

Estimation of Ground State Energies using the Variational Quantum Eigensolver

Charlie Freeman
Cornell University Physics Department
(Dated: March 6, 2024)

The Variational Quantum Eigensolver (VQE) is a hybrid quantum and classical computing algorithm that is used to find the ground state of a physical system. It is particularly useful for simulating larger systems, where it is much more efficient than its classical counterparts, causing it to be able to simulate much larger systems. For the development of this skill, I created algorithms that were able to predict the ground state energy of dihydrogen (H_2) and lithium hydride (LiH) to be -1.13689 ± 10^{-6} Ha and -7.8821 ± 10^{-4} Ha, respectively. These values for the ground state energy are each within the margin of error of the classically computed values, calculated using the full configuration interaction (FCI) method. Predictions of the ground state energy of H_2 were also made successfully at varying bond lengths, although the model broke down at higher bond lengths due to the bond becoming unstable. Using this, the bond length of H_2 was predicted to be $0.742 \pm 0.02\text{\AA}$, in agreement with the FCI prediction. The Python library PennyLane was a very useful tool in building this algorithm.

I. INTRODUCTION

Quantum computing was first proposed in the 1980s by Richard Feynman and Yuri Manin as a way of simulating large physical systems⁽¹⁾. Up until this point, systems of even just a few interacting particles were very difficult to simulate, due to the exponential relationship between size of the system and number of possible quantum states. Systems of about 10-20 active spin orbitals required immense amounts of memory and time to operate on, and systems larger than this became impossible to simulate, even using the world’s most powerful computers. In order to bypass the need for this much memory, the idea of quantum computers was introduced. In such computers, the quantum dynamical processes that are so difficult to simulate classically can be exploited in the machine that is being used for the simulation. Rather than encoding each possible quantum state separately, quantum computers use a single quantum bit (or qubit) to represent each active spin orbital. The qubit, just as the spin orbital it represents, can exist in a superposition of 1 and 0 (representing occupied and unoccupied, respectively), differing from classical bits, which can only be 1 or 0. This allows for a much more efficient way of simulating quantum systems, allowing us to simulate systems much larger than was previously possible.

The VQE was first introduced in 2014 by Peruzzo et al⁽²⁾ as a way to exploit quantum computing to calculate the ground state of a quantum system. It uses the variational principle to calculate the ground state of a quantum system, which has a broad range of applications in fields such as chemistry, material science, and condensed matter physics. For example, knowing the ground state energy is essential in analyzing the stability of a reaction, or determining the excitation energy of a molecule⁽³⁾, both of which have wide ranging applications. There are many different methods for implementing the VQE, but for the purposes of this paper, I will focus on only the methods that can be automated using the Python library PennyLane.

```
import pennylane as qml
from pennylane import numpy as np
#Load the molecular data
dataset = qml.data.load('qchem', molname="H2")[0]
#Define the Hamiltonian and number of qubits
H, qubits = dataset.hamiltonian, len(dataset.hamiltonian.wires)
#Initialize a quantum circuit and define the initial state as the Hartree Fock state
dev = qml.device("lightning.qubit", wires=qubits)
electrons = 2
hf = qml.qchem.hf_state(electrons, qubits)
#Define the circuit as a single double excitation gate
def circuit(param, wires):
    qml.BasisState(hf, wires=wires)
    qml.DoubleExcitation(param, wires=[0, 1, 2, 3])
#Define the optimization algorithm
opt = qml.GradientDescentOptimizer(stepsize=0.4)
```

FIG. 1. Capabilities of PennyLane. We can load molecular properties of H_2 , define the hamiltonian and number of qubits, determine the Hartree Fock state, and define a double excitation gate, all automatically. We can also use the optimization algorithm GradientDescentOptimizer.

II. VQE CONSTRUCTION

A. Hamiltonian

In order to run the VQE algorithm, we first must construct an appropriate hamiltonian to represent the system of interest. We can do this by using the method of second quantization, which looks at each active spin orbital, and tracks whether it is occupied or unoccupied. The second quantized hamiltonian can be written in terms of the creation and annihilation operators as

$$\hat{H} = \sum_{p,q} h_{pq} a_p^\dagger a_q + \sum_{p,q,r,s} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s \quad (1)$$

, where the a^\dagger operators serve to excite an electron to orbital p or orbitals p and q, while the a terms de-excite an electron from orbital q or orbitals r and s. The h_{pq} and h_{pqrs} terms are referred to as the one and two electron integrals, respectively, and can be calculated fairly easily on a classical computer⁽⁴⁾. Only the first two terms of this hamiltonian are generally kept for simplicity. This hamiltonian is in terms of the “fermionic” creation and annihilation operators, which are unable to act on qubits.

In order for our hamiltonian to become compatible with the qubit representation of the wave function, we need to express the hamiltonian in a “spin” basis, such as in terms of Pauli spin matrices. This can be done by using a Jordan-Wigner transformation⁽⁵⁾, which replaces the creation and annihilation operators with sums of Pauli spin matrices:

$$\begin{cases} a^\dagger = \frac{X_j - iY_j}{2} \\ a = \frac{X_j + iY_j}{2} \end{cases} \quad (2)$$

, where X_j and Y_j represent the X and Y Pauli matrices acting on qubit j . This allows us to construct a hamiltonian that is able to act on qubits, which is essential for our algorithm to work properly. It should be noted that, when constructing my algorithm, I automated this process using the Python library PennyLane (see Figure 1).

B. Building the Quantum Circuit

We next must design a quantum circuit that is able to represent the system of interest. In order to do this, we first must define how we will use qubits to label the quantum states. One qubit is assigned to each active spin orbital, which is set to 1 if it is occupied and 0 if it is unoccupied. We can then build the circuit using excitation gates, which allow electrons to jump to a different spin orbital. These gates can be represented in matrix form as a Givens rotation. For example, the double excitation gate acting on the 4 qubit initial state $|1100\rangle$ gives $G_2|1100\rangle = \cos(\theta/2)|1100\rangle - \sin(\theta/2)|0011\rangle$. If we expect the ground state wave function to be equal to this expression for some value of θ , then one double excitation gate is a sufficient circuit to represent the system. More generally, we can build the expected form of the ground state wave function, or ansatz, using a unitary coupled cluster (UCC). This gives the excitation gates in terms of creation and annihilation operators:

$$\begin{cases} \hat{T}_1 = \sum_{ia} t_i^a a_a^\dagger a_i \\ \hat{T}_2 = \sum_{ijab} t_{ij}^{ab} a_a^\dagger a_b^\dagger a_i a_j \end{cases} \quad (3)$$

, where parameters t_i^a and t_{ij}^{ab} are left to optimize. T_1 represents a single electron excitation from orbital i to orbital a , while T_2 represents a double excitation from orbitals i and j to orbitals a and b . For most systems, it is appropriate to ignore all terms of higher order than T_2 . We can once again use the Jordan Wigner transformation to put this into a spin basis, allowing us to write the general ansatz as

$$|\Psi\rangle = e^{\hat{T}}|\Psi_0\rangle \quad (4)$$

, where $\hat{T} = \hat{T}_1 + \hat{T}_2 + \dots$, and Ψ_0 is some initial state—usually the Hartree Fock state in which all electrons occupy their lowest possible orbitals. PennyLane once again can automate most of this, as shown in figure 1.

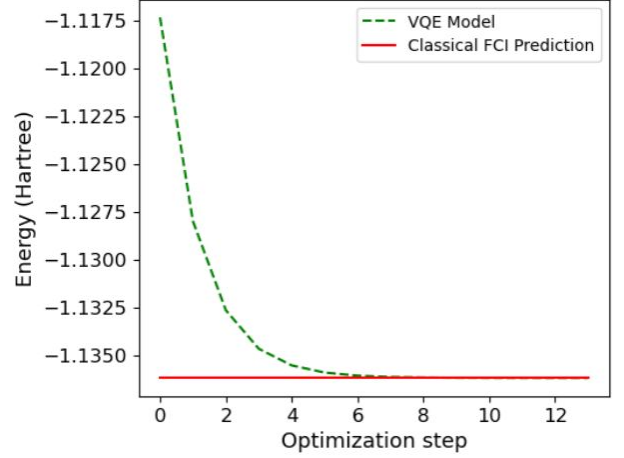


FIG. 2. Expectation value of the H_2 hamiltonian at each optimization step. This yields a ground state energy of -1.13689 ± 10^{-6} Ha, which is in good agreement with the FCI prediction of -1.13690 Ha.

C. Optimization

Now that we have our hamiltonian and our circuit, all that is left is to optimize the circuit parameters in order to calculate the ground state energy of the system. This is done by invoking the variational principle, which states

$$E_0 \leq \langle \Psi | H | \Psi \rangle \quad (5)$$

, for some trial wave function Ψ . This allows us to put an upper bound on the ground state energy by finding the expectation value of the hamiltonian with some given circuit parameters. We first choose an initial state, which is normally the Hartree Fock state where each electron occupies its lowest possible spin orbital. We then use this state as an input to the circuit we constructed, and then find the expectation value of the hamiltonian for this final state. Circuit parameters are then adjusted using some (classical) optimization algorithm (I used PennyLane’s GradientDescentOptimizer), and the process is repeated with the new, more optimal circuit. Once the difference in energies between iterations goes below a certain value (called the tolerance), we assert that the ground state energy has been found!

III. RESULTS AND DISCUSSION

In order to acquire this skill, I first built a simple model of the H_2 molecule, using just one double excitation gate as the circuit. This system only requires 4 qubits and 2 electrons to model— one for each of the two spin states of each electron. In this model, it is assumed that each electron shares the molecular orbital, meaning it does not occupy the atomic orbital of an individual hydrogen atom.

The only possible states the molecule can occupy, then, are the $|1100\rangle$ and $|0011\rangle$ states, by the Pauli exclusion principle. If it were in any other state, then the electrons would be occupying the same spin orbital, which is forbidden. As a result, the wave function will follow the form of $\Psi(\theta) = \cos(\theta/2)|1100\rangle - \sin(\theta/2)|0011\rangle$, which is the same form that we saw earlier from the double excitation gate. Therefore, the double excitation gate can be used as a representation of the H_2 molecule, provided that both electrons occupy the shared molecular orbital. Using this assumption, we can model the ground state with very high accuracy, as shown in figure 3. This model yields a ground state energy of -1.13689 ± 10^{-6} Ha, which it reached after 13 iterations through the optimizing algorithm (see figure 2). This is in very strong agreement with the FCI calculated value of -1.13690 Ha. It also predicts the bond length to be 0.742 ± 0.02 A, which is in perfect agreement with the FCI model.

This assumption breaks down, however, when the ground state energy of the molecule is greater than -1Ha, as the ground state energy of 2 separate hydrogen atoms would now be lower than the lowest possible energy of the electrons occupying the molecular orbital. This would mean that the bond would become unstable, shifting the ground state to be -1 Ha- the ground state of two separate hydrogen atoms. According to both the VQE and FCI models, this transition occurs at a bond length of 1.42 ± 0.02 A. Although the VQE still agrees with the FCI model above this bond length if we assert that the electrons must occupy the molecular orbital, we know that this is not the true minimum energy state for this system (see figure 3). Work remains to be done on formulating a single model that can account for the shift in orbitals that occur in this case, but for now, piecing together the two models will provide a sufficient description of the ground state of an H_2 molecule.

We can also use the VQE to simulate what occurs in larger molecules, such as LiH. This system has 12 active spin orbitals, and therefore requires 12 qubits to model. Since this system is much larger, I used PennyLane's `qchem.excitations()` feature, which gives a list of all the possible excitations within the system, greatly simplifying the circuit building process. Figure 4 shows the iterations of the VQE when analyzing the system, which yields a value for the ground state energy of -7.8821 ± 10^{-4} Ha. This is in reasonable agreement with the FCI value of -7.88268 Ha, yet further from the estimate we obtained for H_2 . This is likely due to the higher uncertainty, which is caused by the higher tolerance of 10^{-4} Ha. This higher tolerance was required in order to allow this simulation to run on my laptop. Any lower tolerance, or any more qubits, and the laptop would crash. There is no obvious reason to believe that, given a lower tolerance, the value would continue to approach the FCI value, especially as the more complex circuit would require more iterations to optimize. However, we are unable to know for sure given the information we have.

Real quantum hardware would be able to model the

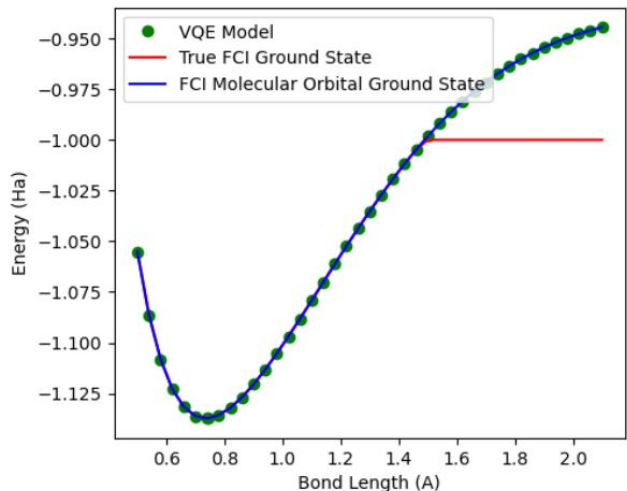


FIG. 3. Ground state energy of H_2 at varying bond lengths. The VQE model predicts a bond length of 0.742 ± 0.02 A, which is in perfect agreement with the FCI model. Note that the true minimum energy can not be greater than -1Ha, despite what the model suggests.

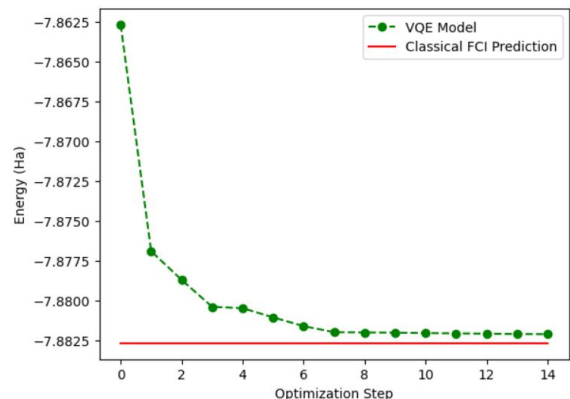


FIG. 4. Expectation value of the LiH hamiltonian at each optimization step. This yields a ground state energy of -7.8821 ± 10^{-4} Ha, which is in reasonable agreement with the FCI prediction of -7.88268 Ha. Note that the tolerance was forced to be lower, resulting in a higher uncertainty, due to the extremely long run time of this simulation.

LiH system with ease, but since I was simply modeling a quantum computer on classical hardware, I am limited to the same realm of classical computation. The inefficiency of the classical computer comes not only from the fact that billions of bits were needed to model the system on the laptop, as opposed to just 12 qubits, but also that the expectation value of the hamiltonian was calculated by brute force. On the classical computer, the expectation value of the hamiltonian was calculated by running the initial state through the circuit many times, and taking the average of the measured hamiltonian. Quantum hardware could accomplish this in just one pass through

the circuit, as it would be able to set each qubit in a superposition of possible states and calculate the expectation value of the hamiltonian based off of those superpositions.

IV. CONCLUSION

Through the development of this skill, I was able to learn the basics of quantum computing, and apply that knowledge to build a variational quantum eigensolver, capable of calculating the ground state energy of a given molecular configuration. I was mostly successful in doing so, with the calculated values for ground state energy

of H_2 and LiH agreeing with the full configuration interaction values calculated classically to within a fraction of a percent. When taken with the well known model of two separate hydrogen atoms, an accurate depiction of the ground state of the H_2 molecule was developed. Molecules larger than LiH were unable to be looked at, however, due to the constraints of simulating a quantum computer on classical computing hardware. It is expected that quantum computers will become much more mainstream within the next couple of decades, and if that occurs, the variational quantum eigensolver could become a widely used simulation tool for systems that otherwise could not be simulated.

-
- [1] Microsoft Learn *Quantum Computing History and Background*, (June 9, 2023)
 - [2] Jules Tilly et al *The Variational Quantum Eigensolver: A review of methods and best practices*, (Physics Reports, November 9, 2021)
 - [3] Michal Stechly *Variational Quantum Eigensolver Explained*, (Musty Thoughts, November 18, 2019)
 - [4] Caleb Clothier *Variational Quantum Eigensolvers*, (Medium, May 7, 2021)
 - [5] Tobias Ried *Jordan Wigner Transformations and Quantum Spin Systems on Graphs*, (Elitenetzwerk Bayern, August 31, 2013)