Master's Thesis (Academic Year 2023)

Beyond Born-Oppenheimer Approximation Quantum Chemistry Computation on a NISQ Device

Keio University
Graduate School of Media and Governance

Austin Huang

Abstract of Master's Thesis of Academic Year 2023

Beyond Born-Oppenheimer Approximation Quantum Chemistry Computation on a NISQ Device

It is widely believed that quantum computers, by their very quantum nature, are well-suited to problems in quantum chemistry. With the rise of computing power in classical computers, beyond Born-Oppenheimer Approximation effects known as nuclear quantum effects have caught greater attention. Simulating these effects classically, however, has poor scaling, and there may be a potential speed-up by using a quantum computer. Current quantum hardware available is limited due to both noise and a lack of qubits. In order to make use of our current quantum hardware, we turn to a hybrid quantum classical approach known as the Variational Quantum Eigensolver (VQE). Unlike previous work which only used statevector simulators, we simulate the Nuclear-Electronic Orbital (NEO) Hamiltonian on a real quantum device as well as statevector simulators and shot-based simulators using VQE. We further explore the viability of the hardware-efficient Ansatz, the influence of the number of entangling layers, and an approach to optimize the initial point via electronic structure calculations. We demonstrate that current devices are far from able to simulate a 12 qubit NEO Hamiltonian, likely due to an overwhelming amount of shot noise and hardware errors. Furthermore, we demonstrate that the accumulated shot-noise alone can prevent the VQE from reaching convergence, despite VQE's supposed relative noise-resilience. Furthermore, this research shows that - even with a perfect quantum computer - reaching chemical accuracy within a reasonable amount of shots is difficult, preventing any real chemical applicability.

Keio University
Graduate School of Media and Governance

Austin Huang

Contents

List of Figures

List of Tables

Chapter 1

Introduction

> Nature isn't classical, dammit, and if you want to make a simulation of Nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem because it doesn't look so easy.
>
> Richard Feynman

1.1. Background

Since the inception of computation, the simulation of chemical, biological, and material systems have been of great interest [1]. However, simulating such systems is not easy for classical computers [1]–[3]. In fact, Richard Feynman, in his 1981 talk, "Simulating Physics with Computers"[4], claimed that to simulate a quantum mechanical system of a certain size required a classical device of exponentially larger size. Certain systems will always remain out of reach, regardless of advancements in algorithms. Indeed, simulating a system of about 50 qubits can already prove to be extremely challenging, requiring state-of-the-art super computers [5].

Enter the quantum computer. Feynman, in the same talk, hypothesized that a quantum computer, by nature of being quantum mechanical itself, could efficiently simulate a quantum mechanical system. There is a caveat to this statement, however. Not all quantum mechanical systems are built equal. Even classically, some are quite easy to simulate, or have clever heuristics or shortcuts that can make them easy [5]–[7]. Therefore, it is important to clarify that it is the simulation of the general quantum mechanical

system that has proven to be difficult [6].

In terms of chemistry, we are confident that quantum computers have an advantage when it comes to systems that require an explicit representation of the wavefunction [6]. This may be due to stringent accuracy requirements, or because the system is highly entangled. In these cases, the exponential growth of the Hilbert Space is classically intractable [8]. Because of this, many are convinced that quantum computers will likely play a growing role in chemical [3], biological [9], and material [10] simulations. It might be that in the future, certain classical subroutines will be replaced with quantum subroutines, and computational chemistry may take on a hybrid-computing framework [6].

As of right now, quantum computers are what we call Noisy Intermediate-Scale Quantum (NISQ) [5] devices. Like the name suggests, the devices are noisy - which means that these computers are not fault-tolerant. They are also of intermediate size - even though they are noisy, their behavior is now beyond what classical computers can simulate. Before starting this research, there seemed to be some promise regarding the utility of these devices. People hypothesized that despite the noisiness, there may be some real world problems that NISQ devices can tackle efficiently. Unfortunately, as of right now, there is very little evidence that suggests that NISQ devices have any commercial value [11], [12]. However, that does not mean that NISQ devices have no value at all. Indeed, they are still useful for exploratory and scientific purposes [13].

1.2. Research Contribution

This thesis is the first to simulate a Nuclear-Electronic Orbital (NEO) Hamiltonian on an actual NISQ device and on shot-based simulators, moving past previous attempts that used statevector simulators [14], [15]. The research explores the viability of the hardware-efficient Ansatz, and examines the influence of the number of entangling layers and the size of parameter space on solution accuracy. Additionally, it investigates the phenomenon of barren plateaus [16], [17] induced by random initialization, emphasizing the necessity for effective initial points. It introduces a modular approach, where one can first optimize the parameters based on the electronic structure, and use those parameters as a starting point for VQE on the NEO Hamiltonian. A critical aspect of this work is the evaluation of optimizer selection, revealing that certain op-

timizers, though successful in noiseless simulations, fail to converge on real devices [18]. It further highlights the difficulty of achieving chemical accuracy even on noiseless shot-based simulators. The study finally concludes on the capabilities of noiseless shot-based simulators and real quantum environments.

1.3. Thesis Structure

The thesis is structured as follows: Chapter 2 serves as an introduction to quantum computation and presents key concepts such as describing the qubit, gate operations and quantum circuits. It will also cover key concepts in quantum chemistry, such as the molecular Hamiltonian, the Born-Oppenheimer Approximation, and second quantization formalism. Chapter 3 defines the problem statement addressed and reviews relevant literature. Chapter 4 offers an overview of computational details, including the workflow and the numerous challenges involved in such a simulation. In Chapter 5, we evaluate the results of our simulation. Finally, Chapter 6 concludes the thesis, and discusses how the ideas from this thesis can be applied generally to many systems.

Chapter 2

Preliminaries

This chapter covers necessary knowledge of quantum computing and quantum chemistry from a computational point of view. It starts with a brief overview of the relationship between quantum chemistry and quantum computing. It then covers the quintessential mathematical tools, such as Dirac notation, that serve as the foundation to do quantum chemistry on a quantum computer. Finally, it provides a gentle amount of chemistry that is absolutely necessary to understanding and engaging with the research presented in this thesis.

2.1. Qubit

A quantum bit, often referred to as a qubit, is the basic unit of computation in quantum computing. It is similar to its classical counterpart, the bit, in that there are only two states: 0 and 1. However, it differs in that, due to its quantum mechanical nature, a qubit has the ability to be in superposition, where it can be both 0 and 1 simultaneously. Not only that, it has the ability to become entangled with other qubits. This difference gives rise to its potential for exponential speedups, but also its fragility [8].

2.1.1 Notation

We use Dirac notation (also known as Bra-Ket notation) to represent qubits. As the simplest example, the zero and one state can be represented as $|0\rangle$ and $|1\rangle$, respectively.

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{2.1}$$

The notation $|\rangle$ is a Ket. $|0\rangle$, for example, is pronounced "ket 0". The ket is a column vector that describes the quantum state of a system. It is a state vector in the Hilbert space. We can also express a single qubit as a linear combination of computational basis states.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle. \tag{2.2}$$

Where the coefficients α and β are complex coefficients, sometimes called *amplitudes*. The sum of α and β are normalized, and the square of either represents the probability that a qubit will be found in said state when a measurement is performed. We will touch on what measurement is in later sections. For now, let's move on.

2.1.2 Bloch Sphere

We can geometrically represent a single qubit in spherical coordinates using what is known as a Bloch Sphere - a three-dimensional unit sphere. It is often used to visualize the behavior of qubits. For example, the $|0\rangle$ or $|1\rangle$ states have the following depictions, as shown in Figure 2.1.

Multiple Qubits

While a single qubit can describe the idea of superposition, it is unable to illustrate *entanglement*. Entanglement, as you'll come to see, is a type of correlation that is stronger than could ever exist between classical systems [8]. Let's consider two qubits. Classically, it is obvious there are only 4 possible states: 00, 01, 10, and 11. Analogously, for qubits, we have $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$. Since all of these can exist in superposition, they are all associated with a complex coefficient or amplitude. The state vector describing two qubits is shown in Equation 2.3:

$$|\psi\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle \tag{2.3}$$

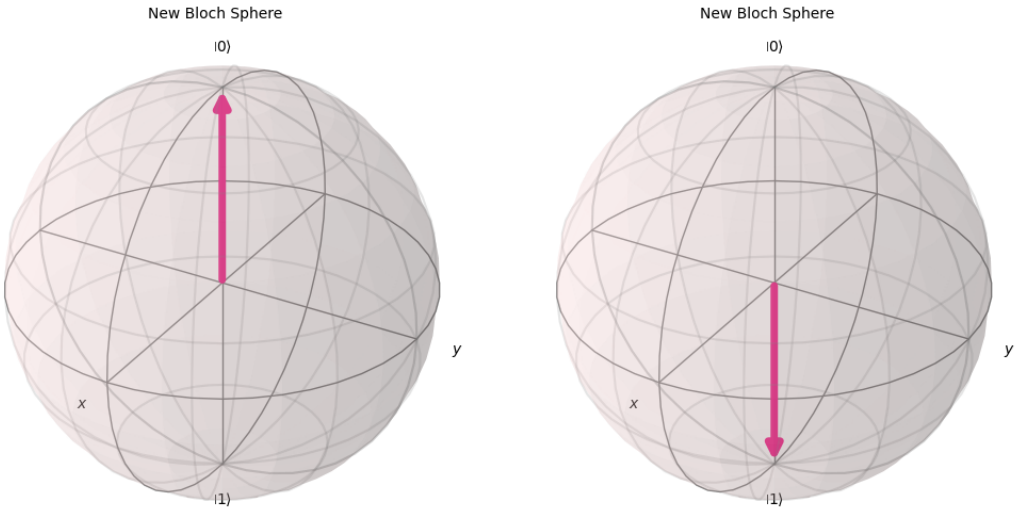


Figure 2.1: Bloch Sphere of $|0\rangle$ and $|1\rangle$

Again, the probabilities must obey the normalization condition, with each probability equalling $|a_x|^2$. It is important to note that each qubit can be measured individually. For example, the probability of measuring 0 for the first qubit would be the sum of the probabilities $|a_{00}|^2 + |a_{01}|^2$.

Bell state

To illustrate entanglement, let's look at the Bell state (Eq. 2.4):

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{2.4}$$

We can see how there are only two possible outcomes. By only measuring the first qubit, we also gain information about the second qubit. In our case, when we measure the first qubit and obtain either 0 or 1, the second qubit will have the same value as the first qubit. In other words, these measurement outcomes are correlated.

This form of correlation by itself is not strange. Suppose we have a pair of shoes, one left and one right. The shoes are placed into boxes such that there is only one shoe in each box. These boxes are then shipped to Alice and Bob, who are at opposite ends of the earth. When Alice opens up her box, she will find her shoe to be the left one, and know that Bob's shoe must be the right one.

Thus, we must consider the key difference between this quantum correlation and classical correlation. In the classical scenario, the shoes are predefined - meaning they had a definite state at the very start. However, in the quantum scenario, the states are not predefined - they are only determined when measured. Thus, for the bell state, when we measure the first qubit - no matter how far the second qubit is - we also determine the state of the second qubit. Additionally, rather than measurement, we can perform other types of operations to our bell state and observe that stronger-than-classical correlation. This type of action is what is referred to as "spooky action at a distance". It is also what won the Nobel Prize in 2022, for proving the universe is not locally real. Turns out, when one is not looking at the moon, the moon really might not exist.

Regardless, it is experiments like these that first suggested to us that the rules of quantum mechanics may allow for information processing beyond that of the classical world [8].

It is here we can once again reference the exponential growth of the Hilbert space. Suppose we have a system of *n* qubits $|x_1x_2...x_n\rangle$. A quantum state of such size is specified by 2^n amplitudes. For even small systems consisting of a few hundred to a few thousand particles, storing such a system would require 2^n bits [8]. Nature does this effortlessly, but for our classical computers this would potentially require more atoms than exists in the universe [8]. So, how can we harness such enormous computational power?

2.2. Quantum Operations

In order to perform computation, it is necessary to be able to manipulate qubits. Physically, there are many ways to implement such a manipulation, but for our purposes, we will think of these operations in its abstract form as unitary matrices. We use the circuit model [8], although other models such as the adiabatic model do exist. In this model, qubits are manipulated via quantum circuits which contain wires and quantum gates. The following section describes common, important quantum gates.

2.2.1 Quantum Gates

Much like how classical computers are controlled with logic gates, quantum computers are controlled via quantum gates. However, unlike classical logic gates, quantum gates are unitary, meaning that they are reversible.

2.2.2 Single Qubit Gates

The following sections will describe important and common single qubit gates. Single qubit gates can be generalized as a rotation around a certain axis n by a certain angle ϕ. For example, Eq 2.5, Eq. 2.6, and Eq. 2.7.

$$Rx(\phi) = \begin{pmatrix} \cos\left(\frac{\phi}{2}\right) & -i\sin\left(\frac{\phi}{2}\right) \\ -i\sin\left(\frac{\phi}{2}\right) & \cos\left(\frac{\phi}{2}\right) \end{pmatrix} \tag{2.5}$$

$$Ry(\phi) = \begin{pmatrix} \cos\left(\frac{\phi}{2}\right) & -\sin\left(\frac{\phi}{2}\right) \\ \sin\left(\frac{\phi}{2}\right) & \cos\left(\frac{\phi}{2}\right) \end{pmatrix} \tag{2.6}$$

$$Rz(\phi) = \begin{pmatrix} e^{-i\frac{\phi}{2}} & 0 \\ 0 & e^{i\frac{\phi}{2}} \end{pmatrix} \tag{2.7}$$

Hadamard Gate

The Hadamard gate is a fundamental quantum operation that puts a qubit into superposition. Its circuit representation is shown in Figure 2.2. Like all quantum gates, it is unitary.

Figure 2.2: Circuit representation of Hadamard gate

In the quantum circuit model, information flows from left to right. The wires carry quantum information, which are manipulated by quantum gates. Each wire is associated with a qubit. If not specified, the initial input state is by default assumed to be $|0\rangle$.

The Hadamard gate can be written in matrix form as Eq. 2.8:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{2.8}$$

Applying the Hadamard gate H on the states $|0\rangle$ and $|1\rangle$ is given by Eq. 2.9 and Eq. 2.10:

$$H|0\rangle = |+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \tag{2.9}$$

$$H|1\rangle = |-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle \tag{2.10}$$

The states $|+\rangle$ and $|-\rangle$ are both states where a qubit is placed into equal superposition. Applying the Hadamard again would recover the $|0\rangle$ and $|1\rangle$ states.

Pauli Gates

The Pauli gates consist of the X gate, Y gate, and Z gate. For mathematical completeness, we also include the identity gate, denoted as I. The I gate does nothing, and its matrix representation is in Eq. 2.11.

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{2.11}$$

Figure 2.3: Circuit representation of a Pauli-X gate

The matrix representation of the Pauli-X gate is given by Eq. 2.12. Its circuit representation is shown in Figure 2.3.

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{2.12}$$

For example:

$$X|0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle. \tag{2.13}$$

The X gate, also known as the bit-flip gate, effectively flips the amplitudes of the states from $|0\rangle$ to $|1\rangle$ and vice versa as shown in Eq. 2.13. In terms of the Bloch sphere, it is a rotation around the X axis by π. In other words, this is the Rx gate, where ϕ is equal to π.

The matrix representations for the Y is shown in Eq. 2.14. Its circuit representation is shown in Figure 2.4:

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \tag{2.14}$$

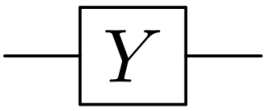


Figure 2.4: Circuit representation of a Pauli-Y gate

The Pauli Z gate, also known as the phase-flip gate, has the following matrix representation, shown in Eq. 2.15. Its circuit representation is shown in Figure 2.5:

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.15}$$

Figure 2.5: Circuit representation of a Pauli-Z gate

The Pauli X, Y, and Z gates are gates around their respective rotation axis by π radians. The Pauli-Y gate is a quantum gate that performs a rotation by π around the y-axis. Similarly, the Pauli-Z gate performs a rotation by π around the z-axis. This leaves

the $|0\rangle$ state unchanged but applies a relative phase flip, transforming the $|1\rangle$ state into $-|1\rangle$.

2.2.3 Two Qubit Gates

In quantum computing, gates that operate on two or more qubits play a crucial role in information transfer and quantum entanglement. Among these, the Controlled-NOT (CNOT) gate and the Controlled-Z (CZ) gate are fundamental.

Controlled-NOT (CNOT) Gate: The CNOT gate, depicted in Figure 2.6, is a two-qubit operation where one qubit acts as the control and the other as the target. If the control qubit is in the state $|1\rangle$, the CNOT gate applies a Pauli X (or NOT) operation to the target qubit. This gate is represented by a 4x4 matrix, as stated in Eq. 2.16:

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{2.16}$$

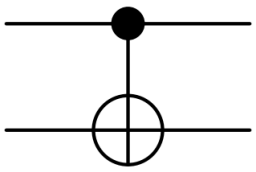


Figure 2.6: Circuit representation of a CNOT gate

In circuit diagrams, the CNOT gate is shown with the control qubit marked by a black dot and the target qubit by a crossed circle. The gate's effect is to flip the target qubit from $|0\rangle$ to $|1\rangle$ (and vice versa) only when the control qubit is $|1\rangle$.

Controlled-Z (CZ) Gate: Similarly, the Controlled-Z or CZ gate, shown in Figure 2.17, involves two qubits. Unlike the CNOT gate, the CZ gate applies a Pauli Z operation to the target qubit when the control qubit is in the state $|1\rangle$. Its matrix form is

stated in Eq. 2.17:

$$\mathrm{CZ} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \tag{2.17}$$

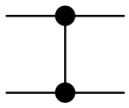


Figure 2.7: Circuit representation of a Controlled-Z gate

The CZ gate is represented in circuit diagrams with both qubits marked by black dots, reflecting the symmetric nature of its operation.

These two gates, CNOT and CZ, are integral in quantum computation for creating entangled states and performing complex quantum operations. They can be found in standard quantum computation textbooks, such as *Quantum Computation and Quantum Information* by Nielsen and Chuang. Additionally, it's notable that the CZ gate can be constructed using two Hadamard gates and one CNOT gate. Finally, it is worth noting that the set containing all single-qubit gates in addition to any two-qubit (e.g. CZ or CNOT) is a universal gate set [8], [19].

2.2.4 Measurement

Measurement is a quintessential part of quantum computation, as it is what we do to extract information about the quantum state. Its circuit representation is shown in Figure 2.8. Unlike measurement in the classical world – which is deterministic – measurement in the quantum world is probabilistic. Unlike all the other operations described above, it is irreversible. It also causes what is known as the *collapse* of the quantum state - meaning that the wave function is no longer in superposition, but rather one of the basis states. For example, if a qubit was in equal superposition between $|0\rangle$ and $|1\rangle$, measurement in the computational basis - the basis quantum computing typically works in - would cause the wavefunction to collapse, thus forcing the qubit to be either in $|0\rangle$ or $|1\rangle$. However, from this single measurement alone it is not clear

if the original qubit was in equal superposition, or instead, for instance, 49% $|0\rangle$ and 51% $|1\rangle$. Thus, it is clear that a *single* measurement of the qubit is insufficient to determine the value of the qubit. In theory, if one had the ability to prepare a quantum state perfectly every time, one could measure said state an infinite amount of times to accurately determine what that state truly was. However, in practice, state preparation and measurement are both imperfect processes, and contribute to the noise we see in quantum computers today.

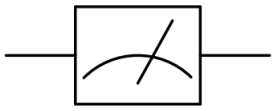


Figure 2.8: Circuit representation of measurement

2.3. Quantum Chemistry

Quantum chemistry, at its core, seeks to understand and predict chemical properties and reactions based on quantum theory, a task traditionally challenging due to the complex behavior of particles at the quantum level. A pivotal component in this field is the general Schrödinger equation, which can be represented in its time-dependent form in Eq. 2.18:

$$i\hbar\frac{\partial}{\partial t}\Psi(\mathbf{r},t) = \hat{H}\Psi(\mathbf{r},t) \tag{2.18}$$

Here, $\Psi(\mathbf{r},t)$ is the time-dependent wave function of the system, $\hat{H}$ is the Hamiltonian operator (representing the total energy of the system), i is the imaginary unit, $\hbar$ is the reduced Planck's constant, and t denotes time.

Focusing on the time-independent aspect of the Hamiltonian offers a more simplified view, essential for many quantum chemistry applications. The time-independent Schrödinger equation, given by Eq. 2.19,

$$\hat{H}\Psi = E\Psi \tag{2.19}$$

is fundamental in solving stationary states of a quantum system, where the Hamiltonian does not explicitly depend on time. This equation is instrumental in understanding the electronic structure of molecules and atoms, paving the way for predicting chemical behavior and properties. Solving the time-independent Hamiltonian allows to calculate the energy eigenvalues of a system. Knowing the possible energy eigenvalues of a system is of great importance - it can help us predict properties about a molecule such as its shape, geometry, reaction rate, and color.

2.3.1 Molecular Hamiltonian

The molecular Hamiltonian can be expressed in atomic units, as shown in Eq. 2.20,

$$\hat{H}_{\text{mol}} = -\sum_i \frac{\nabla^2_{\vec{R}_i}}{2M_i} - \sum_i \frac{\nabla^2_{\vec{r}_i}}{2} - \sum_{i,j} \frac{Z_i}{|\vec{R}_i - \vec{r}_j|} + \sum_{i,j>i} \frac{Z_i Z_j}{|\vec{R}_i - \vec{R}_j|} + \sum_{i,j>i} \frac{1}{|\vec{r}_i - \vec{r}_j|} \quad (2.20)$$

where $\vec{R}_i$, M_i and Z_i indicate the spatial coordinates, masses, and charges of the nuclei in the molecule, and $\vec{r}_i$ are the electronic coordinates.

The first two terms are, respectively, the kinetic energy of the nuclei and the electrons. The latter three terms describe the potential energy of the electron-nucleus attraction, the electron-electron repulsion, and the nucleus-nucleus repulsion. By solving the Schrödinger equation with this Hamiltonian, one can obtain detailed insights into the electronic structure and properties of molecules, which are vital for both theoretical studies and practical applications in chemistry and materials science. Unfortunately, solving this Hamiltonian exactly quickly grows to become intractable. To reduce the exponential computational costs, approximations are of vital importance.

Born Oppenheimer Approximation

The Born-Oppenheimer approximation is based on the assumption that the motion of the nuclei and electrons in a molecule can be separated due to their vast difference in mass. In doing so, we treat all nuclei in a molecule as classical particles, thereby allowing for a clear separation between the electronic and nuclear states (Eq. 2.21):

$$\hat{H}_{\text{mol}} = \hat{H}_{\text{nucl}}(\vec{R}) + \hat{H}_{\text{elec}}(\vec{R}, \vec{r}) \quad (2.21)$$

Similarly, the wave function of the system is separated into nuclear and electronic components (Eq. 2.22):

$$\Psi(\vec{R},\vec{r}) = \phi_{\text{nucl}}(\vec{R}) \cdot \chi_{\text{elec}}(\vec{R},\vec{r}) \tag{2.22}$$

In this framework, for every nuclear configuration $\vec{R}$, a separate electronic eigenvalue problem must be solved (Eq. 2.23):

$$\hat{H}_{\text{elec}}\chi_{\text{elec}}(\vec{R},\vec{r}) = E_{\text{elec}}(\vec{R})\chi_{\text{elec}}(\vec{R},\vec{r}) \tag{2.23}$$

When the BOA is valid, the Electronic Hamiltonian in quantum chemistry becomes Eq. 2.24:

$$\hat{H}_{\text{elec}} = -\sum_i \frac{\nabla^2_{\vec{r}_i}}{2} - \sum_{i,j} \frac{Z_i}{|\vec{R}_i - \vec{r}_j|} + \sum_{i,j>i} \frac{1}{|\vec{r}_i - \vec{r}_j|} \tag{2.24}$$

This formulation significantly simplifies the problem of calculating molecular wavefunctions and energies. By treating the nuclear and electronic components separately, the Born-Oppenheimer Approximation allows for the study of electronic structure with fixed nuclear coordinates, which is a foundational approach in quantum chemistry for understanding molecular behavior and properties.

2.3.2 Second Quantization

In quantum computing, the molecular Hamiltonian is regularly expressed in a standard second-quantized form is shown in Eq. 2.25 :

$$H_{2q} = \sum_{p,q} h_{pq} a_p^\dagger a_q + \sum_{p,q,r,s} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s \tag{2.25}$$

Here, $a_i^\dagger$ and a_i represent the creation and annihilation operators acting on the i-th basis function, respectively. The coefficients h_{pq} and h_{pqrs} are one-body and two-body integrals, which can be efficiently computed (classically) for many choices of basis functions.

The fermionic operators obey the anticommutation relations (Eq. 2.26, **??**:

$$\{a_i, a_j^\dagger\} = a_i a_j^\dagger + a_j^\dagger a_i = \delta_{ij} \tag{2.26}$$

$$\{a_i, a_j\} = \{a_i^\dagger, a_j^\dagger\} = 0 \tag{2.27}$$

The second quantized Hamiltonian holds paramount importance in the realm of quantum chemistry, particularly when contrasted with its first-quantized counterpart. This significance stems from its relative simplicity and direct applicability within the framework of quantum computing as well as its frugality in terms of resources [2]. In essence, the second quantized Hamiltonian is expressed through annihilation and creation operators, which facilitates its transformation into a qubit Hamiltonian, characterized by Pauli terms. Although it is not critical to understand in-depth what the second-quantized Hamiltonian means, the following sections still briefly describe the individual parts that make up this formalism as it is a core component of the computational work detailed in this thesis.

Creation and Annihilation Operators

Creation and annihilation operators play a central role in the second quantization formalism of quantum mechanics. These operators, typically denoted as $a^\dagger$ for the creation operator and a for the annihilation operator, facilitate the representation of quantum states in terms of particle number states.

Vacuum State

The vacuum state, denoted as $|vac\rangle$ or $\langle vac|$, is a fundamental concept in this framework. It represents a state with no particles. This state has no physical meaning. It is but an abstract mathematical entity that is useful for thinking about annihilation and creation operators.

Creating and Annihilating

The action of the annihilation operator on the ith state is defined in Eq. 2.28:

$$a_i|i\rangle = |vac\rangle \tag{2.28}$$

However, when annihilating from the vacuum state (Eq. 2.29), we get 0.

$$a_i|vac\rangle = 0 \tag{2.29}$$

This equation signifies that one cannot annihilate something from the vacuum state, as there is nothing in the vacuum state to be annihilated.

Conversely, the creation operator adds a particle to a given state. Applying the creation operator to the vacuum state yields a single-particle state (Eq. 2.30):

$$a_k^\dagger |vac\rangle = |k\rangle \tag{2.30}$$

This demonstrates the creation of a single-particle state from the vacuum. The newly created state $|k\rangle$ represents a state with one particle.

One-body integrals

In quantum chemistry, one-body and two-body integrals are fundamental mathematical entities that quantify electron interactions within molecular systems. The one-body integral, denoted as h_{pq}, encapsulates the kinetic energy of electrons and their potential energy in the field of atomic nuclei. It is expressed in Eq. 2.31:

$$h_{pq} = \int \psi_p^*(r) \left(-\frac{\hbar^2}{2m} \nabla^2 + V(r) \right) \psi_q(r)\, dr \tag{2.31}$$

Here, $\psi_p(r)$ and $\psi_q(r)$ are the wave functions of the electrons, $\hbar$ is the reduced Planck's constant, m is the electron mass, ∇^2 is the Laplacian operator representing kinetic energy, and $V(r)$ is the potential energy due to the nucleus.

Two-body integrals

In contrast, the two-body integral, represented as h_{pqrs}, describes the electron-electron repulsion energies. This integral is crucial for accounting for the interactions between electron pairs and is given by Eq. 2.32:

$$h_{pqrs} = \int \int \psi_p^*(r_1) \psi_q^*(r_2) \frac{e^2}{|r_1 - r_2|} \psi_r(r_1) \psi_s(r_2)\, dr_1\, dr_2 \tag{2.32}$$

where r_1 and r_2 are the coordinates of the two interacting electrons, and e is the elementary charge. The term $\frac{e^2}{|r_1 - r_2|}$ represents the Coulombic repulsion between the electrons.

Both types of integrals, h_{pq} and h_{pqrs}, are crucial in computational methods of quantum chemistry for accurately calculating molecular energies and properties. For the

purposes of the thesis, it is only necessary to understand that these integrals can be computed efficiently classically, and is a key part of the classical pre-computation necessary for any quantum chemistry workflow.

Chapter 3

Problem Definition

3.1. Why Beyond Born-Oppenheimer

Although the Born-Oppenheimer Approximation can greatly simplify the calculation of molecular systems by decoupling the motion of electrons and nuclei, it does not always hold. ”Neglecting nuclear quantum effects in certain systems, such as those comprised largely of light atoms, can often be a significant source of error” [20]. Beyond Born-Oppenheimer Approximation methods are necessary to simulate hydrogen-bonded systems [21] such as DNA [22] and other biomolecules [23], [24], They are also important in proton-coupled electron transfer reactions [25]–[30], which play a key role in many biological processes [31], [32], such as those in enzymatic reactions [33]. For example, ATP synthesis is powered by a proton gradient across a membrane [34].

Additionally the Born-Oppenheimer Approximation also fails to capture effects such as zero point energy, or tunneling phenomena. These effects lead to significant error in extreme matter conditions, such as where the pressure is extremely high [35] or temperatures are extremely low. Kinetic isotpe effects also become non-existent under the BOA, despite its importance in fields that range from the atmospheric sciences to biochemistry and materials science [20].

Several methodologies, including the Nuclear-Electronic Orbital (NEO) approach [36], pre-Born–Oppenheimer (pBO) quantum theory [37], and Nuclear Orbital plus Molecular Orbital (NO+MO) theory [38], have been developed for an exact quantum description of chemical systems. In this research, we use the NEO approach as it is

able to selectively treat individual protons quantum mechanically [14]. This is especially relevant for near-term NISQ devices, as they will likely not have the resources to treat all nuclei quantum mechanically. Certainly, considering nuclear quantum effects are largely dominated by light atoms, it would make the most sense to pick and choose only the light atoms, such as Hydrogen, to be treated quantum mechanically. On classical computers, NEO methods incur significant computational costs that grow exponentially with system size [14]. Quantum computing, in contrast, offers the ability to both maintain the full wavefunction explicitly, and push for potentially polynomial speed-ups [14]. In this research, we take the NEO approach and simulate the respective NEO Hamiltonian.

3.2. The NEO Method

In the Nuclear Electronic Orbital (NEO) approach, we discard the Born-Oppenheimer approximation. Rather than treat all nuclei as point charges, the NEO approach includes kinetic and potential energy terms only for certain light nuclei in the Hamiltonian. In this work, we label positions of all quantum particles (nuclei and electrons) as $\vec{r}_n$ for nuclei and $\vec{r}_e$ for electrons, respectively. Since the NEO approach can distinguish between quantum nuclei and classical nuclei, in situations where they both exist, classical nuclei positions are represented by $\vec{R}$. In our case, we are not working with any classical nuclei, therefore this is not as important. For the rest of this work, the lower case indices such as i, j, k, l refer to electrons, and the upper case indices such as I, J, K, L to nuclei, regardless of their classical or quantum nature.

The total wave function $\Psi(\vec{r}; \vec{R})$ depends parametrically on classical nuclear coordinates $\vec{R}$. The Schrödinger equation in atomic units [36] is shown in Eq. 3.1:

$$
\begin{aligned}
&\Bigg[-\sum_{i=1}^{N_e} \frac{1}{2}\nabla_i^2 && \text{(Electronic kinetic energy)} \\
&-\sum_{I=1}^{N_n} \frac{1}{2M_I}\nabla_I^2 && \text{(Nuclear kinetic energy)} \\
&-\sum_{i,I} \frac{Z_I}{|\vec{r}_i - \vec{r}_I|} && \text{(Electron-nucleus attraction)} \\
&+\sum_{i<j} \frac{1}{|\vec{r}_i - \vec{r}_j|} && \text{(Electron-electron repulsion)} \\
&-\sum_{i,A} \frac{Z_A}{|\vec{r}_i - \vec{R}_A|} && \text{(Electron-classical nucleus attraction)} \\
&+\sum_{I,A} \frac{Z_I Z_A}{|\vec{r}_I - \vec{R}_A|} && \text{(Quantum nucleus-classical nucleus interaction)} \\
&+\sum_{A<B} \frac{Z_A Z_B}{|\vec{R}_A - \vec{R}_B|} \Bigg] \Psi(\vec{r};\vec{R}) && \text{(Classical nucleus-nucleus repulsion)} \\
&= E\Psi(\vec{r};\vec{R}),
\end{aligned}
\tag{3.1}
$$

where Z_I are nuclear charges, M_I are nuclear masses, and N_e, N_n, and N_c are the total numbers of electrons, quantum nuclei, and classical nuclei, respectively.

The NEO approach solves this equation using electronic and nuclear spin orbitals, $\phi_i(\vec{x}_i) \equiv \phi_i(\vec{r}_i, \vec{s}_i)$ and $\phi_I(\vec{x}_I) \equiv \phi_I(\vec{r}_I, \vec{s}_I)$, respectively, where $\vec{s}_i$ and $\vec{s}_I$ are spin coordinates [14].

These spin orbitals form the basis for the NEO Full Configuration Interaction (NEOFCI) method. The nuclear-electronic wave function is expressed as shown in Eq. 3.2:

$$
\Psi(\vec{x};\vec{R}) = \sum_{\mu,\nu} C_{\mu\nu}\Phi_\mu(\vec{x}_e)\Phi_\nu(\vec{x}_n), \tag{3.2}
$$

where $\Phi_\mu(\vec{x}_e)$ and $\Phi_\nu(\vec{x}_n)$ are Slater determinants or symmetric/antisymmetric products of spin orbitals for electrons and nuclei, respectively [14].

In this work, we consider only the H2 molecule due to its simplicity and ease of simulation. All protons and electrons will be evaluated equally as quantum particles, leading to the following second quantized NEO hamiltonian (Eq. 3.3), [39]

$$\hat{H}_{NEO} = h_{pq}a_p^\dagger a_q + \frac{1}{2}g_{rs}^{pq}a_p^\dagger a_q^\dagger a_s a_r + h_{PQ}a_P^\dagger a_Q + \frac{1}{2}g_{RS}^{PQ}a_P^\dagger a_Q^\dagger a_S a_R - g_{qQ}^{pP}a_p^\dagger a_P^\dagger a_Q a_q \quad (3.3)$$

where h_{pq} and g_{rs}^{pq} refer to the one-body and two-body integrals of electrons, and the upper case equivalents refer to the one-body and two-body integrals of nuclei. The final term is the electron-nuclear repulsion term, where it is a two-body integral of both electron and nuclei.

3.3. Evaluation Metric of Simulation

To our knowledge, no other papers have demonstrated the simulation of a NEO Hamiltonian on any real device, nor shot-based simulator. They have all been tested on noiseless statevector simulators. For the evaluation, we first calculate the exact energies of the systems of interest. We then run the VQE algorithm on a noiseless statevector simulator, noiseless shot-based simulator, and real device and compare the results of each device. We then evaluate the effectiveness of different ansatz parameters, initial points, and optimizers. Finally, we conclude on whether or not current or future devices hold the potential to simulate nuclear quantum effects via the NEO Hamiltonian.

Chapter 4

Computational Details

This chapter covers the specific programs and scripts used to simulate beyond Born-Oppenheimer effects. For reference, the NEO Hamiltonian (previously stated as Equation 3.3) is

$$\hat{H}_{NEO} = h_{pq}a_p^\dagger a_q + \frac{1}{2}g_{rs}^{pq}a_p^\dagger a_q^\dagger a_s a_r + h_{PQ}a_P^\dagger a_Q + \frac{1}{2}g_{RS}^{PQ}a_P^\dagger a_Q^\dagger a_S a_R - g_{qQ}^{pP}a_p^\dagger a_P^\dagger a_Q a_q \quad (3.3)$$

4.1. General Workflow

The general workflow is as follows:

1. **Classical Pre-computation:** In order to implement the NEO Hamiltonian, we first compute the one-body and two-body terms h_{pq}, g_{rs}^{pq}, h_{PQ}, g_{RS}^{PQ}, and g_{qQ}^{pP} classically using a computational chemistry library.

2. **Construct the NEO Hamiltonian:** Build the NEO Hamiltonian - which is in terms of annihilation and creation operators - using the one-body and two-body terms. From now on we will refer to this as the fermionic operator.

3. **Mapping:** The constructed fermionic operator is then mapped to the Qubit basis using the Jordan-Wigner mapping. This is a built-in function provided by Qiskit [40].

4. **Solve** The constructed qubit Hamiltonian (now in Pauli terms) is then solved using either the VQE or by diagonalization. We diagonalize the Hamiltonian in order to verify the validity of both the constructed Fermionic Operator and the mapped Qubit Hamiltonian. We run VQE for our calculations on the real device and simulators.

5. For further details regarding the classical pre-computation and the procedure to construct the Hamiltonian, please refer to the Appendix.

4.1.1 Basis sets

In this work, we choose to use the minimal basis sets of STO6G for the electrons and DZSNB for the nuclei. We choose these basis sets to allow faster iteration and to minimize the computation cost. Larger, more descriptive basis sets are more accurate, but they would also require computational power. For example, upgrading from the minimal basis set to 631G - another common choice - would double the amount of qubits needed to describe our electronic sub-system.

4.1.2 Fermionic to Qubit

To simulate chemical systems in the second quantised representation on a quantum computer, we need to map from operators which act on indistinguishable fermions to operators acting on distinguishable qubits. An encoding method is to map from the fermionic Fock space to the Hilbert space of qubits, such that every fermionic state can be represented by a qubit state [41]. For this work, we will talk only about the Jordan-Wigner mapper as it is the mapper used. It is also relatively straightforward, as it maps one spin-orbital to one qubit.

Jordan-Wigner Mapping

The Jordan-Wigner transformation, introduced by Jordan and Wigner in 1928, maps a spin operator (spin-orbital) to a fermionic operator. It is stored in the state $|0\rangle$ or $|1\rangle$ of a qubit, representing unoccupied and occupied states, respectively. This is formally expressed in Eq. 4.1

$$|f_{M-1}, f_{M-2}, \ldots, f_0\rangle \rightarrow |q_{M-1}, q_{M-2}, \ldots, q_0\rangle, \tag{4.1}$$

where $q_p = f_p \in \{0, 1\}$.

The fermionic operator, which is composed of creation and annihilation operators, introduce a multiplicative phase factor, which is represented in qubit mappings that preserve these characteristics [41]. These operators are defined in Eq. 4.2 and Eq. 4.3.

$$a_p = Q_p \otimes Z_{p-1} \otimes \cdots \otimes Z_0, \tag{4.2}$$

$$a_p^\dagger = Q_p^\dagger \otimes Z_{p-1} \otimes \cdots \otimes Z_0, \tag{4.3}$$

where $Q = |0\rangle\langle 1| = \frac{1}{2}(X + iY)$ and $Q^\dagger = |1\rangle\langle 0| = \frac{1}{2}(X - iY)$. The operator Q or $Q^\dagger$ alters the occupation number of the target spin-orbital, while the series of Z operators account for the exchange phase factor $(-1)^{\sum_{i=0}^{p-1} f_i}$ [41].

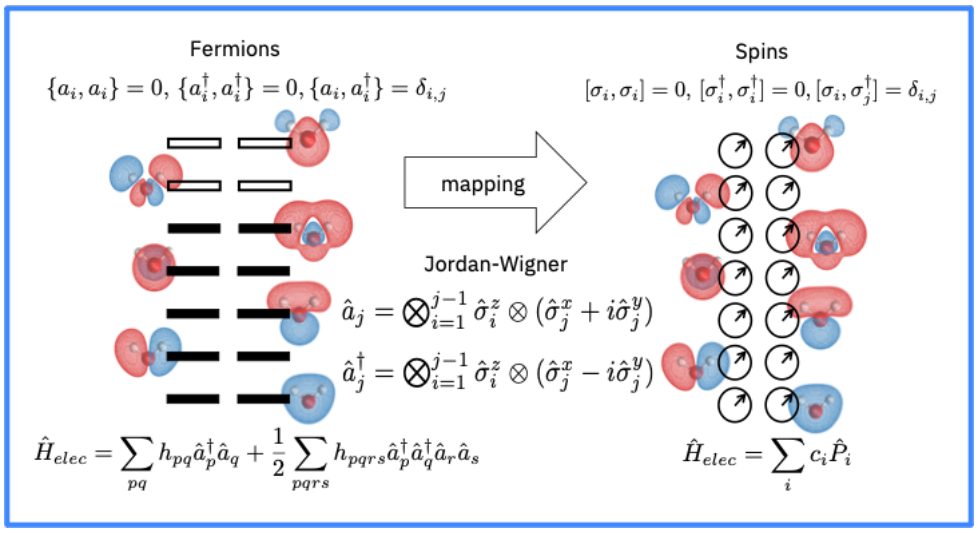


Figure 4.1: The Jordan Wigner Mapping [42]

In the Jordan-Wigner transformation, the second quantised fermionic Hamiltonian is mapped to a linear combination of products of single-qubit Pauli operators [41], shown in Eq. 4.4

$$H = \sum_j h_j P_j = \sum_j h_j \prod_i \sigma_j^i, \tag{4.4}$$

where h_j is a real scalar coefficient, σ_j^i is one of the Pauli operators I, X, Y, or Z, i indicates the qubit the operator acts upon, and j labels the term in the Hamiltonian [41].

4.2. Quantum Computation

The emergence of hybrid quantum-classical (HQC) algorithms marks a significant advancement in computational strategy [16]. These algorithms are designed to optimize the unique strengths of both quantum and classical computing systems. By dividing computational tasks between the two, HQC algorithms capitalize on their respective advantages. This approach is particularly relevant for NISQ (Noisy Intermediate-Scale Quantum) devices, which are tailored to operate with a minimal circuit depth [41]. This feature is crucial for ensuring that the algorithm runs within the limited coherence time of these devices. Moreover, extracting the desired information directly from the quantum states generated in these devices is relatively straightforward [41]. We use a very popular hybrid quantum-classical algorithm, VQE (shown in Figure 5.2, to perform our quantum simulation.

4.2.1 Variational Quantum Algorithms

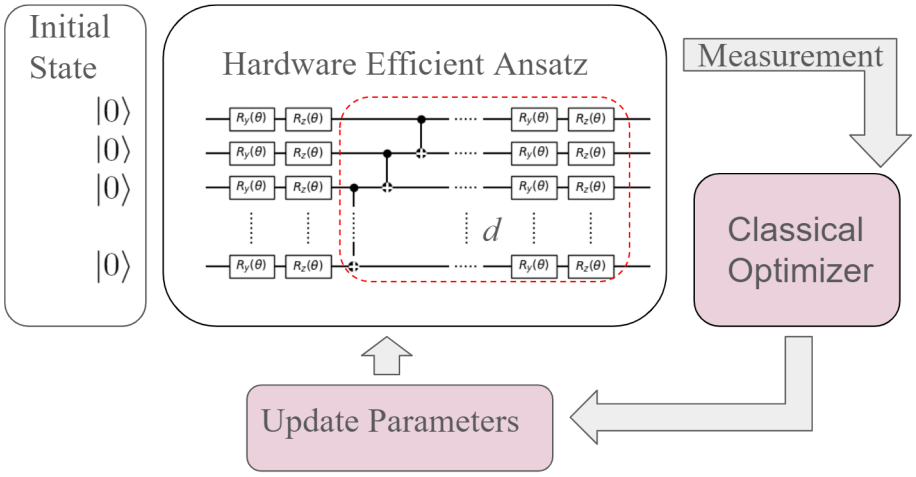


Figure 4.2: Starting from an initial guess for the parameters, the hardware efficient ansatz is prepared using a parametrized circuit on the quantum computer. Outlined in red are the entangling layers, for which repeat d times. Measurements are performed on a computational basis which estimate a cost function E(θ) which returns a value. This value is subsequently passed to a classical optimizer, which then updates the parameters θ and feeds it back into the circuit. This cycle of measurement and parameter update continues until E(θ) converges.

The VQE algorithm is an algorithm that makes use of the quantum computers ability to efficiently maintain a wave function while off-loading the optimization parts to a classical computer [16]. It uses the time-independent variational principle, which states that the expectation value of the Hamiltonian of a system, calculated with any trial wave function, is always greater than or equal to the ground state energy of that Hamiltonian [41]. This means that by varying the parameters of the trial wavefunction, one can minimize this expectation value to approximate the ground state energy as closely as possible [41].

The wave function ansatz can be chosen to take into account the specifications of the quantum hardware in order to reduce gate depth and complexity. These are generally known as Hardware Efficient Ansatz (HEA), and have been used for the simulation of small molecules [43]. However, it is important that the ansatz retain enough expressability to accurately approximate the ground state energy. The energy estimation for the Variational Quantum Eigensolver (VQE) is performed using a technique called *Hamiltonian averaging* [10]. This approach balances coherence time with the ability to sample the quantum computer multiple times [41]. This is why many consider VQE a suitable candidate for simulating the ground states of quantum systems using NISQ devices. The core of the VQE algorithm can be outlined in the following steps:

1. **State Preparation:** A parametrized quantum state $\Psi(\vec{\theta})$ is prepared on the quantum device. This is achieved by applying a parametrized unitary $U(\vec{\theta})$ to an easy-to-prepare initial state $|\Psi_0\rangle$ (e.g., a computational basis state), resulting in $U(\vec{\theta})|\Psi_0\rangle = \Psi(\vec{\theta})$. The parametrized unitary is defined by the choice of ansatz, which should correspond to a family of states that cannot be efficiently represented and manipulated on a classical computer.

2. **Energy Estimation:** The expectation value of the energy $\langle H\rangle(\vec{\theta})$ is estimated using a Hamiltonian averaging procedure. This involves taking measurements of tensor products of Pauli terms corresponding to the qubit representation of the target Hamiltonian.

3. **Classical Feedback:** The parameters $\vec{\theta}$ of the quantum state are updated using a classical non-linear optimization routine.

4. Steps 2 and 3 are repeated until convergence criteria (e.g., energy) are satisfied.

The basic framework of VQE is modular, allowing for various types of extensions and improvements.

Ansatz

The parametrized circuits, or 'ansätze', used in the Variational Quantum Eigensolver (VQE) span a spectrum from hardware-efficient to chemically inspired designs [41]. It can be thought of as an educated guess to the form or structure of the problem at hand. Many times it is inspired by the form of the problem - however, such problem-inspired Ansatz can have deep gate depths, and are not well-suited to the devices we have today. Research comparing the effectiveness of different ansätze for large-scale quantum systems is still in its early stages [41].

Hardware Efficient Ansätze

These ansätze, pioneered in the first VQE experiments, consist of repeated, dense blocks of a few parametrized gates that are straightforward to implement on existing quantum hardware [41]. They aim to create a versatile trial state with as few gates as possible. Such ansätze are well-suited to current quantum computers, which have limited coherence times and gate options. However, they may not be ideal for larger systems as they often ignore the specific details of the chemical systems being simulated. Some recent approaches in this category have focused on integrating chemical principles, like conserving particle number, to enhance their applicability.

Challenges with Hardware Efficient Ansätze

There are notable challenges when using hardware-efficient ansätze. For instance, with random initial parameters, these ansätze can lead to an almost zero energy gradient in many directions in Hilbert space [10]. This issue complicates the optimization process and grows exponentially worse with increasing qubit numbers and circuit depths. Such findings suggest that these ansätze might not be scalable for more complex problems in quantum chemistry [10].

Chemically Inspired Ansätze

Chemically inspired ansätze are adapted from classical computational chemistry methods for efficient execution on quantum computers. A key example is the Unitary Coupled Cluster (UCC) method, which generates a parametrized trial state by considering excitations above a reference state. The UCC method, initially infeasible on classical computers, can be efficiently implemented on quantum computers [44]. It is particularly advantageous due to its fully variational nature and compatibility with multi-reference initial states [44]. The UCC ansatz is often truncated at single and double excitations, known as UCCSD, and requires a significant but manageable number of gates.

Classical Optimizer

Classical optimization plays a pivotal role in the Variational Quantum Eigensolver (VQE), but optimizing complex functions in high-dimensional spaces is challenging. Optimization routines must be fast, accurate, and robust against the significant noise in near-term quantum computers. Classical optimization algorithms are mainly categorized into direct search and gradient-based methods.

Direct Search and Gradient-based Methods

Direct search methods, often more robust to noise, may require more function evaluations than gradient-based methods [45]. Experimental VQE implementations on small systems have shown that direct search methods like Nelder-Mead simplex [46], [47] perform well even with high noise levels.

Challenges with Gradient-based Methods

The simultaneous perturbation stochastic approximation (SPSA) algorithm, an approximate gradient-based method, has been successful in finding ground state energies of small molecules despite uncertainties from shot noise and physical errors [43]. However, traditional gradient descent methods have struggled with high noise levels in current quantum devices [47].

Comparative Studies

Numerical studies comparing optimization algorithms for VQE reveal that while direct search methods and gradient-based methods like LBFGS-B have their strengths, they are prone to becoming trapped in local minima [44]. Effective initialisation strategies, such as using chemically motivated guesses, can mitigate this issue. Stochastic gradient descent [48], variational imaginary time evolution [49], and quantum natural gradient descent [50] have also been explored, though the effectiveness of these methods in realistic noisy conditions remains unclear [41].

Chapter 5

Results

5.1. H^- Calculations

First we verify the validity of our procedure. We begin with a very simple problem - the Hydrogen anion. The hydrogen anion is composed of one proton and two electrons. The simplicity of this system allows us to construct the NEO Hamiltonian by hand. The eigenvalue produced via exact diagonalization of the NEO qubit Hamiltonian is equal to the minimum eigenvalue we find classically. Therefore, we can see that the aforementioned procedure accurately constructs the NEO Hamiltonian in the qubit basis.

Method	Energy/Ha
GAMESS	-0.0955669
Exact Diag.	-0.0955669

Table 5.1: Ground State Energies from GAMESS and diagonalization of Qubit Hamiltonian

5.2. H_2 Calculations

The hydrogen molecule is composed of two protons and two electrons. The nuclear portion and electronic portions are described by 8 spin-orbitals and 4 spin-orbitals respectively. Under the Jordan-Wigner mapping, this leads to a total of 12 qubits. While it

is possible to perform qubit reduction techniques by exploiting the inherent symmetries in the system, that is outside of the scope of this work.

VQE on Statevector Simulator

We compare a total of five common optimizers on a noiseless statevector simulator. Each optimizer is initialized with random parameters. In order to compare the results of our VQE calculation on the real device, we first run VQE on a noise-less statevector simulator. As noted previously, there are mainly three variables that we can tune and adjust. The first is our Ansatz. Although chemistry-inspired Ansatz such as the UCCSD ansatz are relatively popular, they are not suited for the hardware devices we have today. In order to improve the chances our noisy real devices can converge to an accurate solution, we choose to use a hardware-efficient Ansatz, such as the TwoLocal ansatz. In our case, we initialize our TwoLocal ansatz with Ry and Rz as rotational gates, and CNOT as the entangling gates. We use a "reverse-linear" entanglement scheme, which, when used with CNOT gates, produces the same effect as a "full" entanglement scheme. Finally, we try out both three and four entangling layers, and compare the results accordingly in Table 5.2.

The second thing we can adjust is the optimizer of choice. Although VQE is considered to be relatively noise-resilient, the extent of its noise resiliency is affected by the classical optimizer. Some optimizers, such as SPSA, are known to be more noise resilient than others [47]. Interestingly, there is no universal procedure for choosing an optimizer, and the benchmarking data regarding the efficacy of these optimizers is limited [51]. To account for this lack of data, we choose to run our VQE through a variety of optimizers. In doing so, we can compare the optimizers that are sometimes considered "noise-resilient" to those that aren't and from there pick a good optimizer for the VQE calculation on the real device.

The final thing we can do is pick a favorable starting point. Although all of our calculations are done by choosing a random starting point - meaning we start our VQE calculations with a random set of parameters - we could have instead chose a set of parameters from a previous calculation. This procedure is more useful in circumstances where you already have the parameters from a different, but similar calculation, but in our case we can assist our real devices by finding a good starting point from our classical calculations. In our case, we can start with a calculation with only the electronic

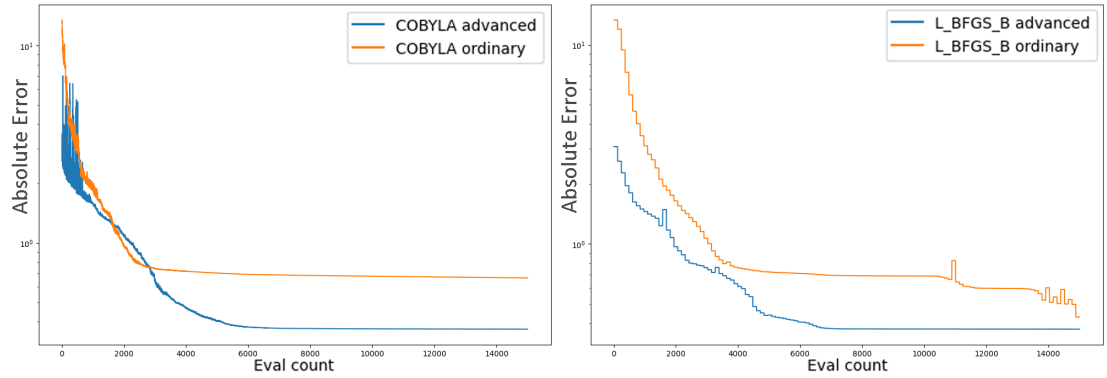
Method	# of Layers	Energy/Ha	Error to Exact
COBYLA	3	-1.007282	0.0470186
	4	-0.3871004	0.6672002
SPSA	3	-0.8979934	0.1563072
	4	-0.3501389	0.7041617
L_BFGS_B	3	-0.6807500	0.3735506
	4	-1.025120	0.0291806
SLSQP	3	-0.6524672	0.4018334
	4	-1.053793	5.076e-4
IMFIL	3	-0.6809014	0.3733992
	4	-1.033556	2.07446e-2

Table 5.2: VQE calculation of H_2 NEO Hamiltonian on noise-less statevector simulator. Here we can see that increasing number of entangling layers does not prevent the optimizer from entering a barren plateau. Additionally, we can see that increasing the number of entangling layers can improve the accuracy of the VQE calculation. The top 3 closest energies were all performed with 4 entangling layers. This suggests that the expanded parameter space provides a notable benefit to the accuracy of the VQE calculation. Furthermore, it suggests that a parameter space that is too small is unable to converge to a sufficiently accurate value.

structure first, and use those parameters to inform our NEO ansatz. In this case, the rest of the parameters for the nuclear subsystem would be set to 0 or close to 0. There is a difficulty here, however, in that one needs to ensure that the parameters are directly matched to the qubits representing their respective subsystem of the NEO Hamiltonian. As seen in figures...We perform this advanced initialization process and show that by choosing a good starting point, we can provide one more advantage to our real devices, and potentially allow it to converge to the real solution.

VQE on noiseless shot-based simulators

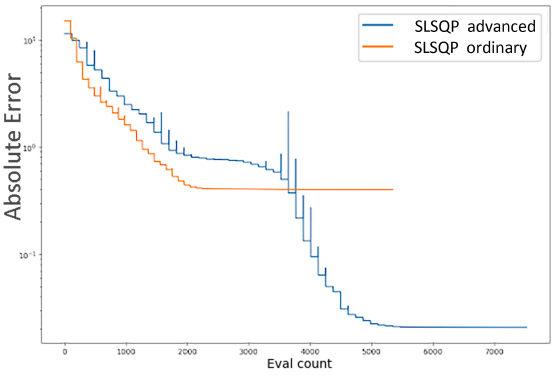
To further emulate the performance on real devices, we trialed several optimizers using random and electronic-structure-based starting points. Before trialing the optimization process, we need to first evaluate the sampling error that arises from using a noiseless shot-based simulator. This allows us to determine a reasonable number of shots for our simulations on the real device. To do this, we use Qiskit's Estimator



(a) COBYLA

(b) L_BFGS_B

(c) SPSA



(d) SLSQP

Figure 5.1: Comparison of Advanced vs Ordinary (random) initialization for various optimizers on statevector simulator

primitive, and use the optimized parameters from the previous statevector simulations. In theory, given infinite shots, we expect the average value to be the same as that of the statevector simulator. However, in our case, we would expect that the average error to decrease as the number of shots increase. Furthermore, we expect the standard deviation to decrease as we increase the number of shots as well.

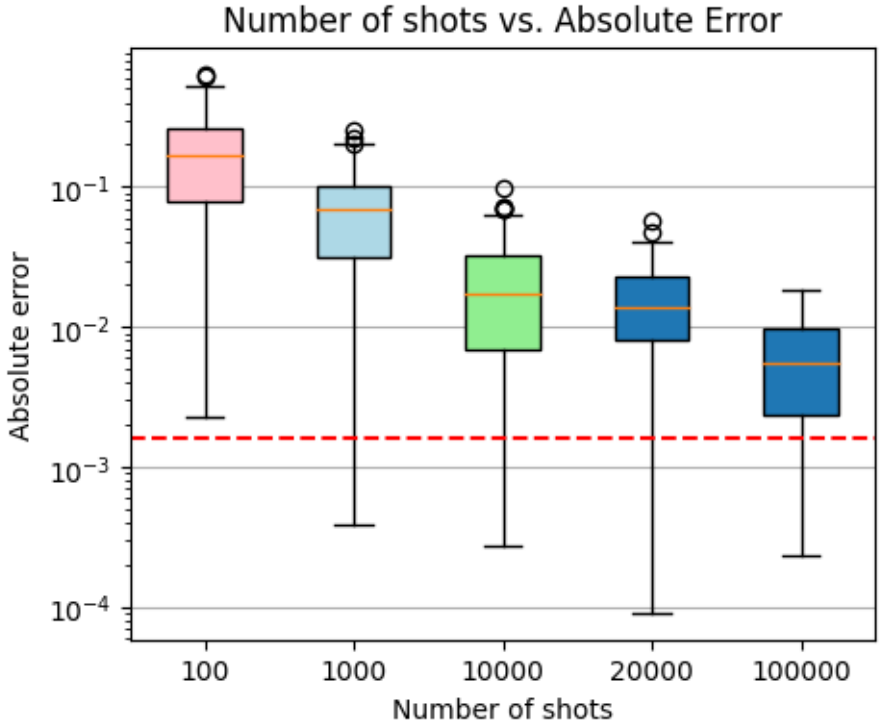


Figure 5.2: Evaluation at optimal parameters on noiseless QASM simulator. Red line is the chemical accuracy, defined at 0.00159 Hartrees.

As expected, we find that the error decreases significantly as we increase the number of shots. Since there is not a huge increase in performance from 10000 to 20000 shots, we used 10000 shots for all of our simulations with the shot-based simulator to save time. Following up, we trialed COBYLA, SLSQP, L_BFGS_B, BOBYQA and IMFIL. The latter two optimizers being recommended from literature as being particularly noise-resilient optimizers [52].

As shown in Table 5.3, the VQE cannot converge to the true value of the NEO Hamiltonian. It appears that shot-noise alone is able to overwhelm the optimization process, and that further studies into better optimizers and mitigating shot-noise is of

great importance. COBYLA, as predicted, performed respectably, which is expected due to the literature suggesting its efficacy in noisy environments [51], [52] . It also suggests that there is no one-size-fits-all optimizer, as the optimizer that performed best in noise-less simulations (SLSQP) performed the worst here in the shot-noise simulation.

Method	Converged	Iterations	Error to Exact
COBYLA	No	4698	1.699836
SLSQP	No	8069	14.13756
L_BFGS_B	No	5398	2.963426
BOBYQA	No	7178	3.230881
IMFIL	No	13923	0.725287

Table 5.3: Five optimizers are trialed on a shot-based simulator for the NEO Hamiltonian simulation using VQE. None are able to reach convergence, although amongst them IMFIL is closest.

VQE on IBM Kawasaki and other real devices

For our attempts on the real system, we initially attempted a fully randomized initial starting position with the same hardware-efficient Ansatz with a gate-depth of 97. At 10000 shots per run and an error mitigation level of 2, we find that the COBYLA optimizer is completely overwhelmed with noise and unable to converge towards the correct answer as expected. It is also important to note that each iteration takes about 8 minutes, meaning a 120 iterations took about 1000 minutes, which is about 16 hours. This is shown in Figure 5.3.

Due to the additional time required from the error mitigation, the next experiment was done without any error mitigation and at 8192 shots. We confirm that even by starting at a pre-optimized starting position (from a previous classical statevector simulation) the COBYLA optimizer is unable to converge to an acceptable answer. This is shown in Figure 5.4. Compared to the random initialization, the eigenvalues of this optimization are much closer to the real values. In this case, each iteration took about 3 minutes.

While changing the initial point is able to improve the accuracy of the optimization, it is nowhere near chemical accuracy. It is clear that the optimization is overwhelmed by noise. In order to check if it might be ”possible” to accurately determine the energies

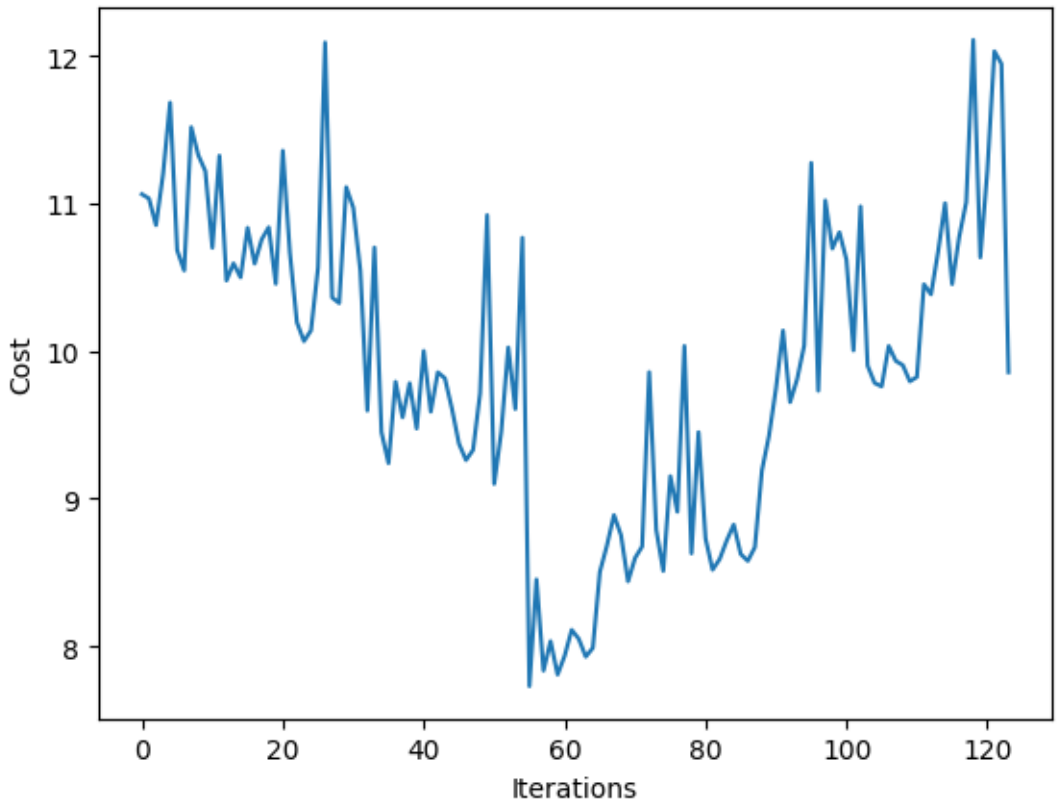


Figure 5.3: Energy Convergence for IBM_Kawasaki with random inital parameters and error mitigation using COBYLA

of our Hamiltonian, we evaluate the accuracy of the device itself. Using optimized parameters, 100000 shots and the highest level of error mitigation offered by IBM, we ran 100 trials, with each trial taking roughly 20 minutes, as shown in Figure 5.5. In doing so, we found that our device is completely incapable of accurately estimating the energy. Since the cost function relies on being able to accurately estimate the energy, we conclude that this type of VQE calculation on a real device is still premature.

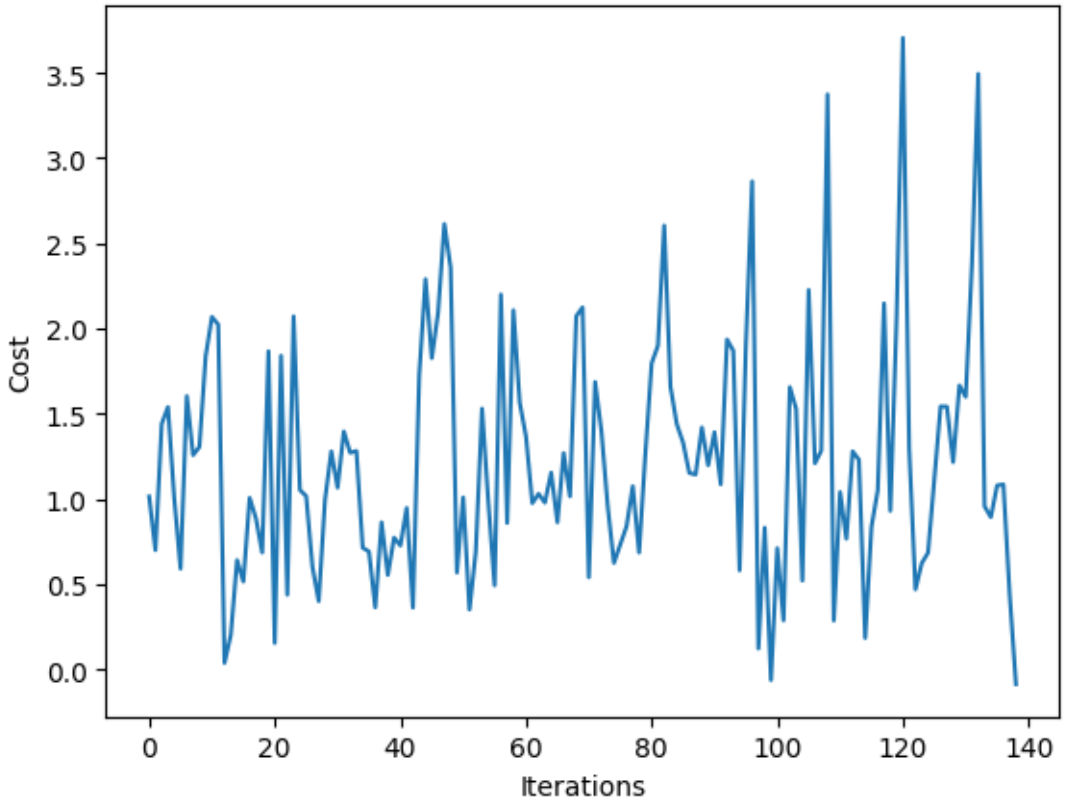


Figure 5.4: Energy Convergence for IBM_Kawasaki with pre-optimized parameters without error mitigation using COBYLA

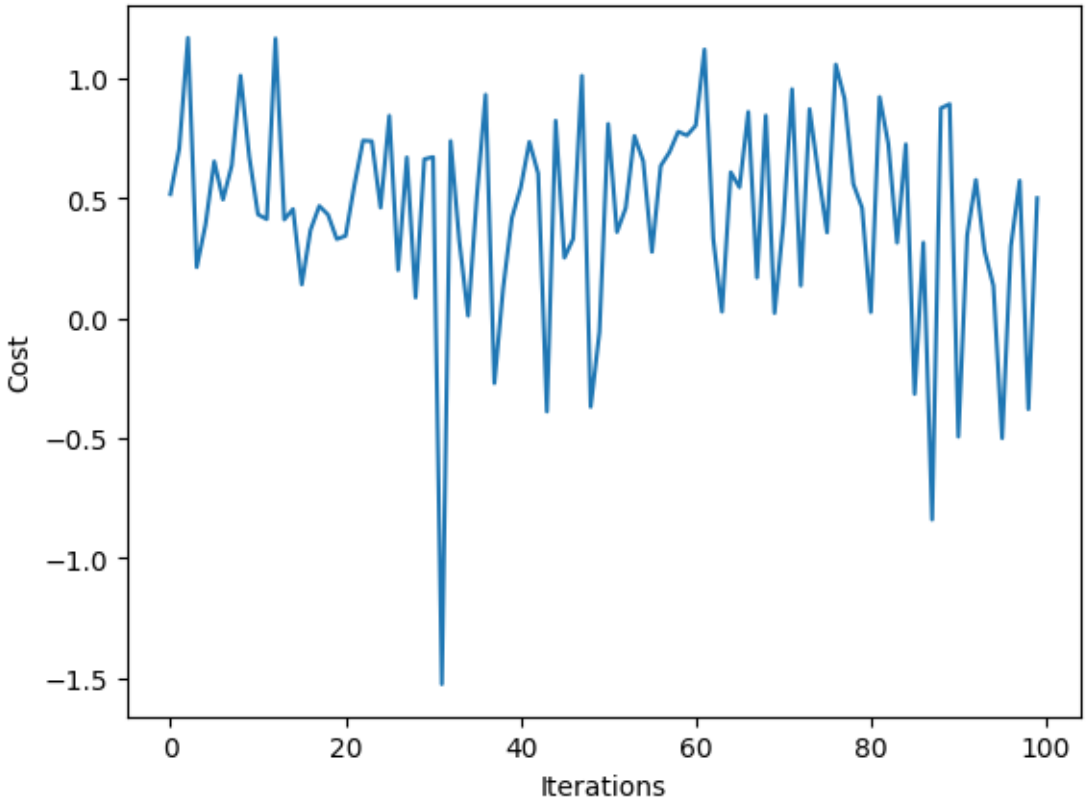


Figure 5.5: Energy Convergence for IBM_Kawasaki with pre-optimized parameters without error mitigation using COBYLA

Chapter 6

Conclusion

6.1. Conclusion

In this thesis, we demonstrated a beyond Born-Oppenheimer Approximation simulation on statevector simulators, shot-based simulators, and noisy real devices. We also demonstrated that one can use the calculations from a previous electronic structure simulation as a good initial point for the nuclear-electronic structure simulation. We conclude that while this type of simulation, even with the best error mitigation techniques, is still far from being realized on a real device. Error rates need to decline dramatically for Ansatz, even those with a gate depth of less than 100, to be viable.

Furthermore, although VQE is considered relatively noise-resilient, we find that even in noiseless shot-based simulators (so-called "perfect" quantum computers), it is not easy to arrive at chemical accuracy using the VQE approach at such gate depths. We arrive at about 10^-2 error with around 10^4 shots. We note that this, however, takes a significant amount of time, even on a simulator. In fact, this would take even more time on a real device that is also noisy. For 10000 shots and no error mitigation, using the quantum runtime that Qiskit provides, each iteration takes about 3 minutes on the quantum device itself. This runtime does not include any of the time that results from queueing, transpilation, or classical optimization. Given that our simulations often require thousands if not ten thousands of iterations, the time cost is still of great concern. Under the very generous assumptions that one needs 10,000 shots at 10,000 iterations to reach chemical accuracy, with each iteration needing about 3 minutes, it would take about 30,000 minutes. That is 500 hours, or about 3 weeks.

Additionally, we investigated the capabilities of a variety of popular optimizers. We find that gradient based optimizers such as SLSQP perform very poorly even in the presence of shot-noise alone. This is in-line with the findings of previous literature [51]. Direct-search optimizers, such as COBYLA, tend to perform better in the presence of noise, but they are not as noise-resilient as they may appear. Finally, IMFIL - which another study [52] evaluated to be fairly noise-resilient - is the most perform of all five tested on shot-based simulators. Unfortunately, none of these optimizers were able to reach convergence on shot-based simulators, indicating a need for accurate but shallow Ansatz as well as ways to mitigate the shot-noise.

6.2. Future Work

Since it is likely that we will remain in the NISQ era for quite some time, in order for NISQ devices to be useable, there will need to be significant needs to improve hardware error rates and error mitigation techniques. Additionally, hardware efficient and gate-efficient methods, such as ADAPT VQE, will grow in importance.

Molecular Systems

On the theoretical side, this type of treatment can also apply to isotope elements such as D_2. We note that on real devices, this is still out of reach, but there is potential in the future to apply the NEO-method to simulate isotope effects and other nuclear quantum effects.

6.3. Code Availability

Available on request.

Acknowledgements

I would like to express my sincere gratitude to my advisor Professor Rodney Van Meter for introducing me to the quantum world and encouraging me to embark on a research project using quantum computers. Without his support and guidance, I would have never been able to simulate quantum chemistry on a real quantum computer. Furthermore, I would like to commend his efforts at AQUA, for he has fostered a group of talented students and faculty that provide valuable insight and opportunities to learn. In particular, I thank Professor Hiroyuki Kusumoto, Dr. Shota Nagayama, Dr. Bernard Ousmane Sane, and Project Associate Professor Michal Hajdušek. I also must acknowledge the help of those in Keio but outside of AQUA – Professor Osamu Nakamura, Professor Jin Mitsugi and others from RG, for their insight has been invaluable.

Additionally, I'd like to thank all the members at AQUA. They are a wonderful group of students who I am proud to teach a little bit of chemistry to every once in a while. I especially thank Naphan Benchasattabuse for his guidance and assistance when I just started learning about quantum computing. His insight has been invaluable. I hope that AQUA continues to grow, continues to educate curious young learners, and continues to advance the field of quantum computing.

This research would not have been possible without the help of the members of Keio Quantum Computing Center (KQCC). I must extend special thanks to Professor Naoki Yamamoto, Professor Kenji Sugisaki, Dr. Shu Kanno, Professor Hiroshi Watanabe, and Dr. Rei Sakuma. Without the help of Professor Kenji Sugisaki, it is likely that I would not have been able to implement the code to create the fermionic operators of the NEO Hamiltonian. Without the help of Professor Hiroshi Watanabe, I would not have been introduced to Professor Haruyuki Nakano, who generously provided me with the codes needed to convert AO integrals to MO integrals.

I would also like to extend my thanks to those at Softbank, especially my supervi-

sors Dr. Ryuji Wakikawa and Yosuke Komiyama. Without Ryuji, I would have never thought about joining Keio University, much less AQUA. Without their enduring support and their commitment to getting me access to real IBM devices, this research would not have been possible. I would also like to thank Koji Kusunoki for his help in navigating Japanese bureaucracy, including helping me join KQCC.

Finally I would like to thank members of the Qiskit community, especially Steve Wood, for always replying promptly.

And of course, I thank all my friends and family for their relentless support.

Bibliography

[1] 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. DOI: 10.1080/00268976.2011.552441.

[2] 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, no. 1, pp. 185–207, 2011, PMID: 21166541. DOI: 10.1146/annurev-physchem-032210-103512. eprint: https://doi.org/10.1146/annurev-physchem-032210-103512. [Online]. Available: https://doi.org/10.1146/annurev-physchem-032210-103512.

[3] A. Aspuru-Guzik *et al.*, "Simulated quantum computation of molecular energies," *Science*, vol. 309, no. 5741, pp. 1704–1707, 2005. DOI: 10.1126/science.1113479.

[4] R. P. Feynman, "Simulating physics with computers," *International Journal of Theoretical Physics*, vol. 21, no. 6/7, pp. 467–488, 1981.

[5] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018. DOI: 10.22331/q-2018-08-06-79.

[6] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya, S. Sim, L. Veis, and A. Aspuru-Guzik, "Quantum chemistry in the age of quantum computing," *Chemical Reviews*, vol. 119, no. 19, pp. 10 856–10 915, 2019, PMID: 31469277. DOI: 10.1021/acs.chemrev.8b00803. eprint: https://doi.org/10.1021/acs.chemrev.8b00803. [Online]. Available: https://doi.org/10.1021/acs.chemrev.8b00803.

[7] J. Preskill, "Quantum computing 40 years later," *arXiv preprint arXiv:2106.10522*, Jun. 2021.

[8] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, 2002, ISBN: 978-0-521-63503-5.

[9] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, "Elucidating reaction mechanisms on quantum computers," *Proceedings of the National Academy of Sciences*, vol. 114, no. 29, pp. 7555–7560, 2017. DOI: 10.1073/pnas.1619152114. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1619152114. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.1619152114.

[10] R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven, and G. K.-L. Chan, "Low-depth quantum simulation of materials," *Phys. Rev. X*, vol. 8, p. 011 044, 1 Mar. 2018. DOI: 10.1103/PhysRevX.8.011044. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevX.8.011044.

[11] S. Chen, J. Cotler, H. Y. Huang, *et al.*, "The complexity of nisq," *Nature Communications*, vol. 14, p. 6001, 2023. DOI: 10.1038/s41467-023-41217-6.

[12] S. Lee, J. Lee, H. Zhai, *et al.*, "Evaluating the evidence for exponential quantum advantage in ground-state quantum chemistry," *Nature Communications*, vol. 14, p. 1952, 2023. DOI: 10.1038/s41467-023-37587-6.

[13] J. J. Meyer, "Fisher information in noisy intermediate-scale quantum applications," *Quantum*, vol. 5, p. 539, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:232404154.

[14] A. Kovyrshin, M. Skogh, A. Broo, S. Mensa, E. Sahin, J. Crain, and I. Tavernelli, "A quantum computing implementation of nuclear-electronic orbital (neo) theory: Toward an exact pre-born–oppenheimer formulation of molecular quantum systems," *Journal of Chemical Physics*, vol. 158, no. 21, p. 214 119, 2023. DOI: 10.1063/5.0150291.

[15] L. Veis, J. Višňák, H. Nishizawa, H. Nakai, and J. Pittner, "Quantum chemistry beyond born–oppenheimer approximation on a quantum computer: A simulated phase estimation study," *International Journal of Quantum Chemistry*, vol. 116, pp. 1328–1336, 2016. DOI: 10.1002/qua.25176.

[16] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G. H. Booth, and J. Tennyson, "The variational quantum eigensolver: A review of methods and best practices," *Physics Reports*, vol. 986, pp. 1–128, 2022, The Variational Quantum Eigensolver: a review of methods and best practices, ISSN: 0370-1573. DOI: https://doi.org/10.1016/j.physrep.2022.08.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0370157322003118.

[17] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *arXiv preprint arXiv:1803.11173*, 2018.

[18] W. Lavrijsen, A. Tudor, J. Müller, C. Iancu, and W. A. de Jong, "Classical optimizers for noisy intermediate-scale quantum devices," *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 267–277, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:215238410.

[19] P. Kaye, R. Laflamme, and M. Mosca, *An Introduction to Quantum Computing*. Oxford University Press, 2007.

[20] T. Markland and M. Ceriotti, "Nuclear quantum effects enter the mainstream," *Nature Reviews Chemistry*, vol. 2, p. 0109, 2018. DOI: 10.1038/s41570-017-0109.

[21] M. Ceriotti *et al.*, "Nuclear quantum effects in water and aqueous systems: Experiment, theory, and current challenges," *Chemical Reviews*, vol. 116, no. 13, pp. 7529–7550, 2016, PMID: 27049513. DOI: 10.1021/acs.chemrev.5b00674. [Online]. Available: https://doi.org/10.1021/acs.chemrev.5b00674.

[22] W. Fang *et al.*, "Inverse temperature dependence of nuclear quantum effects in dna base pairs," *The Journal of Physical Chemistry Letters*, vol. 7, no. 11, pp. 2125–2131, 2016, PMID: 27195654. DOI: 10.1021/acs.jpclett.6b00777. [Online]. Available: https://doi.org/10.1021/acs.jpclett.6b00777.

[23] T. Harris, Q. Zhao, and A. Mildvan, "Nmr studies of strong hydrogen bonds in enzymes and in a model compound," *Journal of Molecular Structure*, vol. 552, no. 1, pp. 97–109, 2000, ISSN: 0022-2860. DOI: https://doi.org/10.1016/S0022-2860(00)00469-5. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022286000004695.

[24] R. H. McKenzie, B. Athokpam, and S. G. Ramesh, "Isotopic fractionation in proteins as a measure of hydrogen bond length," *Journal of Chemical Physics*, vol. 143, no. 4, p. 044 309, 2015. DOI: 10.1063/1.4927391.

[25] S. Hammes-Schiffer, "Proton-coupled electron transfer: Moving together and charging forward," *Journal of the American Chemical Society*, vol. 137, no. 28, pp. 8860–8871, 2015, PMID: 26110700, PMCID: PMC4601483. DOI: 10.1021/jacs.5b04087. [Online]. Available: https://doi.org/10.1021/jacs.5b04087.

[26] M. Sjödin, R. Ghanem, T. Polivka, J. Pan, S. Styring, L. Sun, V. Sundström, and L. Hammarström, "Tuning proton coupled electron transfer from tyrosine: A competition between concerted and step-wise mechanisms," *Physical Chemistry Chemical Physics*, vol. 6, no. 20, pp. 4851–4858, 2004.

[27] R. I. Cukier and D. G. Nocera, "Proton-coupled electron transfer," *Annual Review of Physical Chemistry*, vol. 49, pp. 337–369, 1998, PMID: 9933908. DOI: 10.1146/annurev.physchem.49.1.337. [Online]. Available: https://doi.org/10.1146/annurev.physchem.49.1.337.

[28] S. Hammes-Schiffer, "Theoretical perspectives on proton-coupled electron transfer reactions," *Accounts of Chemical Research*, vol. 34, no. 4, pp. 273–281, 2001, PMID: 11308301. DOI: 10.1021/ar9901117. [Online]. Available: https://doi.org/10.1021/ar9901117.

[29] M. H. Huynh and T. J. Meyer, "Proton-coupled electron transfer," *Chemical Reviews*, vol. 107, no. 11, pp. 5004–5064, 2007, PMID: 17999556, PMCID: PMC3449329. DOI: 10.1021/cr0500030. [Online]. Available: https://doi.org/10.1021/cr0500030.

[30] T. F. Markle, I. J. Rhile, A. G. DiPasquale, and J. M. Mayer, "Probing concerted proton-electron transfer in phenol-imidazoles," *Proceedings of the National Academy of Sciences of the USA*, vol. 105, no. 24, pp. 8185–8190, 2008, PMID: 18212121, PMCID: PMC2448812. DOI: 10.1073/pnas.0708967105. [Online]. Available: https://doi.org/10.1073/pnas.0708967105.

[31] J. Stubbe, D. G. Nocera, C. S. Yee, and M. C. Chang, "Radical initiation in the class i ribonucleotide reductase: Long-range proton-coupled electron transfer?" *Chemical Reviews*, vol. 103, no. 6, pp. 2167–2201, 2003, PMID: 12797828. DOI: 10.1021/cr020421u. [Online]. Available: https://doi.org/10.1021/cr020421u.

[32] S. Hammes-Schiffer and A. V. Soudackov, "Proton-coupled electron transfer in solution, proteins, and electrochemistry," *Journal of Physical Chemistry B*, vol. 112, no. 45, pp. 14 108–14 123, 2008, PMID: 18842015, PMCID: PMC2720037. DOI: 10.1021/jp805876e. [Online]. Available: https://doi.org/10.1021/jp805876e.

[33] J. L. Dempsey, J. R. Winkler, and H. B. Gray, "Proton-coupled electron flow in protein redox machines," *Chemical Reviews*, vol. 110, no. 12, pp. 7024–7039, 2010, PMID: 21082865, PMCID: PMC3005815. DOI: 10.1021/cr100182b. [Online]. Available: https://doi.org/10.1021/cr100182b.

[34] M. Saraste, "Oxidative phosphorylation at the fin de siècle," *Science*, vol. 283, no. 5407, pp. 1488–1493, 1999, PMID: 10066163. DOI: 10.1126/science.283.5407.1488. [Online]. Available: https://doi.org/10.1126/science.283.5407.1488.

[35] M. A. Morales, J. M. McMahon, C. Pierleoni, and D. M. Ceperley, "Nuclear quantum effects and nonlocal exchange-correlation functionals applied to liquid hydrogen at high pressure," *Physical Review Letters*, vol. 110, p. 065 702, 2013. DOI: 10.1103/PhysRevLett.110.065702.

[36] S. Hammes-Schiffer, S. P. Webb, and T. Iordanov, "Multiconfigurational nuclear-electronic orbital approach: Incorporation of nuclear quantum effects in electronic structure calculations," *Journal of Chemical Physics*, vol. 117, pp. 4106–4118, 2002. DOI: 10.1063/1.1501122.

[37] B. Simmen, E. Mátyus, and M. Reiher, "Elimination of the translational kinetic energy contamination in pre-born–oppenheimer calculations," *Molecular Physics*, vol. 111, no. 14-15, pp. 2086–2092, 2013. DOI: 10.1080/00268976.2013.810345.

[38] H. Nakai, "Simultaneous determination of nuclear and electronic wave functions without born–oppenheimer approximation: Ab initio no+mo/hf theory," *International Journal of Quantum Chemistry*, vol. 86, no. 6, pp. 511–517, 2002. DOI: 10.1002/qua.10266.

[39] A. Nykänen, A. Miller, W. Talarico, S. Knecht, A. Kovyrshin, M. Skogh, L. Tornberg, A. Broo, S. Mensa, B. C. B. Symons, E. Sahin, J. Crain, I. Tavernelli, and F. Pavošević, "Toward accurate post-born–oppenheimer molecular simulations on quantum computers: An adaptive variational eigensolver with nuclear-electronic frozen natural orbitals," *Journal of Chemical Theory and Computation*, vol. 19, no. 24, pp. 9269–9277, 2023. DOI: 10.1021/acs.jctc.3c01091.

[40] Qiskit contributors, *Qiskit: An open-source framework for quantum computing*, 2023. DOI: 10.5281/zenodo.2573505.

[41] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, "Quantum computational chemistry," *Rev. Mod. Phys.*, vol. 92, p. 015 003, 1 Mar. 2020. DOI: 10.1103/RevModPhys.92.015003. [Online]. Available: https://link.aps.org/doi/10.1103/RevModPhys.92.015003.

[42] Qiskit Nature Developers, *Qubit mappers*, https://qiskit.org/ecosystem/nature/tutorials/06_qubit_mappers.html, Accessed: [your access date], 2023.

[43] A. Kandala, A. Mezzacapo, K. Temme, *et al.*, "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets," *Nature*, vol. 549, pp. 242–246, 2017. DOI: 10.1038/nature23879.

[44] 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, Oct. 2018. DOI: 10.1088/2058-9565/aad3e4. [Online]. Available: https://dx.doi.org/10.1088/2058-9565/aad3e4.

[45] T. G. Kolda, R. M. Lewis, and V. Torczon, "Optimization by direct search: New perspectives on some classical and modern methods," *SIAM Review*, vol. 45, no. 3, pp. 385–482, 2003. DOI: 10.1137/S003614450242889. eprint: https://doi.org/10.1137/S003614450242889. [Online]. Available: https://doi.org/10.1137/S003614450242889.

[46] C. Hempel, C. Maier, J. Romero, J. McClean, T. Monz, H. Shen, P. Jurcevic, B. P. Lanyon, P. Love, R. Babbush, A. Aspuru-Guzik, R. Blatt, and C. F. Roos, "Quantum chemistry calculations on a trapped-ion quantum simulator," *Phys. Rev. X*, vol. 8, p. 031 022, 3 Jul. 2018. DOI: 10.1103/PhysRevX.8.031022. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevX.8.031022.

[47] A. Peruzzo, J. McClean, P. Shadbolt, *et al.*, "A variational eigenvalue solver on a photonic quantum processor," *Nature Communications*, vol. 5, p. 4213, 2014. DOI: 10.1038/ncomms5213.

[48] R. Sweke, F. Wilde, J. Meyer, M. Schuld, P. K. Faehrmann, B. Meynard-Piganeau, and J. Eisert, "Stochastic gradient descent for hybrid quantum-classical optimization," *Quantum*, vol. 4, p. 314, Aug. 2020, ISSN: 2521-327X. DOI: 10.22331/q-2020-08-31-314. [Online]. Available: http://dx.doi.org/10.22331/q-2020-08-31-314.

[49] S. McArdle, T. Jones, S. Endo, Y. Li, S. C. Benjamin, and X. Yuan, "Variational ansatz-based quantum simulation of imaginary time evolution," *npj Quantum Information*, vol. 5, no. 1, Sep. 2019, ISSN: 2056-6387. DOI: 10.1038/s41534-019-0187-2. [Online]. Available: http://dx.doi.org/10.1038/s41534-019-0187-2.

[50] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, "Quantum natural gradient," *Quantum*, vol. 4, p. 269, May 2020, ISSN: 2521-327X. DOI: 10.22331/q-2020-05-25-269. [Online]. Available: http://dx.doi.org/10.22331/q-2020-05-25-269.

[51] H. Singh, S. Majumder, and S. Mishra, "Benchmarking of different optimizers in the variational quantum algorithms for applications in quantum chemistry," *The Journal of Chemical Physics*, vol. 159, no. 4, p. 044 117, Jul. 2023, ISSN: 0021-9606. DOI: 10.1063/5.0161057. eprint: https://pubs.aip.org/

aip/jcp/article-pdf/doi/10.1063/5.0161057/18065624/044117_1_5.0161057.pdf. [Online]. Available: https://doi.org/10.1063/5.0161057.

[52] W. Lavrijsen, A. Tudor, J. Muller, C. Iancu, and W. de Jong, "Classical optimizers for noisy intermediate-scale quantum devices," in *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2020, pp. 267–277. DOI: 10.1109/QCE49297.2020.00041. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/QCE49297.2020.00041.

Appendix

A. Constructing the Nuclear-Electronic part

Starting with the two-body nuclear-electronic integral obtained from GAMESS and in-house code, we do additional formatting and data parsing to integrate it with Qiskit. We do not show the code for the nuclear and the electronic integrals because that can be done simply with few modifications to built-in qiskit functionality.

```
import numpy as np
import re

from qiskit_nature.second_q.hamiltonians import ElectronicEnergy
from qiskit_nature.second_q.operators import FermionicOp
from qiskit_nature.second_q.mappers import JordanWignerMapper
from qiskit_algorithms import NumPyMinimumEigensolver

# Format of GAMESS integrals is first preprocessed as a tuple with a
    tuple of values and an associated co-efficient e.g. ((4, 4, 2, 2)
    , 2.53724259). For the electronic-nuclear integrals, the first
    two values refer to the nuclei e.g. ((n, n, e, e), 2.53724259)
    They must be mapped separately, and built-in Qiskit code cannot
    handle this case. Here, we assume that the indices for the
    electronic and nuclear portions are already distinct (meaning
    there is no overlap of indices).

data = {}
num_spin_orbitals = [4, 8] ## electronic spin orbitals, nuclear spin
    orbital for H2 under sto6g and DSNZB basis. Depending on the
    system, this will change as well. This must be specified ahead of
     time.

```

```python
# Generate a mapping for 'a' and 'b' based on the number of electronic and nuclear spin orbitals
def generate_mapping(num_spin_orbitals):
    el, nu = num_spin_orbitals
    el = el//2
    nu = nu//2

    mapping = {}
    for i in range(1, el + 1):
        mapping[f'{i}a'] = i - 1  # e.g., '1a' -> 0
        mapping[f'{i}b'] = i - 1 + el  # e.g., '1b' -> max_value

    for j in range(el+1, el+nu+1):
        mapping[f'{j}a'] = j - 1 + el # e.g., '1a' -> 0
        mapping[f'{j}b'] = j - 1 + el + nu  # e.g., '1b' -> max_value

    return mapping


def generate_combinations(inner_tuple):
    # Generate all 'a' and all 'b' combinations for each unique value
    all_a = [f'{value}a' for value in inner_tuple]
    all_b = [f'{value}b' for value in inner_tuple]

    # Generate mixed combinations: first half 'a', second half 'b', and vice versa
    first_half_a_second_half_b = [f'{value}a' if i < len(inner_tuple) / 2 else f'{value}b' for i, value in enumerate(inner_tuple)]
    first_half_b_second_half_a = [f'{value}b' if i < len(inner_tuple) / 2 else f'{value}a' for i, value in enumerate(inner_tuple)]

    # Combine all combinations into a list
    combinations = [all_a, all_b, first_half_a_second_half_b, first_half_b_second_half_a]

    return combinations

# Creates two-body fermionic operator based on values of indices.
# For example, [0,0,1,1] would equal to "+_0 +_0 -_1 -_1".
def generate_fermionic_operator_string(values):
```

```
    operator_str = f'+_{values[0]} +_{values[1]} -_{values[2]} -_{values[3]}'
    return operator_str

def build_fermionic_op_data(input_tuple, num_spin_orbitals):
    inner_tuple, value = input_tuple

    # Generate the mapping
    mapping = generate_mapping(num_spin_orbitals)

    # Generate the combinations
    combinations = generate_combinations(inner_tuple)

    # Convert the combinations to their mapped numeric values
    numeric_combinations = [[mapping[item] for item in combo] for combo in combinations]

    # Swap the second and fourth elements due to Chemist's ordering
    for combo in numeric_combinations:
        combo[1], combo[3] = combo[3], combo[1]

    for combo in numeric_combinations:
        data[generate_fermionic_operator_string(combo)] = value

for integral in en_integrals:
    build_fermionic_op_data(integral, num_spin_orbitals)

fermionic_op_en = FermionicOp(
    data,
    num_spin_orbitals=12,
    copy=False,
)
```

Listing 6.1: Building the electronic-nuclear 2 body coefficients

B. Constructing the NEO Hamiltonian

After constructing the electronic, nuclear, and electronic-nuclear parts, we can add them together using built-in Qiskit functionality.

```
1
2 hamiltonian_n = ElectronicEnergy.from_raw_integrals(one_n,nn_tensor)
    ## onebody and twobody coefficients for nuclei from GAMESS
3 hamiltonian_e = ElectronicEnergy.from_raw_integrals(one_e,ee_tensor)
    ## for electrons
4
5 fermionic_op_n = hamiltonian_n.second_q_op()
6 fermionic_op_e = hamiltonian_e.second_q_op()
7
8 def indice_permutation(fermionic_op, indice, num_spin_orbitals,
    indices_to_increment=None):
9     data = {}
10    for item in fermionic_op.items():
11        res = ""
12        current_index = 0  # Track the current index in the operator
13        for char in item[0]:
14            if char.isnumeric():
15                # If indices_to_increment is None or current_index is
     in indices_to_increment
16                if indices_to_increment is None or current_index in
    indices_to_increment:
17                    char = str(int(char) + indice)
18                current_index += 1
19            res += char
20        data[res] = item[1]
21
22    op = FermionicOp(
23        data,
24        num_spin_orbitals=num_spin_orbitals,
25    )
26    return op
27
28 ### For the fermionic operator of the nucleus portion, we want to
    increment all the indices by 8.
29 ### e.g. from [0 to 3] to [4 to 12]
30 indexed_fermionic_op_n =  indice_permutation(fermionic_op_n, 4, 12)
31
32 fermionic_op = fermionic_op_e + indexed_fermionic_op_n -
    fermionic_op_en
33 jw_mapper = JordanWignerMapper()
```

```
hamiltonian = jw_mapper.map(fermionic_op)
numpy_solver = NumPyMinimumEigensolver()## exact diag...
ans = numpy_solver.compute_minimum_eigenvalue(hamiltonian)
```

Listing 6.2: Constructing and solving the NEO Hamiltonian. The One-body and Two-body tensors are obtained via GAMESS.