Application of Machine Learning Algorithms to Many-body Quantum Dynamics

Alev Orfi

Supervisor: Bill Coish

Collaborator: Felix Fehse

(Dated: April 14, 2020)

Many-particle quantum dynamics is computationally challenging to simulate, since the number of amplitudes required to describe a general quantum state grows exponentially with the number of particles. For some specific many-body problems, machine-learning algorithms have been effective in reducing this complexity [1]. In particular, restricted Boltzmann machines have been used to represent the wavefunction of the system, with fewer degrees of freedom [2] [3]. Using this method, as the system size scales, relatively fewer parameters may be needed to describe the dynamics.

The advantage of machine learning algorithms has been demonstrated on problems which are known to be tractable. We hope to apply this technique to problems where there may be no known simple analytical solution. This project examines the central-spin problem, a model of the interaction between a central spin within an external field, and many environmental spins. A restricted Boltzmann machine was constructed and trained to find the ground state of this central-spin problem for varying system sizes. Two verification methods, one using analytical calculation, the other a numerical approach, were implemented to test the results of this machine-learning algorithm.

Accurate ground-state determination was achieved with the restricted Boltzmann machine reparameterization, showing its ability to model this system. Various minimization algorithms, used to train the restricted Boltzmann machine, were explored. A future goal of this project is to extend this restricted Boltzmann machine to model the dynamics of this spin system, with the hope that this machine-learning reparametrization will reduce the computational challenge of this many-body system.

I. INTRODUCTION

Many-body quantum systems are those which consist of multiple microscopic systems made from interacting particles. The wave function of the entire system must contain sufficient information to describe all the correlations between particles, making analytic calculations of certain large systems impractical. Some examples of these systems include collections of coupled spins in a magnetic material and interacting electrons in a metal. It is important to model the dynamics of these systems to understand how they respond to sudden external changes, such as a change in a local magnetic or electric field. Solving for the dynamics of these systems equates to identifying the eigenstates and eigenvalues of the Hamiltonian of the system. Once these states are found, the time evolution and expectation values of observables can easily be calculated. A common class of many-particle models are spin-$\frac{1}{2}$ systems, which consists of a number of two-level systems. When assuming a time-independent Hamiltonian and in units where $\hbar = 1$, the evolution is described by the Schroedinger equation,

$$\partial_t \left|\psi(t)\right\rangle = -iH \left|\psi(t)\right\rangle . \tag{1}$$

Given the initial state of the system $\psi(0)$, the state at any time is given by

$$\left|\psi(t)\right\rangle = e^{-iHt} \left|\psi(0)\right\rangle . \tag{2}$$

Exact diagonalization of the Hamiltonian, H, allows this exponential to be written in terms of the Hamiltonian's eigenvalues. The time evolution of the state $|\psi\rangle$ can then be represented as a linear combination of the eigenstates $|S\rangle$,

$$\left|\psi(t)\right\rangle = \sum_{\{S\}} C_S(t) \left|S\right\rangle . \tag{3}$$

In a system with N spins, each site is described by a two-dimensional Hilbert space. Thus, in total there are 2^N possible spin configurations. In order to fully represent all interactions, the Hamiltonian is of size $2^N \times 2^N$ with 2^N basis states $|S\rangle$. This exponential scaling leads to an inherent difficulty in solving the time evolution of large systems by exact diagonalization.

This computational complexity problem is broadly one of dimensional reduction. It is more feasible to solve large-N systems if the wavefunction can be expressed with fewer parameters.

Machine learning models have been effective in reducing complexity through inference and pattern recognition. One particular neural network, a restricted Boltzmann machine (RBM), gives a lower-dimensional reparametrization of many-body systems. The RBM gives an alternative form for the coefficients $C_S(t)$ in Eq. (3),

$$C_S(t) \simeq \Psi(S; \mathcal{W}(t)), \tag{4}$$

in terms of network parameters $\mathcal{W} = \{a, b, W_{ij}\}$ and spin variables S, as seen in Fig. 5. This RBM representation has been shown to be an efficient method for finding a ground state estimate and the dynamics of the transverse field Ising model and the antiferromagnetic Heisenberg model [1].

A RBM is a neural network which graphically represents a probability function, in terms of network parameters. The network structure can be seen in Fig. 1. In this project a RBM is extended to parametrize a wave function by using complex network variables. A RBM consists of a number of interacting stochastic variables, known as nodes. There are two layers of these nodes, hidden and visible, which are connected through a matrix of weights, W_{ij}. The visible layer consists of physical variables of the system. In our system these correspond to the spin operators at each spin site, $S = \{\sigma_0^z, \sigma_1^z, ..., \sigma_{N-1}^z\}$, with σ^z the standard Pauli-z operator. The hidden layer consists of M auxiliary spin variables $h_1, h_2, ..., h_M$. The hidden variables are used to capture higher-order correlation between the visible nodes. Both the visible and hidden nodes have corresponding biases, $a_0, ..., a_{N-1}$ and $b_1, ..., b_M$, repectivly. The RBM gives a representation for the coefficients seen in Eq. (4), in terms of these network parameters. Specifically these coefficients are given by [1],

$$\Psi(S; \mathcal{W}(t)) = \sum_{\{h_i\}} e^{\sum_j a_j \sigma_j^z + \sum_i b_i h_i + \sum_{ij} W_{ij} h_i \sigma_j^z}. \tag{5}$$

Summing over the basis states gives the following representation of the wavefunction,

$$|\psi_{\mathrm{RBM}}(\mathcal{W}(t))\rangle = \sum_{\{S\}} \Psi(S; \mathcal{W}(t)) \, |S\rangle \, . \tag{6}$$

In order to properly represent the system, the RBM must be trained to find the ideal set of network parameters, $\mathcal{W} = \{a, b, W_{ij}\}$. The training process differs based on the use of the network. We are considering two uses: ground state determination and dynamics calculations.

To find the ground state of the system the network parameters are determined based on the system's Hamiltonian. This optimization trains the RBM and is performed by minimizing a variational energy expectation value,

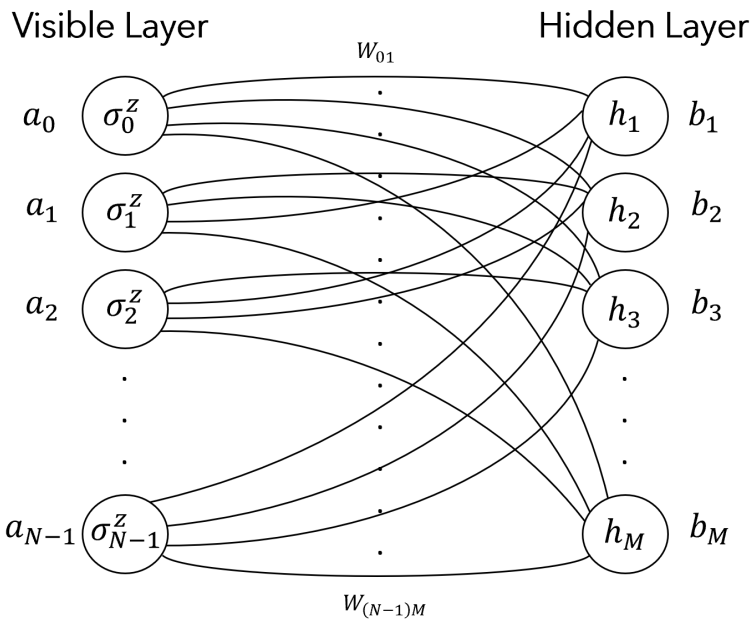

FIG. 1. A restricted Boltzmann machine architecture applied to the central-spin model. The visible layer, shown on the left, is composed of visible spin variables $S = \{\sigma_0^z, \sigma_1^z, ..., \sigma_{N-1}^z\}$. The hidden layer, on the right, holds the hidden variables $h_1, h_2, ..., h_M$.

$$E(\mathcal{W}) = \frac{\langle \psi_{\mathrm{RBM}} | \, H \, | \psi_{\mathrm{RBM}} \rangle}{\langle \psi_{\mathrm{RBM}} | \psi_{\mathrm{RBM}} \rangle}. \tag{7}$$

This variational energy is used analogously to a cost function in standard machine learning applications. After training, a RBM gives an accurate reparametrization of a quantum state [1].

Alternatively, the RBM can be used to find the dynamics of the system by making the network parameters time-dependent, $\mathcal{W}(t)$. At each time step these parameters are trained by minimizing variational residuals defined as,

$$R(\dot{\mathcal{W}}(t)) = \mathrm{dist}(\partial_t \psi_{\mathrm{RBM}}, -iH\psi_{\mathrm{RBM}}). \tag{8}$$

The time derivative of $\mathcal{W}(t)$ is found at each step. Thus, by integration, the dynamics of the system can be found.

One many-body system, the central spin model, would benefit from an efficient numerical modeling approach. This model describes the interactions between N spins, one central spin and $N-1$ bath spins. These bath spins act as an environment and influence the dynamics of the central spin. This description is often used to model the hyperfine interaction between an electron spin in a quantum dot and surrounding nuclear spins [4]. A major issue when using electron spins for quantum

information processing is the decoherence of the state. This decoherence is introduced due to environmental influence. The hyperfine interaction of neighbouring spins must always be considered, even at low temperatures, thus requiring a description of the entire system [5]. We wish to apply this RBM parametrization to get an efficient numerical model of the dynamics of this central spin system [6]. In addition, the central-spin model can be mapped to the Bardeen-Cooper-Schrieffer model of superconductivity [7]. If more efficient modeling methods are found for the central spin problem, they could be additionally applied to problems in non-equilibrium superconductivity.

II. METHODS

A. Mathematical Formalism

The central-spin model is a system of N total spins, where the central spin is subject to an external field B and its interaction with the environmental spins is parameterized by A_k, a set of coupling constants. A model of this system is shown in Fig. 2. This model is represented with the following Hamiltonian,

$$H = BS_0^z + \sum_{k=1}^{N-1} A_k \mathbf{S}_0 \cdot \mathbf{S}_k, \tag{9}$$

where $S_0^z = \frac{1}{2}\sigma_0^z$ is the spin operator for the z-component of the central spin. The vector $\mathbf{S}_k$ is defined as $\mathbf{S}_k = (S_k^x, S_k^y, S_k^z)$ which acts on the k^{th} spin. A simple limit of this model ($A_k = A_0$ constant for all k) can be solved exactly. This analytical solution can then be used to verify results from numerical exact diagonalization, which in turn can be used to verify the results found later with machine learning. The Hamiltonian given in Eq. (9) can also be written in terms of raising and lowering operators, $S_k^+ = S_k^x + iS_k^y$ and $S_k^- = S_k^x - iS_k^y$, acting on the k^{th} spin,

$$H = BS_0^z + A_0 \sum_{k=1}^{N-1} [S_0^z S_k^z + \frac{1}{2}(S_0^+ S_k^- + S_0^- S_k^+)]. \tag{10}$$

1. Product Basis

The central-spin model Hamiltonian is most easily described in terms of N two-dimensional spin-$\frac{1}{2}$ Hilbert spaces that form a 2^N dimensional Hilbert space. The Hilbert space of each spin site is two-dimensional and the total Hilbert space is given by the tensor product of each spin Hilbert space,

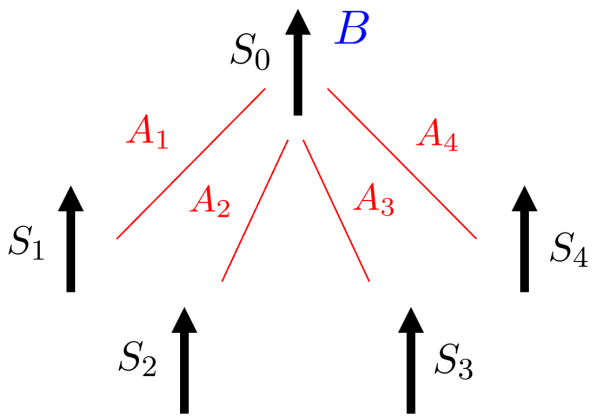

FIG. 2. Diagrammatic representation of the central spin model.

$$\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes ... \otimes \mathcal{H}_N. \tag{11}$$

The spin operator for the z-component of spin on the k^{th} spin site is defined,

$$S_k^z = \mathbb{1} \otimes ... \otimes S^z \otimes ... \otimes \mathbb{1}, \tag{12}$$

where the spin operator is in the k^{th} position of this product. The energy eigenvalues and eigenstates of this Hamiltonian, Eq. (9), can be found with brute-force numerical exact diagonalization. However, as the Hilbert space is 2^N dimensional this becomes exponentially computationally difficult as the number of spin sites increases.

2. Analytic Exact Diagonalization

Instead of considering the entire Hamiltonian, we can instead look at the basis defined in terms of eigenstates of two new operators $\mathbf{S}$ and $\mathbf{J}$. Here, $\mathbf{S} = \sum_{k=1}^{N-1} \mathbf{S}_k$ is the total spin of all bath sites and $\mathbf{J} = \mathbf{S}_0 + \mathbf{S}$ is the total spin of the system. Rewriting the Hamiltonian in terms of the raising and lowering operators, as shown in Eq. (10), it can be seen that $[H, \mathbf{S}^2] = 0$, $[H, S_0^2] = 0$ and $[H, J^z] = 0$, specifying a set of commuting operators. Thus we can define the state $|s, m_s, \sigma\rangle$ where

$\sigma = \pm\frac{1}{2} = \uparrow, \downarrow$ and the definition of s, m_s and m_j are given by the following relations:

$$\mathbf{S}^2 |s, m_s, \sigma\rangle = s(s+1) |s, m_s, \sigma\rangle , \tag{13}$$

$$S^z |s, m_s, \sigma\rangle = m_s |s, m_s, \sigma\rangle , \tag{14}$$

$$S_0^z |s, m_s, \sigma\rangle = \sigma |s, m_s, \sigma\rangle , \tag{15}$$

$$J^z |s, m_s, \sigma\rangle = (m_s + \sigma) |s, m_s, \sigma\rangle = m_j |s, m_s, \sigma\rangle . \tag{16}$$

Consider the two states $|s, m_s, \uparrow\rangle$ and $|s, m_s + 1, \downarrow\rangle$. As they each have the same number of spin sites up, they have the same eigenvalue of J^z, $m_j = m_s + \sigma$. J^z and H share eigenstates. Since there are two orthogonal states for each J^z, corresponding to $\sigma = \frac{1}{2}$ and $\sigma = -\frac{1}{2}$, the Hamiltonian splits into two-dimensional blocks. This can be seen when H is applied to one of these states, the result is a linear combination of the two states associated with the same eigenvalue of J^z.

$$H |s, m_s, \uparrow\rangle = (\frac{1}{2}B + \frac{1}{2}A_0 m_s) |s, m_s, \uparrow\rangle + \frac{1}{2}A_0\sqrt{s(s+1) - m_s(m_s+1)} |s, m_s + 1, \downarrow\rangle \tag{17}$$

$$= u(m_s) |s, m_s, \uparrow\rangle + g(m_s) |s, m_s + 1, \downarrow\rangle , \tag{18}$$

$$H |s, m_s + 1, \downarrow\rangle = \frac{1}{2}A_0\sqrt{s(s+1) - m_s(m_s+1)} |s, m_s, \uparrow\rangle \tag{19}$$

$$- (\frac{1}{2}B + \frac{1}{2}A_0(m_s + 1)) |s, m_s + 1, \downarrow\rangle$$

$$= g(m_s) |s, m_s, \uparrow\rangle - u(m_s + 1) |s, m_s + 1, \downarrow\rangle , \tag{20}$$

where $g(m_s) = \frac{1}{2}A_0\sqrt{s(s+1) - m_s(m_s+1)}$ and $u(m_s) = \frac{1}{2}B + \frac{1}{2}A_0 m_s$. As this 2D space is preserved under this Hamiltonian, an eigenstate of this system is given as a linear combination of these two states; $|\psi\rangle = c_{m_s\uparrow} |s, m_s, \uparrow\rangle + c_{m_s\downarrow} |s, m_s + 1, \downarrow\rangle$. We can find the values of $c_{m_s\uparrow}$ and $c_{m_s\downarrow}$ by diagonalizing the following representation of $H |\psi\rangle = E_{m_j} |\psi\rangle$;

$$\begin{bmatrix} u(m_s) & g(ms) \\ g(ms) & -u(m_s+1) \end{bmatrix} \begin{bmatrix} c_{m_s\uparrow} \\ c_{m_s\downarrow} \end{bmatrix} = E_{m_j} \begin{bmatrix} c_{m_s\uparrow} \\ c_{m_s\downarrow} \end{bmatrix} . \tag{21}$$

Since this matrix is 2×2 we will get two eigenstates. This doublet solution can be labeled as $|m_j, \pm\rangle$ since the two states $|s, m_s, \uparrow\rangle$ and $|s, m_s + 1, \downarrow\rangle$ have the same value of m_j.

We can expand this formulation by considering a basis of our entire system in terms of the states $|s, m_s, \sigma\rangle$. The allowed values of s, the total environmental spin, are related to the number of spins considered. For an N-spin system, $s = \frac{N-1}{2}, \frac{N-1}{2} - 1, ...0$. To further decrease complexity, consider only $s = \frac{N-1}{2}$, known as the highest weight states. In the basis $|s, s, \uparrow\rangle$, $|s, s-1, \uparrow\rangle$, $|s, s, \downarrow\rangle$,

$|s, s-2, \uparrow\rangle$, $|s, s-1, \downarrow\rangle$, ... , $|s, -s, \downarrow\rangle$ the Hamiltonian of the system is block diagonal, with 2×2 blocks identical to that shown in Eq. (21).

$$[H] = \begin{bmatrix} E_{m_j=s+\frac{1}{2}} & & & \\ & \Big[2 \times 2\Big]_{m_j} & 0 & \\ & & \ddots & \\ & 0 & & \Big[2 \times 2\Big]_{m_j} \\ & & & & E_{m_j=-s-\frac{1}{2}} \end{bmatrix}. \tag{22}$$

The blocks in this representation each have a different value of m_j. Each eigenstate can be labelled with m_j and there will be two orthornormal eigenstates for each m_j, thus $|m_j, \pm\rangle$ is a justified labeling of these eigenstates. The maximal and minimal values of m_j, ($m_j = \pm(s + \frac{1}{2})$) correspond the the exact eigenstates of H, rather than doublets. These are the states with all spins up or all down. This formalization allows the Hilbert space to be decomposed in terms of these 2×2 subspaces which can be easily diagonalized, in comparision to diagonalizing the entire 2^N-dimensional space for the product basis.

B. Analytic Verifier

In order to test the validity of the restricted Boltzmann machine, solutions will be compared to those found with brute-force numerical exact diagonalizion. This numerical technique works for any set of coupling constants A_k, but its time and memory cost scales exponentially with the number of spins. It is essential that this numerical method functions correctly, therefore it is additionally compared to the analytic results found by computing the doublet eigenstates $|m_j, \pm\rangle$. This comparison is performed by looking at the time evolution of the expectation value of the S^z operator of the central spin for different initial states. The time evolution of this operator is found through evolving the initial state given by the following relation,

$$\langle S_0^z(t)\rangle = \langle\psi(t)|\, S_0^z\, |\psi(t)\rangle = \sum_{\alpha\beta} e^{-i(E_\alpha - E_\beta)t}\, \langle\psi(0)|\beta\rangle\, \langle\beta|\, S_0^z\, |\alpha\rangle\, \langle\alpha|\psi(0)\rangle\,. \tag{23}$$

Here α and β label eigenstates with associated eigenvalues E_α and E_β, respectively. The state $|\psi(t)\rangle$ is the time-evolved state at time t. These comparisons have been completed for a simplified Hamiltonian with $B = 0$ and $A_0 = 1$. The expectation value of the central spin $\langle S_0^z\rangle$ is compared for $N = 2$ and $N = 3$. The results are shown in Figs. 3 and Figs. 4. The initial state is given in the basis of S_k^z-eigenstates, where the central spin is given by the first spin listed. These comparisons justify the use the numerical exact diagonalization procedure as a verification method for the RBM.

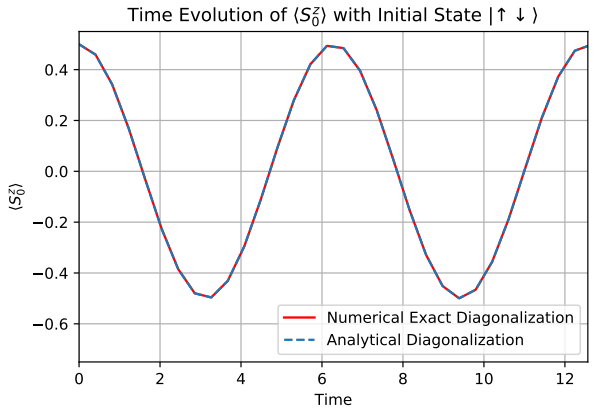

FIG. 3. Evolution of the expectation value of the S_0^z operator acting on the central spin in a two-spin system in units of $(\frac{1}{A_0})$. The initial state used was $|\uparrow\downarrow\rangle$.

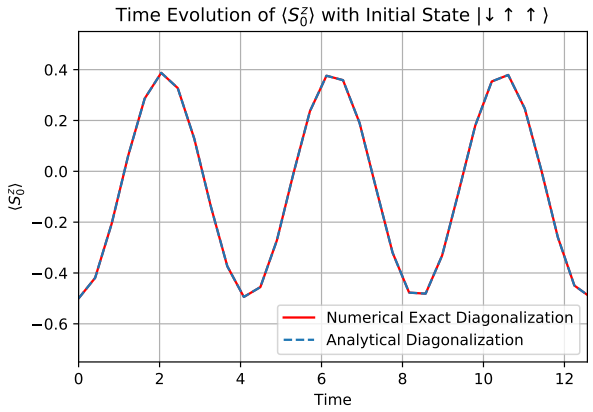

FIG. 4. Evolution of the expectation value of the S_0^z operator acting on the central spin in a three-spin system in units of $(\frac{1}{A_0})$. The initial state used was $|\downarrow\uparrow\uparrow\rangle$.

C. Restricted Boltzmann Machine

The implementation of the restricted Boltzmann machine can be broadly described in three sections: the network definition, the network training, and the network validation. A RBM is used in this context to reparametrize a quantum state through the basis state coefficients outlined in Eq. (5). The network is defined as the state described by these coefficients. Network training is the procedure of finding the optimal RBM parameters to represent a given state. For our purposes, we trained the model to the ground state of the central-spin system. The variational energy, shown in Eq. (7), is minimized through a conjugate gradient algorithm. This minimization method was chosen initially due to ease of implementation with a large number of RBM parameters, using the SciPy optimize library [8]. Other methods were later explored and compared to the results found with the conjugate gradient algorithm in section C of the results.

A conjugate gradient algorithm describes the minimum-energy solution as a linear combination of a set of conjugate vectors. Conjugate vectors, u and v, are defined relative to the Hessian ($\mathcal{H}$) of the variational energy, through the following relation

$$u^T \mathcal{H} v = 0. \tag{24}$$

As each RBM parameter is complex, this algorithm is implemented for both the real and imaginary component of each parameter. These components are initialized with a random value distributed

between zero and one. The gradient of the variational energy is found at this point, defining the first conjugate vector. From there, the RBM parameters are slightly modified and the gradient is then recalculated with these new parameters. This modification is defined such that a second conjugate vector can be found with the new gradient information. This is continued until a basis of these conjugate vectors is found. From there a minimal solution can be easily calculated as it can be expressed as a linear relation to the Hessian of the variational energy,

$$Hx^* = b. \tag{25}$$

with x^* the specific RBM parameters at the minimum and b the vector depending on the function being minimized, in this case the variational energy. As x^* can be written as a linear combination of the conjugate basis, $x^* = \sum_i \alpha_i d_i$, Eq. (25) can be rearranged to find the coefficients α

$$d_k^T b = \sum_i \alpha_i d_k^T H d_i \tag{26}$$

$$= \alpha_k d_k^T H d_k, \tag{27}$$

$$\alpha_k = \frac{d_k^T b}{d_k^T H d_k}. \tag{28}$$

With the conjugate basis found with this procedure, and these coefficients the RBM parameters $\mathcal{W} = \{a, b, W_{ij}\}$ that give the minimum variational energy can be found. Overall this method provides a gradient minimization algorithm that is more efficient than a gradient descent [9].

It is then necessary to quantify the accuracy of these results by comparing them to the verifier which implements exact diagonalization. Two error functions are defined, one for comparing the ground state energy and the other to compare the ground state. The energy error is defined to be the difference between the ground state energy found using the RBM and that found with exact diagonalization,

$$\epsilon_E = |E_{\text{RBM}} - E_{\text{ED}}|. \tag{29}$$

The state error is given in terms of the overlap between the ground state found through numerical exact diagonalization, $|\psi_{\text{ED}}\rangle$, and the ground state found from the RBM, $|\psi_{\text{RBM}}\rangle$,

$$\epsilon_\psi = 1 - |\langle \psi_{\text{RBM}} | \psi_{\text{ED}} \rangle|^2. \tag{30}$$

These two comparisons provide insights into the success of the training of the RBM.

Additional features added to this implementation include a runtime calculator and a ground state degeneracy check. The runtime tracker allows comparison between the RBM and the exact

diagonalization method, as well as between different minimization techniques. As the model is trained to the ground state, if this state is degenerate the variational energy does not have a global minimum and the model cannot be properly trained. Thus, a catch is implemented to warn of a degenerate ground state for the given Hamiltonian parameters.

III. RESULTS

A. Model Validation

The results outlined are for a two-spin system with $B = \pi$ and $A_k = 1$ for all k. Choosing an irrational value of $B = \pi$ reduces the number of energy degeneracies of the system. A restricted Boltzmann machine, with three hidden nodes, was trained using the conjugate gradient algorithm to find the ground state of this system. The success of this minimization is seen in figure Fig. 5. Both the ground state error, shown in Fig. 5 a), and energy error, shown in Fig. 5 b), decrease with each iteration of the minimization algorithm. Both errors fall close to 10^{-5} showing a RBM can accurately represent this many-body state.

In some instances, the training algorithm terminated with a much higher error, ranging from 0.07 up to 0.57. A histogram was performed over 50 runs to understand the cause and frequencies of these higher errors. Fig. 6 shows a histogram of the ground state error, energy error and runtime over these 50 runs. The energy error takes on discrete values, all occurring with significant frequency. This indicates the minimization may be consistently finding a local minimum which restricts its ability to vary. The frequency with which these high errors are realized decreases with the number of hidden nodes of the network. This effect can be seen in Fig. 6, where results are compared for $M = 2$ and $M = 3$. By increasing the number of hidden nodes of the system, the expressivity of the model increases. With an larger number of hidden nodes, there is an increase in the dimension of the space in which the variational energy is minimized. This higher dimensionality may decreases the likelihood of errors due to local minima. As seen in Fig. 6, when $M = 2$, 18% of the runs have an energy error over 0.01. When $M = 3$, this rate decreases to 8%.

B. Minimization Procedure

In addition to the conjugate gradient algorithms, two other minimization techniques were explored. The Nelder-Mead method, which does not used a gradient calculation for the direction of descent, but compares values of the variational energy. It creates a simplex of $n+1$ vertices, with n the dimension

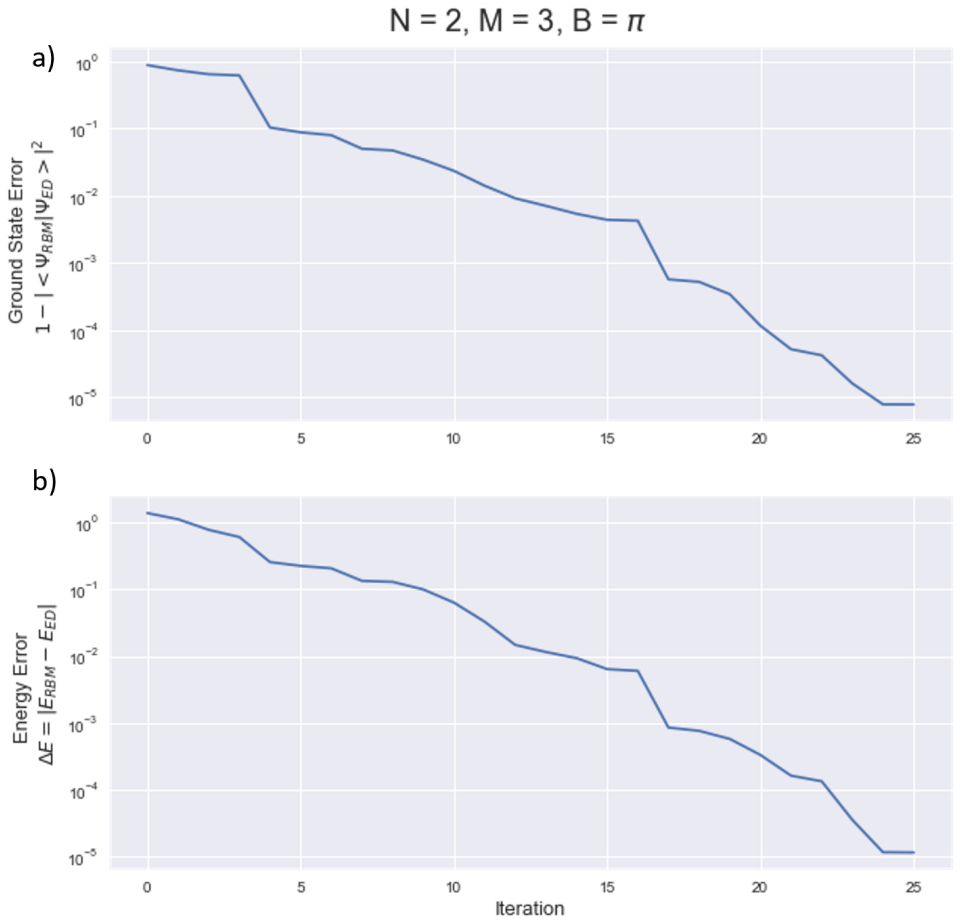

FIG. 5. Ground state and energy error as a function of the iteration of the conjugate gradient minimization. The ground state determination was performed on a system with $N = 2, M = 3, B = \pi$ and $A_k = 1$ for all k.

of the space. The value of the variation energy is sampled at each vertices and a transformation is done to the simplex, depending on these sampled values. These transformations vary the positioning of the simplex in the space, until one of the vertices samples the global minimum. The additional method tested is the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm. This quasi-Newton method depends on approximations of the Hessian matrix through gradient evaluations. The results of these three algorithms are compared in Fig. 7 for a system with $N = 2, M = 1, B = \pi$ and $A_k = 1$ for all k. Interestingly, similar error patterns occur with all three minimization techniques. In some runs errors are on the scale of 10^{-5} where as others around 0.08. In Fig. 7 b) the energy error is compared for each run where the same RBM initialization was used. This demonstrates these local minima are a result of the RBM definition, not due to an inherent weakness of the conjugate gradient method.

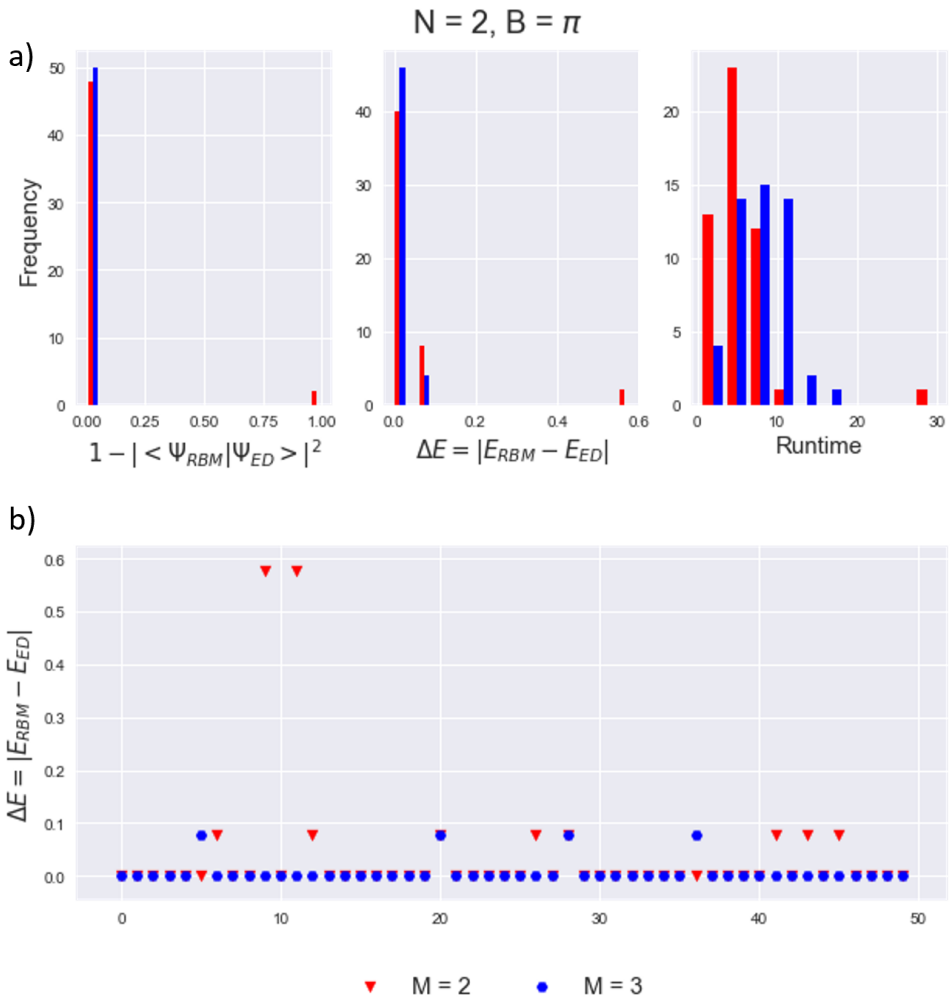

FIG. 6. Comparative histogram of ground state error, energy error and runtime shown in a). Individual sample results are shown in b). The two systems compared have $N = 2, B = \pi$ and $A_k = 1$ for all k, they differ by the number of hidden nodes M.

IV. FUTURE DIRECTIONS

A RBM has been shown to accurately represent the ground state of the central spin system, however, this process is not yet consistent. The source of these local minima needs to be further explored, so that a reliable training method can be found. This may include implementing other minimization algorithms such as stochastic gradient descent or stochastic reconfiguration [1]. From there, the natural extension is to explore larger systems and those with differing coupling constants. With a consistent minimization protocol, the modeling properties of the RBM can be probed. For example, the runtime can be compared for increasing spin numbers to see how the RBM reparametrization scales with system size. The main goal is to use this RBM to model the

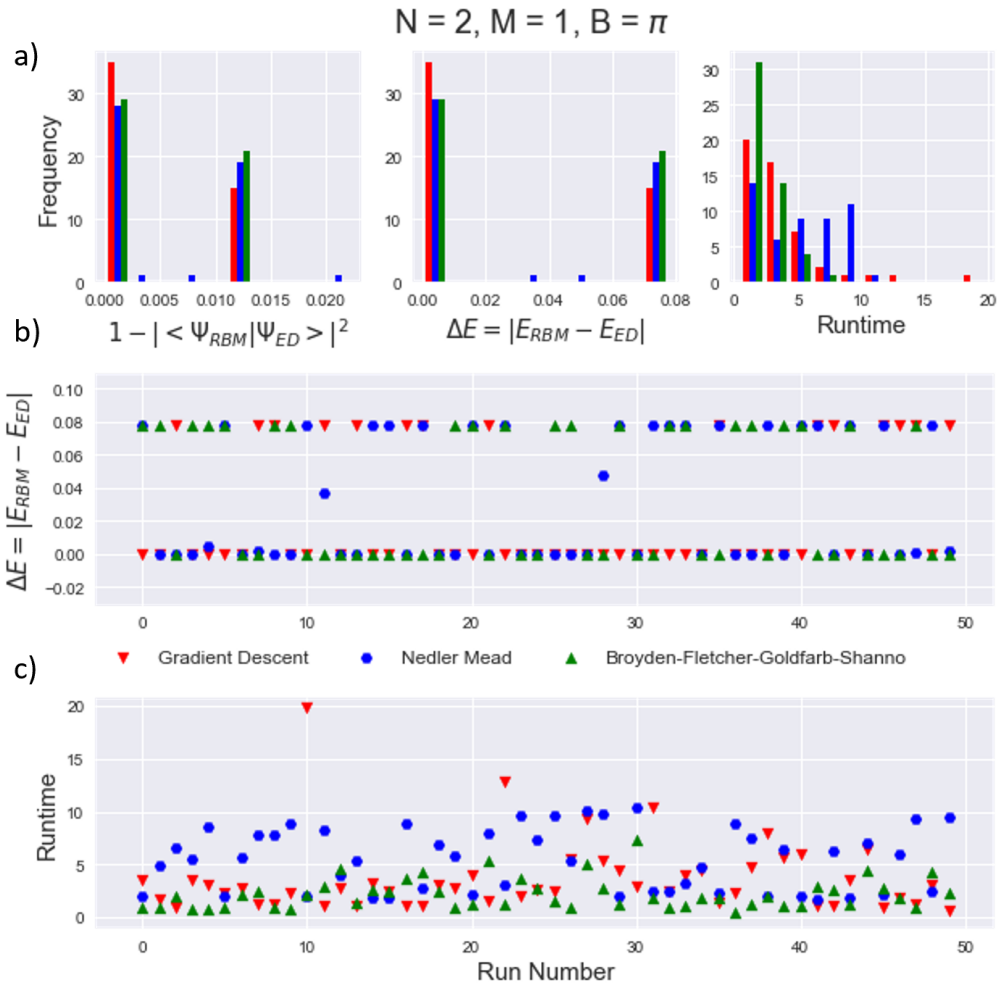

FIG. 7. A histogram of ground state error, energy error and runtime for various minimization algorithms is shown in a). Individual sample results of energy error and runtime are shown in b) and c) respectively. Ran for system with $N = 2, M = 1, B = \pi$ and $A_k = 1$ for all k.

dynamics of the system, a technique shown successful for similar spin models [1]. This would require the adequate implementation of Eq. (8). In addition, we hope to explore the applicability to nonequilibrium superconductivity. Overall, the aim is to explore the capacity of this machine-learning technique in modelling the central spin system.

I plan to pursue these goals over the summer semester.

[1] G. Carleo and M. Troyer, Solving the quantum many-body problem with artificial neural networks, Science **355**, 602 (2017).

[2] M. J. Hartmann and G. Carleo, Neural-network approach to dissipative quantum many-body dynamics, Phys. Rev. Lett. **122**, 250502 (2019).

[3] G. Torlai and R. G. Melko, Latent space purification via neural density operators, Phys. Rev. Lett. **120**, 240503 (2018).

[4] J. Schliemann, A. Khaetskii, and D. Loss, Electron spin dynamics in quantum dots and related nanostructures due to hyperfine interaction with nuclei, Journal of Physics: Condensed Matter **15**, R1809 (2003).

[5] W. A. Coish and D. Loss, Hyperfine interaction in a quantum dot: Non-markovian electron spin dynamics, Phys. Rev. B **70**, 195340 (2004).

[6] M. Bortz and J. Stolze, Exact dynamics in the inhomogeneous central-spin model, Phys. Rev. B **76**, 014304 (2007).

[7] Yuzbashyan, Altshuler, Kuznetsov, and Enolskii, Solution for the dynamics of the BCS and central spin problems, Journal of Physics A: Mathematical and General **38**, 7831 (2005).

[8] P. Virtanen *et al.*, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods **17**, 261 (2020).

[9] J. R. Shewchuk *et al.*, *An introduction to the conjugate gradient method without the agonizing pain* (Carnegie-Mellon University. Department of Computer Science, 1994).