

Simulating Molecules Using a Quantum Recurrent Neural Network and the Variational Quantum Eigensolver

Author: Paul Kottering SID:3033038699

Other Project Members: Pratik Brahma, Kian Jansepar, Joseph Palakapilly, Michael Qi

CS191 Final Project Paper, UC Berkeley

February 4, 2021

1 Introduction

1.1 Project Overview

Recent developments in quantum information science have led to promising new algorithms for applications in quantum chemistry. Given the ease of access to quantum computing cloud simulators, we can build off existing research in the field and run our own algorithms to investigate areas of improvement [1]. As new applications of quantum computers to computational chemistry are being discovered, there has simultaneously been significant progress in quantum machine learning (QML) methods [2]. We wanted to investigate ways that QML could be applied to existing quantum chemistry algorithms. Our project therefore involved two tasks, the first was to research and investigate both quantum chemistry and QML and the second was to implement a new method combining the two advances and evaluate its performance.

1.2 Simulating Molecules

Richard Feynman once said that ‘if you want to make a simulation of nature, you’d better make in quantum mechanical’ and taking this advice we looked to investigate ways that quantum computers can simulate molecules. The two most popular methods for solving problems that simulate the ground state of the Hamiltonian of a molecule are Quantum Phase Estimation (QPE) and the Variational Quantum Eigensolver (VQE). The QPE method finds the energy eigenvalues of a Hamiltonian by encoding them in a measured phase. Despite the higher accuracy that QPE obtains, this method requires fault tolerance. This is because the required QPE coherent circuit depth scales inversely with the precision goal. Recent advances [3] have attempted to combine the two methods, but we will be using the VQE method as it requires only short coherence time and hence is more suitable to the noisy qubits we have today.

1.3 Performance Indicators

The number of qubits and number circuit evaluations required are the two factors that we would like to reduce so that our algorithms are suitable to today’s Noisy Intermediate-Scale Quantum (NISQ) era [4]. For each quantum circuit evaluation required, the probability of introduced error through channels such as bit-flips and depolarizing increases, we therefore look for methods that limit the number of evaluations. Furthermore, even though hardware and control constraints limit the largest operational quantum computers today to 50 qubits, in order for a classical computer to simulate our algorithms we must limit our qubit requirements further. We will therefore be using the qubit requirement and number of circuit evaluations as indicators to measure our method performance.

1.4 Applications

Calculation of energy scales polynomially on quantum computers compared with exponentially on classical computers [5]. These energies and their associated ground state can be used to better understand chemical reaction rates, transition metal catalysis and high temperature superconductivity [6]. More long-term applications also include simulating biological processes and medicine discovery [7]. We will primarily be investigating the ground state of molecules due to the relative computational simplicity, but recent studies have shown similar methods can be applied to excited states [8]. These applications are not just theoretical expectation, but the first electronic structure calculations on a quantum computer have already been performed [9] and more recent investigations have proved successful in simulating more complex molecules [10]. This initial progress indicates that it is a valuable and viable route of investigation.

1.5 Principle Questions

After choosing our project direction we therefore wanted to answer two principle questions. Could a QML assisted VQE method improve the performance over the classic VQE method with respect to qubit requirement and circuit evaluations? Could a QML assisted VQE method allow for easier simulation of more complex molecules?

2 Technical Background

2.1 VQE Method

The VQE method is based on the Rayleigh-Ritz variational principle which states that the normalized expectation value of the Hamiltonian for an arbitrary wave function must be greater or equal to the ground state energy of the Hamiltonian.

$$\frac{\langle \phi(\theta) | H | \phi(\theta) \rangle}{\langle \phi(\theta) | \phi(\theta) \rangle} \geq E_0 \quad (1)$$

The analytical method finds the associated expectation value of a parametrized wavefunction and subsequently performs simple calculus to minimize that expectation with respect to the parameters of the wavefunction. The VQE method is the algorithmic equivalent to this analytical method and its process is described below. The VQE is a hybrid quantum-classical algorithm that uses a quantum computer for state preparation and measurement of the expectation value, and then uses the classical computer to process results and update the parameters of the quantum circuit. The preparation and measurement steps are repeated to obtain the expectation value of the Hamiltonian to a required precision. The quantum computer is re-initialized at each step which means only short coherence times are needed.

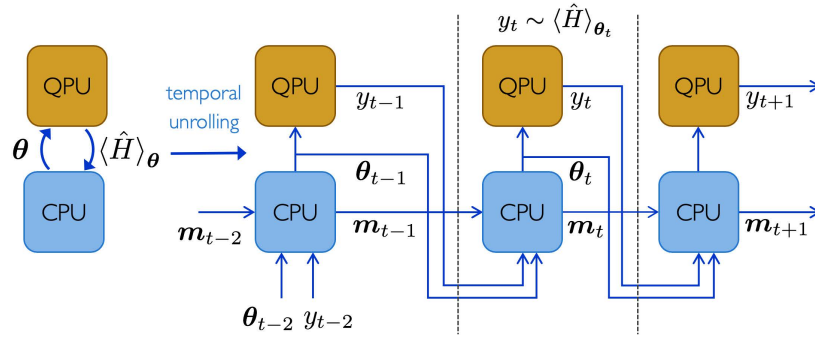


Figure 1: VQE Method Diagram. Quantum circuit evaluations are performed on the quantum processing unit (QPU) whereas the classical optimization strategy is run on the classical central processing unit (CPU). Figure taken from [13]

2.2 Ansatz Choice

The wavefunction that is initially applied is called the ansatz state and it can be prepared by applying a set of parametrized gates to our set of qubits. The choice of ansatz is critical to the performance of the algorithm and will be a focus of much of our discussion. There are generally two types of ansatz which are chosen, the first are chemically motivated ansätze which uses exist-

ing knowledge from classical computational chemistry to build the structure of the ansatz circuit. The other is hardware efficient ansätze which are suitable to implement on NISQ computers and aim to be flexible and minimize required gates and coherence time. Although this is not an area we chose to investigate fully, the discrepancy between the two choices is an interesting area of future research. It has recently been shown that using hardware efficient ansätze with random initial parameters results in the energy gradient being near zero in most directions of Hilbert space [11]. Furthermore, as the number of qubits and circuit depth increases the problems get exponentially worse. Hence hardware efficient ansätze are not suitable for molecule simulation and we chose to use a chemistry motivated ansatz. The above research highlights one of two problems associated with the VQE method that can arise with poor ansatz choice. This first is that instead of finding global minima, we converge to a local minimum. The second is that the number of measurements required for a suitable precision could be too large, so it is unfeasibly computationally expensive. The number of circuit evaluations scales $O(1/\epsilon^2)$ for the VQE method [12].

2.3 Hamiltonian Design

Using the Born-Oppenheimer approximation which treats the nuclei as point charges, the electronic structure Hamiltonian is reduced to.

$$H = - \sum_i \frac{\nabla_i^2}{2} - \sum_{i,I} \frac{Z_I}{|r_i - R_I|} + \frac{1}{2} \sum_{i \neq j} \frac{1}{|r_i - r_j|} \quad (2)$$

The first term is the kinetic energy of the electrons, the second is the Coulomb repulsion between the electrons and nuclei, and the third term is the repulsion between the electrons themselves.

2.4 QRNN

To overcome the ansatz choice issues outlined above, we proposed using QML to learn where to initialize parameters in a process known as meta-learning [13]. This process extracts common structure among instances of a class of problem by training an outer-loop optimizer on many instances of this class which, by exploiting this common structure, improves our ability to solve unseen instances in the class or a related class. We use a recurrent neural network (RNN) optimizer to learn heuristics of our class of problem. During the training stage of the RNN assisted VQE method, the RNN recursively proposes updates to the VQE parameters. The RNN receives as input the previous VQE query’s estimated cost function as well as the information stored internally from previous steps. This then generates a new suggestion for the VQE parameters and hopefully does so in very few steps. The loss function of the RNN when we are applying it to the VQE depends explicitly on Hamiltonian cost function, but the specific architecture is variable. We only use the RNN architecture to learn the problem-class-specific initialization heuristic and leave the fine optimization to a classical optimizer. After the RNN has learnt the VQE method problem-class during the training phase, the RNN is able to find appropriate initial parameters quickly during the test phase.

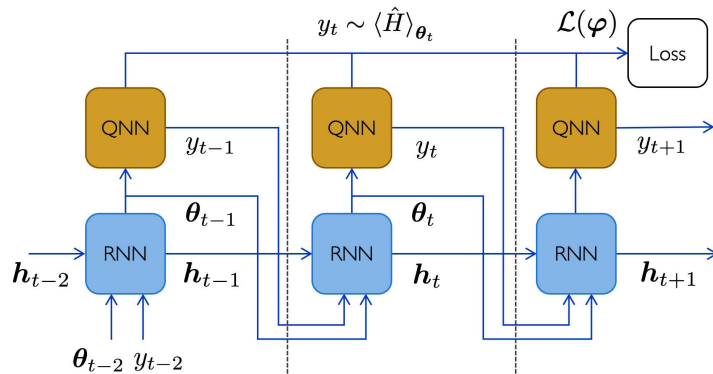


Figure 2: RNN assisted VQE Method Diagram. In order to simulate molecules we use the VQE for the QNN. The memory of the optimizer is encoded in the hidden state of the RNN h . Figure taken from [13]

3 Implementation

To implement the method detailed above, we needed to use a circuit development library that would allow us to build quantum circuits and run simulations. We decided to use Google’s Open Source library called Cirq. For our project we decided to focus on two molecules that were adequately simple such that simulation times were not too long, hydrogen and ethylene. I will now give only a brief summary of the chemistry required to obtain the ansatz and the Hamiltonian quantum circuits, since this was not the focus of our project, but still critical in determining performance.

3.1 Ansatz Implementation

To construct our Hamiltonian and ansatz circuit, we must choose a basis set which can be used to approximate the spin-orbitals of the molecules. The STO-3G basis has 3 primitive Gaussian orbitals fitted to each single Slater-type orbital. Hydrogen has 2 spins and 2 orbitals, hence 4 spin-orbitals and so we must use 4 qubits. Before we can apply our ansatz choice, we must prepare the circuit in a reference state. We know that in the minimal basis hydrogen is discretized into two spatial orbits. After using the mean-field procedure, the best approximation of the ground state is found to be the symmetric superposition of these two orbitals [14]. This can be written in the computation basis as the first two qubits as 1 and the second two as 0.

$$|\psi\rangle_{ref} = X_1 X_0 |0000\rangle = |1100\rangle \quad (3)$$

As we mentioned in the introduction, we will be using a chemically motivated ansatz and one of the most popular choices for this is the so-called unitary coupled cluster ansatz with single double excitation (UCCSD) [15]. The parametrized trial state created by the UCC method results from considering the excitations above the initial reference state. Recent research has shown the

applicability of adapted generalized UCC ansatz to near term quantum computers [16] but for our investigation we use the simplest form.

$$U = e^{(T_1 - T_1^\dagger) + (T_2 - T_2^\dagger)} \quad (4)$$

$$T_1 = \sum_{i \in \text{virt}, \alpha \in \text{occ}} t_{i\alpha} a_i^\dagger a_\alpha \quad (5)$$

$$T_2 = \sum_{i,j \in \text{virt}, \alpha, \beta \in \text{occ}} t_{ij\alpha} a_i^\dagger a_j^\dagger a_\alpha a_\beta \quad (6)$$

In the above equations, occ are the initially occupied orbitals in the Hartree-Fock state, and virt are the initially unoccupied orbitals. The t 's are the parameters that will be optimised in the VQE. These equations can be simplified by splitting the operator with a single Trotter step, using the Jordan-Wigner encoding and separating the resulting Pauli strings [17]. After this simplification, the unitary operator which we will apply to our reference state is given as:

$$U = e^{-i\theta X_3 X_2 X_1 Y_0} \quad (7)$$

We see that our ansatz circuit only has one variable parameter 'theta' which, due to the simplicity of the problem is appropriate, but when more complicated optimization problems are being solved we provide our RNN with a vector of parameters which are simultaneously optimized. Figure 3 shows us the circuit prepared using Cirq which will prepare our ansatz state.

3.2 Hamiltonian Implementation

To evaluate the cost function, which is the expectation value of the Hamiltonian, we must build a quantum circuit for our Hamiltonian. Using the Born-Oppenheimer approximation detailed above, and then using the Jordan-Wigner encoding we can obtain a Hamiltonian for hydrogen.

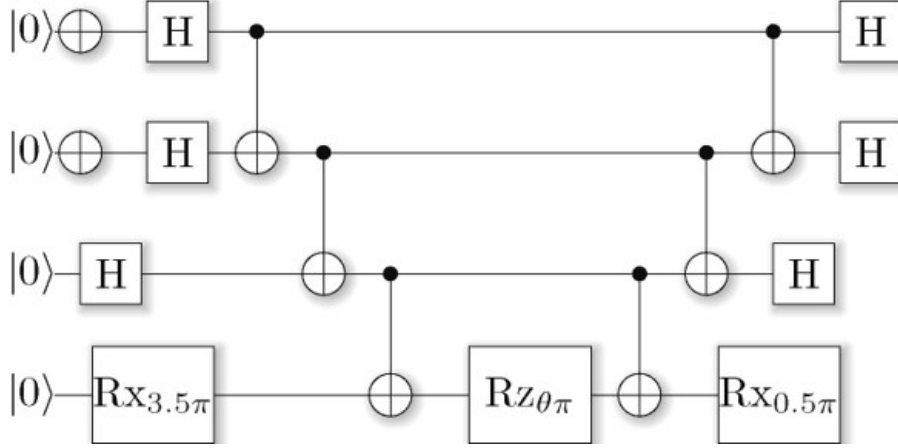


Figure 3: Reference state preparation and ansatz circuit.

$$\begin{aligned}
 H = & h_0 I + h_1 Z_0 + h_2 Z_1 + h_3 Z_2 + h_4 Z_3 + h_5 Z_0 Z_1 + h_6 Z_0 Z_2 + h_7 Z_1 Z_2 + h_8 Z_0 Z_3 + \\
 & h_9 Z_1 Z_3 + h_{10} Z_2 Z_3 + h_{11} Y_0 Y_1 X_2 X_3 + h_{12} X_0 Y_1 Y_2 X_3 + h_{13} Y_0 X_1 X_2 Y_3 + h_{14} X_0 X_1 Y_2 Y_3 \quad (8)
 \end{aligned}$$

We can obtain the Hamiltonian circuit using a computational chemistry package that is fully integrated with Cirq called OpenFermion [18]. For ethylene, the process of forming the Hamiltonian is the same as for the hydrogen molecule, but we want to be able reduce the number of qubits required so we only consider the pi-orbitals and the electrons in those pi-orbitals. Since we have 2 electrons each with 2 orbitals for each spin, we again obtain a 4 qubit Hamiltonian in the STO-3G basis which is the same as the hydrogen Hamiltonian equation given in Equation 8. The exact structure of the ethylene Hamiltonian quantum circuit was not my focus in the project, but it was built using knowledge of the geometry of the ethylene molecule and the OpenFermion Cirq package. If we had considered the total number of spin-orbitals we would have required 28 qubits which would have taken too long to simulate on available resources.

3.3 RNN Implementation

The RNN training and testing was implemented using another package that can be integrated with Cirq called TensorFlow Quantum (TFQ) released in March 2020 [19]. TFQ allows to connect classical and quantum layers of a RNN through seamless backpropagation and directly harnesses existing TensorFlow and Keras tools. TFQ constructs a dataflow graph for our choice of neural network, where the tensors are the directed edges and the operations are the nodes. Having converted the ansatz circuit and the Hamiltonian into tensors using TFQ, we then build our training model using a 5 layered RNN which calculates the expectation value at each layer. The meta-learning RNN loss function uses a weighted history of the expectation values with a preferential weighting given to the expectation value of the last layers. For all of the results presented in this paper, we used 1000 epochs for the training of the RNN.

$$L(\phi) = E[w_t y_t]_{t=1}^T \quad (9)$$

The optimizer used is called an Adam optimizer, which is a stochastic gradient descent method based on adaptive estimation of first-order and second-order moments [20].

3.4 Research Examples

The architecture of our QML-VQE implementation has been adapted from a paper produced by the creators of TensorFlow Quantum that performs RNN meta learning on a Quantum Approximate Optimization Algorithm [13]. We also used an example from OpenFermion-Cirq to assist us with developing the ansatz and Hamiltonian which we adapted for our specific project [18].

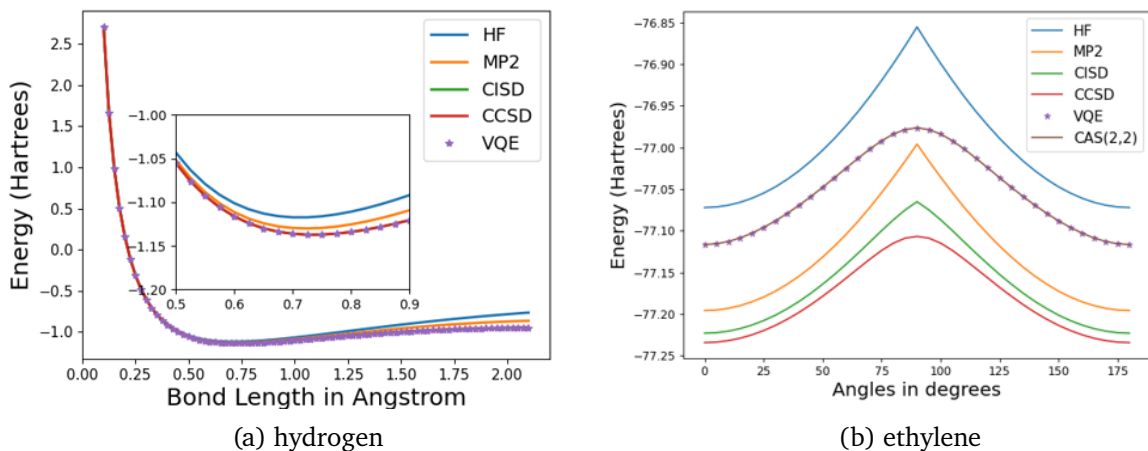


Figure 4: Hydrogen and ethylene VQE Results

4 Results

As described in the previous sections we chose to use hydrogen and ethylene molecules for ground state energy calculation. For hydrogen we chose to vary the bond length to create a training set, whereas for ethylene the torsion angle was the variable used. We initially wanted to test the effectiveness of the VQE without the QML to show our ansatz and Hamiltonian circuits were effective. Figure 4 shows the results of the VQE method alone performed on hydrogen and ethylene compared to various other classical computational methods. We can see that the VQE method is highly successful as we were able to recreate the results found in literature.

We then wanted to test the ability of the RNN to learn the heursitics of the VQE problem for hydrogen and ethylene. We trained the molecule on an arbitrary set of feasible training bounds and then tested the model on a larger range of bond lengths/torsion angles to show the model could extrapolate. Figure 5 shows given the initial parameter provided by the RNN after testing, the associated energies corresponded to the true ground state energy curve. This was a promising result, especially for the extrapolated region, as it showed the RNN had learned the character of the problem class. The random parameter energy values were calculated by taking a random initial parameter from a uniform distribution from 0 to 2π and then calculating the associated

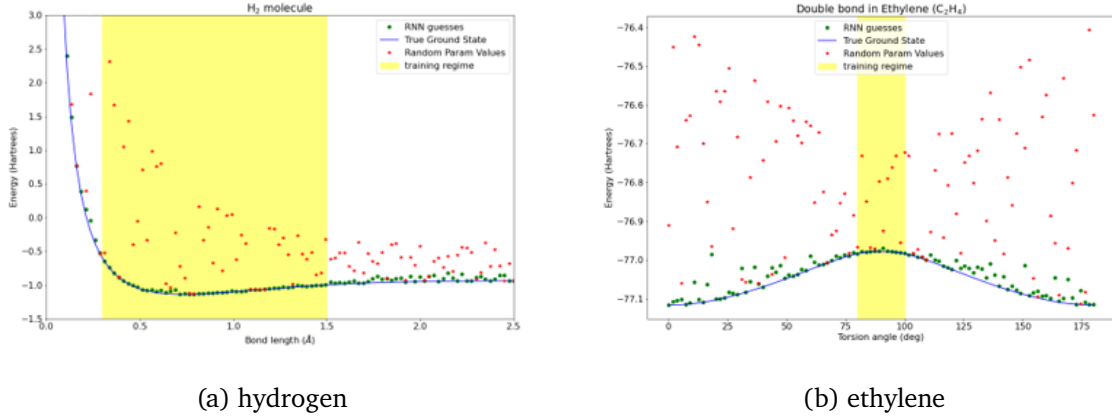


Figure 5: Energies from RNN provided initial parameters.

expectation value of the Hamiltonian for each torsion angle. We do not perform the classic VQE method on those random initial parameters, or on the RNN initial parameter guesses at this point.

Whilst the above results provided confidence that we were not only able to reproduce results from literature, but also that we could use this RNN to find appropriate initial values, in order to answer our principle questions we would have to show that the RNN could assist in solving more complex molecules. To do this we trained the RNN model on the hydrogen molecule over a range of bond lengths and then used the same model to test on the ethylene model. Figure 6 shows the flow diagram for this new train-test regime. Notice that the ansatz circuit is the same for all 3 stages of the RNN assisted VQE method. This is possible as we are able to encode both the hydrogen and ethylene Hamiltonian onto 4 qubits.

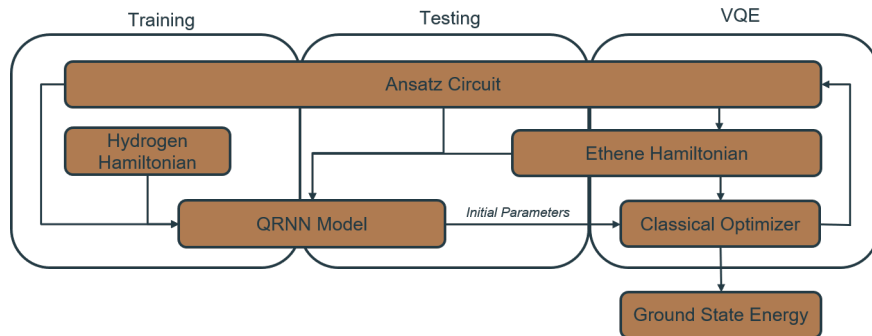


Figure 6: Flow Chart for training on hydrogen and testing on ethylene

The results from the training the RNN on the hydrogen Hamiltonian, and testing on the ethylene Hamiltonian in Figure 7 show that the RNN was able to guess initial parameters that resulted in a ground state energy close to the true ground state for all torsion angles tested. Note, that we are again not applying the VQE method at this point to the initial guess but instead comparing the energies from the RNN initial parameters and random initial parameters.

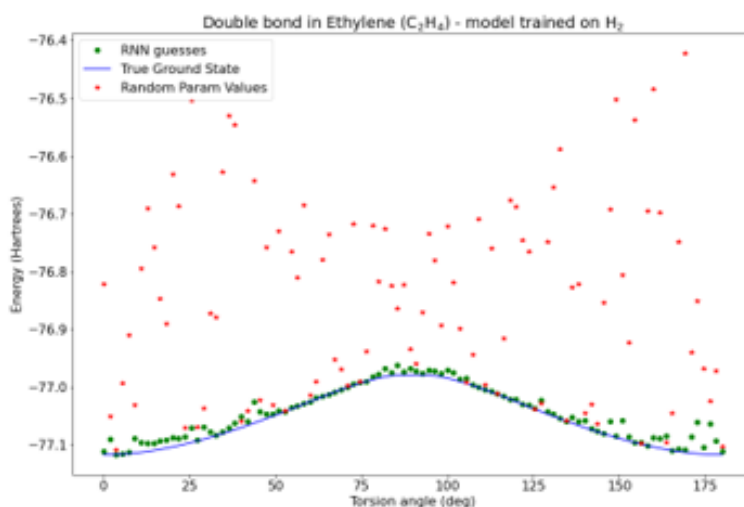


Figure 7: Energy predictions for training on hydrogen and testing on ethylene

Whilst all the team worked collaboratively on every aspect of the project, we focused on different sections. I was heavily involved in the selection of the VQE as the method of simulation, as well as the implementation and choice of ansatz. I was also responsible for building off the work detailed above to do a full analysis of the circuit evaluations required to achieve a desired accuracy. For this I integrated the complete QML-VQE method shown in Figure 6 together and applied the VQE method with and without initial parameters provided by the RNN. After comparing multiple classical optimizers for the VQE method (shown in Figure 8), we chose to use a widely used SLSQP method.

SLSQP is a sequential last squares programming algorithm that uses the Han-Powell quasi-Newton method [21]. Figure 9 shows us the number of circuit evaluations needed for a range of

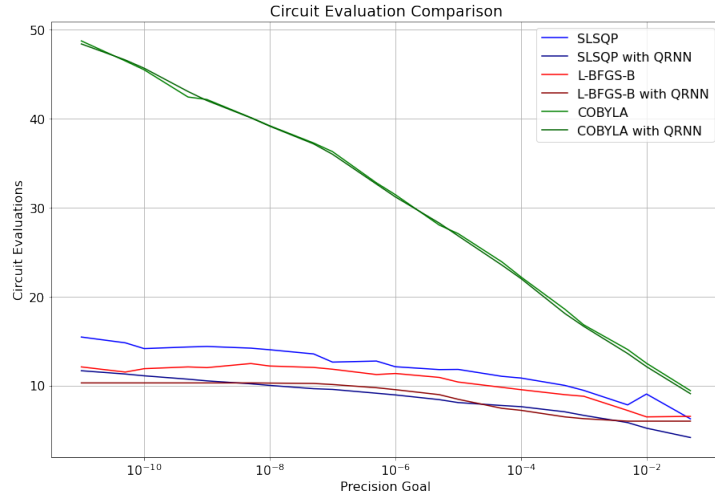


Figure 8: VQE Method Circuit Evaluation Comparison

precision goals in the stopping criterion. We have averaged over the number of circuit evaluations that are performed for each torsion angle of the ethylene molecule. We can see that the number of circuit evaluations required when initializing with the RNN is less than when we use a random initial parameter. However, the total circuit evaluations, including the 5 required for the RNN test, is above the classic VQE for all precision requirements. We do not include the number of circuit evaluations required for training of the RNN.

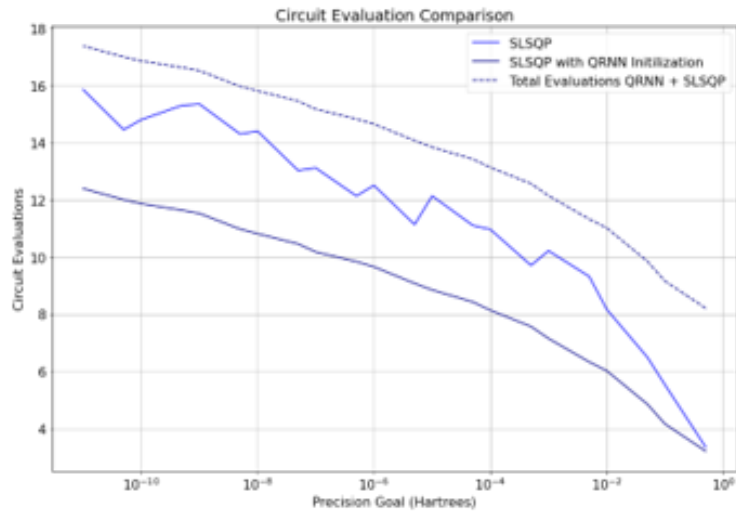


Figure 9: SLSQP Circuit Evaluation Comparison

4.1 Ansatz Circuit with Error

In order to test the robustness of the RNN assisted VQE method against qubit error I encoded random error channels into the ansatz circuit. The two types of error I chose to investigate were bit-flip and phase-flip. A bit-flip represents a scrambling of the amplitude of the qubit in the computational basis whereas the phase-flip represents a scrambling of the phase of the qubit in the computational basis. The channel, the location in the ansatz circuit and the qubit upon which the bit-flip or phase-flip was applied was kept random. The probability of the bit-flip error occurring at some point in the circuit was hard coded as 8 percent and the same for a phase-flip error. Figure 10 shows the energies associated with the initial parameters that the RNN gave with error in the ansatz circuit. The error was present for both the testing and training phase of the RNN. Whilst the 8 percent is a very high error occurrence rate, it proves without question the ability of the RNN assisted method to learn on noisy qubits. We notice from Figure 10 that although the RNN guesses do not follow the true energy curve as closely as the noiseless results, the shape matches the true curve accurately.

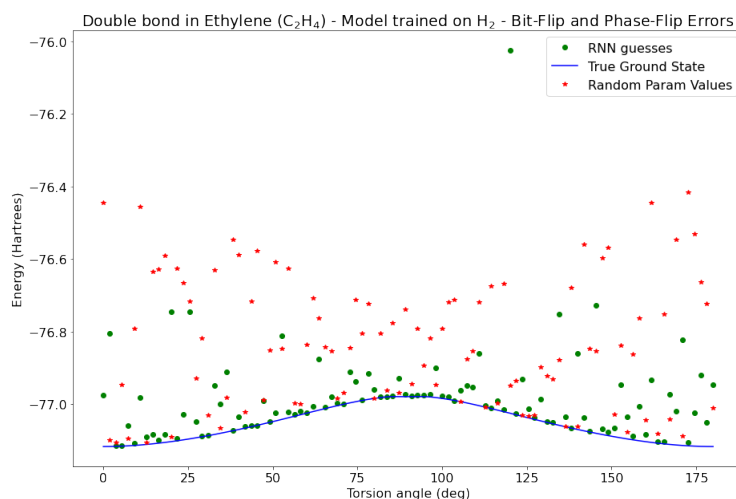


Figure 10: Noisy ansatz circuit RNN parameter energies

5 Discussion

We can see from the results presented that we are able to use the RNN assisted VQE method to scale to more complex molecules. When solving for a complex molecule, if we can reduce the number of circuit evaluations required of the complex molecule Hamiltonian circuit by initializing the ansatz parameters with a RNN training on a Hamiltonian circuit that requires shorter coherence time, then we can improve the robustness of the VQE method against error. Considering the ability of the RNN to guess effective initial parameters, we are confident that applying the RNN method to a more complex ansatz, perhaps with more than one parameter, will mean that the total circuit evaluations will be lower than for the classic VQE method. Due to the remaining relative complexity of performing quantum computation, this RNN method extends the capability of the VQE method.

We noticed from our simulations that the VQE method, both assisted with RNN and without, converged upon the true ground state for all bond lengths/torsion angles very quickly. This is because we tried to limit the ansatz and molecule complexity to show the proof-of-concept, but it is clear that in the future we should evaluate an ansatz with more parameters and on a more complex molecule

5.1 Future Work

The simulation times and lack of hardware accessibility make larger quantum computation difficult, but performance on complex molecules is crucial for determining whether the RNN assisted VQE is effective in the scaling of the VQE method. In previous studies a divide and conquer approach was used to study more complex molecules [22]. This approach uses problem decomposition to break a target molecular system into subsystems which are then individually solved on a quantum computer and then are put together using a classical method. The RNN approach

could be used for both solving the subsystems and potentially even for piecing the individual piece together.

For the implementation of our RNN method we chose a set of hyperparameters such as number of layers, weighting of the loss function between the layers, number of training points, range of training bounds, Adam optimizer weighting and learning rate, number of training epochs etc. Hyperparameter optimization is an active part of machine learning research and future work could perform a study on which RNN architecture reduces VQE circuit evaluations the most effectively.

Recent studies have shown that the VQE method can be used to solve excited states of molecules [8]. This method also uses machine learning by adopting a Quantum Generative Adversarial Network (QGAN) in which a second parametrized circuit is used to find the orthogonal state to the original ground state which has the lowest energy. By basic quantum mechanics, this state must be the first excited state. We could apply our RNN method in combination with the QGAN to assist in searching for the lowest energy orthogonal state.

In today's NISQ era it is important to build algorithms which are robust to noise. Future work could focus on assessing how the RNN assisted VQE method performs when other forms of noise is introduced into the circuit simulation. Furthermore, the architecture of the VQE method optimizer provided in OpenFermion Cirq means that introducing error into the VQE method was not feasible. Therefore, looking at the performance of the error prone ansatz circuit in the VQE method as well as introducing error into the Hamiltonian will further improve the reliability of this method. By reducing the number of circuit evaluations required, the RNN assisted method could perform better than the classic VQE method as the opportunity for error is reduced.

6 Conclusion

For this research project we researched the use of quantum computers in modern chemical applications and investigated some of the methods employed. We then adapted some examples of simulations using a Quantum Recurrent Neural Network and applied them to the Variational Quantum Eigensolver. Our results showed that the neural network was used to rapidly find a global approximate optimum of the ansatz parameters. This then acted as an effective starting point for the classical optimizer, used in the classic VQE method, to search for the local minimum. We also showed that the RNN assisted VQE method had a certain ability to solve more complex molecules after being trained on a simpler molecule. These two conclusions have answered our two principle questions outlined in the introduction. In doing our research and investigation we were also able to identify numerous areas to take this investigation further.

7 Word Count

I decided to include the figures/equations in the main text to aid ease of reading. The word count for the text without figure is 3700 which is greater than 12 pages double-spaced.

References

- [1] Ryan LaRose. Overview and comparison of gate level quantum software platforms. *ArXiv*, abs/1807.02500, 2019.
- [2] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, Sep 2017.
- [3] Daochen Wang, Oscar Higgott, and Stephen Brierley. Accelerated variational quantum eigensolver. *Physical review letters*, 122 14:140504, 2019.
- [4] John Preskill. Quantum computing in the nisc era and beyond. *Bulletin of the American Physical Society*, 2:79, 2018.
- [5] Al’an Aspuru-Guzik, Anthony D. Dutoi, Peter J. Love, and Martin Head-Gordon. Simulated quantum computation of molecular energies. *Science*, 309 5741:1704–7, 2005.
- [6] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon C. Benjamin, and Xiao Yuan. Quantum computational chemistry. *arXiv: Quantum Physics*, 2018.
- [7] Yudong Cao, Jhonathan Romero, and Alán Aspuru-Guzik. Potential of quantum computing for drug discovery. *IBM J. Res. Dev.*, 62:6:1–6:20, 2018.
- [8] Jules Tilly, Glenn Jones, Hongxiang Chen, Leonard Wossnig, and Edward R. Grant. Computation of molecular excited states on ibmq using a discriminative variational quantum eigensolver. *arXiv: Quantum Physics*, 2020.
- [9] P. J. J. O’Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, E. Jeffrey, E. Lucero, A. Megrant, J. Y. Mutus, M. Neeley, C. Neill, C. Quintana, D. Sank, A. Vainsencher, J. Wenner, T. C. White, P. V. Coveney, P. J. Love, H. Neven,

-
- A. Aspuru-Guzik, and J. M. Martinis. Scalable quantum simulation of molecular energies. *Phys. Rev. X*, 6:031007, Jul 2016.
- [10] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Sergio Boixo, Michael Broughton, Bob B. Buckley, David A. Buell, Brian Burkett, Nicholas Bushnell, Yonghong Chen, Zhiqiang Chen, Benjamin Chiaro, Roberto Collins, William Prideaux Courtney, Sean Demura, Andrew Dunsworth, Edward Farhi, Austin G. Fowler, Brooks Foxen, Craig Gidney, Marissa Giustina, Rob Graff, Steve Habegger, Matthew P. Harrigan, Alan Loh Ho, Sabrina N Hong, Trent Huang, William J. Huggins, Lev B. Ioffe, S. V. Isakov, Evan Jeffrey, Zhang Jiang, Chris Jones, Dvir Kafri, Kostyantyn Kechedzhi, Julián Kelly, Seon Wook Kim, Paul Klimov, A. Korotkov, Fedor Kostritsa, David Landhuis, Pavel N. Laptev, Mike Lindmark, Erik Lucero, Orion Martin, John M. Martinis, Jarrod R. McClean, Matt McEwen, Anthony Megrant, Xiao Mi, Masoud Mohseni, Wojciech Mruczkiewicz, Josh Y Mutus, Ofer Naaman, Matthew Gary Neeley, Charles Neill, Hartmut Neven, Murphy Yuezhen Niu, Thomas E O’Brien, Eric P. Ostby, Andre Petukhov, Harald Putterman, Chris Quintana, Pedram Roushan, Nicholas C. Rubin, D Sank, K J Satzinger, Vadim N. Smelyanskiy, Doug Strain, Kevin J. Sung, Marco Szalay, Tyler Y Takeshita, Amit Vainsencher, Theodore C. White, Nathan Wiebe, Zhimin J. Yao, Ping Hung Yeh, and Adam Zalcman. Hartree-fock on a superconducting qubit quantum computer. *arXiv: Quantum Physics*, 2020.
- [11] Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 9, 2018.
- [12] Dave Wecker, Matthew B. Hastings, and Matthias Troyer. Progress towards practical quantum variational algorithms. *Phys. Rev. A*, 92:042303, Oct 2015.

-
- [13] Guillaume Verdon, Michael Broughton, Jarrod R. McClean, Kevin J. Sung, Ryan Babbush, Zhang Jiang, Hartmut Neven, and Masoud Mohseni. Learning to learn with quantum neural networks via classical neural networks. *ArXiv*, abs/1907.05415, 2019.
- [14] Leo P. Kadanoff. More is the same; phase transitions and mean field theories. *Journal of Statistical Physics*, 137:777–797, 2009.
- [15] Rodney J. Bartlett, Stanislaw A. Kucharski, and Jozef Noga. Alternative coupled-cluster ansätze II. The unitary coupled-cluster method. *Chemical Physics Letters*, 155(1):133–140, February 1989.
- [16] Joonho Lee, William J. Huggins, Martin Head-Gordon, and K. Birgitta Whaley. Generalized unitary coupled cluster wave functions for quantum computation. *Journal of chemical theory and computation*, 15 1:311–324, 2019.
- [17] Jonathan Romero, Ryan Babbush, Jarrod R. McClean, Cornelius Hempel, Peter Love, and Al’an Aspuru-Guzik. Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz. *arXiv: Quantum Physics*, page 014008, 2017.
- [18] Jarrod R. McClean, Kevin J. Sung, Ian D. Kivlichan, Yudong Cao, Chengyu Dai, E. Schuyler Fried, Craig Gidney, Brendan Gimby, Pranav Gokhale, Thomas Haner, Tarini Hardikar, Vojtech Havlíček, Oscar Higgott, Cupjin Huang, Josh A. Izaac, Zhang Jiang, Xinle Liu, Sam McArdle, Matthew Neeley, Thomas R O’Brien, Bryan O’Gorman, Isil Ozfidan, Maxwell D Radin, Jhonathan Romero, Nicholas C. Rubin, Nicolas P. D. Sawaya, Kanav Setia, Sukin Sim, Damian S. Steiger, Mark Steudtner, Qiming Sun, Wei Sun, Daochen Wang, Fang Zhang, and Ryan Babbush. Openfermion: The electronic structure package for quantum computers. *arXiv: Quantum Physics*, 2017.
- [19] Michael Broughton, Guillaume Verdon, Trevor McCourt, Antonio Jiménez Martínez,

-
- Jae Hyeon Yoo, S. V. Isakov, Philip Massey, Murphy Yuezhen Niu, Ramin Halavati, Evan Peters, Martin Leib, Andrea Skolik, Michael Streif, David Von Dollen, Jarrod R. McClean, Sergio Boixo, Dave Bacon, Alan Ho, Hartmut Neven, and Masoud Mohseni. Tensorflow quantum: A software framework for quantum machine learning. *ArXiv*, abs/2003.02989, 2020.
- [20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.
- [21] Donald H. Kraft. A software package for sequential quadratic programming. 1988.
- [22] Takeshi Yamazaki, Shunji Matsuura, Ali Narimani, Anushervon Saidmuradov, and Arman Zaribafiyani. Towards the practical application of near-term quantum computers in quantum chemistry simulations: A problem decomposition approach. *ArXiv*, abs/1806.01305, 2018.