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 calculations, 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 as systems become more complex (i.e., more particles), whereas quantum computers could do so in polynomial time. It was then later discovered by David Deutsch of Oxford University, 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".[1]

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. Now, with quantum computation on the rise, people in fields external to physics and information science have begun to discover new applications. One such important application is simulating molecules, also known as quantum chemistry. Simulating quantum chemistry will aid the medical field in a revolutionary way: possibly leading to a cure for cancer or other diseases/ailments. Similarly, quantum computation may be used to eradicate starvation through optimal allocation of resources to people around the world. Similar to a quantum computer's ability to simulate large quantum mechanical system sizes where there are enumerable possible states the system can exist in, quantum computers are efficient at running optimization algorithms where there are very large numbers of states for the system.

For this reason, quantum computation would be useful to distribute resources around the globe. These applications demonstrate the possibility for the utopian reality we've all read and dreamt about: a disease and hunger-free world.

One measurement that is of particular interest in quantum computation is entanglement entropy. Entanglement is a strictly quantum phenomenon where particles within a system cannot be represented mathematically in a way that is completely separate from each other: in short, the particles are fundamentally connected to each other. A more general way to quantify this correlation between partitions of a system is entanglement entropy. Entanglement entropy generalizes from single particles to subsystems. Explicitly, entanglement entropy quantifies the correlations between partitions of a system. Like regular entanglement, entanglement entropy rises from the partitions of the system lacking definite quantum states on their own: i.e., the state of each partition depends on the other. The most general definition is that "Entanglement entropy is a measure of how quantum information is stored in a quantum state".[2] The entanglement piece stems from the uncertainty in the state of the system as a result of the entanglement between partitions. This concept is very similar to classical entropy where the microstate of the system is uncertain. This measure of entanglement entropy was the focus of the research I conducted over the summer of 2023 and continued during the 2023-2024 school year.

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 Undergraduates (REU) programs titled Quantum Algorithms and Optimization (QAO). During this experience, I worked on benchmarking their algorithm for simulating identical, interacting bosons on a lattice. Specifically, I worked with the small scale (2 particles on 2 sites) Bose-Hubbard model in order to confidently extend the simulation to a 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 algorithm to the exact diagonalization (ED) results (Section 3.2.1).

2 Methods

The Bose-Hubbard model simulates identical, interacting bosons on lattice. These bosons (particles with whole integer spin) are allowed to live in the same quantum state as other bosons. In the case of the system we looked into, this meant that the bosons were allowed to live on the same site as other bosons. Therefore, an infinite number of bosons could live on the same site at the same time. This system of bosons on a lattice has applications in quantum computation through its use in describing Rydberg atom quantum hardware, where Rydberg atoms are a specific subset of bosons (atoms in this case). To quote Wu in his 2021 paper, "More generally, any atom (moleculeor semiconductor quantum dot) in a state with a highly-excited electron, i.e., with valance electron in a large principal quantum number n state of several tens to hundreds or even higher, is regarded as Rydberg atom."[3]

The following subsection consists of the by-hand exact-diagonalization calculations I preformed during my research experience at the University of Tennessee Knoxville in the summer of 2023. These calculations are split into two main pieces: the first is for first quantization and the second for second quantization. First quantization consists of labeling the system according to the sites where the bosons exist. Second quantization consists of labeling the system according to the individual particles themselves.

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, we begin with the simpler case: 2^{nd} quantization or spatial entanglement. Additionally, systems bigger than 2 particles on 2 sites make by-hand calculations unreadable. Therefore, all following calculations and data analysis only deal with 2 particles on 2 sites for the Bose-Hubbard model. The basis for 2^{nd} quantization with 2 particles on 2 sites is as follows:

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

where the left-most number corresponds to the number of bosons on site A and the right-most on site B. 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 annihilation operators, respectively. They act as follows:

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

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

J, the hopping term, is also denoted as the tunneling term. This term governs how well the particles are able to move between sites. U, the potential energy term, governs the repulsive force between the bosons. This repulsive potential is a result of the bosons approaching each other in close proximity to where their electrons clouds interact, forcing the bosons in opposite directions. 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 from Eq. (2). 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:

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

Next, the potential part of the full Hamiltonian from Eq. (2) 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:

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

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

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

Where E is the energy of the system and $|\psi\rangle$ is the wavefunction that describes the system. 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 are 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

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

The solutions to Eq. (13) are listed above and are the eigenvalues for the Hamiltonian. 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:

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

In matrix notation:

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

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 given by:

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

Now, we want to calculate the ground state eigenvector. This can be done by substituting 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 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 $U' = \frac{U}{J}$. Finally, the Von Nuemann spatial entanglement entropy is calculated as follows:

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

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

This result shows that the Von Neumann entanglement entropy depends only on the elements of the ground state vector of the system: α, β, and γ.

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

Having obtained the Von Neumann entanglement entropy for 2^{st} quantization, we want to run through the same process with 1^{st} quantization to obtain the particle entanglement. In the case of 1^{st} quantization, we label the particles rather than the sites. The basis for 1^{st} quantization is as follows:

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

where the state numbers (i.e., the base numbers) are the particle labels and the subscripts are the site labels (i.e., which site the particle exists on). 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:

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

where α, β, and γ are the elements of the ground state vector of the system. The full density matrix is

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

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:

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

The only surviving terms from the above calculation are

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

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

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:

$$\hat{\rho}_A - \lambda\hat{I} = 0, \quad (28)$$

where λ is the eigenvalue we are solving for. In matrix form, this becomes:

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

Taking the determinant of this equation yields the characteristic equation:

$$\lambda^2 - \lambda - 2\alpha^2 + 4\alpha^4 + \frac{1}{4} = 0. \quad (30)$$

The eigenvalues can then be calculated using the quadratic equation:

$$\lambda_\pm = \frac{1 \pm \sqrt{1^2 - 4\,(1)\left(-2\alpha^2 + 4\alpha^4 + \frac{1}{4}\right)}}{2} = \frac{1 \pm 2\alpha\sqrt{2 - 4\alpha^2}}{2}. \quad (31)$$

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

$$S_1 = -\mathrm{Tr}\left(\rho_A \ln \rho_A\right) = -\mathrm{Tr}\begin{bmatrix} \lambda_+ & 0 \\ 0 & \lambda_- \end{bmatrix}\begin{bmatrix} \ln\lambda_+ & 0 \\ 0 & \ln\lambda_- \end{bmatrix} \tag{32}$$

$$S_1 = -\lambda_+ \ln\lambda_+ - \lambda_- \ln\lambda_- \tag{33}$$

where λ_+ and λ_- are defined by Eq. (31). Similar to the Von Neumann entanglement entropy calculation for 2^{nd} quantization, the entanglement entropy for 1^{st} quantization also depends only on the elements of the ground state vector for the system.

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

$$S_\alpha = \frac{1}{1-\alpha}\ln\left(\mathrm{Tr}\left(\rho_A^\alpha\right)\right), \tag{34}$$

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 $\alpha \to 1$.[4] 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).

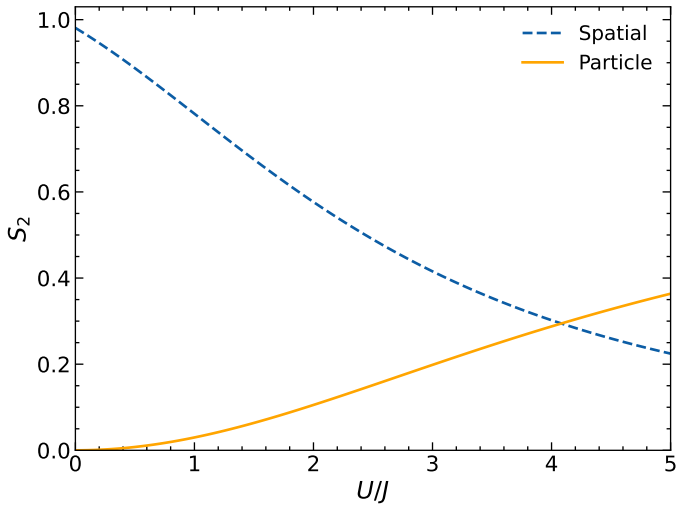

Figure 1: The figure above is a graph of the Renyi entanglement entropy for both spatial and particle bipartitions versus the ratio of the interaction term to the hopping term. Again, this is for the 2 particle, 2 site Bose-Hubbard model.

3.1 PIGSFLI Algorithm

PIGSFLI stands for Path Integral Ground State (Quantum Monte Carlo) for Lattice Implementation. Quoting Casiano-Diaz in his paper, "Path Integral Ground State (PIGS) quantum Monte Carlo utilizes the projection of a trial wave function in imaginary time to obtain stochastically exact results for the ground state of a quantum many-body system."[5] For our uses, the algorithm operates for 2^{nd} quantization measurements in the Bose-Hubbard model.

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. (35) and shown by Fig. 2:

$$D = \frac{(N + L - 1)!}{(N)!\,(L - 1)!}, \tag{35}$$

where N is the number of particles in the system and L is the number of sites in the system. 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.

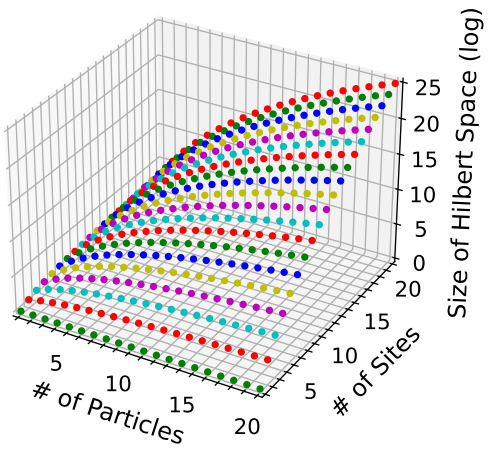

Figure 2: The figure above shows the size of the Hilbert space as a function of number of sites and particles. This 3-dimensional plot for the number of combinations in the Hilbert space goes up to 20 sites and 20 particles.

3.2 Results

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). 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 expectation values of the kinetic and potential energies are as follows:

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

where ψ is the wavefunction, $\hat{H}_K$ is the kinetic energy operator, and $\hat{H}_P$ is the potential energy operator.

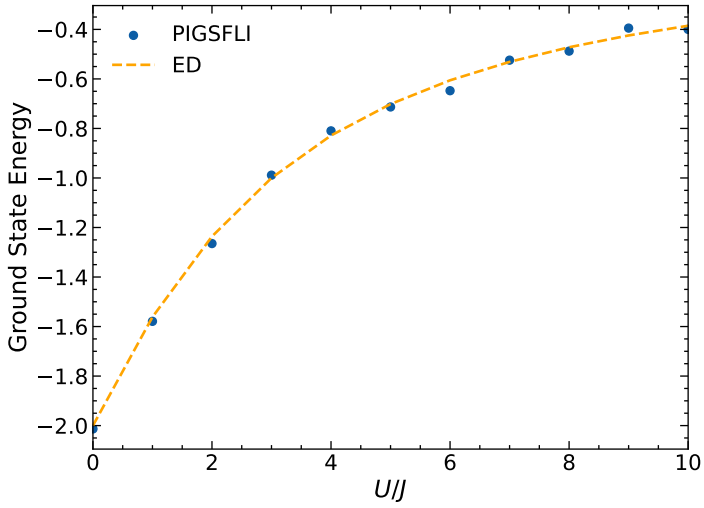

Figure 3: A plot of the ground state energy versus the ratio between U, the potential term, and J, the hopping term for both exact diagonalization and PIGSFLI algorithm. From this plot, the PIGSFLI reults match very well to our exact-diagonalization calculation.

A good way to ensure the PIGSFLI algorithm appears correct from a general physics sense is to analyze the occupancy graphs for each site: occupancy is essentially the probability of that site being occupied by a particle. Figures 4 to 7 below show the occupancy graphs for low and high interaction.

Figures 4 and 5 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.

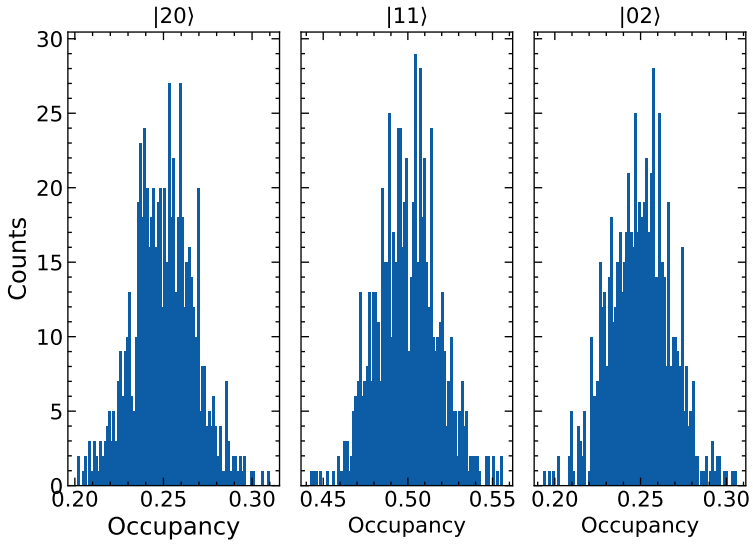

Figure 4: The figure above shows the histogram of the occupation number for the particle bipartition as a function of the ratio of the interaction term to the hopping term. This graph specifically shows the case where the interaction term is a tenth of the hopping term.

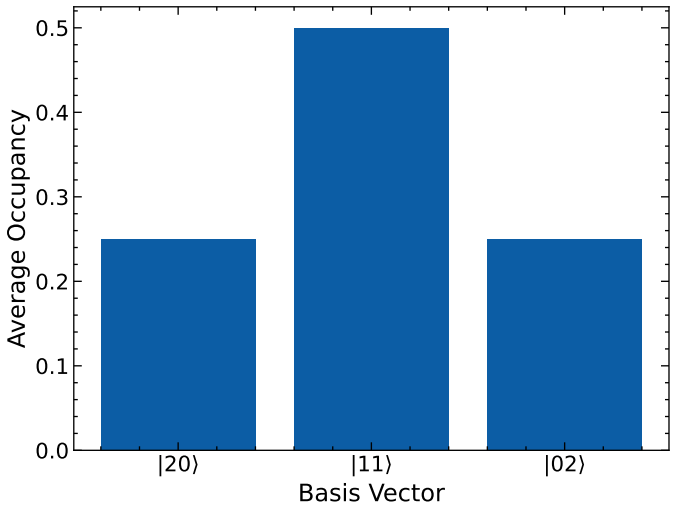

Figure 5: The figure above shows the average occupation number for the spatial bipartition as a function of the ratio of the interaction term to the hopping term. This graph specifically shows the case where the interaction term is a tenth of the hopping term.

Figures 6 and 7 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.

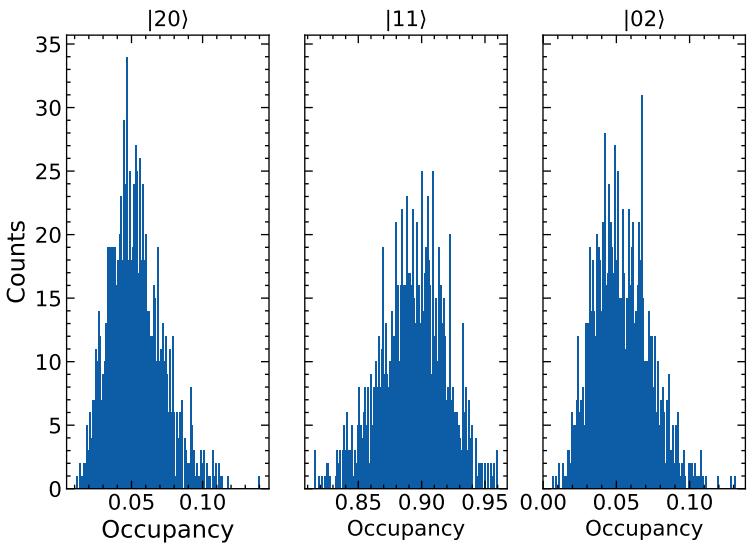

Figure 6: The figure above shows the histogram of the occupation number for the spatial bipartition as a function of the ratio of the interaction term to the hopping term. This graph specifically shows the case where the interaction term is 10 times the hopping term.

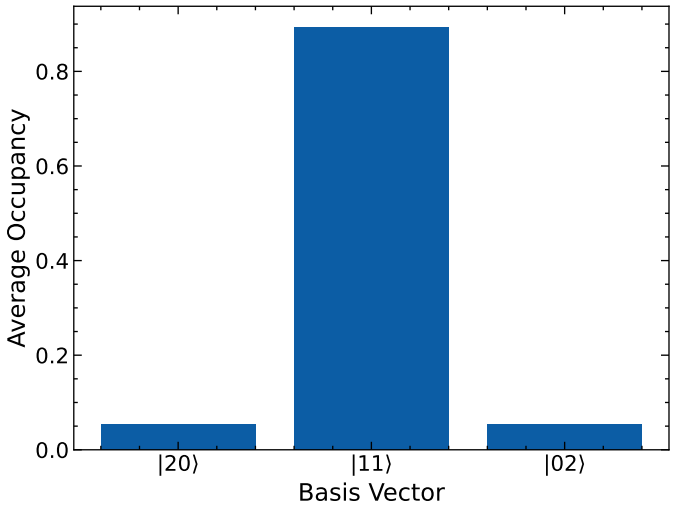

Figure 7: The figure above shows the average occupation number for the spatial bipartition as a function of the ratio of the interaction term to the hopping term. This graph specifically shows the case where the interaction term is 10 times the hopping term.

Figure 5 demonstrates that the system does not converge on any specific state but

rather has equal probability of existing in either of the two unique basis states: $|20\rangle$ (equivalently for 2 particles on site B, $|02\rangle$) or $|11\rangle$. Figure 7 demonstrates that the system does converge on the $|11\rangle$ when the ratio of the interaction term to the hopping term is high: 10 in our case. Since this is the result that we expected, we know that the physics built into the algorithm is functioning as desired.

3.2.1 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). Specifically, this plot contains both data points generated by the PIGSFLI algorithm and the curve we would expect theoretically from Eq. (35).

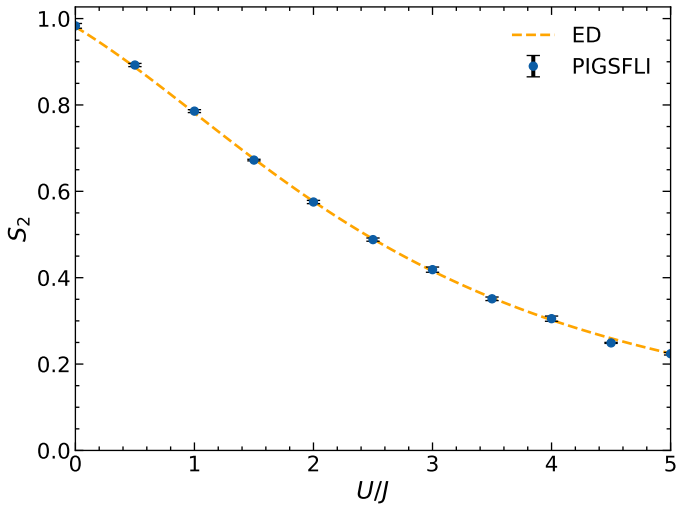

Figure 8: The second Rényi spatial entanglement entropy as a function of U/J for both exact diagonalization (ED) and PIGSFLI. This plot shows excellent agreement between the PIGSFLI estimation and that of our ED calculation, showing that PIGSFLI is a viable option for computation on much larger system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

As stated in the caption above, this plot shows excellent agreement between the

measurement values expected theoretically and the measurement values estimated by the PIGSLFI algorithm. For this reason, we are confident in extending the code to larger system sizes with the same accuracy achieved from the small scale system we tested.

4 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 Knoxville, I was exposed to machine learning as a tool for physicists and other scientists alike. The professor I worked with, Dr. Adrian Del Maestro, provided me with the lecture videos and other course materials which he used to educate during covid.

4.1 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.

4.2 Basic Neural Networks

Basic neural networks consist of layers, or sets of horizontal node (2 layers in Fig. 9), with their own weights and biases that determine how the information is propagated through the architecture to the output.

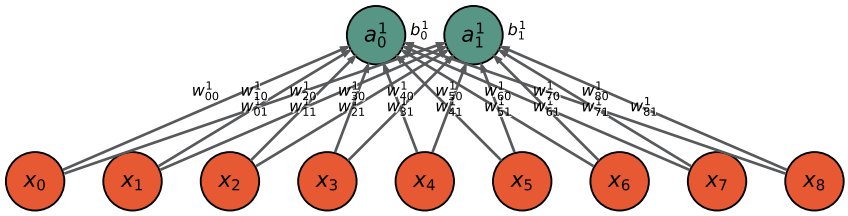

Figure 9: The figure above is an example of a simple feed-forward neural network, taking in nine inputs and returning two outputs. The x nodes are input values, a nodes are nodes in layers following the input layer, w values are the weights of each connection between nodes, and b values are the biases or activation values of each node.

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).

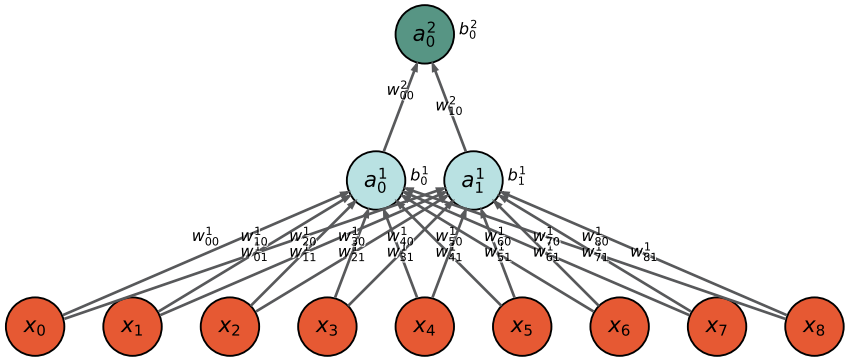

Figure 10: The figure above is an example of a simple feed-forward deep neural network, taking in nine inputs and returning one output with one hidden layer between the input and output layers. The x nodes are input values, a nodes are nodes in layers following the input layer, w values are the weights of each connection between nodes, and b values are the biases or activation values of each node.

4.3 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:

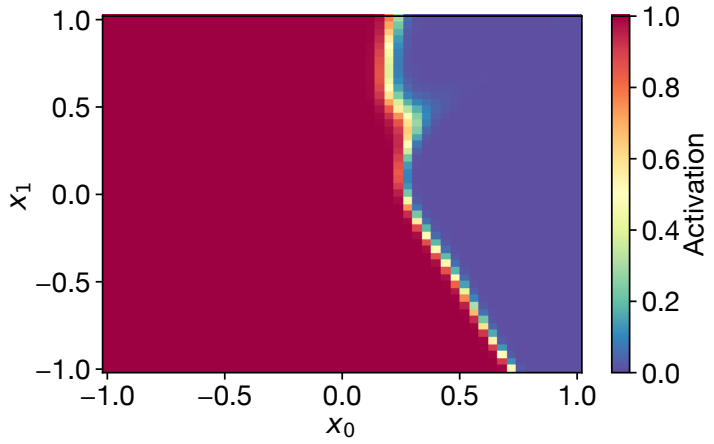

Figure 11: The figure above is the activation output as a function of the two input parameters x_0 and x_1, which were randomly generated. The activation is calculated through use of the sigmoid function. Specifically, this is a heat map where the color is associated with the activation level.

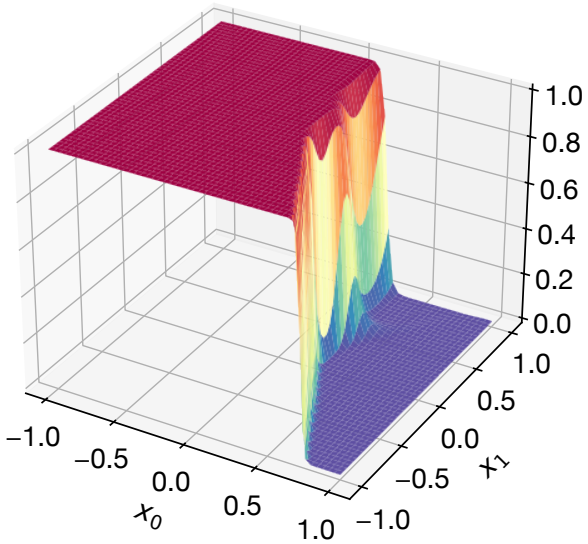

Figure 12: The figure above is the activation output as a function of the two parameters x_0 and x_1, which were randomly generated. The activation is calculated through use of the sigmoid function. Specifically, this is a heat map in 3-dimensional space where the activation is the on the vertical axis.

4.4 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,[6] which is a Python library for numerical computing.

4.5 Linear Regression

4.5.1 Linear Regression Example

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

$$N(t) = N(0)e^{-t/\tau}, \tag{36}$$

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

$$N(t) = \ln\left(N(0)\right) - \frac{1}{\tau}t. \tag{37}$$

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

$$F = w_o + w_1 t, \tag{38}$$

were F is the function we are 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 in Fig. 13:

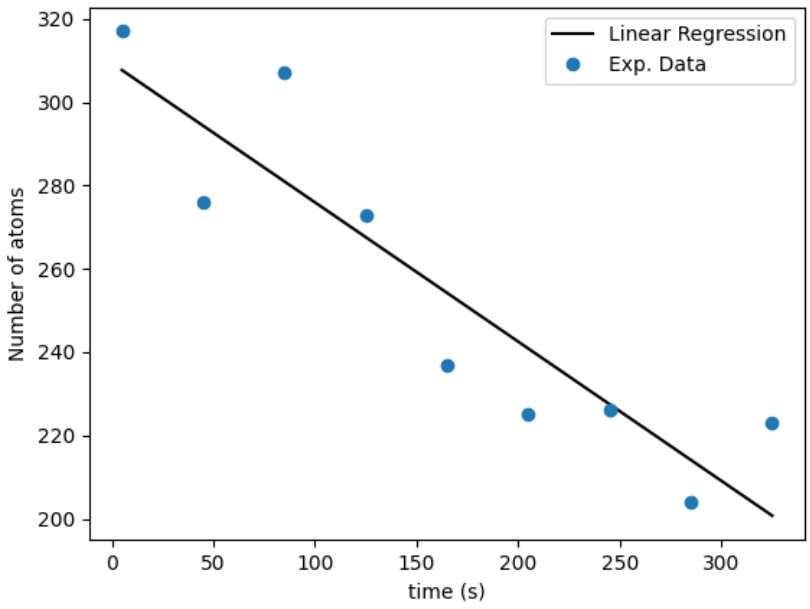

Figure 13: This figure is a plot of the number of atoms versus time passed in seconds. The blue data points in the figure above represent the experimental data and the black line represents the linear fit to this data using a simple feed-forward neural network.

5 Conclusion

After demonstrating that the PIGSFLI algorithms operates as expected according to theory, I was given the opportunity to continue working with the DelMaestro group virtually through their weekly Zoom research meetings. During the time I worked with the group virtually, I profiled the PIGSFLI algorithm to find where it was spending more time than necessary. Although this search was not fully successful, I learned much about both how the code operates and many aspects of C++ which I had not yet learned. In the Fall of

2024, I will be attending the University of Tennessee Knoxville for their P.h.D. program in condensed matter theory, where I will continue to work with the DelMaestro research group. Not only will I most likely continue work with the PIGSFLI code and the Bose-Hubbard model, but I may also act as a teaching assistant for Dr. Del Maestro's Introduction to Machine Learning for Scientists. I look forward in continuing work with the DelMaestro group on the Bose-Hubbard model and machine learning.

References

[1] Singh, P. A study on the basics of Quantum Computing (2005).

[2] Hartman, T. Lectures on Quantum Gravity and Black Holes (2015).

[3] Wu, X. *et al.* A concise review of Rydberg atom based quantum computation and quantum simulation (2021).

[4] Wang, Y.-X., Mu, L.-Z., Vedral, V. & Fan, H. Entanglement Renyi alpha-entropy (2016).

[5] Emanuel, C.-D., Herdman, C. & Del Maestro, A. PIGSFLI: A Path Integral Ground State Monte Carlo Algorithm for Entanglement of Lattice Bosons (2022).

[6] Harris, C. R. *et al.* Array programming with NumPy. *Nature* **585**, 357–362 (2020). URL https://doi.org/10.1038/s41586-020-2649-2.