Quantum Computing Final Project

Tristan Larkin

May 3, 2025

1 Problem Statement

Using cloud quantum computing services, compute with two significant figures the energy of the ground state of a Fermi Hubbard Hamiltonian with $t = U = 1$. To be specific, the Hamiltonian is

$$\hat{H} = -t \sum_{(i,j)\in E} \sum_{\sigma\in\{\uparrow,\downarrow\}} \hat{a}^\dagger_{i,\sigma}\hat{a}_{j,\sigma} + U \sum_{v\in V} \hat{n}_{v,\uparrow}\hat{n}_{v,\downarrow}$$

where $\hat{a}^\dagger_{v,s}$ ($\hat{a}_{v,s}$) is the Fermionic creation (annihilation) operator which creates (destroys) a spin-s electron at site v and $\hat{n}_{v,s}$ is the Fermionic number operator which counts the number of spin-s electrons at site v. There are 6 sites which are connected as the connectivity of an octahedron. Phrased differently, V represents the set of 6 vertices and E represents the set of edges (all pairs of connected vertices) of the octahedron. In the above equation edge (v_i,v_j) and (v_j,v_i) both need to be counted in E. This Hamiltonian is a common approximation to model molecules in quantum chemistry.

While this specific case is classically solvable by diagonalizing the Hamiltonian, the dimensions matrix representation of this Hamiltonian increases exponentially as we add vertices. Hopefully, quantum computers will allow us to find the ground state of this Hamiltonian tractably. Unfortunately, current quantum computers are noisy and have a limited number of qubits. The literature refers to these as Noisy Intermediate-Scale Quantum (NISQ) computers. The goal of this project is to use a NISQ computer to find the ground state of the stated Hamiltonian.

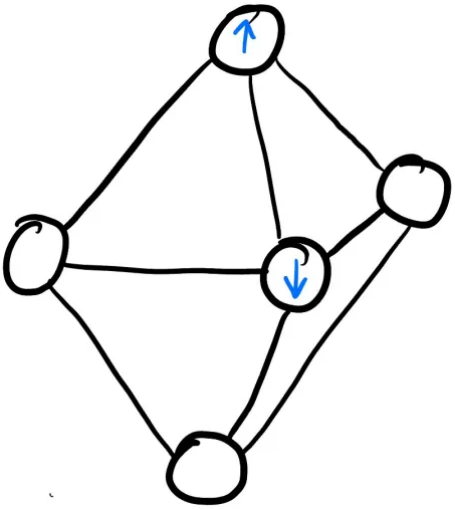

Figure 1: The Hamiltonian allows for electrons to be on any of the six vertices of an octahedron in either spin up or spin down.

2 Methodology

First, this problem is classically solvable. The solution can be found by diagonalizing the second quantized Hamiltonian which comes out to be a 4096×4096 matrix. This only takes around 2-3 minutes and gives all energy eigen states. While this fact is useful for getting an answer ($E_0 \approx -7.847$), it is not a use of a quantum computer. This problem's classical complexity grows quickly as we add nodes to the lattice. The goal here was to explore how we would do this with a quantum computer in the hopes of getting a more tractable way of solving for the ground state.

I wanted to use a Variational Quantum Eigensolver (VQE) algorithm to find the ground state of the Hamiltonian [1]. VQE takes advantage of the fact the there exists no state $|\psi\rangle$ such that the

expectation value of the Hamiltonian is smaller than the ground state. Explicitly this can be written as

$$\langle\psi|\,\hat{H}\,|\psi\rangle > \langle\text{Ground}|\,\hat{H}\,|\text{Ground}\rangle = E_0$$

By creating some parameterized circuit, which we call the ansatz, we can vary the parameters to continuously lower the expectation value of the Hamiltonian with help of a minimizer. Specifically, I used a 'scipy' minimizer and the "Cobyla" method. I want to minimize the energy of the the ansatz,so my cost function is just the expectation value of the ansatz. To get the expectation value I need to measure the state many times and infer the expectation value from the results. The number of times I repeat the measurement is called the shots, and increasing this number will decrease stochastic error that comes from the nature of quantum measurements.

There are two issues with naively approaching the problem with VQE. Firstly, since I only had access to NISQ computers, I have to be very aware of the noise that will be introduced into my state. I used the IBM Sherbrooke machine, which is implemented using $E\hat{C}R$, $\hat{1}$, $\hat{R}_Z$, $\sqrt{\hat{X}}$, and $\hat{X}$. From the information IBM gives on the Sherbrooke machine, we can see that the error in $E\hat{C}R$ dominates over the single qubit operations. This supports that we should try first and foremost to minimize the number of $E\hat{C}R$ applications. There is also error in the single qubit gates so reducing the total number of gates should also be beneficial. There is also a relaxation time (T1) and dephasing time (T2) to consider. From what I could tell, the time it takes to run the gates is small enough compared to T1 and T2 that it should not be an issue with a small enough circuit, but I still tried to make the circuit as fast as I could.

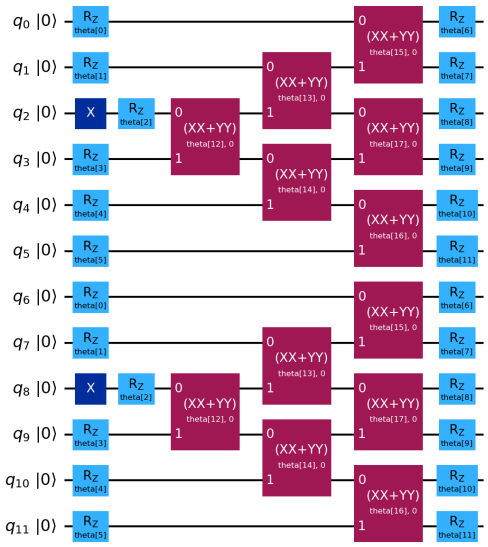

Figure 2: The ansatz that was sent to the quantum computer. the $(\hat{X}\hat{X} + \hat{Y}\hat{Y})$ gates are entangling gates that conserve the number of ones.

Our Hamiltonian was given in second quantized form, and I used the `FermionicOp` class in `qiskit-nature` to build the Hamiltonian operator. My ansatz was designed with a few things in mind. First, we were given that there are only two electrons in the ground state. The Hamiltonian is also completely agnostic as to what spin is what, so we should expect the ground state to have equal parts of opposite spin basis states.

Figure 2 shows my ansatz. The basic circuit is really only a six qubit state, and I repeat that state twice. This comes from the up-down symmetry, since the way I mapped my Hamiltonian onto qubits has the first six representing the up states and the second six representing the down states. There are 18 parameters for the minimizer, six $\hat{R}_Z$ gates in the front, six entangling $\exp(\hat{X}\hat{X} + \hat{Y}\hat{Y})\theta$ gates that fully connect my circuit, then six more $\hat{R}_Z$ gates. This form limits the possible resulting state to be a single up and single down in any combination, but enforces that exchanging the up and down electrons will not change the energy. This was a good choice for limiting the possible states the optimizer had to explore while still converging quickly and using a small number of entangling gates. When transpliled into the `ibm_sherbrook` base gates this circuit has the following gates:

$$\hat{R}_Z : 151 \;\;\&\;\; \sqrt{\hat{X}} : 91 \;\;\&\;\; E\hat{C}R : 24 \;\;\&\;\; \hat{X} : 7$$

It came to this through a mixture of testing and exploiting known attributes of the solution, but one thing I cannot explain is the improvement I get adding the sixth entangling gate. I expected the minimum number of entangling gates to be optimal, but adding the sixth gate to the middle two qubits significantly improved my performance on the quantum computer. I expect it produces

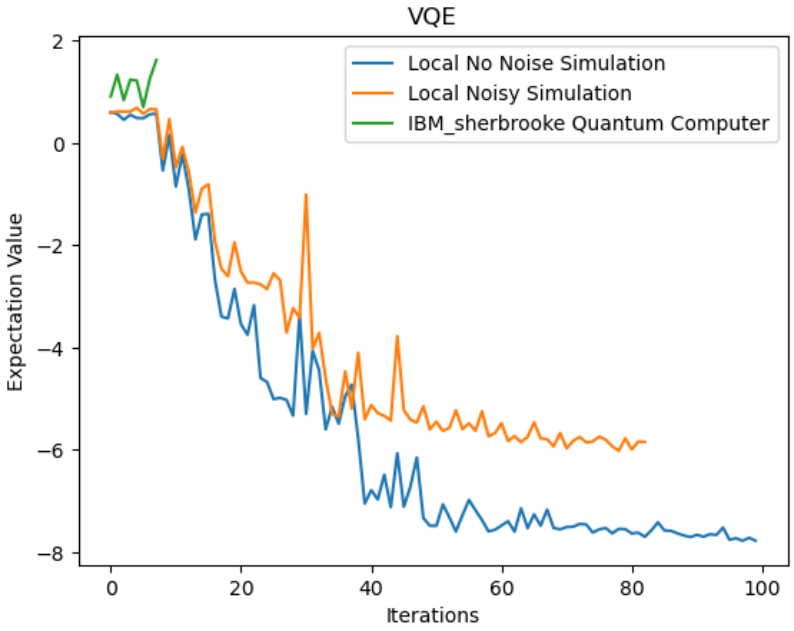

Figure 3: Iterative VQE results using three different backends. The local simulations started with the same initial conditions while the quantum computer started with a different random start.

a more stable state that is less susceptible to errors, but that is just speculation. I wanted to keep the total number of parameters small enough to help the optimizer have to explore less options.

My goal was to perform VQE on my classical machine to find how quickly I could get my ansatz to converge close to the answer, then use a noisy simulation to do the same, and finally perform VQE on an actual quantum computer. I also wanted to test the ansatz to see how well the quantum computer could create the state and get the expectation value. I was not able to get any decent results using the actual quantum computer due to the time constraint on the IBM machines.

3 Results

Using the ansatz I developed in the Methodology section, I ran VQE with the `Estimator` class using a classical simulation backend with no artificial noise (see blue plot in Figure 3). This method found parameters that got very close to the ground state energy which we could check with by diagonalizing the Hamiltonian. The next step was to ensure that the ansatz circuit could reasonably prepare the intended state with noise and have the expectation value measured.

I ran this many times with randomized initial parameters and generally got my classical simulation to converge to around -7.6 to -7.8. Depending on the day and small variations, these final optimized states would produce expectation values on the quantum computer between -5.5 and -7.5 with the lowest I ever recorded at -7.54597. This showed that my ansatz does produce results that are close to the expected on a quantum computer.

With that in mind I ran VQE with a noise simulator (see Figure 3, orange) and got similar results to the no noise simulation for a while before it diverges and gets stuck around -6. It was hard to tell how long each iteration would take on the real quantum computer so I sent it in with slightly reduced shots and let it eat up the rest of my time. This turned out to be futile as it did not even reach an iteration count that I would expect to see any results (I also accidentally started it as a different initial condition).

4 Analysis

I did not do any manual error mitigation but I did look into how my circuit was being affected. Using the `Sampler` class I was able to run my ansatz with pre-found parameters on the quantum computer 15000 times and look at the states it measured. My state only allows for the 36 states that have a single up electron and single down electron. In my sampler I got that 10489 states measured were in one of the 36 states we expect, and 4511 were in a state that we do not expect. These are fairly strong errors so it explains part of why we expect to see significant deviation from the noiseless case.

According to IBM [2], the Sherbrooke machine has 7.266×10^{-3} median error in $E\hat{C}R$ gates, 2.203×10^{-4} median error in $\sqrt{\hat{X}}$ gates, and 1.587×10^{-2} median readout error. Each qubit has around 2 $E\hat{C}R$ applications on it, which would mean around 1.4×10^{-2} error on them plus the 16×10^{-2} readout error implies that the overall error on each qubit is around 3×10^{-2}. There are on average nine $\sqrt{\hat{X}}$ gates per qubit bringing total $\sqrt{\hat{X}}$ error contribution to 0.02. If we measure 12 qubits each time then we expect around 0.38 error per total measurement, which is close to what we see in the sampling.

5 Conclusion

My circuit is almost minimally entangled and is designed to run quickly on the quantum computer I was able to use. The state preparation and measurement seemed fairly close and my later analysis of sampling error matched about what the error rates provided predicted. I do not know how well my VQE would have done had it had more time to run, but I do not expect to have gotten near the answer since the noisy simulation got stuck (Figure 3). Overall I was not able to use a quantum computer to solve for the ground state energy, but I think that the process I used to reduce my ansatz is the type of thinking that would be needed to approach this with more computing time. The fact that I was able to solve the answer exactly allowed me to test the results I got, which would not be available in a problem that really needed to be solved on a quantum computer. However, I think my analysis of the symmetries in the Hamiltonian would still have led me to similar conslusion.

References

[1] Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, and Jonathan Tennyson. The variational quantum eigensolver: A review of methods and best practices. *Physics Reports*, 986:1–128, November 2022.

[2] Quantum IBM. ibm_sherbrooke details. `https://quantum.ibm.com/services/resources?system=ibm_sherbrooke`, 2025.