Bose-Hubbard Model & Machine Learning

A thesis submitted to
the Department of Physics
at Saint Vincent College
in partial fulfillment
for the degree of
Bachelor of Science

By
Will Mallah
May 2024

Abstract

Although quantum computation is still in the early stages, there are several promising architectures. One of these quantum computing architectures uses Rydberg atoms as qubits. Rydberg atoms, a specific type of Boson, as qubits on a chip, can be modeled by the Bose-Hubbard model. For this reason, we investigated the Bose-Hubbard model. More specifically, we benchmarked code written to simulate this system to determine its accuracy at low scales for extension to higher scale systems. In doing so, it was found that the code operates successfully for the 2 particle, 2 site system. Therefore, we can use the code on higher scale systems with high confidence in the accuracy of its results. Additionally, machine learning models can be implemented to speed up the estimations of desired values within the algorithm.

Contents

1 Introduction

Richard Feynman was the first to conceptualize the idea of using a quantum mechanical system to perform calculations. Not only would this machine be able to perform calculation, but it would also be capable of simulating physical quantum mechanical problems. Through further analysis, Feynman found that quantum computers could solve problems which could not be solved by classical computers, as the calculations may last longer than the age of the universe. More specifically, classical computers require exponentially growing time to solve quantum mechanical many body problems, whereas quantum computers could do so in polynomial time. It was then later discovered by Deutsch, a man whose name is sprinkled through many quantum computing methods and algorithms, that a general-purpose quantum computer is theoretically possible. Moreover, he "showed that any physical process, in principle could be modelled perfectly by a quantum computer".[?]

The first major application of quantum computation was discovered by Peter Shor, famous for Shor's factorization algorithm: this algorithm was used to successfully factor huge numbers rapidly. Since all information on classical computers is kept safe through encryption methods of factoring large prime numbers, this breakthrough gained the attention of many around the world.

One measurement that is of particular interest in quantum computation is entanglement entropy: "Entanglement entropy is a measure of how quantum information is stored in a quantum state".[?]

During the summer of 2023, I worked with Dr. Adrian Del Maestro and his research group at the University of Tennessee Knoxville while attending one of their Research Experience for Undergradutes (REU) programs titled Quantum Alogorithms and Optimization (QAO). During this experience, I worked on benchmarking their algorithm for simulating identical, interacting bosons on a lattice. More specifically, I worked with the small scale

(2 particles on 2 sites) in order to confidently extend the simulation to higher scale (larger system sizes). I began my work by studying the theory of the Bose-Hubbard model (Section 2.1). After learning the theory, I began simulating the system using the PIGSFLI algorithm (Section 3.1). I then compared the results of the PIGSFLI algorithmn to the exact diagonalization (ED) results (Section 3.1.1).

2 Methods

2.1 Bose-Hubbard Calculations

2.1.1 2^{nd} Quantization Von Nuemann Entanglement Entropy Calculation

To get a better initial feeling for this calculation, begin with the simpler case: 2^{nd} quantization or spatial entanglement. The basis for 2^{nd} quantization is as follows:

$$|20\rangle, |11\rangle, |02\rangle. \tag{1}$$

The full Hamiltonian for the Bose-Hubbard system is

$$\hat{H} = -J \sum_i b_i^\dagger b_{i+1} + b_{i+1}^\dagger b_i + \frac{U}{2} \sum_i n_i(n_i - 1) - \mu \sum_i n_i, \tag{2}$$

where J is the hopping term, U is the potential energy, and μ is the chemical potential. $b^\dagger$ and b are the creation and annhilation operators, respectively. They act as follows:

$$b^\dagger|n\rangle = \sqrt{n+1}\ \ |n+1\rangle \tag{3}$$

$$b|n\rangle = \sqrt{n}\ \ |n-1\rangle. \tag{4}$$

The full Hamiltonian can be calculated by calculating the kinetic and potential pieces separately, then simply adding them together. Start by calculating the kinetic energy piece of the full matrix Hamiltonian. The kinetic energy part of the full hamiltonian above for the 2 particle, 2 site system is

$$\hat{H}_{KE} = -J(b_1^\dagger b_2 + b_2^\dagger b_1). \tag{5}$$

The matrix elements of the kinetic energy part of the Hamiltonian are calculated as follows:

$$\begin{aligned}
\langle 20|H_{KE}|20\rangle &= 0 \\
\langle 11|H_{KE}|20\rangle &= -\sqrt{2}J \\
\langle 02|H_{KE}|20\rangle &= 0 \\
\langle 20|H_{KE}|11\rangle &= -\sqrt{2}J \\
\langle 11|H_{KE}|11\rangle &= 0 \\
\langle 02|H_{KE}|11\rangle &= -\sqrt{2}J \\
\langle 20|H_{KE}|02\rangle &= 0 \\
\langle 11|H_{KE}|02\rangle &= -\sqrt{2}J \\
\langle 02|H_{KE}|02\rangle &= 0.
\end{aligned}$$

From these matrix elements, the full kinetic energy piece of the Hamiltonian is written as:

$$\hat{H}_{KE} = \begin{pmatrix} 0 & -\sqrt{2} & 0 \\ -\sqrt{2} & 0 & -\sqrt{2} \\ 0 & -\sqrt{2} & 0 \end{pmatrix}. \tag{6}$$

Next, the potential part of the hamiltonian can be calculated as follows:

$$\hat{H}_P = \frac{U}{2}\sum_i n_i(n_i - 1) = \frac{U}{2}[n_1(n_1 - 1) + n_2(n_2 - 1)], \tag{7}$$

where $n_i = b_i^\dagger b_i$. The elements of the potential part of the Hamiltonian matrix are:

$$\langle 20|H_P|20\rangle = U$$

$$\langle 11|H_P|20\rangle = 0$$

$$\langle 02|H_P|20\rangle = 0$$

$$\langle 20|H_P|11\rangle = 0$$

$$\langle 11|H_P|11\rangle = 0$$

$$\langle 02|H_P|11\rangle = 0$$

$$\langle 20|H_P|02\rangle = 0$$

$$\langle 11|H_P|02\rangle = 0$$

$$\langle 02|H_P|02\rangle = U.$$

From these matrix elements, the full potential energy part of the Hamiltonian is written as:

$$\hat{H}_P = \begin{pmatrix} U & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & U \end{pmatrix}. \tag{8}$$

The separate kinetic and potential energy pieces of the matrix Hamiltonian can be combined together to make the full matrix as follows:

$$\hat{H} = \begin{pmatrix} U & -\sqrt{2} & 0 \\ -\sqrt{2} & 0 & -\sqrt{2} \\ 0 & -\sqrt{2} & U \end{pmatrix}. \tag{9}$$

Now that we have the full matrix Hamiltonian, we can find the energy states of the system by solving the Schrodinger equation. The general form of the Schrodinger equation is

$$\hat{H}|\psi\rangle = E|\psi\rangle. \tag{10}$$

This equation can then be manipulated to look like the following equation:

$$\left(\hat{H} - E\hat{I}\right)|\psi\rangle = 0. \tag{11}$$

Since we're not interested in the solution where $|\psi\rangle = 0$, then $\hat{H} - E\hat{I}$ must be zero. This yields the following characteristic equation:

$$\hat{H} - E\hat{I} = 0. \tag{12}$$

This characteristic equation can be solved via expanding the matrix minors and solving the resulting polynomial, where

$$-E^2 + 2UE^2 + 4J^2E - U^2E - 4J^2U = 0. \tag{13}$$

Solving this equation yields

$$\begin{aligned} E_1 &= U \\ E_2 &= \frac{1}{2}\left(U - \sqrt{16J^2 + U^2}\right) \\ E_3 &= \frac{1}{2}\left(U + \sqrt{16J^2 + U^2}\right). \end{aligned}$$

These are the eigenvalues for the Hamiltonian calculated earlier. Looking at the three energies listed above, it is obvious that E_2 is the ground state energy since it is the lowest of the three. Before the entanglement entropy can be calculated, the density matrix and

reduced density matrix must be found. The full density matrix is defined as $\rho = |\psi\rangle\langle\psi|$. The calculation is as follows:

$$\hat{\rho} = (\alpha|20\rangle + \beta|11\rangle + \gamma|02\rangle)\,(\alpha^*\langle 20| + \beta^*\langle 11| + \gamma^*\langle 02|)\,. \tag{14}$$

Putting our result in matrix notation yields the following:

$$\hat{\rho} = \begin{bmatrix} \alpha\alpha^* & \alpha\beta^* & \alpha\gamma^* \\ \beta\alpha^* & \beta\beta^* & \beta\gamma^* \\ \gamma\alpha^* & \gamma\beta^* & \gamma\gamma^* \end{bmatrix}. \tag{15}$$

The reduced density matrix is calculated as follows:

$$\hat{\rho}_A = \sum_{n=0}^{2} {}_B\langle n|\psi\rangle\langle\psi|n\rangle_B = {}_B\langle 0|\psi\rangle\langle\psi|0\rangle_B + {}_B\langle 1|\psi\rangle\langle\psi|1\rangle_B + {}_B\langle 2|\psi\rangle\langle\psi|2\rangle_B \tag{16}$$

$$\hat{\rho}_A = |\alpha|^2{}_A|2\rangle\langle 2|_A + |\beta|^2{}_A|1\rangle\langle 1|_A + |\gamma|^2{}_A|0\rangle\langle 0|_A. \tag{17}$$

These elements written in matrix notation are as follows:

$$\hat{\rho}_A = \begin{bmatrix} |\alpha|^2 & 0 & 0 \\ 0 & |\beta|^2 & 0 \\ 0 & 0 & |\gamma|^2 \end{bmatrix} \tag{18}$$

Now, we want to calculate the ground state eigenvector. This can be done by plugging our ground state eigenvalue into equation [10] and solving the resulting augmented matrix. This matrix is as follows:

$$\begin{bmatrix} U - \frac{1}{2}\left(U - \sqrt{16J^2 + U^2}\right) & -\sqrt{2}J & 0 \\ -\sqrt{2}J & -\frac{1}{2}\left(U - \sqrt{16J^2 + U^2}\right) & -\sqrt{2}J \\ 0 & -\sqrt{2}J & U - \frac{1}{2}\left(U - \sqrt{16J^2 + U^2}\right) \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = 0. \tag{19}$$

Solving the above equation provides the solutions for α, β, & γ, which are as follows:

$$\alpha = \frac{2}{\sqrt{U'^2 + U'\sqrt{U'+16}+16}} = \gamma$$
$$\beta = \frac{U' + \sqrt{U'+16}}{\sqrt{2}\sqrt{U'^2 + U'\sqrt{U'+16}+16}},$$

where U' is defined as $U' = \frac{U}{J}$ for simplicity. Finally, the Von Nuemann spatial entanglement entropy is calculated as follows:

$$S_1 = -Tr\left(\rho_A \ln \rho_A\right) = -Tr \begin{bmatrix} |\alpha|^2 \ln|\alpha|^2 & 0 & 0 \\ 0 & |\beta|^2 \ln|\beta|^2 & 0 \\ 0 & 0 & |\gamma|^2 \ln|\gamma|^2 \end{bmatrix} \tag{20}$$

$$S_1 = -\left[|\alpha|^2 \ln|\alpha|^2 + |\beta|^2 \ln|\beta|^2 + |\gamma|^2 \ln|\gamma|^2\right] \tag{21}$$

2.1.2 1^{st} Quantization Von Nuemann Entanglement Entropy Calculation

Now, we want to run through the same process with 1^{st} quantization to obtain the particle entanglement. The basis for 1^{st} quantization is as follows:

$$|1_1 2_1\rangle, |1_1 2_2\rangle, |1_2 2_1\rangle, |1_2 2_2\rangle. \tag{22}$$

We begin by calculating the full Hamiltonian, which behaves the same way for 1^{st} quantization as it did for 2^{nd} quantization:

$$\hat{H} = \begin{bmatrix} \langle 1_1 2_1|\hat{H}|1_1 2_1\rangle & \langle 1_1 2_1|\hat{H}|1_1 2_2\rangle & \langle 1_1 2_1|\hat{H}|1_2 2_1\rangle & \langle 1_1 2_1|\hat{H}|1_2 2_2\rangle \\ \langle 1_1 2_2|\hat{H}|1_1 2_1\rangle & \langle 1_1 2_2|\hat{H}|1_1 2_2\rangle & \langle 1_1 2_2|\hat{H}|1_2 2_1\rangle & \langle 1_1 2_2|\hat{H}|1_2 2_2\rangle \\ \langle 1_2 2_1|\hat{H}|1_1 2_1\rangle & \langle 1_2 2_1|\hat{H}|1_1 2_2\rangle & \langle 1_2 2_1|\hat{H}|1_2 2_1\rangle & \langle 1_2 2_1|\hat{H}|1_2 2_2\rangle \\ \langle 1_2 2_2|\hat{H}|1_1 2_1\rangle & \langle 1_2 2_2|\hat{H}|1_1 2_2\rangle & \langle 1_2 2_2|\hat{H}|1_2 2_1\rangle & \langle 1_2 2_2|\hat{H}|1_2 2_2\rangle \end{bmatrix} \tag{23}$$

$$\hat{H} = \begin{bmatrix} U & -\sqrt{2} & -\sqrt{2} & 0 \\ -\sqrt{2} & 0 & 0 & -\sqrt{2} \\ -\sqrt{2} & 0 & 0 & -\sqrt{2} \\ 0 & -\sqrt{2} & -\sqrt{2} & U \end{bmatrix}. \tag{24}$$

The full density matrix is then given by $|\psi\rangle\langle\psi|$, where ψ, the ground state, is given by:

$$\psi = \alpha|1_1 2_1\rangle + \frac{\beta}{\sqrt{2}}\left(|1_1 2_2\rangle + |1_2 2_1\rangle\right) + \gamma|1_2 2_2\rangle.$$

$$\hat{\rho} = \begin{bmatrix} |\alpha|^2 & \frac{\alpha\beta^*}{\sqrt{2}} & \frac{\alpha\beta^*}{\sqrt{2}} & \alpha\gamma^* \\ \frac{\beta\alpha^*}{\sqrt{2}} & \frac{|\beta|^2}{2} & \frac{|\beta|^2}{2} & \frac{\beta\gamma^*}{\sqrt{2}} \\ \frac{\beta\alpha^*}{\sqrt{2}} & \frac{|\beta|^2}{2} & \frac{|\beta|^2}{2} & \frac{\beta\gamma^*}{\sqrt{2}} \\ \gamma\alpha^* & \frac{\gamma\beta^*}{\sqrt{2}} & \frac{\gamma\beta^*}{\sqrt{2}} & |\gamma|^2 \end{bmatrix}. \tag{25}$$

We then needed to calculate the reduced density matrix, which is given by tracing out the degrees of freedom of the second particle. The reduced density matrix is as follows:

$$\hat{\rho}_A = \sum_{n=1}^{2} {}_B\langle n|\psi\rangle\langle\psi|n\rangle_B = {}_B\langle 2_1|\psi\rangle\langle\psi|2_1\rangle_B + {}_B\langle 2_2|\psi\rangle\langle\psi|2_2\rangle_B \tag{26}$$

$$\begin{aligned}\hat{\rho}_A = &\left(|\alpha|^2 + \frac{|\beta|^2}{2}\right) {}_A|1_1\rangle\langle 1_1|_A + \left(\frac{\alpha\beta^* + \beta\gamma^*}{\sqrt{2}}\right) {}_A|1_1\rangle\langle 1_2|_A \\ &+ \left(\frac{\beta\alpha^* + \gamma\beta^*}{\sqrt{2}}\right) {}_A|1_2\rangle\langle 1_1| + \left(\frac{|\beta|^2}{2} + |\gamma|^2\right) {}_A|1_2\rangle\langle 1_2|_A.\end{aligned}$$

$$\hat{\rho}_A = \begin{bmatrix} {}_A\langle 1_1|\hat{\rho_A}|1_1\rangle & {}_A\langle 1_1|\hat{\rho_A}|1_2\rangle \\ {}_A\langle 1_2|\hat{\rho_A}|1_1\rangle & {}_A\langle 1_2|\hat{\rho_A}|1_2\rangle \end{bmatrix} = \begin{bmatrix} |\alpha|^2 + \frac{|\beta|^2}{2} & \frac{\alpha\beta^* + \beta\gamma^*}{\sqrt{2}} \\ \frac{\beta\alpha^* + \gamma\beta^*}{\sqrt{2}} & \frac{|\beta|^2}{2} + |\gamma|^2 \end{bmatrix} \hat{\rho}_A = \begin{bmatrix} \frac{1}{2} & \sqrt{2}\alpha\beta \\ \sqrt{2}\alpha\beta & \frac{1}{2} \end{bmatrix}. \tag{27}$$

To get the Von Nuemann entanglement entropy, the eigenvalues of the above matrix are needed. To obtain these eigenvalues, the following equation must be solved:

$$\hat{\rho}_A - \lambda \hat{I} = 0. \tag{28}$$

In matrix form, this becomes:

$$\begin{bmatrix} \frac{1}{2} - \lambda & \sqrt{2}\alpha\beta \\ \sqrt{2}\alpha\beta & \frac{1}{2} - \lambda \end{bmatrix} = 0. \tag{29}$$

Taking the determinant of this equation yields the characteristic equation:

$$\frac{1}{4} - \lambda + \lambda^2 - 2\alpha^2\beta^2 = 0 \tag{30}$$

$$\lambda^2 - \lambda - 2\alpha^2\left(1 - 2\alpha^2\right) + \frac{1}{4} = 0 \tag{31}$$

$$\lambda^2 - \lambda - 2\alpha^2 + 4\alpha^4 + \frac{1}{4} = 0. \tag{32}$$

The eigenvalues can then be solved for using the quadractic equation:

$$\lambda_\pm = \frac{1 \pm \sqrt{1^2 - 4\,(1)\left(-2\alpha^2 + 4\alpha^4 + \frac{1}{4}\right)}}{2} = \frac{1 \pm 2\alpha\sqrt{2 - 4\alpha^2}}{2}. \tag{33}$$

The Von Nuemann particle entanglement entropy is then calculated as follows:

$$S_1 = -Tr\left(\rho_A \ln \rho_A\right) = -Tr \begin{bmatrix} \lambda_+ & 0 \\ 0 & \lambda_- \end{bmatrix} \begin{bmatrix} \ln \lambda_+ & 0 \\ 0 & \ln \lambda_- \end{bmatrix} \tag{34}$$

$$S_1 = -\lambda_+ \ln \lambda_+ - \lambda_- \ln \lambda_-. \tag{35}$$

3 Discussion

Although calculations above in section 2 show the Von Neumann entanglement entropy, the Renyi entanglement entropy is also a useful measure of entanglement. The Renyi entanglement entropy is defined as

$$S_\alpha = \frac{1}{1-\alpha} \ln\left(\mathrm{Tr}\left(\rho_A^\alpha\right)\right), \tag{36}$$

where ρ_A is the reduced density matrix of the subsystem A and α is a parameter that can be varied. The Renyi entanglement entropy is a generalization of the Von Neumann entanglement entropy, which is the case when $\alpha \to 1$ [REF HERE (Wang, 2016)]. Fig. 1 is a graph of the 2_{nd} Renyi entanglement entropy (i.e., the case when $\alpha = 2$) versus the ratio of the interaction term (U) to the hopping term (J).

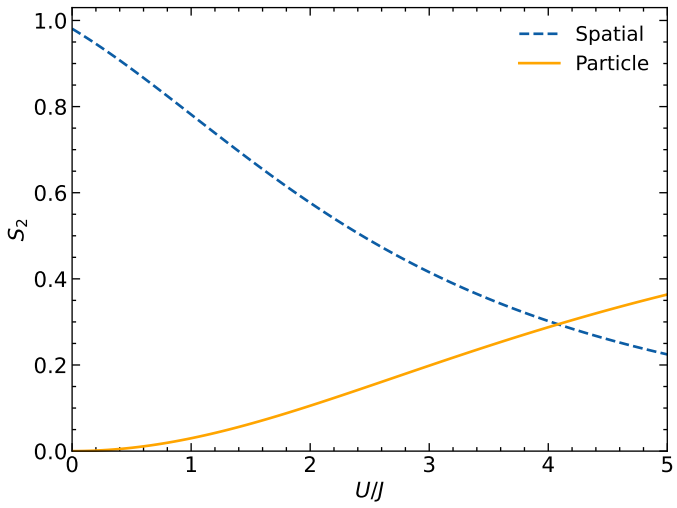

Figure 1: The figure above is a graph of the Renyi entanglement entropy for both spatial and particle bipartitions versus the ratio of the interaction term to the hopping term.

3.1 PIGSFLI Algorithm

A Monte Carlo simulation is a stochastic (random) method of integration, which allows for accurate estimations of desired values. Monte Carlo is necessary for situations where exact calculations are too computationally expensive, such as in the exact diagonalization of the reduced density matrix for high particle/site number Bose Hubbard systems. For example, the size of the Hilbert space is calculated by Eq. (37) and shown by Fig. 2:

$$D = \frac{(N + L - 1)!}{(N)!\,(L - 1)!}. \tag{37}$$

In this example, the Hilbert space size is calculated for a system with N particles and L sites. The Hilbert space size is the number of possible states that the system can be in. For example, a system with $N = 2$ particles and $L = 2$ sites has a Hilbert space size of 3. This means that there are 3 possible states that the system can be in. The largest system size shown by Fig. 2 is 20 particles and 20 sites, which has a Hilbert space size of 6.8×10^{10} or about 68 billion possible states. For this reason, exact diagonalization is not feasible for large system sizes and Monte Carlo methods are necessary with classical computation.

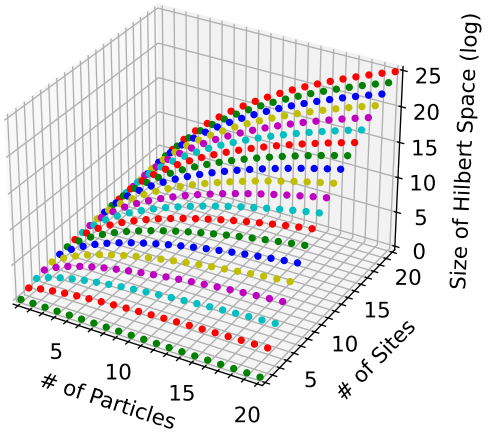

Figure 2: The figure above shows the size of the Hilbert space as a function of number of sites and particles. This 3-dimensional plot for the number of combinations in the Hilbert space goes up to 20 sites and 20 particles.

Fig. 3 shows the total (both kinetic and potential) ground state energy of the 2 particle, 2 site system versus the ratio of the interaction term (U) to the hopping term (J). More specifically, this plot contains both data points generated by the PIGSFLI algorithm and the curve we would expect theoretically from adding the expectation values of both the kinetic and potential energies. The expecation values of the kinetic and potential energies are as follows:

$$K.E. = \langle\psi|\hat{H}_K|\psi\rangle$$

$$P.E. = \langle\psi|\hat{H}_P|\psi\rangle$$

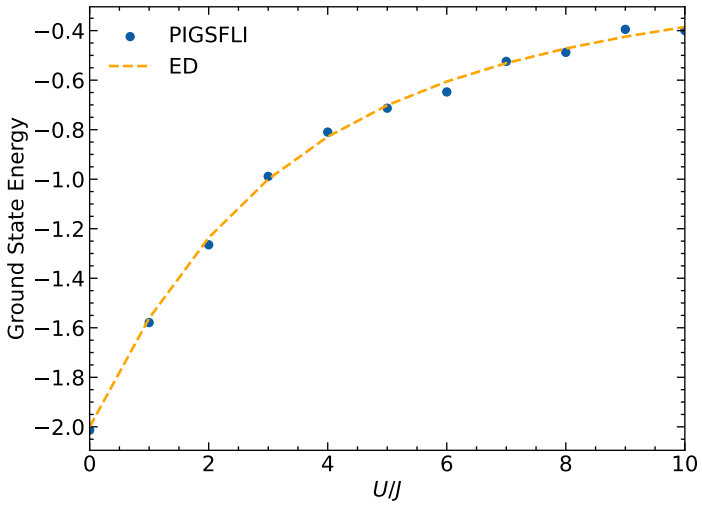

Figure 3: A plot of the ground state energy versus the ratio between U, the potential term, and J, the hopping term for both exact diagonalization and PIGSFLI algorithm. From this plot, the PIGSFLI reults match very well to our ED calculation.

A good way to ensure the PIGSFLI algorithm appears correct from a general physics sense, it is important to look at the occupancy graphs for each site: occupancy is essentially the probability of that site being occupied by a particle. These plots are shown in Figs. 4 to 7 below.

The first two plots show the case where the interaction term is small. Since the interaction term is repulsive, the particles should exist on the sites with equal probability.

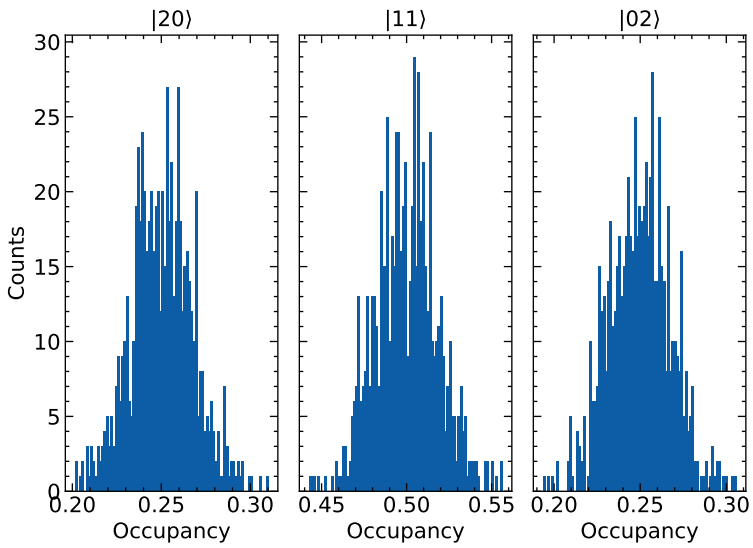

Figure 4: The figure above shows the histogram of the occupation number for the particle bipartition as a function of the ratio of the interaction term to the hopping term. This graph specifically shows the case where the interaction term is a tenth of the hopping term.

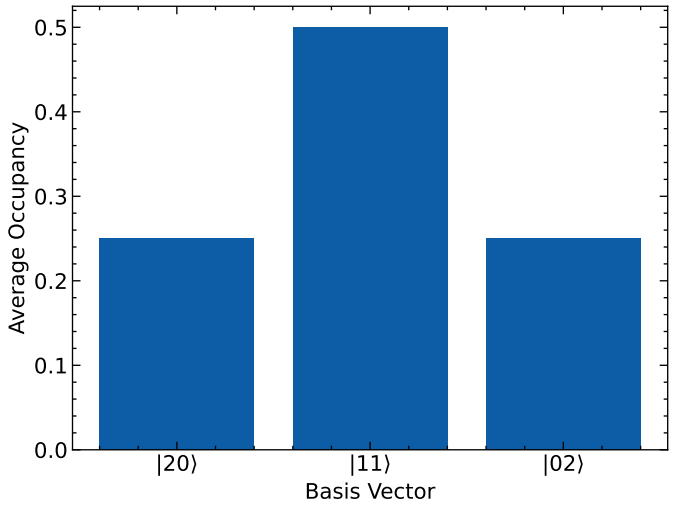

Figure 5: The figure above shows the average occupation number for the spatial bipartition as a function of the ratio of the interaction term to the hopping term. This graph specifically shows the case where the interaction term is a tenth of the hopping term.

The second two plots show the case where the interaction is much larger. Since the

interaction term is repulsive, the particles should converge on the $|11\rangle$ state: the state where they exist the farthest distance from each other.

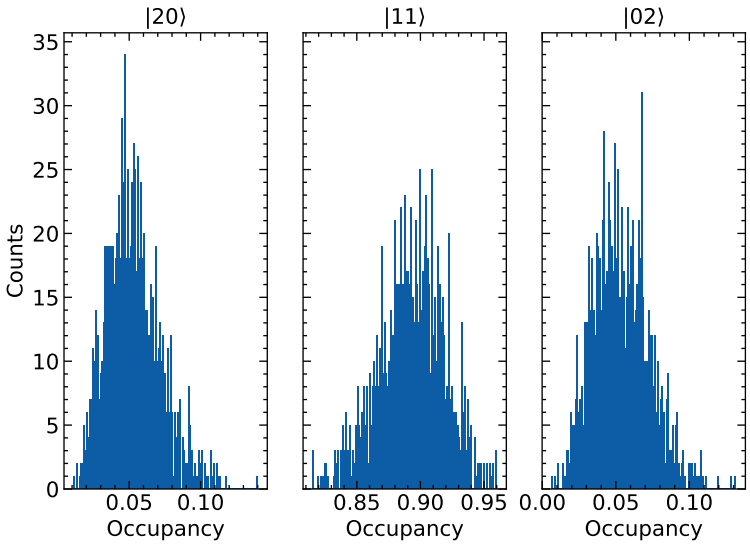

Figure 6: The figure above shows the histogram of the occupation number for the spatial bipartition as a function of the ratio of the interaction term to the hopping term. This graph specifically shows the case where the interaction term is 10 times the hopping term.

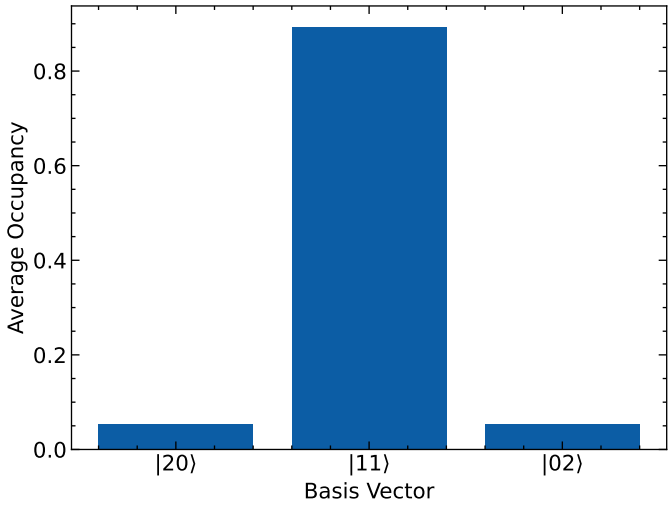

Figure 7: The figure above shows the average occupation number for the spatial bipartition as a function of the ratio of the interaction term to the hopping term. This graph specifically shows the case where the interaction term is 10 times the hopping term.

3.1.1 Comparison of PIGSFLI to Exact Diagonalization

Fig. 8 is a plot of the second Renyi entanglement entropy versus the ratio of the interaction term (J) to the hopping term (U). More specifically, this plot contains both data points generated by the PIGSFLI algorithm and the curve we would expect theoretically from Eq. (37).

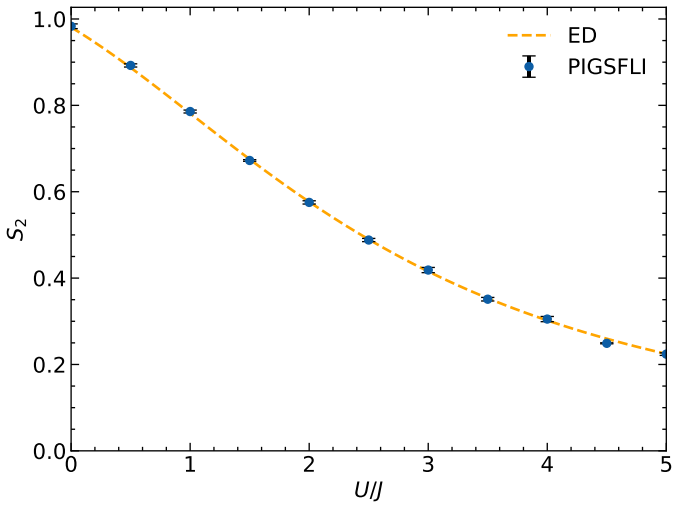

Figure 8: The second Rényi spatial entanglement entropy as a function of U/J for both exact diagonalization (ED) and PIGSFLI. This plot shows excellent agreement between the PIGSFLI estimation and that of our ED calculation, showing that PIGSFLI is a viable option for computation on much larger system sizes.

3.2 Introduction to Machine Learning for Scientists

While working with the DelMaestro group and attending lectures held during the QAO REU at the University of Tennessee Koxville, I was exposed to machine learning as a tool for physicists and other scientists alike.

3.2.1 Applications of DNN to Physics

Application: Learning the Energy Potential

Description: Train network on small system sizes (which we know the exact solution for), then extrapolate for application of larger system sizes.

Application: Phase Discrimination

Description: Supervised and un-supervised machine learning used to classify phases of matter.

Application: Variational Ansatz

Description: Method of guessing the wave function to solve the Schrodinger equation for a given potential quickly.

Application: Solving the Schrodinger Equation

Description: Train a neural network to solve the Schrodinger equation for a given potential.

Application: Speeding up Monte Carlo

Description: Monte Carlo update algorithms are highly dependent on the model. Therefore, a neural network can be trained to predict the next state of the system, which can speed up the Monte Carlo simulation.

3.2.2 Basic Neural Networks

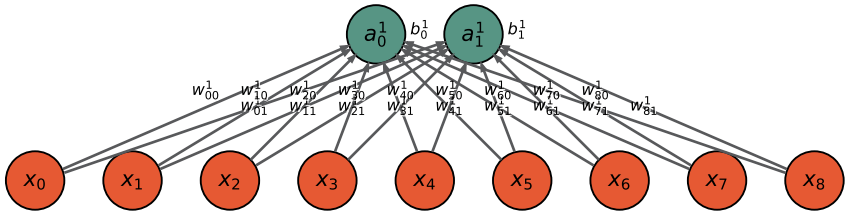

Figure 9: The figure above is an example of a simple feed-forward neural network, taking in nine inputs and returning two outputs.

Neural Network: non-linear function of many variables that depends on a large number of parameters

Deep Neural Network: a neural network with one or more hidden layers (i.e., not an input or output layer)

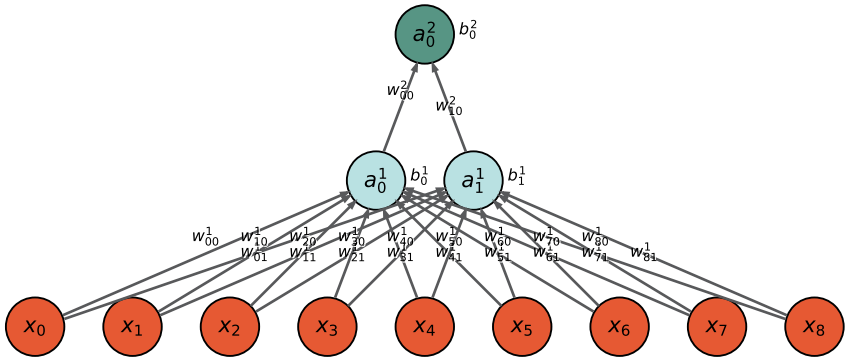

Figure 10: The figure above is an example of a simple feed-forward deep neural network, taking in nine inputs and returning one output with one hidden layer between the input and output layers.

3.2.3 Visualizing Feed Forward

Some examples to visualize complexity generated from simple feed forward neural network with 2 input parameters and 2 hidden layers with 200 nodes each are shown below:

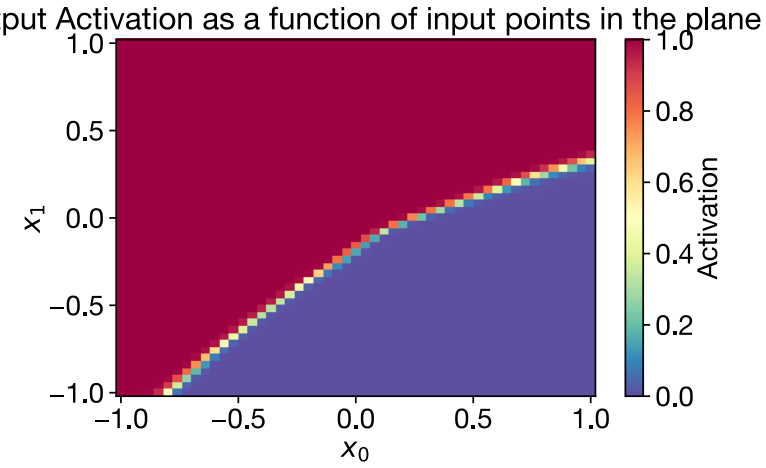

Figure 11: The figure above is the activation output as a function of the two input parameters x_0 and x_1, which were randomly generated. The activation is calculated through use of the sigmoid function. Specifically, this is a heat map where the color is associated with the activation level.

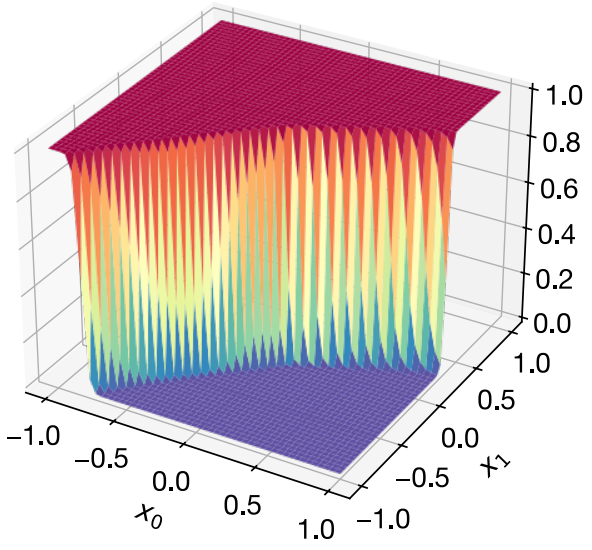

Figure 12: The figure above is the activation output as a function of the two parameters x_0 and x_1, which were randomly generated. The activation is calculated through use of the sigmoid function. Specifically, this is a heat map in 3-dimensional space where the activation is the on the vertical axis.

3.2.4 Batch Processing

Multiple batches of data can be processed at once, which can be more efficient than processing one set of data at a time. This is especially useful when running the neural network on a GPU, which can process multiple batches of data in parallel. The most common way of achieving batch processing is through linear algebra methods with Numpy, which is a Python library for numerical computing.

3.2.5 Linear Regression

3.2.5.1 Linear Regression Example

The radioactive decay of an unknown sample is described by the following equation:

$$N(t) = N(0)e^{-t/\tau}, \tag{38}$$

where $N(t)$ is the number of atoms at time t, $N(0)$ is the number of atoms at time $t = 0$, t is the time in seconds, and τ is the time constant of the decay. The above equation can be rearranged into a linear relationship of the form

$$N(t) = \ln\left(N(0)\right) - \frac{1}{\tau}t. \tag{39}$$

For our simple feed forward neural network, this is similar to the following form:

$$F = w_o + w_1 t, \tag{40}$$

were F is the function we're trying to fit, w_0 and w_1 are the weights, which are (potentially non-linear) functions of the unknown parameters $N(0)$ and τ, and t is the time in seconds. Plotting the original data as well as the linear regression calculation, we find the following:

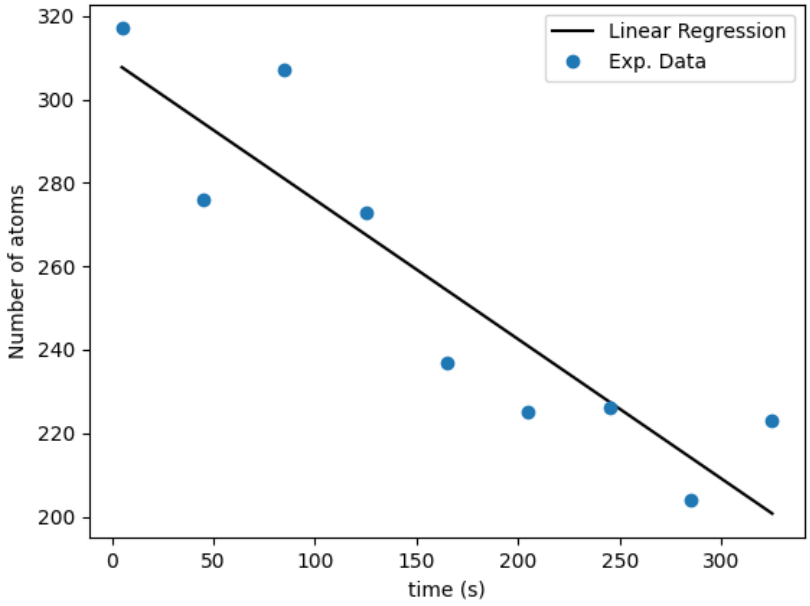

Figure 13: This figure is a plot of the number of atoms versus time passed in seconds. The blue data points in the figure above represent the experimental data and the black line represents the linear fit to this data using a simple feed-forward neural network.

4 Appendix A: More interesting details

Here are details in an Appendix.