i \prod_{a_i} X_{a_i}^{(j,k)} \prod_{b_i} Y_{b_i}^{(j,k)} \prod_{c_i} Z_{c_i}^{(j,k)} \prod_{d_i} I_{d_i}^{(j,k)} \tag{2.10}$$

It can be verified following the rules above if $|\Psi\rangle$ and $|\phi\rangle$ are constructed as in equation 2.3 and 2.4:

$$\langle\Psi|H'_{(j,k)}|\Psi\rangle = \langle\phi_{b(j)}|H|\phi_{b(k)}\rangle \tag{2.11}$$

Furthermore, for all $j = k$ we can simplify to:

$$
\begin{aligned}
X_i^{(j,j)} &= \frac{1-\sigma_z^{i_j}\sigma_z^{i_j}}{2} = 0 \quad & Y_i^{(j,j)} &= \mathbf{i}\frac{\sigma_z^{i_j}-\sigma_z^{i_j}}{2} = 0 \\
Z_i^{(j,j)} &= \frac{\sigma_z^{i_j}+\sigma_z^{i_j}}{2} = \sigma_z^{i_j} \quad & I_i^{(j,j)} &= \frac{1+\sigma_z^{i_j}\sigma_z^{i_j}}{2} = I
\end{aligned}
\tag{2.12}
$$

31

Thus, if we add the sign $S(\text{j})$ and $S(k)$, we can get:

$$
\begin{aligned}
\langle\Psi| & \sum_{\text{j}=1}^{r}\sum_{k=1}^{r} H'_{(\text{j},k))}S(\text{j})S(k)|\Psi\rangle \\
&= \sum_{\text{j}=1}^{r}\sum_{k=1}^{r}\langle\phi_{b(\text{j})}|H|\phi_{b(k)}\rangle S(\text{j})S(k) \\
&= \sum_{J,K}^{J\neq K, J,K\leq 2^n} b_J b_K \langle\phi_J|H|\phi_K\rangle + \sum_{J}^{J\leq 2^n} b_J^2\langle\phi_{\text{j}}|H|\phi_J\rangle \\
&= \sum_m b_m^2\langle\psi^1|H|\psi^1\rangle \approx \sum_m b_m^2\langle\psi|H|\psi\rangle
\end{aligned}
\tag{2.13}
$$

where j k are indexes of the n qubits in the $r\times n$ qubits and J, K are indexes of the basis of n qubits. $|\psi^1\rangle = \sum_J \frac{b_J}{\sqrt{\sum_m b_m^2}}|\phi_J\rangle$ and $|\psi\rangle = \sum_J a_J|\phi_J\rangle$ (for $\sum_m b_m^2 = 0$ case $|\psi^1\rangle = |\psi\rangle = 0$). By increasing r we can use $|\psi^1\rangle$ to approximate $|\psi\rangle$.

Through above we have established a mapping of operators between n qubits space and $r\times n$ qubits space by constructing $H' = \sum_{\text{j},k}^{\text{j},k\leq r} H'_{(\text{j},k)}S(\text{j})S(k)$. The expectation values of the original operator and the mapped operator have a difference of a constant $\sum_m b_m^2$ which can be calculated on the mapped state by constructing a new operator C where:

- $C = \sum_{\pm}(\sum_{\text{i}}(\prod_{k=1_{\text{i}}}^{n_{\text{i}}}\frac{1\pm\sigma_z^k}{2})S(\text{i}))^2$

- $\sum_{\pm}$ over all combination of positive and negative signs of each digit in each i_{th} of the n-qubit in space $|\Psi\rangle$.

- $\sum_{\text{i}}$ over all n-qubit collection in $|\Psi\rangle$ to check whether each n qubits is in a certain state.

- $\prod_{k=1_{\text{i}}}^{n_{\text{i}}}$ over each qubits of i_{th} n qubits in $|\Psi\rangle$.

- $\frac{1\pm\sigma_z^k}{2}$ is to check whether k_{th} qubits of i_{th} n-qubit subspace is in a certain state. $\frac{1+\sigma_z^k}{2}$ is 1 when the k_{th} qubits is in $|0\rangle$ state and 0 otherwise.

Here, we present an algorithm combing $H' = \sum_{\text{j},k}^{\text{j},k\leq r} H'_{(\text{j},k)}$ and C to calculate the ground state of the initial Hamiltonian H. We mark the expectation value of $|\psi^1\rangle$ for H as λ_1. The expectation value of mapped $|\Psi\rangle$ for H' is $\sum_m b_m^2\lambda_1$. Thus, if we choose a λ and construct a Hamiltonian $H' - \lambda C$. The expectation value of $|\Psi\rangle$ for $H' - \lambda C$ is $\sum_m b_m^2(\lambda_1 - \lambda)$

Algorithm 1

for i from 0 to $\lfloor \frac{r}{2} \rfloor$ **do**

    Set the signs of the first i n qubits to be negative and the others to be positive.

    Construct H' and C. Set λ to be a large number (at least larger than the minimal λ_1 of all λ_1 with current signs excluding case $\sum_m b_m^2 = 0$ to avoid case $\sum_m b_m^2 = 0$).

    while True **do**

        Calculate $H' - \lambda C$ and get the ground state $|\Psi'\rangle$ and the corresponding ground state eigenvalue $\sum_m b_m^2(\lambda_1 - \lambda)$.

        if $\sum_m b_m^2(\lambda_1 - \lambda) \geq 0$ **then**

            Break the while loop

        end if

        Calculate C on $|\Psi'\rangle$ to get $\sum_m b_m^2$ then get λ_1. Set λ to be λ_1.

    end while

    Set λ_2 to be λ and λ_2 is the smallest approximating eigenvalue with current signs.

end for

The smallest one among all λ_2 is the approximating ground state energy of H.

Theorem 2: The algorithm will stop at the minimum achievable expectation value of H by the mapping in finite iterations.

Proof: The algorithm will continue only if we find a mapped state $|\Psi\rangle$ of $H' - \lambda C$ with eigenvalue $\sum_m b_m^2(\lambda_1 - \lambda) < 0$, which would give us $\lambda_1 - \lambda < 0$. The algorithm will set $\lambda = \lambda_1$ and start a new iteration. Because we first set λ to be a large number (at leaset larger than the minimal λ_1 all λ_1 with current signs excluding case $\sum_m b_m^2 = 0$), during the first iteration, we will have the minimal eigenvalue of $H' - \lambda C$ as a negative value and λ will be set to a possible λ_1. As long as there is a possible λ_1 which does not correspond to case $\sum_m b_m^2 = 0$ and is smaller than current λ, we always have minimal eigenvalue of $H' - \lambda C$ is negative. Each time λ is set to a smaller value and decreases monotonically.

We will finally come to the minimum achievable expectation value by the mapping for the finite number of expectation values. If and only if λ is the minimal λ_1 of all λ_1 with current signs excluding case $\sum_m b_m^2 = 0$, the minimal eigenvalue of $H' - \lambda C$ will become 0

and the algorithm will stop. Thus λ will decrease for each iteration and will finally stop only if become the smallest achievable expectation value (excluding case $\sum_m b_m^2 = 0$).

Theorem 3: To account for the sign, we just need to set i from 0 to $\lfloor \frac{r}{2} \rfloor$ and set signs of the first i_{th} n qubits to be negative and the others to be positive in $|\Psi\rangle$.

Proof: If the number of of n qubits with negative sign is larger than the number of n qubits with positive sign, we can always make all negative signs to positive signs and all positive signs to negative signs without changing the final results. If we have an n qubits in the $|\Psi\rangle$ with negative sign, where $|\Psi\rangle$ has total i n qubits with negative sign. If this n qubits are not in first i n qubits, we can rearrange it to the first i n qubits by exchanging it with n qubits in first i n qubits which has positive sign without changing the final results. Thus all combination can be reduced to the combination stated in this Theorem 3.

Thus, we have established a transformation from an initial Hamiltonian to a k-local diagonal Hamiltonian and presented an algorithm to calculate the minimum achievable expectation value of initial Hamiltonian using the diagonal Hamiltonian.

2.3   Reduce K-local Diagonal Hamiltonian to 2-local Ising Hamiltonian

Here we present the procedure to reduce the locality of H' from k-local to a 2-local Ising Hamiltonian. For x, y, $z \in \{0, 1\}$ [26], [27]:

$$
\begin{aligned}
xy = z \text{ iff } xy - 2xz - 2yz + 3z = 0 \\
xy \neq z \text{ iff } xy - 2xz - 2yz + 3z > 0
\end{aligned}
\tag{2.14}
$$

Thus, the 3-local $x_1 x_2 x_3$ can be transformed to 2-local:

$$
\begin{aligned}
min(x_1 x_2 x_3) = min(x_4 x_3 + x_1 x_2 - 2x_1 x_4 - 2x_2 x_4 + 3x_4) \quad x_1, x_2, x_3, x_4 \in \{0, 1\} \\
min(-x_1 x_2 x_3) = min(-x_4 x_3 + x_1 x_2 - 2x_1 x_4 - 2x_2 x_4 + 3x_4) \quad x_1, x_2, x_3, x_4 \in \{0, 1\}
\end{aligned}
\tag{2.15}
$$

Theorem 4: $min(k x_1 x_2 x_3 + f(x) = g_1(x)) = min(k x_4 x_3 + |k|(x_1 x_2 - 2x_1 x_4 - 2x_2 x_4 + 3x_4) + f(x) = g_2(x, x_4))$ where k is a non-zero real value and $f(x)$ is polynomial of all variables (including x_1, x_2, x_3 and other variables, excluding x_4). The set of variables x makes $g_1(x)$ minimum will also make $g_2(x, x_4)$ with certain x_4 minimum and vice versa.

Proof: If there exists a set of variables x' makes $g_1(x')$ to be minimum, we can always make $g_2(x', x_4') = g_1(x')$ by choosing $x_4' = x_1' x_2'$ where x_1', x_2' are values of x_1, x_2 in x'.

If there exists a set of variables x'' and x_4'' makes $g_2(x'', x_4'')$ to be minimum, then $g_1(x'') \leq g_2(x'')$:

1. If $x_4'' = x_1'' x_2''$, $g_1(x'') = k x_1'' x_2'' x_3'' + f(x'') = k x_4'' x_3'' + |k|(x_1'' x_2'' - 2 x_1'' x_4'' - 2 x_2'' x_4'' + 3 x_4'') + f(x'') = g_2(x'', x_4)$ where x_1'', x_2'', x_3'' are values of x_1, x_2, x_3 in x'',

2. If $x_4'' \neq x_1'' x_2''$, $g_2(x'', x_4) = k x_4'' x_3'' + |k|(x_1'' x_2'' - 2 x_1'' x_4'' - 2 x_2'' x_4'' + 3 x_4'') + f(x'') \geq = k x_4'' x_3'' + |k| + f(x'') \geq k x_1'' x_2'' x_3'' + f(x'') = g_1(x'')$ where x_1'', x_2'', x_3'' are values of x_1, x_2, x_3 in x'',.
   This is valid because $|k|(x_1'' x_2'' - 2 x_1'' x_4'' - 2 x_2'' x_4'' + 3 x_4'') \geq |k|$ and $k x_4'' x_3'' - k x_1'' x_2'' x_3'' \geq -|k|$.

Thus, we have $g_1(x') = g_2(x', x_4') \geq g_2(x'', x_4'') \geq g_1(x'') \geq g_1(x')$. Thus we have $g_1(x') = g_2(x', x_4') = g_2(x'', x_4'') = g_1(x'')$, or any set of variables x makes $g_1(x)$ minimum will also make $g_2(x, x_4)$ with certain x_4 minimum and vice versa.

Thus, we can obtain the reduction of the locality for the Hamiltonian H' using equation 2.15 to reduce the locality. For example, for $\sigma_z^1 \sigma_z^2 \sigma_z^3$, we can use equation 2.15 to get:

$$min(\frac{1 + \sigma_z^1}{2} \times \frac{1 + \sigma_z^2}{2} \times \frac{1 + \sigma_z^3}{2})$$
$$= min(\frac{1}{4}((1 + \sigma_z^4)(1 + \sigma_z^3) + (1 + \sigma_z^1)(1 + \sigma_z^2) - 2(1 + \sigma_z^1)(1 + \sigma_z^4) - 2(1 + \sigma_z^2)(1 + \sigma_z^4) + 6(1 + \sigma_z^4)))$$
$$= min(\frac{1}{4}(4 - \sigma_z^1 - \sigma_z^2 + \sigma_z^3 + 3\sigma_z^4 + \sigma_z^1 \sigma_z^2 + \sigma_z^3 \sigma_z^4 - 2\sigma_z^1 \sigma_z^4 - 2\sigma_z^2 \sigma_z^4))$$

$$(2.16)$$

Or by Theorem 4 we can get:

$$min(\sigma_z^1 \sigma_z^2 \sigma_z^3)$$
$$= min(8\frac{1 + \sigma_z^1}{2} \times \frac{1 + \sigma_z^2}{2} \times \frac{1 + \sigma_z^3}{2} - 8\frac{1 + \sigma_z^1 + \sigma_z^2 + \sigma_z^3 + \sigma_z^1 \sigma_z^2 + \sigma_z^1 \sigma_z^3 + \sigma_z^2 \sigma_z^3}{8})$$
$$= min(8 \times \frac{1}{4}(4 - \sigma_z^1 - \sigma_z^2 + \sigma_z^3 + 3\sigma_z^4 + \sigma_z^1 \sigma_z^2 + \sigma_z^3 \sigma_z^4 - 2\sigma_z^1 \sigma_z^4 - 2\sigma_z^2 \sigma_z^4)$$
$$- (1 + \sigma_z^1 + \sigma_z^2 + \sigma_z^3 + \sigma_z^1 \sigma_z^2 + \sigma_z^1 \sigma_z^3 + \sigma_z^2 \sigma_z^3))$$
$$= min(7 - 3\sigma_z^1 - 3\sigma_z^2 + \sigma_z^3 + 6\sigma_z^4 + \sigma_z^1 \sigma_z^2 + 2\sigma_z^3 \sigma_z^4 - 4\sigma_z^1 \sigma_z^4 - 4\sigma_z^2 \sigma_z^4 - \sigma_z^1 \sigma_z^3 - \sigma_z^2 \sigma_z^3)$$

$$(2.17)$$

According to Theorem 4, $\{\sigma_1, \sigma_2, \sigma_3\}$ making left term minimum will also make the right term minimum with certain σ_4 and vice versa.

35

Using the same approach for $-\sigma_z^1\sigma_z^2\sigma_z^3$ we can get:

$$
\begin{aligned}
min(&-\sigma_z^1\sigma_z^2\sigma_z^3) \\
&= min(5 - \sigma_z^1 - \sigma_z^2 - \sigma_z^3 + 2\sigma_z^4 + 3\sigma_z^1\sigma_z^2 - 2\sigma_z^3\sigma_z^4 - 4\sigma_z^1\sigma_z^4 - \\
&\quad 4\sigma_z^2\sigma_z^4 + \sigma_z^1\sigma_z^3 + \sigma_z^2\sigma_z^3)
\end{aligned}
\tag{2.18}
$$

According to Theorem 4, $\{\sigma_1, \sigma_2, \sigma_3\}$ making left term minimum will also make the right term minimum with certain σ_4 and vice versa. Furthermore, there are many other implementations to reduce the locality, for example, D-Wave Ocean software [28].

2.4   Numerical Example

Here we present an example of our transformation and use our algorithm to find the achievable minimum expectation value of the mapping to better illustrate the whole transformation. The Hamiltonian describes a simple model of two spin $\frac{1}{2}$ electrons with an exchange coupling constant J in an effective transverse magnetic field of strength B. This simple model has been used to discuss the entanglement for H_2 molecule [29]. The general Hamiltonian for such a system is given by:

$$
H = -\frac{J}{2}(1+\gamma)\sigma_x^1\sigma_x^2 - \frac{J}{2}(1-\gamma)\sigma_y^1\sigma_y^2 - B\sigma_z^1 - B\sigma_z^2
\tag{2.19}
$$

When $r = 2$ the mapping Hamiltonian for each term can be written as:

$$
\begin{aligned}
H'_{(1,1)} &= -B\sigma_z^1 - B\sigma_z^2 \\
H'_{(2,2)} &= -B\sigma_z^3 - B\sigma_z^4 \\
H'_{(1,2)} &= -\frac{J}{2}(1+\gamma)\frac{1-\sigma_z^1\sigma_z^3}{2}\frac{1-\sigma_z^2\sigma_z^4}{2} - \frac{J}{2}(1-\gamma)\frac{\sigma_z^1-\sigma_z^3}{2}\frac{\sigma_z^4-\sigma_z^2}{2} \\
&\quad - B\frac{\sigma_z^1+\sigma_z^3}{2}\frac{1+\sigma_z^2\sigma_z^4}{2} - B\frac{\sigma_z^2+\sigma_z^4}{2}\frac{1+\sigma_z^1\sigma_z^3}{2} \\
H'_{(2,1)} &= H'_{(1,2)}
\end{aligned}
\tag{2.20}
$$
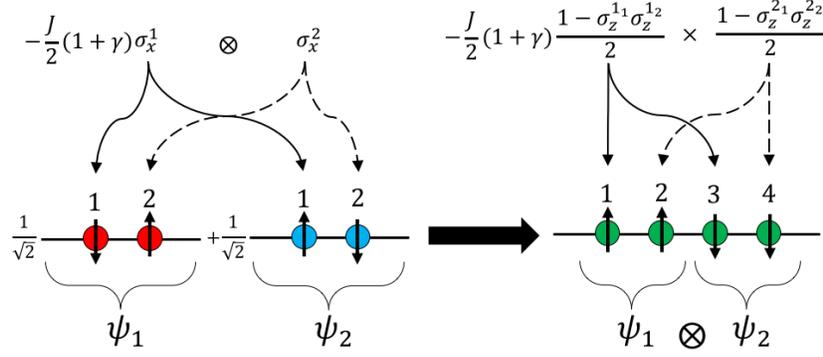
Figure 2.3. The mapped Hamiltonian $-J/2(1+\gamma)\sigma_x^1\sigma_x^2$ between different basis.

Thus, the mapping Hamiltonian H' as well as the C used in the calculation can be written as:

$$H' = \sum_{j,k}^{j,k\leq 2} H'_{(j,k)}S(j)S(k) \tag{2.21}$$

$$
\begin{aligned}
C = (&\frac{1+\sigma_z^1}{2}\frac{1+\sigma_z^2}{2}S(1) + \frac{1+\sigma_z^3}{2}\frac{1+\sigma_z^4}{2}S(2))^2 \\
+ (&\frac{1+\sigma_z^1}{2}\frac{1-\sigma_z^2}{2}S(1) + \frac{1+\sigma_z^3}{2}\frac{1-\sigma_z^4}{2}S(2))^2 \\
+ (&\frac{1-\sigma_z^1}{2}\frac{1+\sigma_z^2}{2}S(1) + \frac{1-\sigma_z^3}{2}\frac{1+\sigma_z^4}{2}S(2))^2 \\
+ (&\frac{1-\sigma_z^1}{2}\frac{1-\sigma_z^2}{2}S(1) + \frac{1-\sigma_z^3}{2}\frac{1-\sigma_z^4}{2}S(2))^2
\end{aligned}
\tag{2.22}
$$

We show the procedure of our algorithm (we set $B = 0.001$, $J = -0.1$ and $\gamma = 0$) with $r = 2$:

1. For $S(1) = S(2) = 1$, first we choose $\lambda = 1000$, we get the minimum eigenvalue of $H'-1000C$ is -4000.008 with $|\Psi\rangle = |0000\rangle$. Thus we get $\sum_m b_m^2 = 4$ then $\lambda_1 = -0.002$.

2. We set $\lambda = -0.002$, we get the minimum eigenvalue of $H'+0.002C$ is 0. We stop here.

3. For $-S(1) = S(2) = 1$, first we choose $\lambda = 1000$, we get the minimum eigenvalue of $H' - 1000C$ is -2000.2 with $|\Psi\rangle = |0110\rangle$. Thus we get $\sum_m b_m^2 = 2$ then $\lambda_1 = -0.1$.

4. We set $\lambda = -0.1$, we get the minimum eigenvalue of $H' + 0.1C$ is 0. We stop here.

5. We get the minimum eigenvalue of H is $min(-0.1, -0.002) = -0.1$.

We also present the result of mapping the above Hamiltonian H with $B = 0.001$, $J = -0.821R^{5/2}e^{-2R}$ [29] and $\gamma = 0$ as in Figure 2.4.
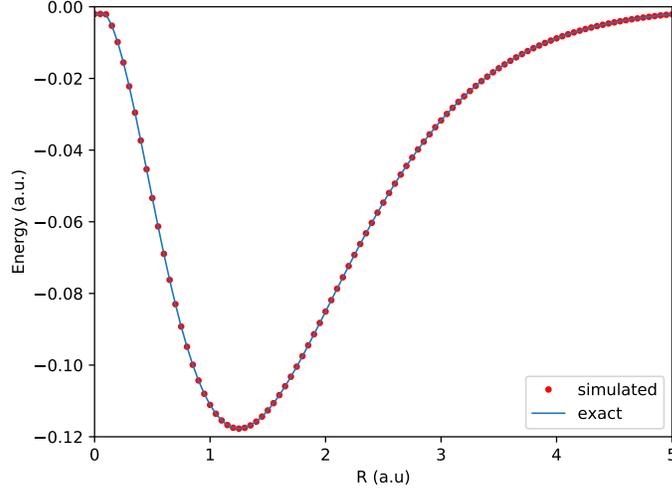
Figure 2.4. Comparing the ground state energy from exact (atomic units) of the original Hamiltonian H, as a function of the internuclear distance R (solid line) with the results of the transformed Hamiltonian H'.

2.5 Mapping H$_2$ Hamiltonian to Ising Hamiltonian

Here, we treat the Hydrogen molecule in a minimal basis STO-6G. we can write the second quantization Hamiltonian of H$_2$:

$$H_{H_2} = h_0 + \sum_{ij} h_{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{ijkl} h_{ijkl} a_i^\dagger a_j^\dagger a_k a_l \qquad (2.23)$$

h_0 is the nuclei repulsion energy. The one and two-electron integrals are giving by:

$$\begin{aligned} h_{ij} &= \int d\vec{r} \chi_i^*(\vec{r})(-\frac{1}{2}\nabla^2 - \sum_A \frac{Z_A}{|\vec{r} - \vec{R}_A|})\chi_j(\vec{r}) \\ h_{ijkl} &= \int d\vec{r_1} d\vec{r_2} \chi_i^*(\vec{r_1})\chi_j^*(\vec{r_2})\frac{1}{r_{12}}\chi_k(\vec{r_2})\chi_l(\vec{r_1}) \end{aligned} \qquad (2.24)$$

where $\chi_i(\vec{r})$, $\chi_j(\vec{r})$, $\chi_k(\vec{r})$, $\chi_l(\vec{r})$ are molecular spin orbital wave functions. Z_A is A_{th} nuclear charge and $\vec{R}_A$ is position of A_{th} nucleus.

By using binary code transformation with checksum code to save to qubits utilizing the spin symmetry, we can rewrite the Hamiltonian as:

$$H_{H_2} = g_0 + g_1\sigma_z^1 + g_2\sigma_z^2 + g_3\sigma_z^1\sigma_z^2 + g_4\sigma_x^1\sigma_x^2 \tag{2.25}$$

By applying the mapping method described above, we can get the Hamiltonian H' consisting of only σ_z (where 1_i and 2_i means the 1 and 2 qubits of i_{th} 2 qubits) as well as C needed for the algorithm:

$$
\begin{aligned}
H' = \sum_i & g_0 + g_1\sigma_z^{1_i} + g_2\sigma_z^{2_i} + g_3\sigma_z^{1_i}\sigma_z^{2_i} \\
+ \sum_{i \neq j} \Big[& g_0 \frac{(1 + \sigma_z^{1_i}\sigma_z^{1_j})(1 + \sigma_z^{2_i}\sigma_z^{2_j})}{4} S(i)S(j) \\
+ & g_1 \frac{(\sigma_z^{1_i} + \sigma_z^{1_j})(1 + \sigma_z^{2_i}\sigma_z^{2_j})}{4} S(i)S(j) \\
+ & g_2 \frac{(\sigma_z^{2_i} + \sigma_z^{2_j}))(1 + \sigma_z^{1_i}\sigma_z^{1_j})}{4} S(i)S(j) \\
+ & g_3 \frac{(\sigma_z^{1_i} + \sigma_z^{1_j})(\sigma_z^{2_i} + \sigma_z^{2_j})}{4} S(i)S(j) \\
+ & g_4 \frac{(1 - \sigma_z^{1_i}\sigma_z^{1_i})(1 - \sigma_z^{2_i}\sigma_z^{2_j})}{4} S(i)S(j) \Big]
\end{aligned}
\tag{2.26}
$$

$$C = \sum_{\pm}\Big(\sum_i \frac{(1 \pm \sigma_z^{1_i})(1 \pm \sigma_z^{2_i})}{4} S(i)\Big)^2 \tag{2.27}$$

For each time when calculating the new Hamiltonian $H' - \lambda C$ we can use the method in 2.3 to reduce the Hamiltonian to an Ising-type Hamiltonian which is able to be implemented on D-Wave quantum computer.

2.6 Results and Discussion

2.6.1 Numerical Results

To illustrate this proposed method, we present the calculations for the Hydrogen molecule H_2, the Helium dimer He_2, HeH^+ diatomic molecule and the LiH molecule. First, we convert

the second quantization Hamiltonian in the minimal basis set (STO-6G) to the spin Hamiltonian of $(\sigma_x, \sigma_y, \sigma_z)$. Then we use our transformed Hamiltonian in $r \times n$ qubits to obtain a diagonal k-local Hamiltonian of σ_z terms and we can get:

$$H' = c + \sum_i h'_i \sigma_z^i + \sum_{ij} h_{ij} \sigma_z^i \sigma_z^j + \sum_{ijk} h_{ijk} \sigma_z^i \sigma_z^j \sigma_z^k + ... \tag{2.28}$$

where c is a constant. Furthermore, we can reduce the locality to get a 2-local Ising Hamiltonian with the constant c' by introducing ancilla qubits:

$$H'' = c' + \sum_i a_i \sigma_z^i + \sum_{ij} a_{ij} \sigma_z^i \sigma_z^j \tag{2.29}$$

We use the binary code transformation [10] implemented in OpenFermion [30] with checksum code to save two qubits by spin symmetry when transforming to the Pauli matrices format. For H_2 the final Pauli matrices format Hamiltonian has 2 qubits and we set $r = 10$. For He_2 the final Pauli matrices format Hamiltonian has 2 qubits and we set $r = 10$. One should note that the Hamiltonian of He_2 after binary code transformation is already an Ising-type Hamiltonian because we are using the minimal basis set. But we still use our transformation on it to show our method also works for Ising-type electronic structure Hamiltonian. For HeH^+ the final Pauli matrices format Hamiltonian has 2 qubits and we set $r = 10$. For LiH, we use the active space approach to assume the two lowest energy spin orbitals are always occupied and first six highest energy spin orbitals are always unoccupied and the final Pauli matrices format Hamiltonian has 2 qubits. We set $r = 10$.

The result presented below is calculated by the algorithm in 2.2.2. The k-local diagonal Hamiltonian and the transformed Ising-type Hamiltonian have same ground state (on original qubits without introduced ancilla qubits) and same ground state energy, which have been proved by section 2.3. Due to the large size of the Ising-type Hamiltonian and limited computer resource, as well as we only need the ground state on original qubits and the ground state energy, the ground state (on original qubits without introduced ancilla qubits) and ground state energy of the Ising-type Hamiltonian are simulated by the direct diagonalization of k-local diagonal Hamiltonian.
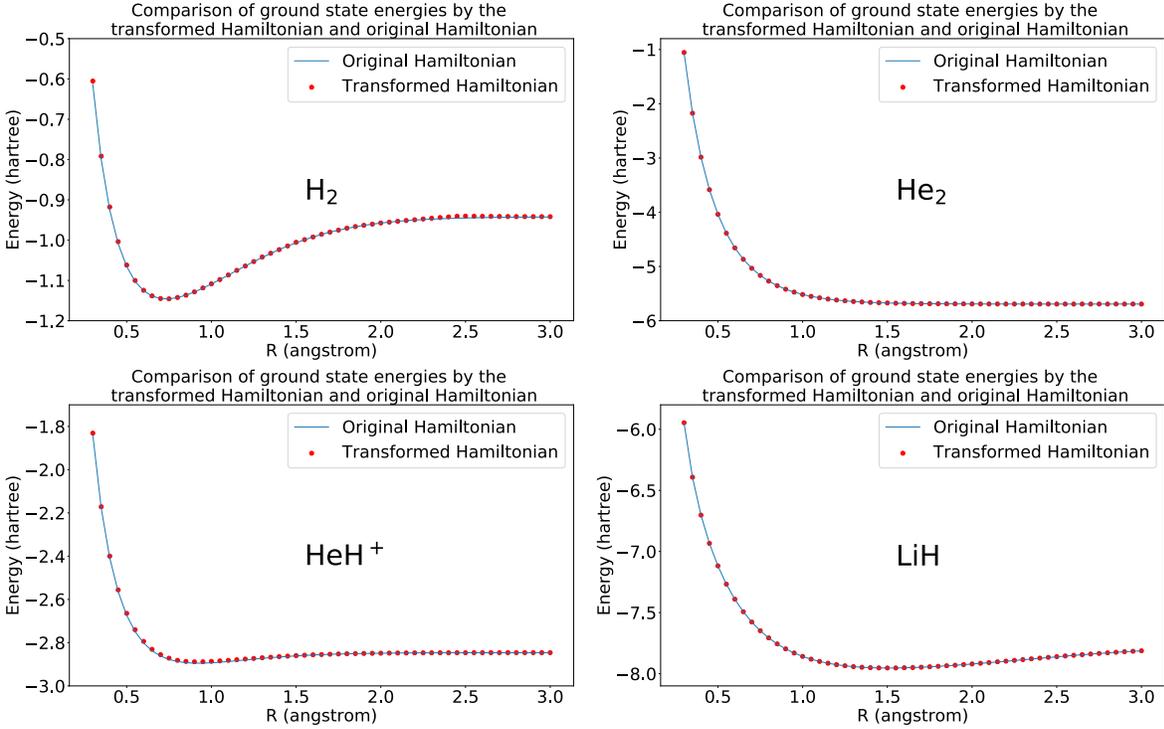
Figure 2.5. Comparing the numerical results of ground state energy obtained by the Ising-type Hamiltonian (k-local diagonal Hamiltonian) with the ground state energies obtained from diagonalization of H_2, He_2, HeH^+ and LiH Hamiltonian as one varies the internuclear distance R.

2.6.2 Discussion

The requirement of qubits can be derived by the several steps of constructing the final Hamiltonian. For example, for a Hamiltonian with n total spin orbitals taken into consideration, transforming Hamiltonian from second quantization to Pauli operators requires $O(n)$ qubits and $O(n^4)$ terms. Mapping Hamiltonian from Pauli matrices format H to k-local diagonal Hamiltonian H' requires $O(rn)$ qubits and $O(2^n r^2 n^4)$ terms. The C in the algorithm requires $O(rn)$ qubits and $O(r^2 2^{2n})$ terms. The both exponential numbers of terms will introduce a large number of ancilla qubits when reduced to 2-local.

Furthermore, we cannot clearly give a relationship between r and wanted accuracy. Same as the variational method which tries to get an optimal result during certain condition, this mapping tries to approach the wanted ground eigenstate by a new state to achieve an optimal result by repeating r times. Thus we cannot calculate the errors between our optimal result

and exact result because we have no idea what is the exact result as the variational method. But by the calculation we can see that by increasing r, we can achieve more accuracy [31]. This is clear because by increasing r we have more space to repeat which will help us to make b_i approach a_i.

The meaning of our method mainly lies in we have successfully mapping a normal Hamiltonian to a k-local diagonal Hamiltonian which is much easier to implement in experiment. Nowadays implementing σ^x and σ^y is impossible in adiabatic quantum computing like D-Wave. Our method gives a possibility of easy implementing Hamiltonian and do require difficult calculation like diagonalizing matrix which is as hard as calculating eigenstates. Moreover, the recent experimental results for simple few electrons diatomic molecules presented by the IBM group have shown that a hardware-efficient optimizer implemented on a 6-qubit superconducting quantum processor is capable of producing the potential energy surfaces of such molecules [13]. The development of efficient quantum hardware and the possibility of mapping the electronic structure problem into an Ising-type Hamiltonian may grant efficient ways to obtain exact solutions to the Schrödinger equation, this being one of the most daunting computational problems present in both chemistry and physics.

3. QUANTUM-ASSISTED RESTRICTED BOLTZMANN FOR ELECTRONIC STRUCTURE CALCULATIONS

Machine learning techniques are demonstrably powerful tools displaying remarkable success in compressing high dimensional data [32], [33]. These methods have been applied to a variety of fields in both science and engineering, from computing excitonic dynamics [34], energy transfer in light-harvesting systems [35], molecular electronic properties [36], surface reaction network [37], learning density functional models [38] to classify phases of matter, and the simulation of classical and complex quantum systems [39]–[45]. Modern machine learning techniques have been used in the state space of complex condensed-matter systems for their abilities to analyze and interpret exponentially large data sets [40] and to speed-up searches for novel energy generation/storage materials [46], [47].[1]

Quantum machine learning [48] - hybridization of classical machine learning techniques with quantum computation – is emerging as a powerful approach allowing quantum speed-ups and improving classical machine learning algorithms [49]–[53]. Recently, Wiebe *et. al.* [54] have shown that quantum computing is capable of reducing the time required to train a restricted Boltzmann machine (RBM), while also providing a richer framework for deep learning than its classical analogue. The standard RBM models the probability of a given configuration of visible and hidden units by the Gibbs distribution with interactions restricted between different layers. Here, we focus on an RBM where the visible and hidden units assume $\{+1, -1\}$ forms [55], [56].

Accurate electronic structure calculations for large systems continue to be a challenging problem in the field of chemistry and material science. Toward this goal — in addition to the impressive progress in developing classical algorithms based on *ab initio* and density functional methods — quantum computing based simulation have been explored [14], [22], [57]–[60]. Recently, Kivlichan *et. al.* [61] show that using a particular arrangement of gates (a fermionic swap network) it is possible to simulate electronic structure Hamiltonian with linear depth and connectivity. These results present significant improvement on the cost

---

[1]Contents in this chapter are reprinted and adapted from Xia, R., & Kais, S. (2018). Quantum machine learning for electronic structure calculations. Nature communications, 9(1), 1-6.

of quantum simulation for both variational and phase estimation based quantum chemistry simulation methods.

Recently, Troyer and coworkers proposed using a restricted Boltzmann machine to solve quantum many-body problems, for both stationary states and time evolution of the quantum Ising and Heisenberg models [55]. However, this simple approach has to be modified for cases where the wave function's phase is required for accurate calculations [56].

In this chapter, we propose a three-layered RBM structure that includes the visible and hidden layers, plus a new layer correction for the signs of coefficients for basis functions of the wave function. We will show that this model has the potential to solve complex quantum many-body problems and to obtain very accurate results for simple molecules as compared with the results calculated by a finite minimal basis set, STO-3G. We also employed a quantum algorithm to help the optimization of training procedure.

3.1   Improved RBM for Electronic Structure Problems

3.1.1   Previous Work of RBM to Solve Quantum Many Body Problems

We will begin by briefly outlining the original RBM structure adapted from [55]. For a given Hamiltonian, H, and a trial state, $|\phi\rangle = \sum_x \phi(x)|x\rangle$, the expectation value can be written as[55]:

$$\langle H \rangle = \frac{\langle \phi | H | \phi \rangle}{\langle \phi | \phi \rangle} = \frac{\sum_{x,x} \langle \phi | x \rangle \langle x | H | x \rangle \langle x | \phi \rangle}{\sum_x \langle \phi | x \rangle \langle x | \phi \rangle} = \frac{\sum_{x,x} \overline{\phi(x)} \langle x | H | x \rangle \phi(x)}{\sum_x |\phi(x)|^2} z \qquad (3.1)$$

where $\phi(x) = \langle x | \phi \rangle$ will be used throughout this letter to express the overlap of the complete wave function with the basis function $|x\rangle$, $\overline{\phi(x)}$ is the complex conjugate of $\phi(x)$.

We can map the above to a RBM model with visible layer units σ_z^1, σ_z^2... σ_z^n and hidden layer units h_1, h_2... h_m with σ_z^i, $h_j \in \{-1, 1\}$. We use the visible unit σ_z^i to represent the spin state of the qubit i – if $\sigma_z^i = -1$ it represents qubit i in $|0\rangle$ and if $\sigma_z^i = 1$ it represents qubit i in $|1\rangle$. The total spin state of n qubits is represented by the basis $|x\rangle = |\sigma_z^1 \sigma_z^2 ... \sigma_z^n\rangle$.

$\phi(x) = \sqrt{P(x)}$ where $P(x)$ is the probability for x from the distribution determined by the RBM. The probability of a specific set $x = \{\sigma_z^1, \sigma_z^2...\sigma_z^n\}$ is:

$$P(x) = \frac{\sum_{\{h\}} e^{(\sum_i a_i \sigma_z^i + \sum_j b_j h_j + \sum_{i,j} w_{ij} \sigma_z^i h_j)}}{\sum_x \sum_{\{h\}} e^{(\sum_i a_i \sigma_z^i + \sum_j b_j h_j + \sum_{i,j} w_{ij} \sigma_z^i h_j)}} \tag{3.2}$$

Within the above a_i and b_j are trainable weights for units σ_z^i and h_j. w_{ij} are trainable weights describing the connections between σ_z^i and h_j (see Figure 3.1)

By setting $\langle H \rangle$ as the objective function of this RBM, we can use the standard gradient decent method to update parameters, effectively minimizing $\langle H \rangle$ to obtain the ground state energy.

3.1.2 Improved RBM Structure for Electronic Structure Problem

However, previous prescriptions considering the use of RBMs for electronic structure problems have found difficulty as $\phi(x_i)$ can only be non-negative values. We have thus appended an additional layer to the neural network architecture to compensate for the lack of sign features specific to electronic structure problems.

We propose an RBM with three layers. The first layer, σ_z, describes the parameters building the wave function. The h's within the second layer are parameters for the coefficients for the wave functions and the third layer s, represents the signs associated $|x\rangle$:

$$s(x) = s(\sigma_z^1, \sigma_z^2...\sigma_z^n) = tanh(\sum_i d_i \sigma_z^i + c) \tag{3.3}$$

The s uses a non-linear function $tanh$ to classify whether the sign should be positive or negative. Because we have added another function for the coefficients, the distribution is not solely decided by RBM. We also need to add our sign function into the distribution. Within this scheme, c is a regulation and d_i are weights for σ_z^i. (see Figure 3.1). Our final objective function, now with $|\phi\rangle = \sum_x \phi(x)s(x)|x\rangle$, becomes:

$$\langle H \rangle = \frac{\sum_{x,x} \overline{\phi(x)s(x)} \langle x|H|x\rangle \phi(x)s(x)}{\sum_x |\phi(x)s(x)|^2} \tag{3.4}$$

45
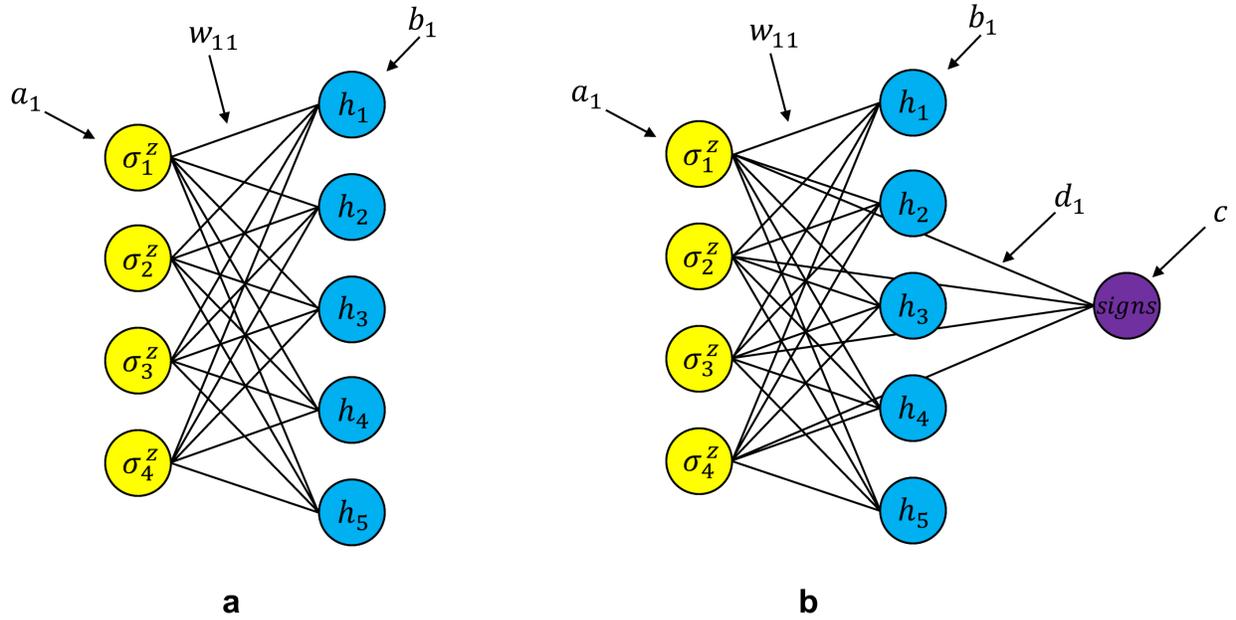
Figure 3.1. (a) Original RBM for quantum many-body problems. (b) Our improved RBM for electronic structure calculations.

After setting the objective function, the learning procedure is performed by sampling to get the distribution of $\phi(x)$ and calculating to get $s(x)$. We then proceed to calculate the joint distribution determined by $\phi(x)$ and $s(x)$. The gradients are determined by the joint distribution and we use gradient decent method to optimize $\langle H \rangle$. Calculating the the joint distribution is efficient because $s(x)$ is only related to x.

3.1.3 Calculating Gradients for Improved RBM Structure

For an electronic structure Hamiltonian prepared by second quantization and transformed to Pauli matrices format, H, and a trial wave function, $|\phi\rangle = \sum_x \phi(x)s(x)|x\rangle$, the expectation value can be written as[62]:

$$\langle H \rangle = \frac{\langle \phi | H | \phi \rangle}{\langle \phi | \phi \rangle} = \frac{\sum_{x,x} \overline{\phi(x)s(x)} \langle x | H | x \rangle \phi(x)s(x)}{\sum_x |\phi(x)s(x)|^2} \tag{3.5}$$

x is a combination of $\{\sigma_z^1, \sigma_z^2...\sigma_z^n\}$ and $|x\rangle = |\sigma_z^1\sigma_z^2...\sigma_z^n\rangle$. If we set $\Phi(x) = \phi(x)s(x)$, because $\phi(x)$ and $s(x)$ are all real value functions, then the gradient can be calculated as[55], [62]:

$$\partial_{p_k}\langle H\rangle = \frac{\sum_x(\partial_{p_k}\Phi(x))\langle x|H|\phi\rangle + \sum_x\langle\phi|H|x\rangle(\partial_{p_k}\Phi(x))}{\sum_x|\Phi(x)|^2}$$
$$- \frac{\sum_x\Phi(x)\langle x|H|\phi\rangle}{\sum_x|\Phi(x)|^2}\frac{\sum_x((\partial_{p_k}\Phi(x))\Phi(x) + \Phi(x)\partial_{p_k}\Phi(x))}{\sum_x|\Phi(x)|^2} \quad (3.6)$$

If we set $E_{loc}(x) = \frac{\langle x|H|\phi\rangle}{\Phi(x)}$ and $D_{p_k}(x) = \frac{\partial_{p_k}\Phi(x)}{\Phi(x)}$, the gradient can be written as[62]:

$$\partial_{p_k}\langle H\rangle = \frac{\sum_x D_{p_k}(x)E_{loc}(x)|\Phi(x)|^2 + \sum_x E_{loc}(x)D_{p_k}(x)|\Phi(x)|^2}{\sum_x|\Phi(x)|^2}$$
$$- \frac{\sum_x|\Phi(x)|^2E_{loc}(x)}{\sum_x|\Phi(x)|^2}\frac{\sum_x(D_{p_k}(x) + D_{p_k}(x))|\Phi(x)|^2}{\sum_x|\Phi(x)|^2} \quad (3.7)$$
$$= 2\langle E_{loc}D_{p_k}\rangle - 2\langle E_{loc}\rangle\langle D_{p_k}\rangle$$

where $\langle...\rangle$ represent the expectation value of distribution determined by $\Phi(x)$. $\langle x|H|\phi\rangle = \langle\phi|H|x\rangle$ for that H is a real symmetric matrix because we used real-valued STO-3G basis set as well as the real-valued transformation to Pauli matrices format.

p_k is the parameters a_i, b_j, w_{ij}, d_i, c for k_{th} iterations. Thus we have[62]:

$$D_{a_i}(x) = \frac{1}{2}\sigma_z^i - \frac{1}{2}\langle\sigma_z^i\rangle_{RBM},$$
$$D_{b_j}(x) = \frac{1}{2}tanh(\theta_j) - \frac{1}{2}\langle h_j\rangle_{RBM},$$
$$D_{w_{ij}}(x) = \frac{1}{2}tanh(\theta_j)\sigma_z^i - \frac{1}{2}\langle\sigma_z^i h_j\rangle_{RBM}, \quad (3.8)$$
$$D_c(x) = 1/s(x) - s(x),$$
$$D_{d_i}(x) = \sigma_z^i(1/s(x) - s(x)),$$

where $\theta_j = \sum_i w_{ij}\sigma_z^i + b_j$. $\langle...\rangle_{RBM}$ represents the distribution determined solely by RBM. We do not need to calculate the second term of D_{a_i}, D_{b_j} and $D_{w_{ij}}$ for that they will be cancelled when calculating the gradient $\partial_{p_k}\langle H\rangle$. We use the gradient decent method to optimize our RBM, yielding the minimum corresponding to the ground energy:

$$p_{k+1} = p_k - \alpha_k\partial_{p_k}\langle H\rangle \quad (3.9)$$

47

where α_k is the learning rate for k_{th} iteration, controlling the convergence rate. We can continue iterating until we reach the maximum number of iterations. The gradient is estimated by the distribution calculated by sampling.

3.1.4 Quantum Algorithm to Train RBM

We propose a quantum algorithm to sample the distribution determined by RBM. The probability for each combination $y = \{\sigma_z, \ h\}$ can be written as:

$$P(y) = \frac{e^{\sum_i a_i \sigma_z^i + \sum_j b_j h_j + \sum_{i,j} w_{ij} \sigma_z^i h_j}}{\sum_y e^{\sum_i a_i \sigma_z^i + \sum_j b_j h_j + \sum_{i,j} w_{ij} \sigma_z^i h_j}} \quad (3.10)$$

Instead of $P(y)$, we try to sample the distribution $Q(y)$ as:

$$Q(y) = \frac{e^{\frac{1}{k}(\sum_i a_i \sigma_z^i + \sum_j b_j h_j + \sum_{i,j} w_{ij} \sigma_z^i h_j)}}{\sum_y e^{\frac{1}{k}(\sum_i a_i \sigma_z^i + \sum_j b_j h_j + \sum_{i,j} w_{ij} \sigma_z^i h_j)}} \quad (3.11)$$

where k is an adjustable constant with different values for each iteration and is chosen to increase the probability of successful sampling. In our simulation, it is chosen as $O(\sum_{i,j} |w_{ij}|)$.

We employed a quantum algorithm to sample the Gibbs distribution from the quantum computer. This algorithm is based on sequential applications of controlled-rotation operations, which used an ancilla qubit showing whether the sampling for $Q(y)$ is successful[54].

This two-step algorithm uses one system register (with $n + m$ qubits in use) and one scratchpad register (with one qubit in use) as shown in Figure 3.2.

All qubits are initialized as $|0\rangle$ at the beginning. The first step is to use R_y gates to get a superposition of all combinations of $\{\sigma_z, h\}$ with $\theta_i = 2arcsin(\sqrt{\frac{e^{a_i/k}}{e^{a_i/k} + e^{-a_i/k}}})$ and $\gamma_j = 2arcsin(\sqrt{\frac{e^{b_j/k}}{e^{b_j/k} + e^{-b_j/k}}})$:

$$\otimes_i R_y(\theta_i)|0_i\rangle \otimes_j R_y(\gamma_j)|0_j\rangle|0\rangle = \sum_y \sqrt{O(y)}|y\rangle|0\rangle$$

where $O(y) = \frac{e^{\sum_i a_i \sigma_z^i/k + \sum_j b_j h_j/k}}{\sum_y e^{\sum_i a_i \sigma_z^i/k + \sum_j b_j h_j/k}}$ and $|\phi_y\rangle$ corresponds to the combination $|y\rangle = |\sigma_z^1..\sigma_z^n h_1...h_m\rangle$.

The second step is to calculate $e^{w_{ij}\sigma_z^i h_j}$. We use controlled-rotation gates to achieve this. The idea of sequential controlled-rotation gates is to check whether the target qubit is in state $|0\rangle$ or state $|1\rangle$ and then rotate the corresponding angle (Figure 3.2). If qubits σ_z^i and h_j are in $|00\rangle$ or $|11\rangle$, the ancilla qubit is rotated by $R_y(\theta_{ij,1})$ and otherwise by $R_y(\theta_{ij,2})$, with $\theta_{ij,1} = 2arcsin(\sqrt{\frac{e^{w_{ij}/k}}{e^{|w_{ij}|/k}}})$ and $\theta_{ij,2} = 2arcsin(\sqrt{\frac{e^{-w_{ij}/k}}{e^{|w_{ij}|/k}}})$. Each time after one $e^{w_{ij}\sigma_z^i h_j}$ is calculated, we do a measurement on the ancilla qubit. If it is in $|1\rangle$ we continue with a new ancilla qubit initialized in $|0\rangle$, otherwise we start over from the beginning.

All controlled rotation gates can be expressed as below:

$$C^2 R_{w_{ij},1} = C_{\sigma_z^i,h_j} \otimes R_y(2arcsin(\sqrt{e^{w_{ij}/k}e^{-|w_{ij}|/k}})) + (D_{\sigma_z^i,h_j} + E_{\sigma_z^i,h_j} + F_{\sigma_z^i,h_j}) \otimes I$$
$$C^2 R_{w_{ij},2} = D_{\sigma_z^i,h_j} \otimes R_y(2arcsin(\sqrt{e^{-w_{ij}/k}e^{-|w_{ij}|/k}})) + (C_{\sigma_z^i,h_j} + E_{\sigma_z^i,h_j} + F_{\sigma_z^i,h_j}) \otimes I$$
$$C^2 R_{w_{ij},3} = E_{\sigma_z^i,h_j} \otimes R_y(2arcsin(\sqrt{e^{-w_{ij}/k}e^{-|w_{ij}|/k}})) + (C_{\sigma_z^i,h_j} + D_{\sigma_z^i,h_j} + F_{\sigma_z^i,h_j}) \otimes I$$
$$C^2 R_{w_{ij},4} = F_{\sigma_z^i,h_j} \otimes R_y(2arcsin(\sqrt{e^{w_{ij}/k}e^{-|w_{ij}|/k}})) + (C_{\sigma_z^i,h_j} + D_{\sigma_z^i,h_j} + E_{\sigma_z^i,h_j}) \otimes I$$

$$(3.12)$$

where $C_{\sigma_z^i,h_j} = B_{\sigma_z^i} \otimes B_{h_j}$, $D_{\sigma_z^i,h_j} = A_{\sigma_z^i} \otimes B_{h_j}$, $E_{\sigma_z^i,h_j} = B_{\sigma_z^i} \otimes A_{h_j}$, $F_{\sigma_z^i,h_j} = A_{\sigma_z^i} \otimes A_{h_j}$ and

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Between the calculation of two w_{ij}, we need to do a measurement on the ancilla qubit to make sure the state of system qubits collapse to the wanted state. Measuring ancilla qubit in $|1\rangle$ means the state of system qubits collapse to the wanted state as we initialize the ancilla qubit in $|0\rangle$.

We then do the measurement, if and only if the ancilla qubit is in $|1\rangle$ we continue with a new ancilla qubit initialized in $|0\rangle$, otherwise we start from beginning. The probability of success is very large since we choose k as a large number.

After we finish all measurements, the distribution should be $Q(y)$. We just measure the first $n+m$ qubits of the system register to obtain the probability distribution. After we get the distribution, we calculate all probabilities to the power of k and normalize to get the Gibbs distribution. The whole procedure can be seen from the flow chart Figure 3.3.
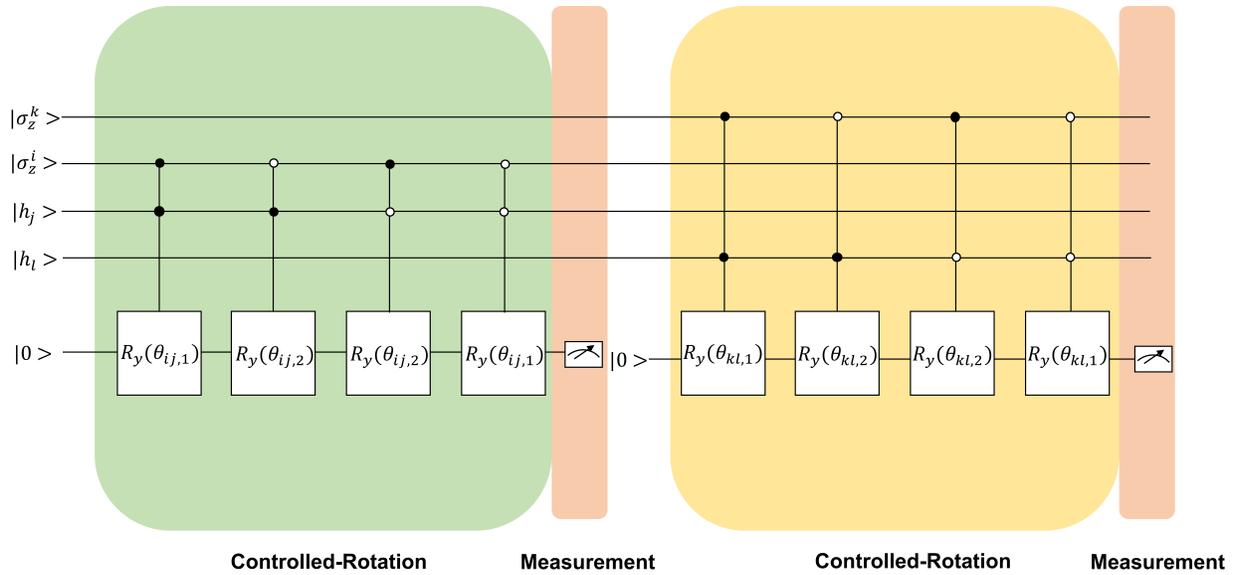
Figure 3.2. The example circuit for the controlled-rotation gate approach with measurements.
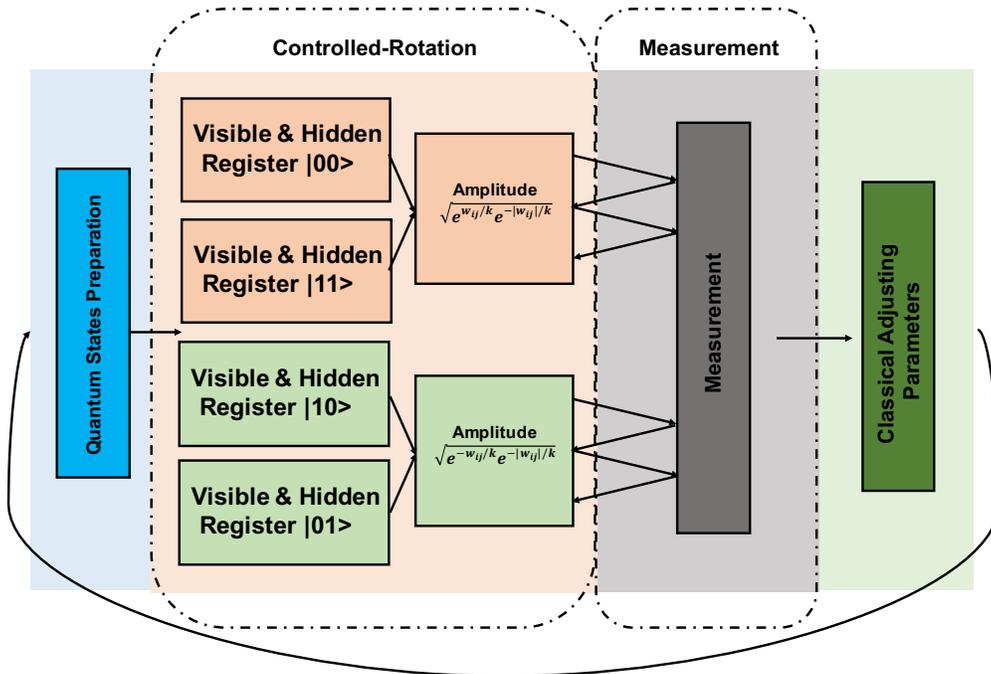
Figure 3.3. The algorithmic flow chart of the quantum algorithm based on sequential controlled-rotations gates.

3.1.5 Complexity and Successful Probability of the Algorithm

The successful probability P can be written as:

$$P = \frac{\sum_{\sigma_z, h} e^{\frac{1}{k}(\sum_i a_i \sigma_z^i + \sum_j b_j h_j + \sum_{i,j} w_{ij} \sigma_z^i h_j)}}{\sum_{\sigma_z, h} e^{\frac{1}{k}(\sum_i a_i \sigma_z^i + \sum_j b_j h_j)} e^{\frac{1}{k}(\sum_{i,j} |w_{ij}|)}} \geq \frac{e^{\frac{-1}{k}(\sum_{i,j} |w_{ij}|)}}{e^{\frac{1}{k}(\sum_{i,j} |w_{ij}|)}} = \frac{1}{e^{\frac{1}{k}(\sum_{i,j} 2|w_{ij}|)}} \qquad (3.13)$$

If we choose $k = 1$, we are sampling the targeted distribution determined by RBM. However, the probability of successful sampling is $\frac{1}{e^{\frac{1}{k}(\sum_{i,j} 2|w_{ij}|)}}$ which is exponentially small. To solve this problem we choose $k = O(\sum_{i,j} |w_{ij}|)$, we have $P \geq \frac{1}{e^{O(1)}}$ which means the lower bound of probability of successful sampling is a constant. In the simulation, we choose $k = max(\frac{1}{2}\sum_{i,j}|w_{ij}|, 1)$ because larger k introduces larger sampling errors. This particular choice of k gives us lower bound of success as e^{-4}.

For a controlled-controlled-U gate or $C^2(U)$ conditioned by $|11\rangle$ to apply U on the target qubit, it can be decomposed as the below Figure 3.4 [2]:
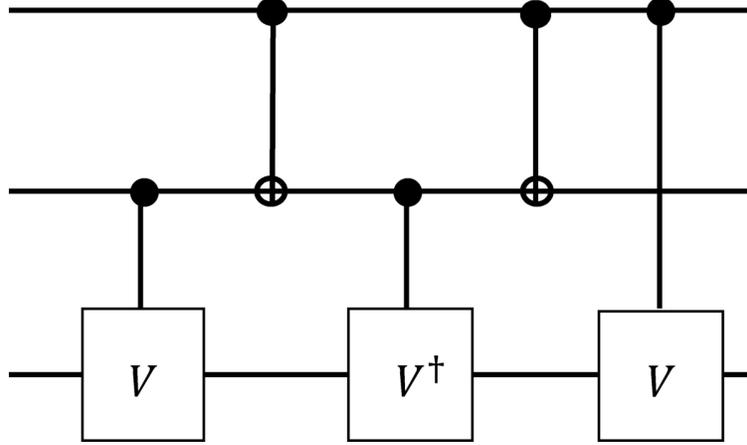
Figure 3.4. The decomposition of the $C^2(U)$ gate. $V^2 = U$.

In our algorithm, $U = R_y(\theta)$, thus we can choose $V = R_y(\theta/2)$ to achieve the decomposition. $C^2(U)$ conditioned by $|00\rangle$, $|10\rangle$ or $|01\rangle$ can be achieved by adding X gates on controlling qubits. For each w_{ij}, we have 4 $C^2(U)$ which means the gates complexity scales to $O(mn)$ and the number of qubits for our algorithms scales to $O(mn)$, which can be reduced to $O(m+n)$ if considering qubit reuse. Because the lower bound of probability of successful sampling is constant, if the number of successful sampling is N_s, the complexity for each

iteration is $O(N_s mn)$. The only error comes from the error of sampling if not considering noise in the quantum computer.

3.2 Results and Discussion

3.2.1 Numerical Results

In the numerical simulation, we treat H_2 molecule with 2-electrons in a minimal basis STO-3G and use the Jordan-Wigner transformation[5]. The final Hamiltonian is of 4 qubits. We treat LiH molecule with 4-electrons in a minimal basis STO-3G and use the Jordan-Wigner transformation[5]. We assumed the first two lowest energy spin orbitals are occupied by electrons and first six highest energy spin orbitals are never occupied and the the final Hamiltonian is of 4 qubits. We treat H_2O molecule with 10-electrons in a minimal basis STO-3G. We assume the first four lowest energy orbitals are occupied by electrons and first two highest energy orbitals are never occupied all time. We also use the spin symmetry in [10], [13], [63] to reduce another two qubits by using the binary code transformation with checksum code [10]. With the reduction of the number of qubits, finally we have 6 qubits Hamiltonian [64], [65]. All calculations of integrals in second quantization and transformations of electronic structure are done by OpenFermion[30] and Psi4[66].

We now present the results derived from our RBM for H_2, LiH and H_2O molecules. It can clearly be seen from Figure 3.5 that our three layer RBM yields very accurate results comparing to the disorganization of transformed Hamiltonian which is calculated by a finite minimal basis set, STO-3G. Points deviating from the ideal curve are likely due to local minima trapping during the optimization procedure. This can be avoided in the future by implementing optimization methods which include momentum or excitation, increasing the escape probability from any local features of the potential energy surface.

Further discussion about our results should mention instances of transfer learning. Transfer learning is a unique facet of neural network machine learning algorithms describing an instance (engineered or otherwise) where the solution to a problem can inform or assist in the solution to another similar subsequent problem. Given a diatomic Hamiltonian at a specific intermolecular separation, the solution yielding the variational parameters — which

are the weighting coefficients of the basis functions — are adequate first approximations to those parameters at a subsequent calculation where the intermolecular separation is a small perturbation to the previous value.

In the transfer learning as in Figure 3.5 (d), we proceed from the last point to the first point, and use the parameters from the calculations of the previous point. Except for the last point in the Figure 3.5 (d), we use 1/40 of the iterations for the last point in the calculations initiated with transferred parameters from previous iterations of each points and still achieve a good result. We also see that the local minimum is avoided if the starting point achieve a good minimum.

In simulation we use $k = max(1, \frac{1}{2}\sum_{i,j}|w_{ij}|)$ to make the probability becomes $O(\frac{1}{e^4})$, which needs constant number of measurements to get enough successful sampling. After we get the distribution, we need to calculate all distribution to the power of k and normalize to get the wanted distribution. k is a large number, which is around 5 at final for H_2, LiH and H_2O in our simulation. In the simulation, the only error comes from the statistical errors by the sampling. When k is large, calculating the power of k introduces large errors in the simulation and causes fluctuations in final steps of optimization. Furthermore, there are many different errors and noise on current quantum devices, for example, gate errors, measurement errors and statistical errors. Calculating power of k will introduce large errors if k is large. To decrease errors when calculating power of k, some techniques have to be introduced to decrease errors, for example, error mitigation on quantum computer and increasing number of sampling. Because we investigated small molecule system, k is not very large and the quantum algorithm is efficient. For large k, our quantum algorithm may require other error correction techniques and large number of sampling, which may not be efficient.
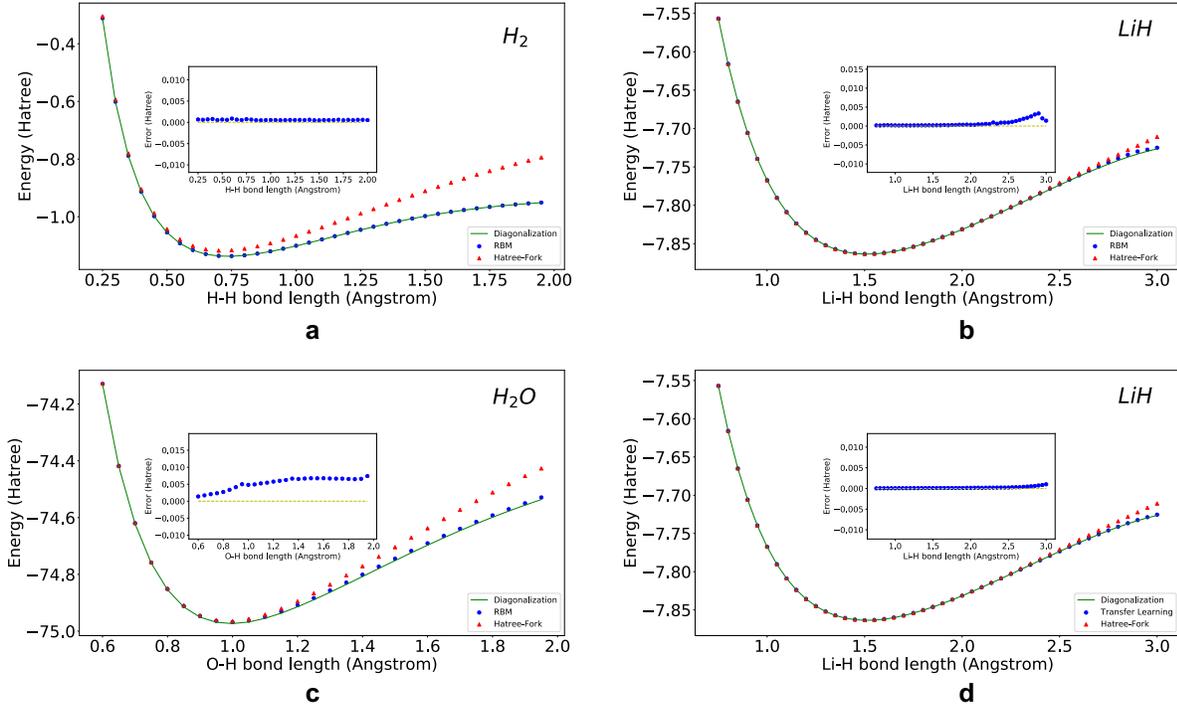
Figure 3.5. Results of calculating ground state energy of H_2, LiH and H_2O. **a**, **b**, **c** are the results of H_2 ($n = 4$, $m = 8$), LiH ($n = 4$, $m = 8$) and H_2O ($n = 6$, $m = 6$) calculated by our three layer RBM compared with exact diagonalized results of the transformed Hamiltonian. **d** is the result of LiH ($n = 4$, $m = 8$) calculated by the Transfer Learning method. We use STO-3G as basis to compute the molecular integrals for the Hamiltonian. Bond length represents inter-atomic distance for the diatomic molecules and the distance O-H of the water molecule with fixed bond angle. The data points of RBM are minimum energies of all energies calculated during the whole optimization. In the simulation, the distribution of RBM by the quantum algorithm is obtained by sampling.

3.2.2 Discussion

In the current simulation, H_2 requires 13 qubits with the number of visible units $n = 4$, the number of hidden units $m = 8$ and additional 1 reusing ancilla qubits . LiH requires 13 qubits with the number of visible units $n = 4$, the number of hidden units $m = 8$ and additional 1 reusing ancilla qubits. H_2O requires 13 qubits with the number of visible units $n = 6$, the number of hidden units $m = 6$ and additional 1 reusing ancilla qubits. The order of scaling of qubits for the system should be $O(m + n)$ with reusing ancilla qubits. The

number of visible units n is equal to the number of spin orbitals. The choice of the number of hidden units m is normally constant integer times of n which gives us a scaling of $O(n)$ with reusing ancilla qubits . Thus, the scaling of the qubits increases polynomially with the number of spin orbitals. Also, the complexity of gates $O(n^2)$ scales polynomially with the number of spin orbitals.

In conclusion, we present a combined quantum machine learning approach to perform electronic structure calculations. Here, we have a proof of concept and show results for small molecular systems. Screening molecules to accelerate the discovery of new materials for specific application is demanding since the chemical space is very large! For example, it was reported that the total number of possible small organic molecules that populate the 'chemical space' exceed 10^{60}[67], [68]. Such an enormous size makes a thorough exploration of chemical space using the traditional electronic structure methods impossible. Moreover, in a recent perspective[69]in Nature Reviews Materials the potential of machine learning algorithms to accelerate the discovery of materials was pointed out. Machine learning algorithms have been used for material screening. For example, out of the GDB-17 data base, consisting of about 166 billion molecular graphs, one can make organic and drug-like molecules with up to 17 atoms and 134 thousand smallest molecules with up to 9 heavy atoms were calculated using hybrid density functional (B3LYP/6-31G(2df,p). Machine learning algorithms trained on these data, were found to predict molecular properties of subsets of these molecules [70]–[72].

With the rapid development of larger-scale quantum computers and the possible training of some machine units with the simple dimensional scaling results for electronic structure, quantum machine learning techniques are set to become powerful tools to perform electronic structure calculations and assist in designing new materials for specific applications.

4. HYBRID QUANTUM-CLASSICAL NEURAL NETWORK FOR CALCULATING GROUND STATE ENERGIES OF MOLECULES

Quantum computing has shown its great potential in advancing quantum chemistry research [73]. Many quantum algorithms have been proposed to solve quantum chemistry problems [64], [65], [74], such as Phase Estimation Algorithm [6], [58], [59], [75] to calculate eigenstate energies of simple molecules, the Variational Quantum Eigensolver (VQE) [13], [14], [76] to solve electronic structure problems and quantum algorithms for open quantum dynamics [77]. Using quantum computing techniques to perform machine learning tasks [48] has also received much attention recently including quantum data classification [49], [50], quantum generative learning [78], [79], and quantum neural network approximating non-linear functions [80]. So far, applying the various quantum machine learning techniques to quantum chemistry is a natural extension [81], [82]. However, previous studies focused solely on quantum circuits with only a few non-linear operations, which are introduced by data encoding [80], [83] or repeated measurements until success [84]. Moreover, a recent research shows increasing layers of parameterized quantum circuit (PQC) would reach saturation and may not improve the performance when the number of layers is large enough. FurthermoreThe non-linearity is the most important part for the classical neural network[85] which makes neural networks able to produce complex results[32], [84], [86]. Therefore, a non-linear operation is needed for the quantum neural network.[1]

To solve this problem, here we introduce a new hybrid quantum classical neural network, by combining quantum computing and classical computing with measurements between the parameterized quantum circuits. In this paper, we first give a detailed description of the whole structure of the hybrid quantum classical neural network. We then present numerical simulations by using the new hybrid quantum classical neural network to calculate ground state energies of different molecular systems. The calculated ground state energies are very

---

[1]Contents in this chapter are reprinted and adapted from Xia, R., & Kais, S. (2020). Hybrid Quantum-Classical Neural Network for Calculating Ground State Energies of Molecules. Entropy, 22(8), 828.

accurate, which demonstrate the potential of the proposed hybrid quantum classical neural network to generate potential energy surfaces.

We propose a new structure of quantum-classical hybrid neural network by connecting the quantum part (quantum layer) with the classical part (classical layer). For a classical neural network, each artificial neuron is normally constructed by linear connected layers, with nonlinear activation functions connected at the end, as shown in the left part of Figure 4.1. In this work, we replace the linear part by the quantum circuit as shown in the right side of Figure 4.1 to take advantage of possible speedup in quantum computation. In the meantime, we use expectation values of operators by measurements, which are nonlinear operations, to serve as the activation function. In this neural network set-up, the quantum circuit can be viewed as the quantum layer and the expectation values by measurements can be viewed as the classical layer. The input data is first encoded into quantum states and calculated by the quantum layer. The outputs are extracted as the expectation values by measurements. The two steps can be repeated several times to construct a hybrid multi-layer neural network. In our construction, the quantum layer is enabled by parameterized quantum circuits (PQC) [87]. We will give details about the hybrid quantum-classical neural network in the following sections. Note, in this chapter, the qubits are indexed starting from 0 instead of 1 to be consistent with indexes of simulation software we are using.

4.1 Quantum Layer

The quantum layer is enabled by a parameterized quantum circuit consisting of parameterized quantum gates, which allows the PQC to be optimized by adjusting the parameters to approximate wanted results. PQC has been widely used in many areas of quantum computing and quantum machine learning, such as in VQE [13], [14], [76], quantum autoencoder [81], and quantum generative learning [78]. In the following section, we will provide details of the quantum layer including encoding classical data into quantum circuits and parameterized quantum circuits.
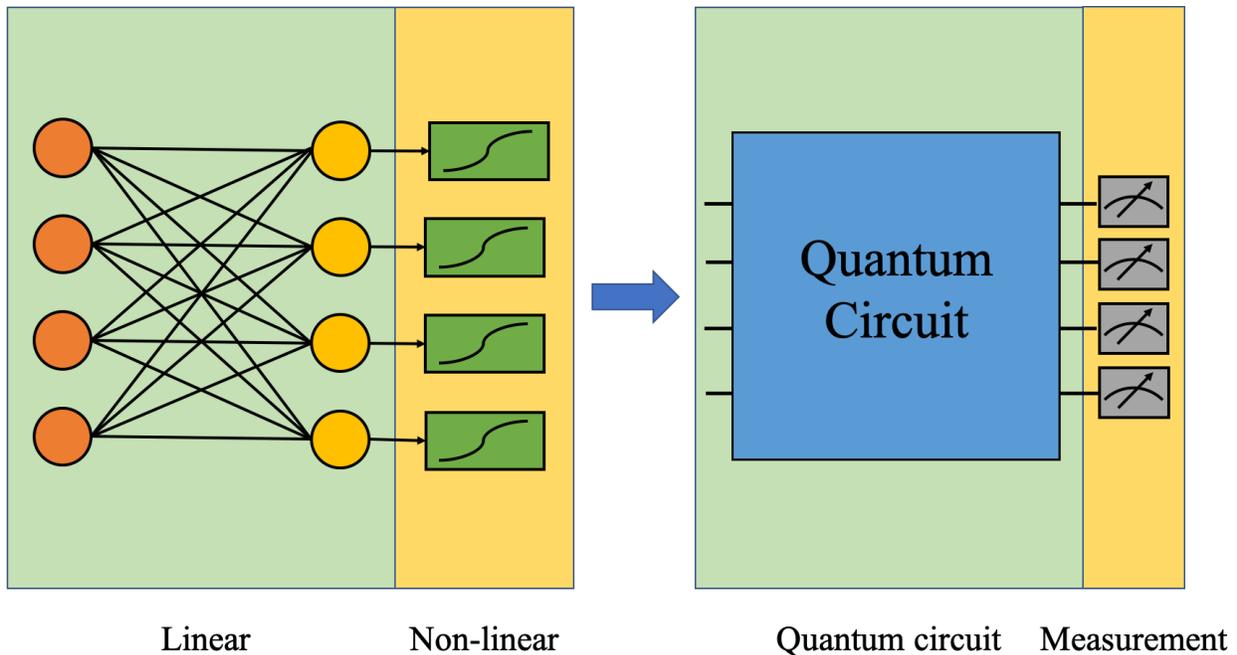
Figure 4.1. In the proposed quantum-classical hybrid neural network, the linear part in the classical neural network is replaced by the quantum circuits and the nonlinear part is replaced by measurements.

4.1.1 Data Encoding

To implement the quantum layer, the first step is to encode the input classical data into a quantum state. Variational encoding [83] has been proposed to reduce the depth of quantum circuits and has been widely used in many quantum machine learning techniques [80], [83], [88], [89]. Variational encoding is used to prepare a set of quantum gates with parameters generated by the input data and then initialize the state from the basic state with all qubits as $|0\rangle$ with these gates. For an array of data $\{a_0, a_1, ...a_{n-1}\}$, an example of variational encoding to encode n qubits is to prepare the gate G as:

$$G = \otimes_{i=0}^{n-1} g_i(f_i(a_i)) \tag{4.1}$$

where g_i is a set of single qubit quantum gates on qubits i and f_i is a classical function to encode a_i as the parameter of g_i. The encoded state would be $G|0\rangle^{\otimes n}$. One simple example is given in our numerical simulations: we take the bond length, a, as the encoding

58

data for each qubit. We choose f_i as the identity function and g_i as $R_y H$, where R_y is the rotation-y gate and H is the Hadamard gate. Thus, the encoded quantum state would be $(\otimes_{i=0}^{n-1} R_y(a)H)|0\rangle^{\otimes n}$.

In most variational encoding the depth of the circuit needed to encode the data would be $O(1)$ [88] for that the number of quantum gates to initialize the quantum state is fixed, which makes variational encoding more suitable for Noisy Intermediate-Scale Quantum (NISQ) devices [11]. Furthermore, recently it has been shown how the variational encoding may help to introduce nonlinearity features in quantum circuits [83], [90]. Variational encoding can only be implemented at the beginning of the quantum circuit, but connections between multiple PQC also need to be nonlinear. To enable nonlinear connections, we introduce measurements as connections between multiple PQC. In the numerical simulations, we will be using the variational encoding to perform the simulation and discuss implementing the quantum circuits on NISQ device.

4.1.2 Parameterized Quantum Circuit

A parameterized quantum circuit, also known as a variational quantum circuit [76], [87], is a quantum circuit consisting of parameterized gates with fixed depth. This is the main part of the quantum layer to perform the calculation. The parameterized quantum circuit consists of one-qubit gates as well as $CNOT$. Some more complicated gates may also be used in PQC which can be decomposed into one qubit gates and $CNOT$ [2]. In general, an n qubits PQC can be written as:

$$U(\vec{\theta})|\psi\rangle = (\prod_{i=1}^{m} U_i)|\psi\rangle \qquad (4.2)$$

where $U(\vec{\theta})$ is the set of universal gates and m is the number of quantum gates. $\vec{\theta}$ is the set of parameters $\{\theta_0, \theta_1 .... \theta_{k-1}\}$, where k is the total number of parameters and $|\psi\rangle$ is the encoded quantum state after data encoding. For each unitary gate U_i, it may be a quantum gate which does not require parameters or a quantum gate which takes parameters. Examples of

the unitary gate taking parameters are rotational gates, $R_x(\theta)$, $R_y(\theta)$, and $R_z(\theta)$, which are given by:

$$R_x(\theta) = e^{-i\frac{\theta}{2}\sigma_x} \qquad R_y(\theta) = e^{-i\frac{\theta}{2}\sigma_y} \qquad R_z(\theta) = e^{-i\frac{\theta}{2}\sigma_z} \tag{4.3}$$

where σ_x, σ_y, and σ_z are Pauli matrices. The operation of U can be modified by changing parameters $\vec{\theta}$. Thus, the output state can be optimized to approximate the wanted state by changing parameters $\vec{\theta}$. By optimizing the parameters used in $U(\vec{\theta})$, PQC approximates the wanted quantum states.

4.2 Classical Layer

The classical layer in our construction of the quantum-classical hybrid neural network is to serve as the activation function connecting different quantum layers. To achieve nonlinearity, the classical layer is enabled by measurements—expectation values of operators on each qubit of the PQC, for example, $\langle \sigma_z^i \rangle$ of each qubit i as the classical layer, which would also serve as nonlinear operations. Expectation values of operators can save complexity because quantum tomography is exponentially hard. Though the expectation values of operators may lose some information compared to quantum tomography, some work used expectation values of operators as connections between quantum computation and classical computation and showed great success [91], which indicates expectation values of operators are capable of extracting useful information from quantum circuits.

4.3 Numerical Simulations

To demonstrate the power of the proposed quantum-classical hybrid neural network, we present results for calculating the ground state energies of simple molecular systems: H_2, LiH, and BeH_2. The inputs for the unsupervised learning are bond lengths and the outputs are the ground state energies. The whole procedure consists of first training the neural network with some bond lengths and then testing the neural network with other bond lengths to generate the whole potential energy curve.

4.3.1 Constructions of the Quantum Layer

The quantum layer consists of two parts: the variational encoding part and PQC part. We choose to use the variational encoding to decrease the depth of the quantum circuit so that it can be implemented on NISQ devices. The construction of the quantum layer follows [88], [91]. The input state is initialized as $(\otimes_{i=0}^{n-1} R_y(a)H)|0\rangle^{\otimes n}$, where a is the bond length, H is the Hadamard gate, and R_y is the rotation-y gate. We only have one bond length while the number of qubits of the PQC is n; we decided to follow the variational encoding in [88] to encode each qubit with same value. The number of qubits n is equal to the number of qubits of the corresponding Hamiltonian. The quantum computation part is to use a simple PQC consisting of R_y and $CNOT$ gate. The j$_{th}$ layer L_n^j of the n qubits PQC can be written as:

$$
\begin{aligned}
L_n^j &= (\otimes_{i=0}^{n-1} R_y(w_{i+n\times j})) \times (CNOT_{n-3,n-2}...CNOT_{3,4}CNOT_{1,2}) \\
&\times (CNOT_{n-2,n-1}...CNOT_{2,3}CNOT_{0,1}) \qquad n \text{ is even} \\
L_n^j &= (\otimes_{i=0}^{n-1} R_y(w_{i+n\times j})) \times (CNOT_{n-2,n-1}...CNOT_{3,4}CNOT_{1,2}) \\
&\times (CNOT_{n-3,n-2}...CNOT_{2,3}CNOT_{0,1}) \qquad n \text{ is odd}
\end{aligned}
\tag{4.4}
$$

where w are adjustable parameters, R_y represents rotation-y gate on the qubit i, and $CNOT_{m,n}$ represents $CNOT$ gate with m as the control qubit and n is the target qubit. To achieve better entanglement of the qubits before appending nonlinear operations, the n qubits PQC has n repeated layers in our simulation or the PQC can be written as $\prod_{j=0}^{n-1} L_n^j$. By optimizing the parameters, the general PQC tries to approximate arbitrary states so that it can be used for different specific molecules. The construction of the PQC for three qubits is illustrated in the blue part of Figure 4.2, and the construction of the PQC for four qubits is illustrated in the blue part of Figure 4.3.

4.3.2 Constructions of the Classical Layer

The classical layer is enabled by expectation values of the operators. In our numerical simulations, we are using $\langle \sigma_z^i \rangle$ for qubit i as the classical layer. The outputs from the classical layer will be encoded into another quantum layer. The second quantum layer is the same as the first one except for the data encoding part it would be $\otimes_{i=0}^{n-1} R_y(b_i\pi)H$, where b_i is
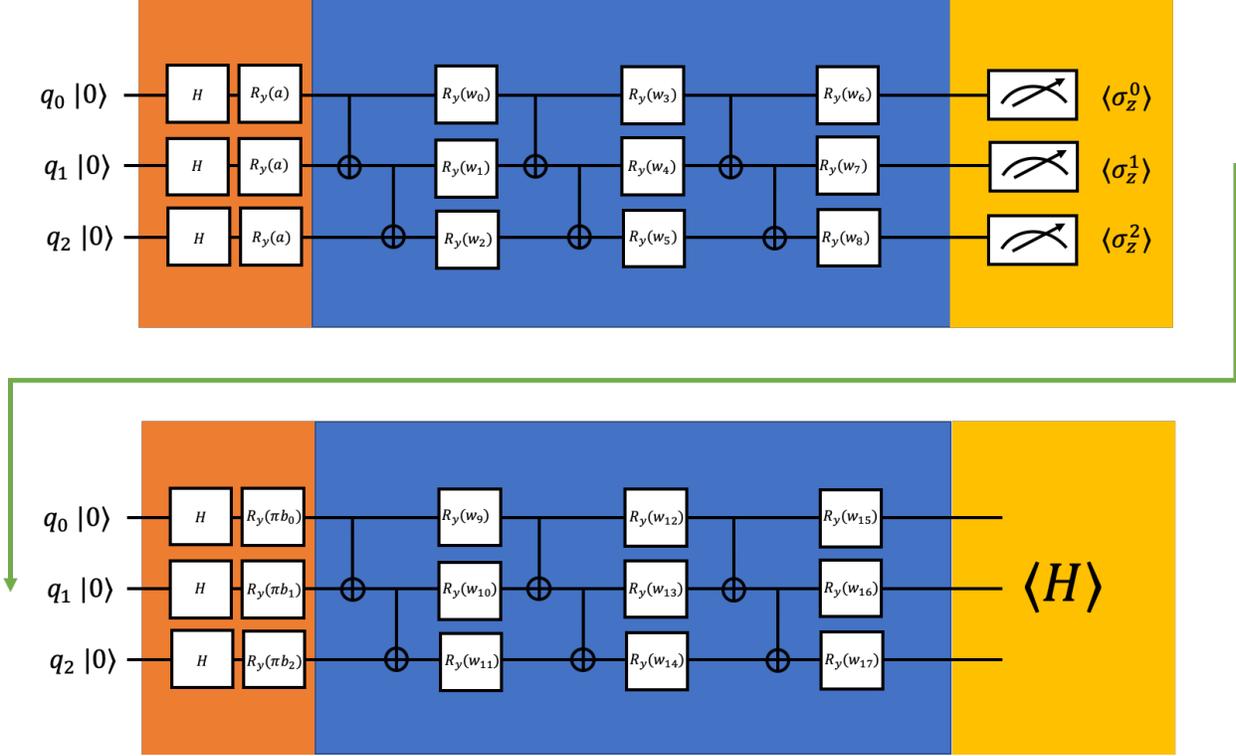
Figure 4.2. The example constructions of the proposed hybrid quantum-classical neural network for 3 qubits (odd qubits number). The orange parts are the data encoding, the blue parts are parameterized quantum circuits, and the yellow parts are measurements. The first measurements serve as nonlinear operations connecting two PQC. a is the input bond length, bs are the expectation values of σ_z, and ws are adjustable parameters.

the measured expectation value from qubit i. We multiply each b_i with π when encoding to change the range of the encoding data from $[-1, 1]$ to $[-\pi, \pi]$ [92]. The construction of our proposed hybrid quantum-classical neural network is illustrated in Figure 4.3.

4.3.3 Cost Function

The cost function is defined as:

$$f = \sum_j \langle \phi_j | H_j | \phi_j \rangle \tag{4.5}$$

where j represents the j_{th} input bond length of the training bond lengths. $|\phi_j\rangle$ is the final state of the proposed hybrid quantum-classical neural network with the input as the j_{th} input bond
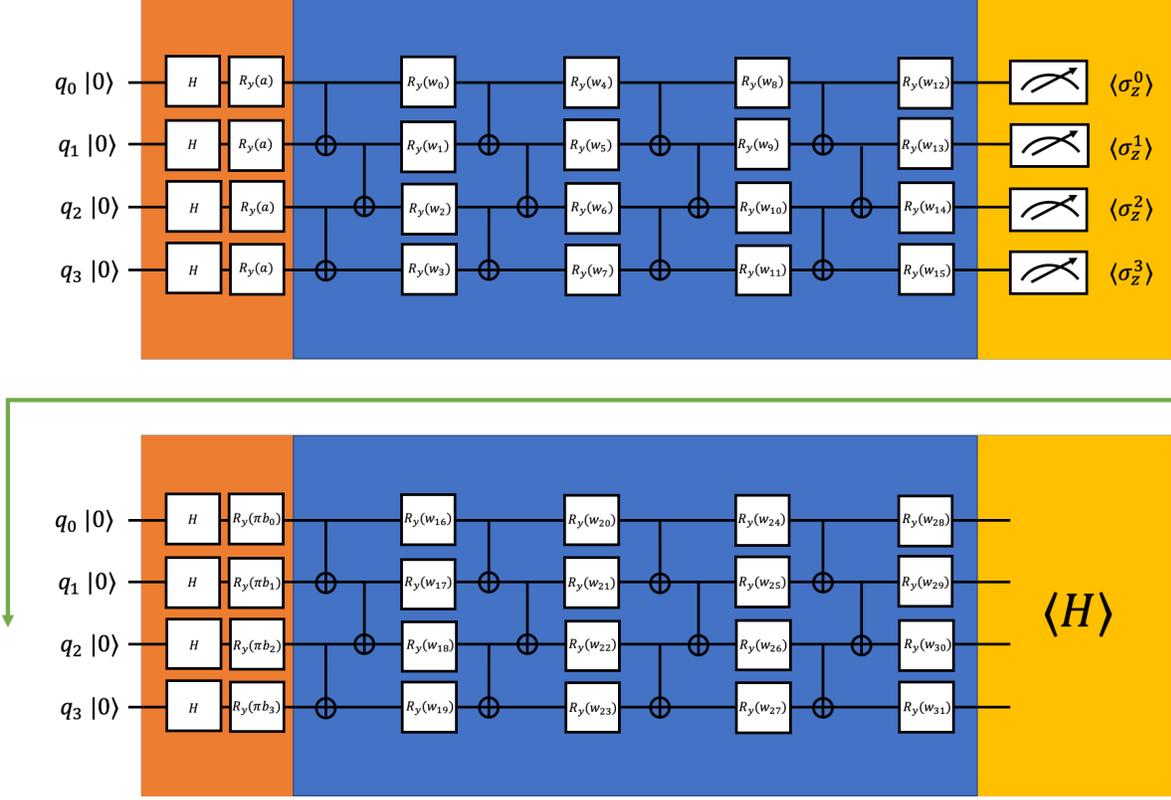
Figure 4.3. The example constructions we use for the 4 qubits H_2 calculation (even qubits number). The orange parts are the data encoding, the blue parts are parameterized quantum circuits, and the yellow parts are measurements. The first measurements serve as nonlinear operations connecting two PQC. a is the input bond length, bs are the expectation values of σ_z, and ws are adjustable parameters.

length and H_j is the Hamiltonian corresponding to the j_{th} input bond length. The idea of the cost function is similar to VQE: by optimizing the parameters, the expectation energy of $|\phi_j\rangle$ is minimized to approximate the ground state energy. The evaluation of the Hamiltonian can be done by techniques in [13]. The Hamiltonian can be written as the sum of tensor products of Pauli matrices $H = \sum_i c_i P_i$, where c_i is the coefficient and P_i is the tensor product of Pauli matrices. Instead of evaluating the whole Hamiltonian, we can evaluate each term of the Hamiltonian and the expectation of the Hamiltonian can be obtained by $\langle H \rangle = \sum_i c_i \langle P_i \rangle$, which does not need quantum tomography or take exponential complexity. The whole training procedure is done by taking a set of bond lengths and corresponding

Hamiltonian and minimizing the cost function as equation 4.5. After the training, we test the model with other bond lengths.

4.3.4 Simulation Results

The Hamiltonian of the molecule systems can be derived by transforming the corresponding second quantization Hamiltonian into sum of tensor products of Pauli matrices. For H_2, we use the Jordan–Wigner transformation [5] to get a 4-qubit Hamiltonian. We decided to apply the complete active space (CAS) approach [93], [94], which divides the orbitals into inactive orbitals such as always occupied low energy orbitals and always unoccupied high energy orbitals, and active orbitals, to reduce the number of qubits of LiH and BeH_2 Hamiltonian [13], [65] and the reduced Hamiltonian is only of the active orbitals. For LiH, we assume the first two lowest energy spin orbitals are always occupied and use the binary code transformation [10] with checksum code considering spin symmetry to save two qubits. We get an 8-qubit LiH Hamiltonian. For BeH_2, we assume the first two lowest energy spin orbitals are always occupied and the first two highest energy spin orbitals are never occupied, and use the binary code transformation [10] with checksum code considering spin symmetry to save two qubits. We get an 8-qubit BeH_2 Hamiltonian.

In the simulation, H_2 used four qubits and 32 parameters. LiH and BeH_2 both used eight qubits and 128 parameters. The gate and parameter complexity of the proposed hybrid quantum-classical neural network in this simulation is $O(n^2)$, where n is the number of qubits of the Hamiltonian. Here, we present the results using our proposed hybrid quantum-classical neural network for ground state energies of H_2, LiH, and BeH_2 in Figures 4.4 and 4.5. We can see from these figures that the training data points converge very close to the diagonalization results without pre-known ground state information of the transformed Hamiltonian in Pauli matrices format. Furthermore, after training, by inputting the other bond lengths we can also get good approximating ground state energies with optimized parameters. BeH_2 has some deviation when the bond length is large, which may be solved by improving the parameterized quantum circuit. For example, the work in [95], which discusses expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms, shows

that increasing the depth of PQC will increase the expressibility and different constructions of PQC have also different expressibility.
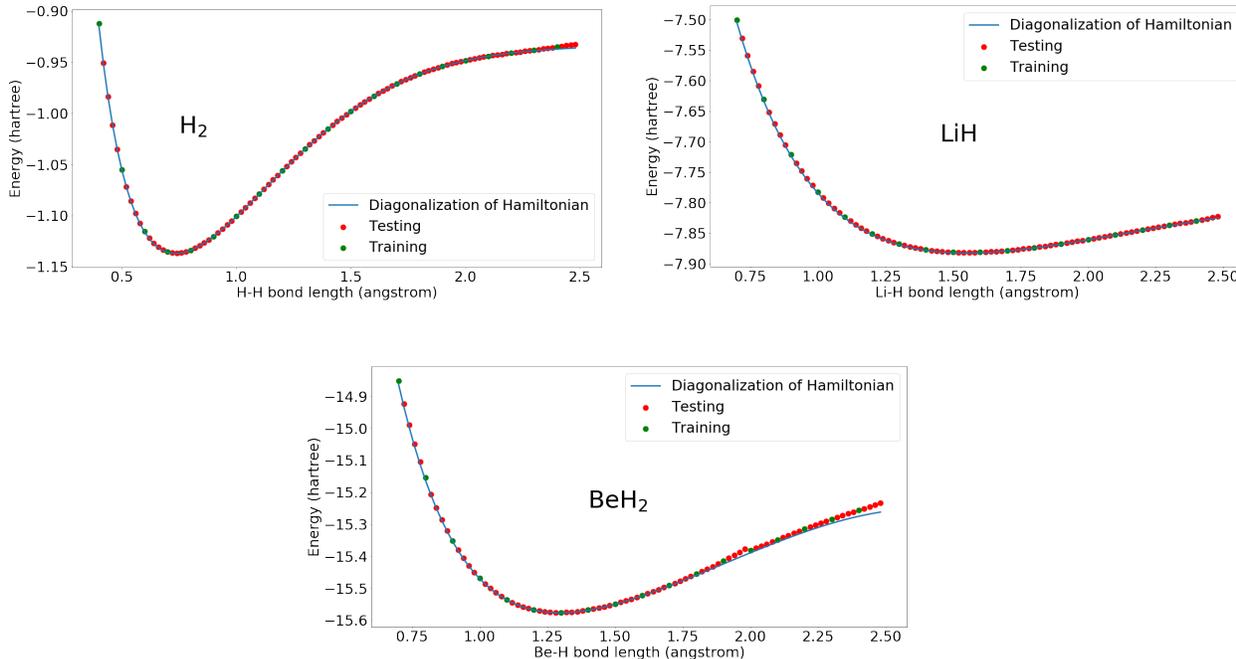
Figure 4.4. Ground state energies of H_2, LiH, and BeH_2 calculated by the proposed hybrid quantum- classical neural network.

Furthermore, to show that the intermediate nonlinear measurements improve the performance, we present the comparison of the results of our proposed hybrid quantum-classical neural network and quantum neural network removing intermediate measurements. The setting of the quantum neural network removing intermediate measurements is illustrated in Figure 4.7.

In Figure 4.6, we present the comparison of the results of our proposed hybrid quantum-classical neural network and quantum neural network removing intermediate measurements. The proposed hybrid quantum-classical neural network and quantum neural network, removing intermediate measurements, are trained with same set of bond lengths as in Figure 4.4. We can see without the intermediate nonlinear measurements, the quantum neural network can only achieve bad results. However, by adding the intermediate nonlinear measurements, the results converge closely to the diagonalization results.
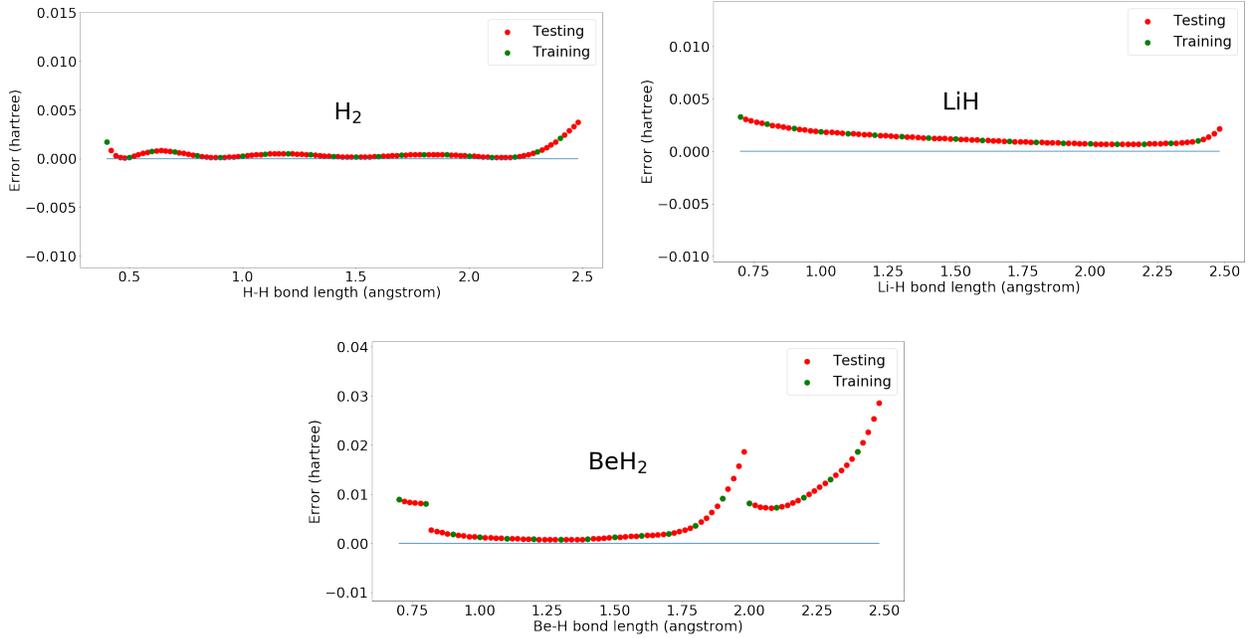
Figure 4.5. Errors of ground state energies of H_2, LiH, and BeH_2 calculated by the proposed hybrid quantum-classical neural network.

The parameters of the proposed hybrid quantum-classical neural network and quantum neural network removing intermediate measurements, are initialized from a Gaussian distribution with standard deviation as 0.1 and mean as 0. Because different initialization of parameters will result in different starting of the optimization and may lead to different final results, to eliminate the effects of parameter initialization, here we present the quantitative comparison of the two constructions with four different parameter initialization from same Gaussian distribution with different random seeds. All are trained with the same set of the training bond lengths as in Figure 4.4. In the Table 4.1, we can see that our proposed quantum neural network performs better than the quantum neural network without intermediate measurements. Our simulation results show that adding intermediate nonlinear measurements would help to improve the expressibility of the PQC. Furthermore, adding intermediate measurements would also decrease the circuit depth which makes it more suitable for current NISQ devices.
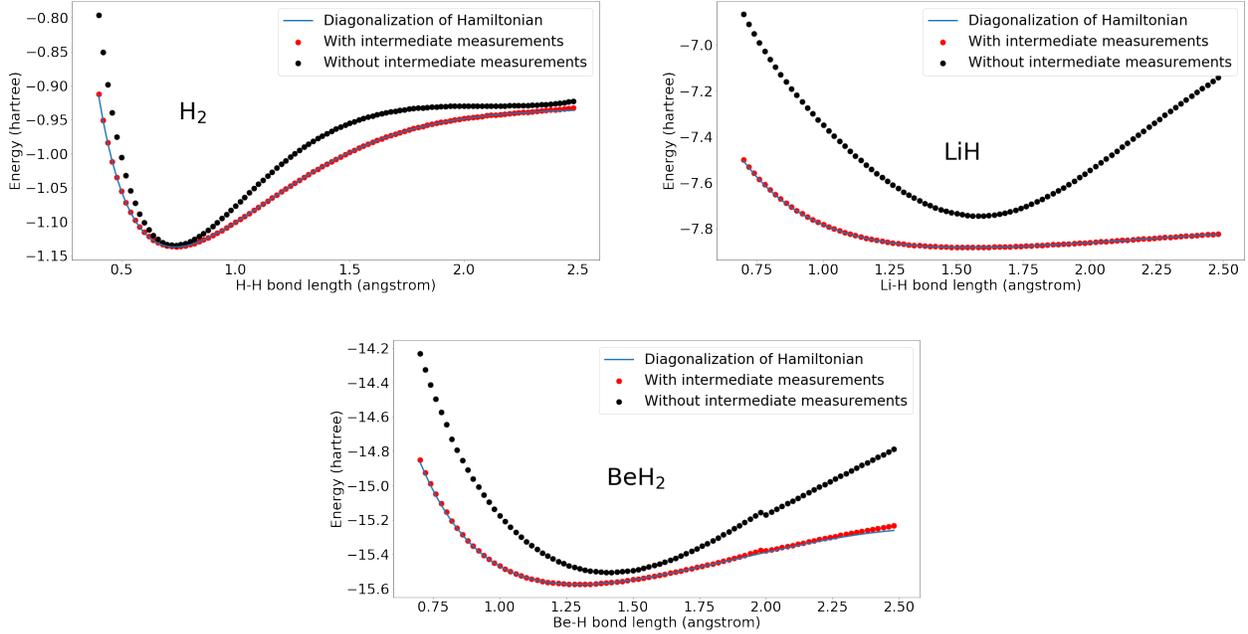
Figure 4.6. Results of H_2, LiH, and BeH_2 by the proposed hybrid quantum-classical neural network and the quantum neural network removing intermediate measurements. With intermediate measurements represents the results by our proposed hybrid quantum-classical neural network. Without intermediate measurements represents the quantum neural network removing the intermediate measurements. Both are trained with same set of bond lengths as in Figure 4.4 and same parameter initialization.
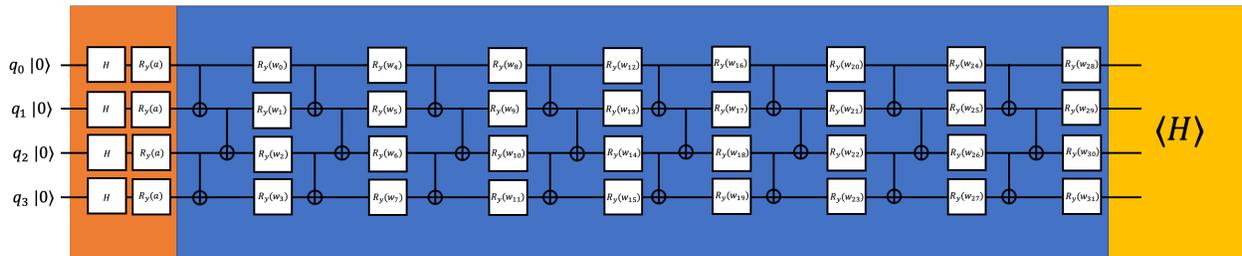
Figure 4.7. Constructions of the quantum neural network removing intermediate measurements for H_2.

4.4 Materials and Methods

Orbital integrals in the second quantization Hamiltonian are calculated by STO-3G minimal basis using PySCF [96] and the transformation is done by OpenFermion [30]. The

Table 4.1. Results for the proposed hybrid quantum-classical neural network and the quantum neural network removing intermediate measurements. With intermediate measurements represents the proposed hybrid quantum-classical neural network. Without intermediate measurements represents the quantum neural network removing intermediate measurements. $\sum_{training} Error$ represents the sum of the error of the calculated ground state energies on the training set. $\sum_{testing} Error$ represents the sum of the error of the calculated ground state energies on the testing set. Each result is calculated by 4 different parameter initialization and presented as means and standard deviations. It can be seen that adding intermediate measurements to introduce nonlinear options would help to improve the performance. The results presented are round to four decimal places.

Constructions	$\sum_{training} Error$	$\sum_{testing} Error$
With intermediate measurements (H_2)	0.0271 ± 0.0246	0.1178 ± 0.1061
Without intermediate measurements (H_2)	0.6296 ± 0.0151	2.2755 ± 0.0677
With intermediate measurements (LiH)	0.0287 ± 0.0038	0.1178 ± 0.0190
Without intermediate measurements (LiH)	4.7638 ± 1.4444	19.1479 ± 5.7715
With intermediate measurements (BeH_2)	0.1253 ± 0.0552	0.5613 ± 0.2483
Without intermediate measurements (BeH_2)	3.7280 ± 0.6497	14.8440 ± 2.3747

simulation is done by Qiskit [97]. The tensor production orders in OpenFermion and Qiskit are opposite. For a n qubits, the tensor production order in OpenFermion is $q_0 \otimes q_1 ... \otimes q_{n-1}$, while the tensor production order in Qiskit is $q_{n-1} \otimes q_{n-2} ... \otimes q_0$. We decided to follow the tensor production order in OpenFermion. In simulation, we treat the qubit indexed in Qiskit reversely. For n qubits, the qubit indexed as q_0 in Qiskit is treated as q_{n-1}, the qubit indexed as q_1 in Qiskit is treated as q_{n-2}, etc. By doing this, we change the tensor production order in Qiskit same as OpenFermion. The optimization is performed by the Broyden–Fletcher–Goldfarb–Shanno algorithm [98] using finite difference gradients implemented in SciPy [99] with maximum 500 iterations and gradient norm tolerance to stop as 10^{-5}. In the simulation, the expectation of the operator is simulated by matrix production of the operator matrix and the Hamiltonian can be treated as a single operator. To save the simulation time, instead of evaluating each $\langle P_i \rangle$ to get $\langle H \rangle = \sum_i c_i \langle P_i \rangle$, we treat H as a single operator and only evaluate once.

4.5   Conclusions

In this work, we proposed a new hybrid quantum-classical neural network by combing PQC and measurements to achieve nonlinear operations in quantum computing. We have shown that the proposed hybrid quantum-classical neural network can be trained to obtain the electronic energies at certain bond lengths and then generate the whole potential energy curve. The results of H_2, LiH, and BeH_2 are very accurate and demonstrate the power of the proposed hybrid quantum-classical neural network.

Furthermore, we show that the intermediate nonlinear measurements are very important in comparison with quantum neural network removing the intermediate measurements. The intermediate nonlinear measurements can reduce the circuit depth and are more suitable for NISQ devices. Although the method is used to generate one-dimensional potential energy curves, the approach is general and could be generalized to generate multidimensional potential energy surfaces, for example, changing the inputs from the bond lengths to multidimensional coordinates. This will be done in future work.

5. QUBIT COUPLED CLUSTER SINGLES AND DOUBLES VARIATIONAL QUANTUM EIGENSOLVER ANSATZ

Quantum computing has been developing rapidly in recent years as a promising new paradigm for solving many problems in science and engineering. One major potential application of quantum computing is solving quantum chemistry problems [100] such as electronic structure of molecules, which has received a lot of research interest and achieved a big success in both algorithmic development and experimental implementation. The early development of electronic structure calculations was based on the quantum phase estimation algorithm developed by Kitaev [101], Abrams and Lloyd [102] and used to find spectrum of simple molecular systems [57], [59], [65], [75], [103], [104]. More recently, hybrid classical-quantum algorithms have been developed such as the variational quantum eigenslover (VQE) [12], [76], [105], [106] and quantum machine learning techniques [82] for electronic structure calculations. Moreover, many experiments have been conducted on quantum computers to show that electronic structure calculations of simple molecules are possible on current Noisy Intermediate-Scale Quantum (NISQ) devices [11], [13], [14].[1]

One of the most promising quantum algorithms to perform electronic structure calculations is based on unitary coupled cluster [107] singles and doubles (UCCSD), which implements the quantum computer version of UCCSD as the VQE ansatz [76], [94], [108] to calculate the ground state from a Hartree-Fock reference state. The results from UCCSD VQE achieve high accuracy [94], [100], [109], [110]. However, the gate complexity for first order trotterization UCCSD VQE is up-bounded to $O(n^5)$ [94], [100] using Jordan-Wigner transformation where n is the number of qubits of the Hamiltonian. This makes it difficult to implement on current NISQ devices. Some strategies developed may be used to reduced the complexity, for example, the ordering and parallelization techniques in [111] can reduce the circuit depth by $O(n)$ [94] and low-rank factorization [112] can reduce the gate complexity to $O(n^4)$. Here we introduce a new VQE ansatz based on the particle preserving exchange gate [108], [113] to achieve qubit excitations, which has gate complexity up-bounded to $O(n^4)$

---

[1]Contents in this chapter are reprinted and adapted from Xia, R., & Kais, S. (2020). Qubit coupled cluster singles and doubles variational quantum eigensolver ansatz for electronic structure calculations. Quantum Science and Technology, 6(1), 015001. Copyright (2020) IOP Publishing Ltd.

and has comparable accuracy compared to first order trotterization UCCSD VQE. By reducing the gate complexity, QCCSD VQE ansatz – qubit coupled cluster singles and doubles (QCCSD) VQE ansatz, might be more favorable for current NISQ devices.

The rest of the chapter is organized as follows: The first section gives a brief introduction to the method of UCCSD VQE ansatz. Then we give a detailed description of QCCSD VQE ansatz. We also show QCCSD VQE is a simplified version of the first order trotterization UCCSD VQE. Finally, we give the numerical simulation results of BeH_2, H_2O, N_2, H_4 and H_6 using first order trotterization UCCSD VQE and QCCSD VQE ansatz.

5.1 UCCSD VQE

The electronic structure Hamiltonian can be written in second quantization as:

$$H = h_0 + \sum_{ij} h_{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{ijkl} h_{ijkl} a_i^\dagger a_j^\dagger a_k a_l \tag{5.1}$$

where the one-electron integrals h_{ij} and the two-electron integrals h_{ijkl} can be calculated by orbital integrals.

Using Jordan-Wigner transformation we can rewrite the Hamiltonian in the Pauli matrices form:

$$H = \sum_i c_i P_i \tag{5.2}$$

where c_i are general coefficients and P_i are the tensor product Pauli matrices σ^x, σ^y, σ^z and 2×2 identity matrix.

In unitary coupled clustered single double excitations, we can calculate the ground state from the Hartree-Fock reference state by excitation operators of the form:

$$|\phi\rangle = e^{T(\vec{\theta}) - T^\dagger(\vec{\theta})} |\phi_{HF}\rangle \tag{5.3}$$

where $T(\vec{\theta}) = T_1(\vec{\theta_1}) + T_2(\vec{\theta_2})$ is the excitation operator, $|\phi_{HF}\rangle$ is the Hartree-Fock reference state and $\vec{\theta}$ is the set of adjustable parameters. The single excitation operator can be written as $T_1(\vec{\theta_1}) = \sum_{i,j} \theta_{ij} a_i^\dagger a_j$ and the double excitation operator can be written as $T_2(\vec{\theta_2}) =$

$\sum_{i,j,k,l} \theta_{ijkl} a_i^\dagger a_j^\dagger a_k a_l$. We can minimize $\langle \phi | H | \phi \rangle$ to get the ground state energy by optimizing $\vec{\theta}$.

Considering an n qubits Hamiltonian, the number of spin orbitals is n and the total number of excitation terms in T is $O(\binom{N_{occ}}{2} \times \binom{N_{virt}}{2})$, where N_{occ} is the number of occupied spin orbitals, N_{virt} is the number of virtual spin orbitals. $n = N_{occ} + N_{virt}$ is the number of qubits of the Hamiltonian or the total number of spin orbitals.

The first order trotterization UCCSD operator can be written as:

$$\mathrm{e}^{T(\vec{\theta})-T^\dagger(\vec{\theta})} \approx \prod_{i,j} \mathrm{e}^{\theta_{ij}(a_i^\dagger a_j - a_j^\dagger a_i)} \times \prod_{i,j,k,l} \mathrm{e}^{\theta_{ijkl}(a_i^\dagger a_j^\dagger a_k a_l - a_l^\dagger a_k^\dagger a_j a_i)} \tag{5.4}$$

To map the first order trotterization UCCSD to quantum computer, we use same transformation, Jordan-Wigner transformation, as we do for the Hamiltonian to transform creation and annihilation operators into Pauli matrices. Each term in equation 5.4 can be implemented as unitary quantum gates by Jordan-Wigner transformation. Since the cost of Jordan-Wigner transformation for each term is $O(n)$ [100], the gate complexity for the first order trotterization UCCSD VQE is $O(\binom{N_{occ}}{2} \times \binom{N_{virt}}{2} \times n) < O(n^5)$ using Jordan-Wigner transformation [94], [100].

UCCSD VQE has shown high accuracy in electronic structure calculations [94], [100], [109], [110]. However, one problem of the UCCSD VQE is the large complexity. The first order trotterization UCCSD VQE has up-bounded $O(n^4)$ terms and $O(n^5)$ gate complexity using Jordan-Wigner transformation. Here, we propose a new coupled cluster singles and doubles VQE ansatz using the particle preserving exchange gate [108], [113]. The gate complexity of QCCSD ansatz scales as $O(\binom{N_{occ}}{2} \times \binom{N_{virt}}{2}) < O(n^4)$. In the numerical simulations, we show that QCCSD ansatz can achieve comparable accuracy to the first order trotterization UCCSD VQE.

5.2 QCCSD VQE Ansatz

After Jordan-Winger transformation, each qubit represents whether the corresponding spin orbital is occupied or not. When qubit i is in $|0\rangle$, spin orbital i is not occupied and

when qubit i is in $|1\rangle$ spin orbital i is occupied. Thus we can write down a particle preserving exchange gate $U_{\mathrm{e}x}$ [108], [113] between two qubits as:

$$U_{\mathrm{e}x}(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\theta & -sin\theta & 0 \\ 0 & sin\theta & cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The particle preserving exchange gate $U_{\mathrm{e}x}$ will not change the total number occupation when applied to arbitrary states. Suppose we have two qubits in $|10\rangle$, which represents that the first spin orbital is occupied and the second spin orbital is not occupied. If we apply $U_{\mathrm{e}x}$ to this state we have:

$$U_{\mathrm{e}x}(\theta)|10\rangle = cos\theta|10\rangle - sin\theta|01\rangle \tag{5.5}$$

which corresponds to a single excitation between one spin occupied and one virtual spin orbitals.

We can also write down a particle preserving exchange gate $U'_{\mathrm{e}x}$ between four qubits as in Figure 1. Suppose we have four qubits in $|1010\rangle$, which represents the first and the third spin orbitals are occupied while the second and the fourth spin orbitals are not occupied. If we apply $U'_{\mathrm{e}x}$ to this state we have:

$$U'_{\mathrm{e}x}(\theta)|1010\rangle = cos\theta|1010\rangle - sin\theta|0101\rangle \tag{5.6}$$

which corresponds to a double excitation between two occupied and two virtual orbitals.

$$U'_{\mathrm{ex}}(\theta) = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & cos\theta & 0 & 0 & 0 & 0 & -sin\theta & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & sin\theta & 0 & 0 & 0 & 0 & cos\theta & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}$$

Figure 5.1. Matrix of $U'_{\mathrm{ex}}(\theta)$

We can write down an operator U by $U_{\mathrm{e}x}$ and $U'_{\mathrm{e}x}$ to achieve single and double excitations from the Hartree-Fock reference state:

$$|\Phi\rangle = U(\vec{\Theta})|\phi_{HF}\rangle = \prod_{\mathrm{i,j}} U_{ex,\mathrm{i,j}}(\theta_{\mathrm{ij}}) \prod_{i,j,k,l} U'_{ex,\mathrm{i,j},k,l}(\theta_{\mathrm{ij}kl})|\phi_{HF}\rangle \tag{5.7}$$

$U_{ex,\mathrm{i,j}}$ represents $U_{\mathrm{e}x}$ between qubits i j where qubit i represents the occupied orbital and qubit j represents the virtual orbital. $U'_{ex,\mathrm{i,j},k,l}$ represents $U'_{\mathrm{e}x}$ between qubits i j k l where qubit i, k represent occupied spin orbitals and qubit j, l represent the virtual spin orbitals. $\vec{\Theta}$ is the set of adjustable parameters. We can minimize $\langle\Phi|H|\Phi\rangle$ to get the ground state energy by optimizing $\vec{\Theta}$.

74

$U_{\mathrm{e}x}(\theta)$ and $U'_{\mathrm{e}x}(\theta)$ can be decomposed into elementary quantum gates with gate complexity $O(1)$ because the sizes of matrices of $U_{\mathrm{e}x}(\theta)$ and $U'_{\mathrm{e}x}(\theta)$ are $O(1)$. A possible decomposition of $U_{\mathrm{e}x}(\theta)$ is by Gray code [2]:
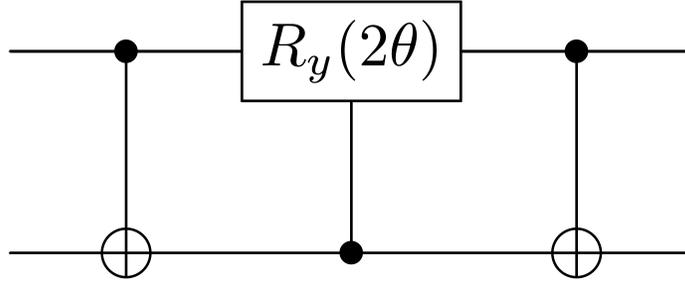
Figure 5.2. Decomposition of $U_{\mathrm{e}x}(\theta)$ by Gray code.

A possible decomposition of $U'_{\mathrm{e}x}(\theta)$ as in [114] is shown in Figure 5.3. A number of groups have shown how to reduce the gate complexity of coupled cluster methods [115]–[117] on quantum computer and may be able to be applied to QCCSD VQE. Recently, O'Gorman et al. [118] show that by using fermionic swap networks one can reduce the circuit depth to $O(n^{k-1})$ when implementing set of k qubits gates on n logical qubits, which may reduce the complexity of QCCSD VQE by a factor of n. This is a possible future improvement but out of the scope of this paper. Recently, Yordanov et al [114] proposed a new decomposition of UCCSD VQE into two steps: first applying the qubit excitation gates, which are the particle preserving exchange gates in our QCCSD VQE used for single and double excitations though termed differently, then applying CNOT gates to include the parity information. In our simulation, $U_{\mathrm{e}x}(\theta)$ and $U'_{\mathrm{e}x}(\theta)$ are implemented as single unitary gates in Qiskit [119].
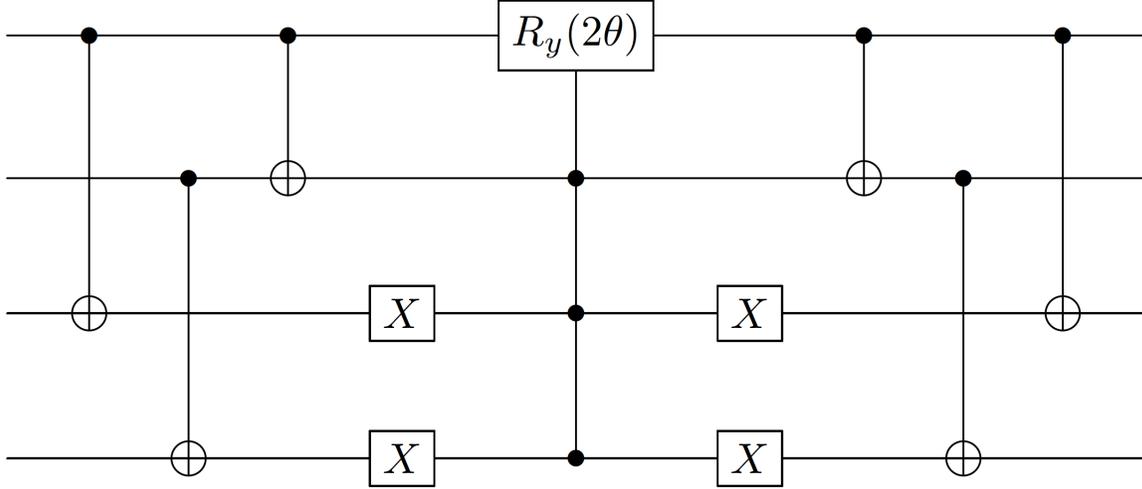
Figure 5.3. Decomposition of $U'_{ex}(\theta)$ follows in [114]

5.2.1 Excitation List Selection

One important part of the proposed VQE is to choose the excitation list, or to decide between which spin orbitals the excitation will occur. Spin preserving VQE ansatzes, which preserve the net spin magnetization s_z, have been widely studied [120], [121]. We use the same strategy and choose the excitation list only allowing spin preserving exicitations. As shown in [110], the ordering of excitation operators in the Trotterized UCCSD VQE may have impact on the final results. To eliminate the effect of the operator ordering, we choose the same ordering as in the implementation of first order trotterization UCCSD VQE in Qiskit [119]. The ordering is also presented in the below algorithm.

Algorithm 2 Coupled cluster singles and doubles VQE considering spin preserving

---

1: **for** orbital$_i$ in spin-up occupied orbitals **do**

2:     **for** orbital$_j$ in spin-up virtual orbitals **do**

3:         Construct U_{ex} between qubit i and j.

4:     **end for**

5: **end for**

6: **for** orbital$_k$ in spin-down occupied orbitals **do**

7:     **for** orbital$_l$ in spin-down virtual orbitals **do**

8:         Construct U_{ex} between qubit k and l.

9:     **end for**

10: **end for**

11: **for** orbital$_i$ in spin-up occupied orbitals and orbital$_j$ in spin-up virtual orbitals **do**

12:     **for** orbital$_k$ in spin-down occupied orbitals and orbital$_l$ in spin-down virtual orbitals **do**

13:         Construct U'_{ex} between qubit i j k and l.

14:     **end for**

15: **end for**

16: **for** orbital$_i$ in spin-up occupied orbitals and orbital$_k$ in spin-up virtual orbitals **do**

17:     **for** orbital$_j$ (j > i) in spin-up occupied orbitals and orbital$_l$ ($l > k$) in spin-up virtual orbitals **do**

18:         Construct U'_{ex} between qubit i k j and l.

19:     **end for**

20: **end for**

21: **for** orbital$_i$ in spin-down occupied orbitals and orbital$_k$ in spin-down virtual orbitals **do**

22:     **for** orbital$_j$ (j > i) in spin-down occupied orbitals and orbital$_l$ ($l > k$) in spin-down virtual orbitals **do**

23:         Construct U'_{ex} between qubit i k j and l.

24:     **end for**

25: **end for**

The term complexity of our ansatz scales as $O(\binom{N_{occ}}{2} \times \binom{N_{virt}}{2})$. The required elementary quantum gates for $U_{\text{e}x}$ and $U'_{\text{e}x}$ are both $O(1)$. Thus the gate complexity of our ansatz scales as $O(\binom{N_{occ}}{2} \times \binom{N_{virt}}{2}) < O(n^4)$. One should note that, for linear connectivity, if no extra strategies are applied, the straightforward compilation will make the complexity of proposed QCCSD VQE scale up to $O(n^5)$. However, this complexity can be reduced by applying strategies for the compilation as done for example in the generalized swap network [118]. Moreover, a recent study [120] shows that considering the total spin s preserving may also help to achieve better accuracy, QCCSD VQE ansatz may also be able to be modified to preserve the total spin s, which will be done in the future work.

5.2.2 Relation to UCCSD VQE Ansatz

Here, we present that QCCSD VQE ansatz is a simplified version of UCCSD VQE ansatz. Consider a single excitation term in first order trotterization UCCSD VQE:

$$e^{\theta(a_\text{i}^\dagger a_\text{j} - a_\text{j}^\dagger a_\text{i})} \tag{5.8}$$

Without loss of generality, we can require $\text{i} \in virt$ and $\text{j} \in occ$ where $virt$ represents virtual orbitals and occ represents occupied orbitals. Furthermore we can set $\text{i} > \text{j}$ and if using Jordan-Wigner transformation we get:

$$e^{\frac{\text{i}\theta}{2}\sigma_y^\text{j}\sigma_x^\text{i}\otimes_{a=\text{j}+1}^{\text{i}-1}\sigma_z^a} e^{\frac{-\text{i}\theta}{2}\sigma_x^\text{j}\sigma_y^\text{i}\otimes_{a=\text{j}+1}^{\text{i}-1}\sigma_z^a} \tag{5.9}$$

In equation 5.9, $\otimes_{a=\text{j}+1}^{\text{i}-1}\sigma_z^a$ counts for the parity of qubits from $\text{j}+1$ to $\text{i}-1$. If we remove the parity term we get:

$$e^{\frac{\text{i}\theta}{2}\sigma_y^\text{j}\sigma_x^\text{i}} e^{\frac{-\text{i}\theta}{2}\sigma_x^\text{j}\sigma_y^\text{i}} = U_{\text{ex,j,i}}(-\theta) \tag{5.10}$$

Thus our particle conservation exchange gate $U_{\text{ex,j,i}}$ counts for the single excitation term of qubits j i in first order trotterization UCCSD VQE without considering the parity of qubits from $\text{j}+1$ to $\text{i}-1$. One should be aware that if different order of spin orbitals is used,

the single excitation term with parity terms removed may equal to $U_{ex}(\theta)$. Also, consider a double excitation term in first order trotterization UCCSD VQE:

$$e^{\theta(a_i^\dagger a_j^\dagger a_k a_l - a_l^\dagger a_k^\dagger a_j a_i)} \tag{5.11}$$

Without loss of generality, we can require $i < j \in virt$ and $l < k \in occ$. Furthermore we can choose the order $j > k > i > l$ and if using Jordan-Wigner transformation we get:

$$e^{\frac{-i\theta}{8}\sigma_x^l \sigma_x^i \sigma_x^k \sigma_y^j \otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a} e^{\frac{-i\theta}{8}\sigma_x^l \sigma_y^i \sigma_x^k \sigma_x^j \otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a}$$

$$e^{\frac{-i\theta}{8}\sigma_x^l \sigma_y^i \sigma_y^k \sigma_y^j \otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a} e^{\frac{-i\theta}{8}\sigma_y^l \sigma_y^i \sigma_x^k \sigma_y^j \otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a}$$

$$e^{\frac{i\theta}{8}\sigma_x^l \sigma_x^i \sigma_y^k \sigma_x^j \otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a} e^{\frac{i\theta}{8}\sigma_y^l \sigma_x^i \sigma_x^k \sigma_x^j \otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a} \tag{5.12}$$

$$e^{\frac{i\theta}{8}\sigma_y^l \sigma_x^i \sigma_y^k \sigma_y^j \otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a} e^{\frac{i\theta}{8}\sigma_y^l \sigma_y^i \sigma_y^k \sigma_x^j \otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a}$$

In equation 5.12 $\otimes_{a=l+1}^{i-1}\sigma_z^a \otimes_{a=k+1}^{j-1}\sigma_z^a$ counts for the parity of qubits from $l+1$ to $i-1$ and from $k+1$ to $j-1$. If we remove the parity term we get:

$$e^{\frac{-i\theta}{8}\sigma_x^l \sigma_x^i \sigma_x^k \sigma_y^j} e^{\frac{-i\theta}{8}\sigma_x^l \sigma_y^i \sigma_x^k \sigma_x^j} e^{\frac{-i\theta}{8}\sigma_x^l \sigma_y^i \sigma_y^k \sigma_y^j} e^{\frac{-i\theta}{8}\sigma_y^l \sigma_y^i \sigma_x^k \sigma_y^j}$$

$$e^{\frac{i\theta}{8}\sigma_x^l \sigma_x^i \sigma_y^k \sigma_x^j} e^{\frac{i\theta}{8}\sigma_y^l \sigma_x^i \sigma_x^k \sigma_x^j} e^{\frac{i\theta}{8}\sigma_y^l \sigma_x^i \sigma_y^k \sigma_y^j} e^{\frac{i\theta}{8}\sigma_y^l \sigma_y^i \sigma_y^k \sigma_x^j} = U'_{ex,l,i,k,j}(-\theta) \tag{5.13}$$

Thus our particle preserving exchange gate $U'_{ex,l,i,k,j}$ counts for the double excitation term of qubits l,k,i,j in first order trotterization UCCSD VQE without considering the parity of qubits from $l+1$ to $i-1$ and from $k+1$ to $j-1$. One should be aware that if different order of spin orbitals is used, the double excitation term with parity terms removed may equal to $U'_{ex}(\theta)$. QCCSD VQE is the simplified version of first order trotterization UCCSD VQE. The reduced gate complexity of our VQE comes from removing the parity term in UCCSD VQE. Recently S. E. Smart et al [122] presented an efficient ansatz for two-electron system, showing that fermionic double excitations can be simplified to qubit double excitations in the two-electron system, which indicates the QCCSD VQE has the same double excitation terms as the UCCSD VQE for a two-electron system.

5.3 Numerical Simulation Results

In this section, we present numerical results of BeH_2, H_2O, N_2, H_4 and H_6 by using QCCSD VQE with gates U_{ex} and U'_{ex}. To compare performance of QCCSD VQE, we also present the results by using first order trotterization UCCSD VQE implemented by Qiskit [119]. For each numerical simulation, the orbital integrals are calculated using STO-3G minimal basis by PySCF [96] and the Hamiltonian is obtained by Jordan-Wigner transformation. The optimization is performed by the sequential least squares programming (SLSQP) algorithm [123]. The input state is the Hartree -Fock reference state and all parameters are initialized as 0 for both ansatzes. The bounds for all parameters for both ansatzes are set to $[-\pi, \pi]$. The energy thresholds for convergence is set to 10^{-6} Hartree with maximum 500 iterations. The noiseless simulation is done by Qiskit [119] with version 0.14.1. In the figures in this section, QCCSD VQE represent the proposed qubit coupled cluster singles and doubles VQE ansatz while UCCSD VQE represents the first order trotterization UCCSD VQE ansatz implemented by Qiskit [119].

Complete active space (CAS) approach [93], which divides the space to active spin orbitals and inactive spin orbitals, has been applied to reduce the qubits of molecule Hamiltonian in quantum simulation. To investigate the effect of the size of active space, we compare the performance of QCCSD ansatz for different sizes of active spaces for the same molecule.

For BeH_2 we choose three different active spaces: First 2 lowest energy spin orbitals are always filled and first 2 highest energy spin orbitals are always empty, corresponding to 10 active spin orbitals with 4 electrons or 10 qubits Hamiltonian. First 2 lowest energy spin orbitals are always filled, corresponding to 12 active spin orbitals with 4 electrons or 12 qubits Hamiltonian. No spin orbitals are always filled or always empty, corresponding to 14 active spin orbitals with 6 electrons or 14 qubits Hamiltonian. We compare the errors between the ground state energies from the VQE results and the ground state energies from the diagonalization of the corresponding Hamiltonian as in Figure 5.4. We can see that, although the size of the active space is increased,our QCCSD VQE achieves similar accuracy as the first order trotterization UCCSD VQE.
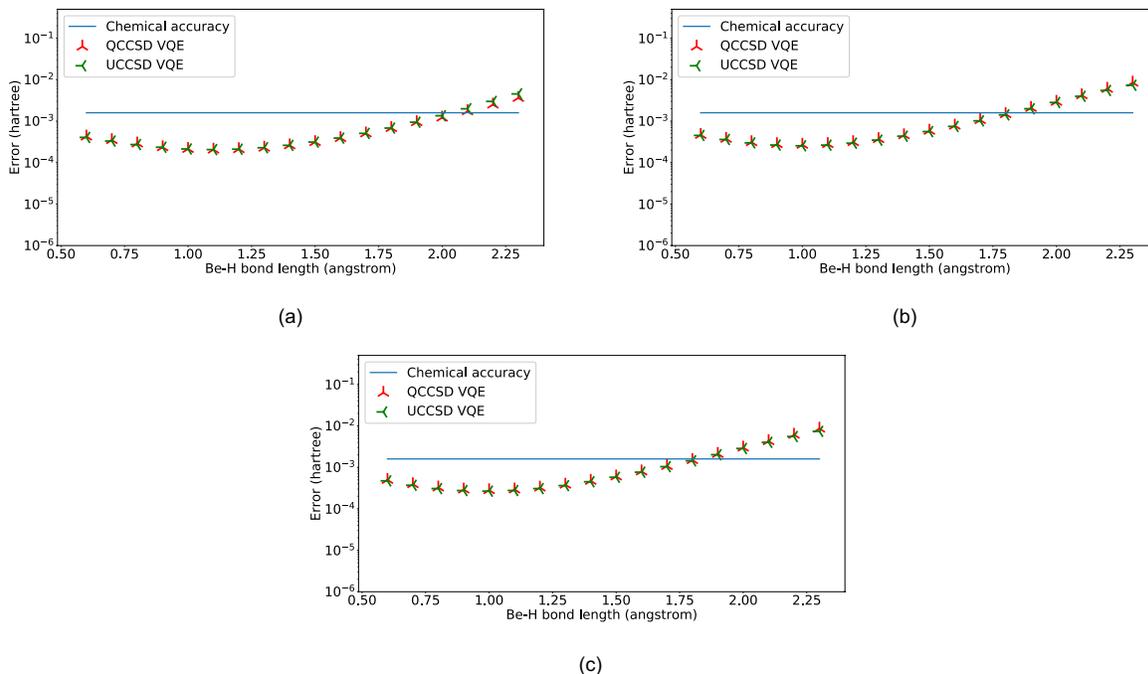
Figure 5.4. VQE results of BeH$_2$ by QCCSD VQE compared with first order trotterization UCCSD VQE. (a) The errors of ground state energies of 10 qubits BeH$_2$ Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE. (b) The errors of ground state energies of 12 qubits BeH$_2$ Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE. (c) The errors of ground state energies of 14 qubits BeH$_2$ Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE.

For H$_2$O we choose three different active spaces: First 4 lowest energy spin orbitals are always filled, corresponding to 10 active spin orbitals with 6 electrons or 10 qubits Hamiltonian; First 2 lowest energy spin orbitals are always filled, corresponding to 12 active spin orbitals with 8 electrons or 12 qubits Hamiltonian and no spin orbitals are always filled or always empty, corresponding to 14 active spin orbitals with 10 electrons or 14 qubits Hamiltonian. We compare the errors between the ground state energies from the VQE results and the ground state energies from the diagonalization of the corresponding Hamiltonian as in Figure 5.5 for different Hamiltonian. For 10 qubits H$_2$ Hamiltonian our qubit coupled cluster VQE achieves almost the same accuracy as the first order trotterization UCCSD VQE except at one point. For 12 and 14 qubits H$_2$ Hamiltonian, with increased

81

size of active space,our QCCSD VQE performs a little worse compared to the first order trotterization UCCSD VQE the error increased from 10^{-5} to 10^{-3} Hartree, but the error is still within or around the chemical accuracy.
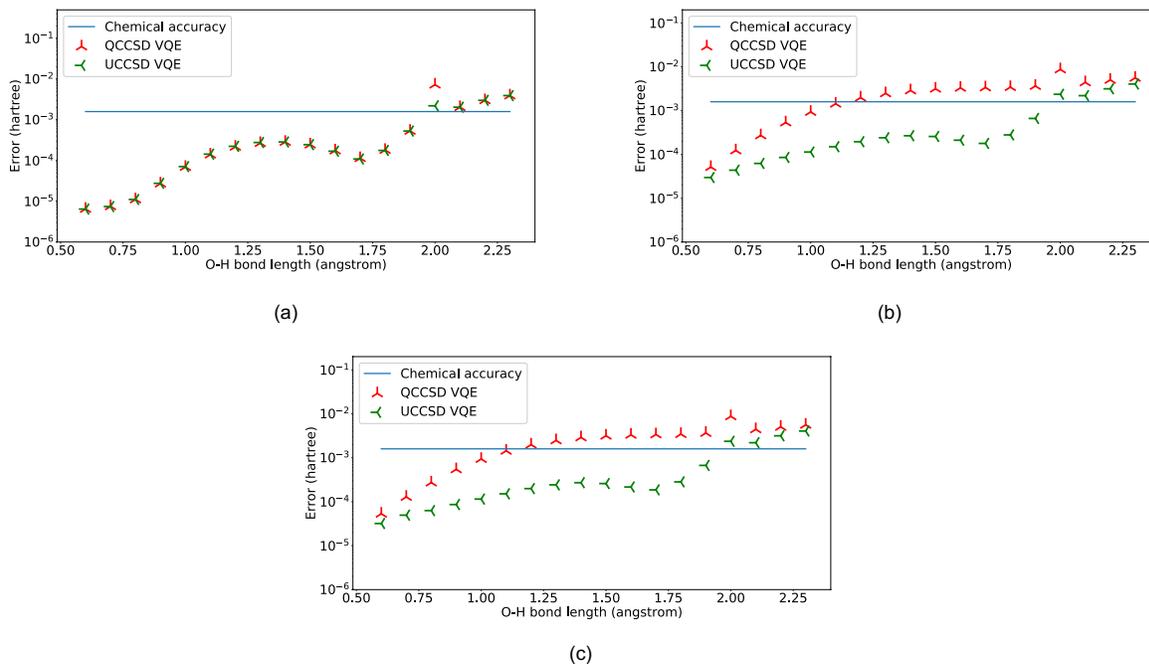
Figure 5.5. VQE results of H_2O by QCCSD VQE compared with first order trotterization UCCSD VQE. (a) The errors of ground state energies of 10 qubits H_2O Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE. (b) The errors of ground state energies of 12 qubits H_2O Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE. (c) The errors of ground state energies of 14 qubits H_2O Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE.

For N_2 we choose four different active spaces: First 8 lowest energy spin orbitals are always filled and first 2 highest energy spin orbitals are always empty, corresponding to 10 active spin orbitals with 6 electrons or 10 qubits Hamiltonian; First 8 lowest energy spin orbitals are always filled, corresponding to 12 active spin orbitals with 6 electrons or 12 qubits Hamiltonian; First 6 lowest energy spin orbitals are always filled, corresponding to 14 active spin orbitals with 8 electrons or 14 qubits Hamiltonian. First lowest energy 4 spin orbitals are always filled, corresponding to 16 active spin orbitals with 10 electrons or 16

qubits Hamiltonian. We compare the errors between the ground state energies from the VQE results and the ground state energies from the diagonalization of the corresponding Hamiltonian as in Figure 5.6 for different Hamiltonian. For 10 qubits N_2 Hamiltonian our QCCSD VQE achieves almost same or even better accuracy as the first order trotterization UCCSD VQE except one point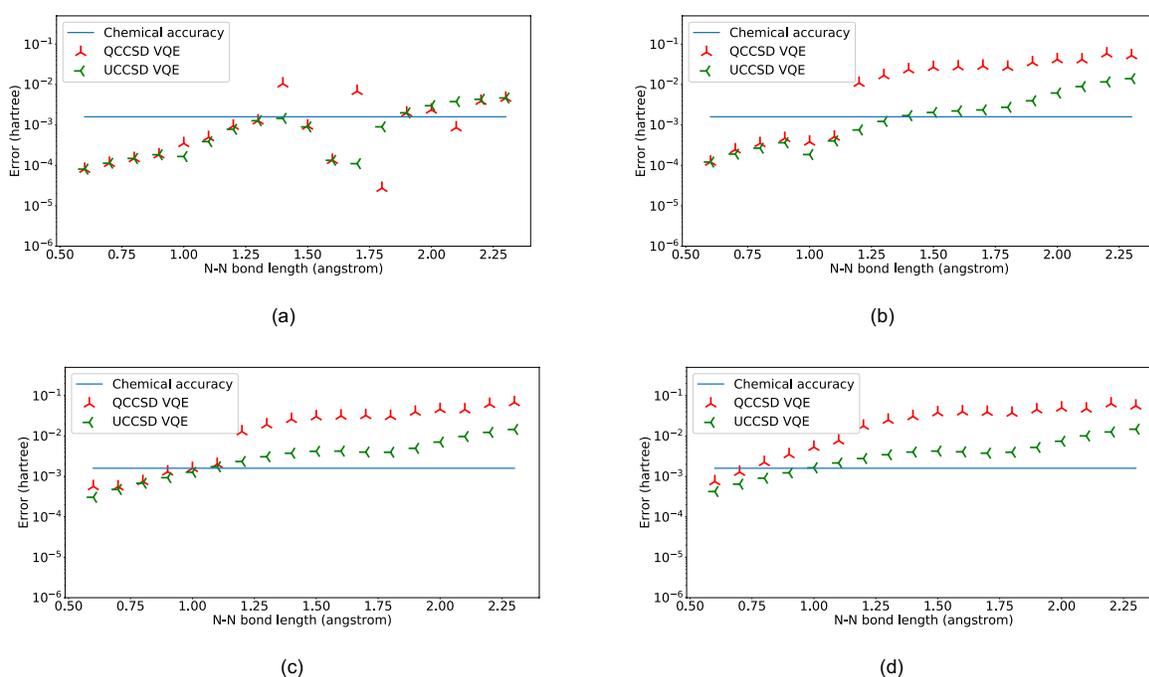. For 12, 14 and 16 qubits N_2 Hamiltonian, with increased size of active space,our QCCSD VQE performs worse compared to the first order trotterization UCCSD VQE. This indicates that the removal of parity terms in excitation operators may affect accuracy of the the couple cluster method for larger system size.

Figure 5.6. VQE results of N_2 by QCCSD VQE compared with first order trotterization UCCSD VQE. (a) The errors of ground state energies of 10 qubits N_2 Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE. (b) The errors of ground state energies of 12 qubits N_2 Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE. (c) The errors of ground state energies of 14 qubits N_2 Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE. (d) The errors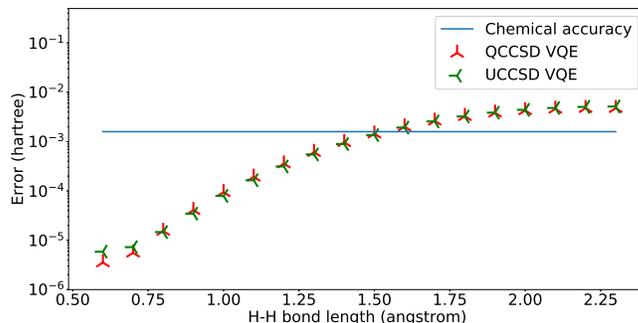 of ground state energies of 16 qubits N_2 Hamiltonian calculated by QCCSD VQE compared with first order trotterization UCCSD VQE.

For H_4 chain we do not have any restrictions on the spin orbitals, corresponding to 8 active spin orbitals and 4 active electrons. We also show the error between the ground state energies from VQE results and the ground state energies from the diagonalization of the corresponding Hamiltonian as in Figure 5.7.our QCCSD VQE achieves the same level accuracy compared to the first order trotterization UCCSD VQE.

Figure 5.7. The errors of ground state energies of H_4 calculated by QCCSD VQE compared with UCCSD VQE.

For H_6 chain we do not have any restrictions on the spin orbitals, corresponding to 12 active spin orbitals and 6 active electrons. We also show the error between the ground state energies from VQE results and the ground state energies from the diagonalization of the corresponding Hamiltonian as in Figure 5.8. our QCCSD VQE achieves same level accuracy compared to the first order trotterization UCCSD VQE.
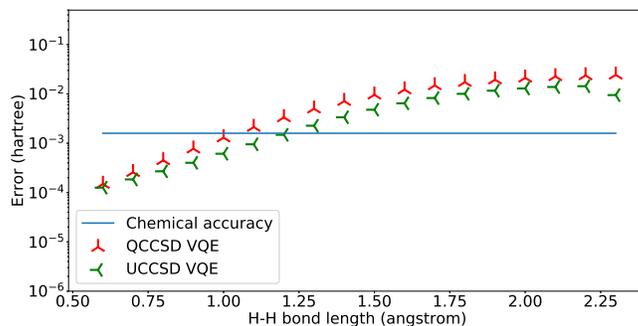
Figure 5.8. The errors of ground state energies of H_6 calculated by QCCSD VQE compared with UCCSD VQE.

5.4 Discussion and Conclusion
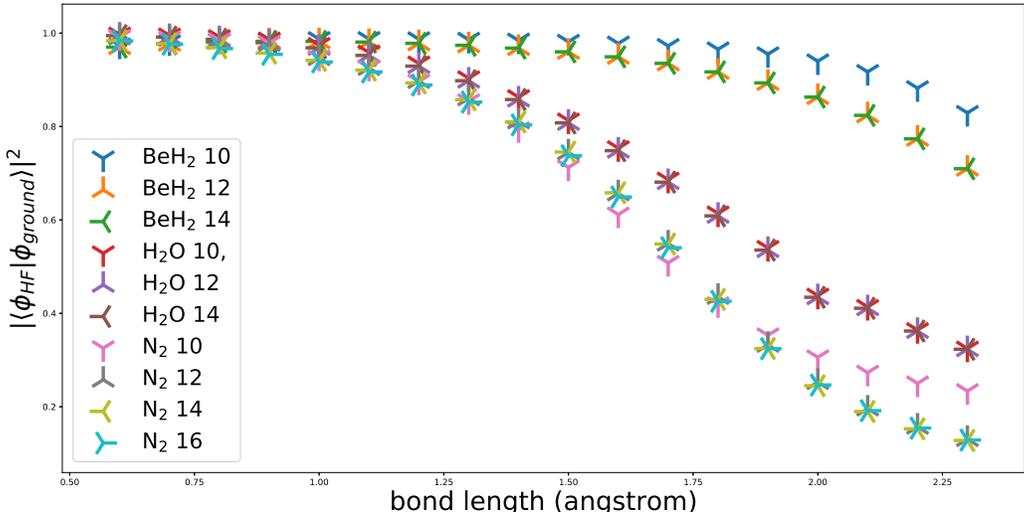
Figure 5.9. The overlap $|\langle\phi_{HF}|\phi_{ground}\rangle|^2$ where $|\phi_{HF}\rangle$ is the input Hartree-Fock state. and $|\phi_{ground}\rangle$ is the exact ground state obtained by diagonalization of the corresponding Hamiltonian for BeH_2, H_2O and N_2 of different active spaces. BeH_2 10 represents the BeH_2 10 qubits Hamiltonian. BeH_2 12 represents the BeH_2 12 qubits Hamiltonian. BeH_2 14 represents the BeH_2 14 qubits Hamiltonian. H_2O 10 represents the H_2O 10 qubits Hamiltonian. H_2O 12 represents the H_2O 12 qubits Hamiltonian. H_2O 14 represents the H_2O 14 qubits Hamiltonian. N_2 10 represents the N_2 10 qubits Hamiltonian. N_2 12 represents the N_2 12 qubits Hamiltonian. N_2 14 represents the N_2 14 qubits Hamiltonian. N_2 16 represents the N_2 16 qubits Hamiltonian.

In simulations, we have shown that increasing the size of active space will have little effect on the accuracy of the qubit coupled cluster VQE for BeH_2. Our QCCSD VQE can still achieve good results for larger active space for H_2O but performs worse than UCCSD VQE for N_2. Here we present the overlap $|\langle\phi_{HF}|\phi_{ground}\rangle|^2$ where $|\phi_{HF}\rangle$ is the input Hartree-Fock state. and $|\phi_{ground}\rangle$ is the exact ground state obtained by diagonalization of the corresponding Hamiltonian for BeH_2, H_2O and N_2 of different active spaces in Figure 5.9. We can see that the overlaps for BeH_2 with different sizes of active space are large, which may indicate very few excitation operators and small amplitudes of excitation operators are needed to approximate the exact ground state and removal of parity terms will have little effect on

results. However, for N_2, the overlap for N_2 is small when the size of active space increases and bond length is large, which may indicate that a large portion of excitation operators and large amplitudes of excitation operators are needed to approximate the exact ground state, thus removal of parity terms may have some effects on results.

In conclusion, we have introduced a new VQE ansatz based on the particle preserving exchange gate [108], [113]. We have shown QCCSD VQE has reduced gate complexity from up-bounded to $O(n^5)$ of UCCSD VQE to up-bounded to $O(n^4)$ if using Jordan-Wigner transformation. In numerical simulations of BeH_2, H_2O, N_2, H_4 and H_6, we have shown that QCCSD VQE have achieved comparable accuracy compared to UCCSD VQE. With reduced complexity and high accuracy, QCCSD VQE ansatz might provide a new promising direction to implement electronic structure calculations on NISQ devices with chemical accuracy.

REFERENCES

[1] R. P. Feynman, "Simulating physics with computers," *International journal of theoretical physics*, vol. 21, no. 6, pp. 467–488, 1982.

[2] M. A. Nielsen and I. Chuang, *Quantum computation and quantum information*, 2002.

[3] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. (2000). "Quantum computation by adiabatic evolution." arXiv: `quant-ph/0001106`.

[4] A. Messiah, *Quantum Mechanics [Vol 1-2]*. 1964.

[5] E. Fradkin, "Jordan-wigner transformation for quantum-spin systems in two dimensions and fractional statistics," *Physical review letters*, vol. 63, no. 3, p. 322, 1989.

[6] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, "Simulation of electronic structure hamiltonians using quantum computers," *Molecular Physics*, vol. 109, no. 5, pp. 735–750, 2011.

[7] J. T. Seeley, M. J. Richard, and P. J. Love, "The bravyi-kitaev transformation for quantum computation of electronic structure," *The Journal of chemical physics*, vol. 137, no. 22, p. 224 109, 2012.

[8] S. B. Bravyi and A. Y. Kitaev, "Fermionic quantum computation," *Annals of Physics*, vol. 298, no. 1, pp. 210–226, 2002.

[9] A. Tranter, P. J. Love, F. Mintert, and P. V. Coveney, "A comparison of the bravyi–kitaev and jordan–wigner transformations for the quantum simulation of quantum chemistry," *Journal of chemical theory and computation*, vol. 14, no. 11, pp. 5617–5630, 2018.

[10] M. Steudtner and S. Wehner, "Fermion-to-qubit mappings with varying resource requirements for quantum simulation," *New Journal of Physics*, vol. 20, no. 6, p. 063 010, 2018.

[11] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, p. 79, 2018.

[12] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, "The theory of variational hybrid quantum-classical algorithms," *New Journal of Physics*, vol. 18, no. 2, p. 023 023, 2016.

[13] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets," *Nature*, vol. 549, no. 7671, p. 242, 2017.

[14] P. J. O'Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding, *et al.*, "Scalable quantum simulation of molecular energies," *Physical Review X*, vol. 6, no. 3, p. 031 007, 2016.

[15] D. R. Herschbach, J. S. Avery, and O. Goscinski, *Dimensional scaling in chemical physics*. Springer Science & Business Media, 2012.

[16] F. Iachello and R. D. Levine, *Algebraic theory of molecules*. Oxford University Press, 1995.

[17] A. Szabo and N. S. Ostlund, *Modern quantum chemistry: introduction to advanced electronic structure theory*. Courier Corporation, 1989.

[18] J. D. Whitfield, P. J. Love, and A. Aspuru-Guzik, "Computational complexity in electronic structure," *Physical Chemistry Chemical Physics*, vol. 15, no. 2, pp. 397–411, 2013.

[19] N. Schuch and F. Verstraete, "Computational complexity of interacting electrons and fundamental limitations of density functional theory," *Nature Physics*, vol. 5, no. 10, p. 732, 2009.

[20] J. D. Whitfield and Z. Zimborás, "On the np-completeness of the hartree-fock method for translationally invariant systems," *The Journal of chemical physics*, vol. 141, no. 23, p. 234 103, 2014.

[21] S. Lloyd, "Universal quantum simulators," *Science*, pp. 1073–1078, 1996.

[22] R. Babbush, P. J. Love, and A. Aspuru-Guzik, "Adiabatic quantum simulation of quantum chemistry," *Scientific reports*, vol. 4, p. 6603, 2014.

[23] S. Boixo, T. F. Rønnow, S. V. Isakov, Z. Wang, D. Wecker, D. A. Lidar, J. M. Martinis, and M. Troyer, "Evidence for quantum annealing with more than one hundred qubits," *Nature Physics*, vol. 10, no. 3, p. 218, 2014.

[24] M. W. Johnson, M. H. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk, *et al.*, "Quantum annealing with manufactured spins," *Nature*, vol. 473, no. 7346, p. 194, 2011.

[25] R. McWeeny and B. T. Sutcliffe, *Methods of molecular quantum mechanics*. Academic press, 1969, vol. 2.

[26] I. G. Rosenberg, "Reduction of bivalent maximization to the quadratic case.," *Cahiers du Centre d'etudes de Recherche Operationnelle*, pp. 71–74, 1975.

[27] Z. Bian, F. Chudak, W. G. Macready, L. Clark, and F. Gaitan, "Experimental determination of ramsey numbers," *Physical Review Letters*, vol. 111, no. 13, p. 130 505, 2013.

[28] *D-wave's ocean tools.* [Online]. Available: `https://github.com/dwavesystems/dwave-ocean-sdk`.

[29] Z. Huang and S. Kais, "Entanglement as measure of electron–electron correlation in quantum chemistry calculations," *Chemical physics letters*, vol. 413, no. 1, pp. 1–5, 2005.

[30] J. McClean, N. Rubin, K. Sung, I. D. Kivlichan, X. Bonet-Monroig, Y. Cao, C. Dai, E. S. Fried, C. Gidney, B. Gimby, *et al.*, "Openfermion: The electronic structure package for quantum computers," *Quantum Science and Technology*, 2020.

[31] M. Streif, F. Neukart, and M. Leib, "Solving quantum chemistry problems with a d-wave quantum annealer," in *International Workshop on Quantum Technology and Optimization Problems*, Springer, 2019, pp. 111–122.

[32] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[33] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[34] F. Häse, S. Valleau, E. Pyzer-Knapp, and A. Aspuru-Guzik, "Machine learning exciton dynamics," *Chemical Science*, vol. 7, no. 8, pp. 5139–5147, 2016.

[35] F. Häse, C. Kreisbeck, and A. Aspuru-Guzik, "Machine learning for quantum dynamics: Deep learning of excitation energy transfer properties," *Chemical science*, vol. 8, no. 12, pp. 8419–8426, 2017.

[36] G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, and O. A. Von Lilienfeld, "Machine learning of molecular electronic properties in chemical compound space," *New Journal of Physics*, vol. 15, no. 9, p. 095 003, 2013.

[37] Z. W. Ulissi, A. J. Medford, T. Bligaard, and J. K. Nørskov, "To address surface reaction network complexity using scaling relations machine learning and dft calculations," *Nature communications*, vol. 8, p. 14 621, 2017.

[38] F. Brockherde, L. Vogt, L. Li, M. E. Tuckerman, K. Burke, and K.-R. Müller, "Bypassing the kohn-sham equations with machine learning," *Nature communications*, vol. 8, no. 1, p. 872, 2017.

[39] L. Wang, "Discovering phase transitions with unsupervised learning," *Physical Review B*, vol. 94, no. 19, p. 195 105, 2016.

[40] J. Carrasquilla and R. G. Melko, "Machine learning phases of matter," *Nature Physics*, vol. 13, no. 5, p. 431, 2017.

[41] P. Broecker, J. Carrasquilla, R. G. Melko, and S. Trebst, "Machine learning quantum phases of matter beyond the fermion sign problem," *Scientific reports*, vol. 7, no. 1, p. 8823, 2017.

[42] K. Ch'ng, J. Carrasquilla, R. G. Melko, and E. Khatami, "Machine learning phases of strongly correlated fermions," *Physical Review X*, vol. 7, no. 3, p. 031 038, 2017.

[43] E. P. Van Nieuwenburg, Y.-H. Liu, and S. D. Huber, "Learning phase transitions by confusion," *Nature Physics*, vol. 13, no. 5, p. 435, 2017.

[44] L.-F. Arsenault, A. Lopez-Bezanilla, O. A. von Lilienfeld, and A. J. Millis, "Machine learning for many-body physics: The case of the anderson impurity model," *Physical Review B*, vol. 90, no. 15, p. 155 136, 2014.

[45] A. G. Kusne, T. Gao, A. Mehta, L. Ke, M. C. Nguyen, K.-M. Ho, V. Antropov, C.-Z. Wang, M. J. Kramer, C. Long, *et al.*, "On-the-fly machine-learning for high-throughput experiments: Search for rare-earth-free permanent magnets," *Scientific reports*, vol. 4, p. 6367, 2014.

[46] P. De Luna, J. Wei, Y. Bengio, A. Aspuru-Guzik, and E. Sargent, "Use machine learning to find energy materials," *Nature*, vol. 552, no. 7683, pp. 23–25, 2017.

[47] J. N. Wei, D. Duvenaud, and A. Aspuru-Guzik, "Neural networks for the prediction of organic chemistry reactions," *ACS central science*, vol. 2, no. 10, pp. 725–732, 2016.

[48] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, p. 195, 2017.

[49] S. Lloyd, M. Mohseni, and P. Rebentrost. (2013). "Quantum algorithms for supervised and unsupervised machine learning." arXiv: `1307.0411`.

[50] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Physical review letters*, vol. 113, no. 13, p. 130 503, 2014.

[51] H. Neven, G. Rose, and W. G. Macready. (2008). "Image recognition with an adiabatic quantum computer i. mapping to quadratic unconstrained binary optimization." arXiv: 0804.4457.

[52] H. Neven, V. S. Denchev, G. Rose, and W. G. Macready, "Training a binary classifier with the quantum adiabatic algorithm," 2008. arXiv: 0811.0416.

[53] H. Neven, V. S. Denchev, G. Rose, and W. G. Macready. (2009). "Training a large scale classifier with the quantum adiabatic algorithm." arXiv: 0912.0779.

[54] N. Wiebe, A. Kapoor, and K. M. Svore, "Quantum deep learning," *Quantum Information & Computation*, vol. 16, no. 7-8, pp. 541–587, 2016.

[55] G. Carleo and M. Troyer, "Solving the quantum many-body problem with artificial neural networks," *Science*, vol. 355, no. 6325, pp. 602–606, 2017.

[56] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, "Neural-network quantum state tomography," *Nature Physics*, vol. 14, no. 5, p. 447, 2018.

[57] S. Kais, "Introduction to quantum information and computation for chemistry," *Quantum Information and Computation for Chemistry*, pp. 1–38, 2014.

[58] A. Daskin and S. Kais, "Direct application of the phase estimation algorithm to find the eigenvalues of the hamiltonians," *Chemical Physics*, 2018.

[59] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, "Simulated quantum computation of molecular energies," *Science*, vol. 309, no. 5741, pp. 1704–1707, 2005.

[60] I. Kassal, J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, "Simulating chemistry using quantum computers," *Annual review of physical chemistry*, vol. 62, pp. 185–207, 2011.

[61] I. D. Kivlichan, J. McClean, N. Wiebe, C. Gidney, A. Aspuru-Guzik, G. K.-L. Chan, and R. Babbush, "Quantum simulation of electronic structure with linear depth and connectivity," *Physical review letters*, vol. 120, no. 11, p. 110 501, 2018.

[62] G. Carleo, M. Troyer, G. Torlai, R. Melko, J. Carrasquilla, and G. Mazzola, "Neural-network quantum states," *Bulletin of the American Physical Society*, 2018.

[63] S. Bravyi, J. M. Gambetta, A. Mezzacapo, and K. Temme. (2017). "Tapering off qubits to simulate fermionic hamiltonians." arXiv: 1701.08213.

[64] R. Xia, T. Bian, and S. Kais, "Electronic structure calculations and the ising hamiltonian," *The Journal of Physical Chemistry B*, vol. 122, no. 13, pp. 3384–3395, 2017.

[65] T. Bian, D. Murphy, R. Xia, A. Daskin, and S. Kais, "Quantum computing methods for electronic states of the water molecule," *Molecular Physics*, pp. 1–14, 2019.

[66] R. M. Parrish, L. A. Burns, D. G. Smith, A. C. Simmonett, A. E. DePrince III, E. G. Hohenstein, U. Bozkaya, A. Y. Sokolov, R. Di Remigio, R. M. Richard, *et al.*, "Psi4 1.1: An open-source electronic structure program emphasizing automation, advanced libraries, and interoperability," *Journal of chemical theory and computation*, vol. 13, no. 7, pp. 3185–3197, 2017.

[67] C. M. Dobson, "Chemical space and biology," *Nature*, vol. 432, no. 7019, p. 824, 2004.

[68] L. C. Blum and J.-L. Reymond, "970 million druglike small molecules for virtual screening in the chemical universe database gdb-13," *Journal of the American Chemical Society*, vol. 131, no. 25, pp. 8732–8733, 2009.

[69] D. P. Tabor, L. M. Roch, S. K. Saikin, C. Kreisbeck, D. Sheberla, J. H. Montoya, S. Dwaraknath, M. Aykol, C. Ortiz, H. Tribukait, *et al.*, "Accelerating the discovery of materials for clean energy in the era of smart automation," *Nat. Rev. Mater.*, 2018.

[70] D. P. Tabor, L. M. Roch, S. K. Saikin, C. Kreisbeck, D. Sheberla, J. H. Montoya, S. Dwaraknath, M. Aykol, C. Ortiz, H. Tribukait, *et al.*, "Accelerating the discovery of materials for clean energy in the era of smart automation," *Nat. Rev. Mater.*, vol. 3, pp. 5–20, 2018.

[71] O. A. von Lilienfeld, "Quantum machine learning in chemical compound space," *Angewandte Chemie International Edition*, vol. 57, no. 16, pp. 4164–4169, 2018.

[72] R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. Von Lilienfeld, "Quantum chemistry structures and properties of 134 kilo molecules," *Scientific data*, vol. 1, p. 140 022, 2014.

[73] S. Kais, *Quantum information and computation for chemistry*. John Wiley & Sons, 2014.

[74] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. Sawaya, *et al.*, "Quantum chemistry in the age of quantum computing," *Chemical reviews*, 2018.

[75] H. Wang, S. Kais, A. Aspuru-Guzik, and M. R. Hoffmann, "Quantum algorithm for obtaining the energy spectrum of molecular systems," *Physical Chemistry Chemical Physics*, vol. 10, no. 35, pp. 5388–5393, 2008.

[76] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'brien, "A variational eigenvalue solver on a photonic quantum processor," *Nature communications*, vol. 5, p. 4213, 2014.

[77] Z. Hu, R. Xia, and S. Kais, "A quantum algorithm for evolving open quantum dynamics on quantum computing devices," *Scientific reports*, vol. 10, no. 1, pp. 1–9, 2020.

[78] P.-L. Dallaire-Demers and N. Killoran. (2018). "Quantum generative adversarial networks." arXiv: 1804.08641.

[79] S. Lloyd and C. Weedbrook, "Quantum generative adversarial learning," *Physical review letters*, vol. 121, no. 4, p. 040502, 2018.

[80] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," *Physical Review A*, vol. 98, no. 3, p. 032309, 2018.

[81] J. Romero, J. P. Olson, and A. Aspuru-Guzik, "Quantum autoencoders for efficient compression of quantum data," *Quantum Science and Technology*, vol. 2, no. 4, p. 045001, 2017.

[82] R. Xia and S. Kais, "Quantum machine learning for electronic structure calculations," *Nature Comuunications*, 2018.

[83] M. Schuld and N. Killoran, "Quantum machine learning in feature hilbert spaces," *Physical review letters*, vol. 122, no. 4, p. 040504, 2019.

[84] Y. Cao, G. G. Guerreschi, and A. Aspuru-Guzik. (2017). "Quantum neuron: An elementary building block for machine learning on quantum computers." arXiv: 1711.11240.

[85] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.

[86] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the national academy of sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.

[87] Y. Du, M.-H. Hsieh, T. Liu, and D. Tao. (2018). "The expressive power of parameterized quantum circuits." arXiv: 1810.11922.

[88] J. Romero and A. Aspuru-Guzik. (2019). "Variational quantum generators: Generative adversarial quantum machine learning for continuous distributions." arXiv: 1901.00848.

[89] Y. Levine, O. Sharir, N. Cohen, and A. Shashua. (2018). "Bridging many-body quantum physics and deep learning via tensor networks." arXiv: 1803.09780.

[90] W. J. Huggins, P. Patil, B. Mitchell, K. B. Whaley, and M. Stoudenmire, "Towards quantum machine learning with tensor networks," *Quantum Science and technology*, 2018.

[91] A. Mari, T. R. Bromley, J. Izaac, M. Schuld, and N. Killoran, "Transfer learning in hybrid classical-quantum neural networks," *Quantum*, vol. 4, p. 340, 2020.

[92] E. Stoudenmire and D. J. Schwab, "Supervised learning with tensor networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 4799–4807.

[93] B. O. Roos, P. R. Taylor, and P. E. Sigbahn, "A complete active space scf method (casscf) using a density matrix formulated super-ci approach," *Chemical Physics*, vol. 48, no. 2, pp. 157–173, 1980.

[94] J. Romero, R. Babbush, J. R. McClean, C. Hempel, P. J. Love, and A. Aspuru-Guzik, "Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz," *Quantum Science and Technology*, vol. 4, no. 1, p. 014 008, 2018.

[95] S. Sim, P. D. Johnson, and A. Aspuru-Guzik, "Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms," *Advanced Quantum Technologies*, vol. 2, no. 12, p. 1 900 070, 2019.

[96] Q. Sun, T. C. Berkelbach, N. S. Blunt, G. H. Booth, S. Guo, Z. Li, J. Liu, J. D. McClain, E. R. Sayfutyarova, S. Sharma, S. Wouters, and G. K.-L. Chan, *Pyscf: The python-based simulations of chemistry framework*, 2017.

[97] G. Aleksandrowicz, T. Alexander, P. Barkoutsos, L. Bello, Y. Ben-Haim, D. Bucher, F. J. Cabrera-Hernández, J. Carballo-Franquis, A. Chen, C.-F. Chen, J. M. Chow, A. D. Córcoles-Gonzales, A. J. Cross, A. Cross, J. Cruz-Benito, C. Culver, S. D. L. P. González, E. D. L. Torre, D. Ding, E. Dumitrescu, I. Duran, P. Eendebak, M. Everitt, I. F. Sertage, A. Frisch, A. Fuhrer, J. Gambetta, B. G. Gago, J. Gomez-Mosquera, D. Greenberg, I. Hamamura, V. Havlicek, J. Hellmers, Ł. Herok, H. Horii, S. Hu, T. Imamichi, T. Itoko, A. Javadi-Abhari, N. Kanazawa, A. Karazeev, K. Krsulich, P. Liu, Y. Luh, Y. Maeng, M. Marques, F. J. Martín-Fernández, D. T. McClure, D. McKay, S. Meesala, A. Mezzacapo, N. Moll, D. M. Rodríguez, G. Nannicini, P. Nation, P. Ollitrault, L. J. O'Riordan, H. Paik, J. Pérez, A. Phan, M. Pistoia, V. Prutyanov, M. Reuter, J. Rice, A. R. Davila, R. H. P. Rudy, M. Ryu, N. Sathaye, C. Schnabel, E. Schoute, K. Setia, Y. Shi, A. Silva, Y. Siraichi, S. Sivarajah, J. A. Smolin, M. Soeken, H. Takahashi, I. Tavernelli, C. Taylor, P. Taylour, K. Trabing, M. Treinish, W. Turner, D. Vogt-Lee, C. Vuillot, J. A. Wildstrom, J. Wilson, E. Winston, C. Wood, S. Wood, S. Wörner, I. Y. Akhalwaya, and C. Zoufal, *Qiskit: An open-source framework for quantum computing*, 2019. DOI: 10.5281/zenodo.2562110.

[98] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[99] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[100] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. Sawaya, *et al.*, "Quantum chemistry in the age of quantum computing," *Chemical reviews*, vol. 119, no. 19, pp. 10 856–10 915, 2019.

[101] A. Y. Kitaev. (1995). "Quantum measurements and the abelian stabilizer problem." arXiv: `quant-ph/9511026`.

[102] D. S. Abrams and S. Lloyd, "Simulation of many-body fermi systems on a universal quantum computer," *Physical Review Letters*, vol. 79, no. 13, p. 2586, 1997.

[103] A. Daskin and S. Kais, "Decomposition of unitary matrices for finding quantum circuits: Application to molecular hamiltonians," *The Journal of chemical physics*, vol. 134, no. 14, p. 144 112, 2011.

[104] A. Daskin, A. Grama, G. Kollias, and S. Kais, "Universal programmable quantum circuit schemes to emulate an operator," *The Journal of chemical physics*, vol. 137, no. 23, p. 234 112, 2012.

[105] M.-H. Yung, J. Casanova, A. Mezzacapo, J. Mcclean, L. Lamata, A. Aspuru-Guzik, and E. Solano, "From transistor to trapped-ion computers for quantum chemistry," *Scientific reports*, vol. 4, p. 3589, 2014.

[106] R. Barends, L. Lamata, J. Kelly, L. García-Álvarez, A. Fowler, A. Megrant, E. Jeffrey, T. White, D. Sank, J. Mutus, *et al.*, "Digital quantum simulation of fermionic models with a superconducting circuit," *Nature communications*, vol. 6, no. 1, pp. 1–7, 2015.

[107] R. J. Bartlett and M. Musiał, "Coupled-cluster theory in quantum chemistry," *Reviews of Modern Physics*, vol. 79, no. 1, p. 291, 2007.

[108] P. K. Barkoutsos, J. F. Gonthier, I. Sokolov, N. Moll, G. Salis, A. Fuhrer, M. Ganzhorn, D. J. Egger, M. Troyer, A. Mezzacapo, *et al.*, "Quantum algorithms for electronic structure calculations: Particle-hole hamiltonian and optimized wave-function expansions," *Physical Review A*, vol. 98, no. 2, p. 022 322, 2018.

[109] Y. Shen, X. Zhang, S. Zhang, J.-N. Zhang, M.-H. Yung, and K. Kim, "Quantum implementation of the unitary coupled cluster for simulating molecular electronic structure," *Physical Review A*, vol. 95, no. 2, p. 020501, 2017.

[110] H. R. Grimsley, D. Claudino, S. E. Economou, E. Barnes, and N. J. Mayhall, "Is the trotterized uccsd ansatz chemically well-defined?" *Journal of Chemical Theory and Computation*, vol. 16, no. 1, pp. 1–6, 2019.

[111] D. Poulin, M. B. Hastings, D. Wecker, N. Wiebe, A. C. Doberty, and M. Troyer, "The trotter step size required for accurate quantum simulation of quantum chemistry," *Quantum Information & Computation*, vol. 15, no. 5-6, pp. 361–384, 2015.

[112] M. Motta, E. Ye, J. R. McClean, Z. Li, A. J. Minnich, R. Babbush, and G. K.-L. Chan. (2018). "Low rank representations for quantum simulation of electronic structure." arXiv: 1808.02625.

[113] D. C. McKay, S. Filipp, A. Mezzacapo, E. Magesan, J. M. Chow, and J. M. Gambetta, "Universal gate for fixed-frequency qubits via a tunable bus," *Physical Review Applied*, vol. 6, no. 6, p. 064007, 2016.

[114] Y. S. Yordanov, D. R. M. Arvidsson-Shukur, and C. H. W. Barnes. (2020). "Efficient quantum circuits for quantum computational chemistry." arXiv: 2005.14475.

[115] I. G. Ryabinkin, T.-C. Yen, S. N. Genin, and A. F. Izmaylov, "Qubit coupled cluster method: A systematic approach to quantum chemistry on a quantum computer," *Journal of chemical theory and computation*, vol. 14, no. 12, pp. 6317–6326, 2018.

[116] I. G. Ryabinkin, R. A. Lang, S. N. Genin, and A. F. Izmaylov, "Iterative qubit coupled cluster approach with efficient screening of generators," *Journal of Chemical Theory and Computation*, vol. 16, no. 2, pp. 1055–1063, 2020.

[117] R. A. Lang, I. G. Ryabinkin, and A. F. Izmaylov. (2020). "Iterative qubit coupled cluster method with involutory linear combinations of pauli products." arXiv: 2002.05701.

[118] B. O'Gorman, W. J. Huggins, E. G. Rieffel, and K. B. Whaley. (2019). "Generalized swap networks for near-term quantum computing." arXiv: 1905.05118.

[119] G. Aleksandrowicz, T. Alexander, P. Barkoutsos, L. Bello, Y. Ben-Haim, D. Bucher, F. Cabrera-Hernández, J. Carballo-Franquis, A. Chen, C. Chen, *et al.*, "Qiskit: An open-source framework for quantum computing," *Accessed on: Mar*, vol. 16, 2019.

[120] B. T. Gard, L. Zhu, G. S. Barron, N. J. Mayhall, S. E. Economou, and E. Barnes, "Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm," *npj Quantum Information*, vol. 6, no. 1, pp. 1–9, 2020.

[121] I. O. Sokolov, P. K. Barkoutsos, P. J. Ollitrault, D. Greenberg, J. Rice, M. Pistoia, and I. Tavernelli, "Quantum orbital-optimized unitary coupled cluster methods in the strongly correlated regime: Can quantum algorithms outperform their classical equivalents?" *The Journal of Chemical Physics*, vol. 152, no. 12, p. 124 107, 2020.

[122] S. E. Smart and D. A. Mazziotti, "Efficient two-electron ansatz for benchmarking quantum chemistry on a quantum computer," *Physical Review Research*, vol. 2, no. 2, p. 023 048, 2020.

[123] D. Kraft, "A software package for sequential quadratic programming," *Forschungsbericht-Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt*, 1988.