Modeling a Quantum Many Body Problem Using a Restricted Boltzmann Machine

Dalen, Bendik Steinsvåg and Pinjusic, Daniel

(Dated: Monday 7th June, 2021)

Presented is a study on the viability of using restricted Boltzmann machines to emulate the wave function of a quantum mechanical many body problem, specifically for two electrons in harmonic oscillator trap. We tested both with and without a repulsive interaction between the electrons and found that the RBM performed excellent in the non-interactive case, where we got almost exactly the same energy as the expected 2 a.u. In the interactive case the RBM performed well, producing energies around 3.2 a.u, with the expected result being 3 a.u.

I. INTRODUCTION

Restricted Boltzmann machines (RBM) have a wide range of applications, for example in statistical tasks such as classification [8] and feature learning [3]. G. Carleo and M. Troyer (2017)[2] recently introduced the idea of using a restricted Boltzmann machine to also represent a quantum mechanical wave function. In their paper they applied a RBM to the quantum mechanical spin lattice systems of the Ising model and Heisenberg model, with high accuracy.

The aim of this paper is to further test the viability of using RBMs to model quantum mechanical systems, by using one to emulate a system of two interacting electrons confined to move in a harmonic oscillator trap. More specifically, we are interested in estimating an upper bound for the ground state energy of the system using a specific trial wave function.

This project builds on a paper by B.S. Dalen and A.G. Eriksen (2021)[4], where they model a hard sphere Bose gas using a variational Monte Carlo (VMC) estimation to emulate the wave function. We use a similar approach here, but replace the VMC approach with a RBM and also only model two electrons. Some familiarity with that paper is recommended before reading on, as some of the concepts introduced there will not be reintroduced here.

In this paper, we start by explaining some of the theory that relates to this project, such as Bolzmann machines, the Hamiltonian and the chosen trial wave function, however, we refer the reader to the aforementioned article[4] for the theory on the Metropolis algorithm, importance sampling, etc. We then go on to describe our implementations of said theory before presenting our results, which we subsequently discuss in the discussion section before finally making our concluding remarks in the conclusion section.

All code used in solving the project can be found at the following Github repository: `https://github.com/bendkok/Project_2-FYS4411` [1]

II. THEORY AND ALGORITHMS

A. Physical Problem

The physical problem we will be studying here is a system of two interacting electrons confined to move in a harmonic oscillator trap. We will only use two electrons so that we don't have to worry about the Pauli exclusion principle. However the problem can be generalised to instead study a n-body boson problem, such as the Bose gas studied in the paper by B.S. Dalen and A.G. Eriksen (2021)[4]. We will also only model the problem in two dimensions.

The Hamiltonian of such a system takes the form

$$\hat{H} = \sum_{i=1}^{N} \left(-\frac{1}{2}\nabla_i^2 + \frac{1}{2}\omega^2 r_i^2 \right) + \sum_{i<j} \frac{1}{r_{ij}}. \tag{1}$$

This can be separated into the standard harmonic oscillator part and the repulsive interaction part,

$$\hat{H}_0 = \sum_{i=1}^{N} \left(-\frac{1}{2}\nabla_i^2 + \frac{1}{2}\omega^2 r_i^2 \right), \tag{2}$$

$$\hat{H}_1 = \sum_{i<j} \frac{1}{r_{ij}}, \tag{3}$$

where $r_{ij} = |\boldsymbol{r}_1 - \boldsymbol{r}_2|$ and $r_i = \sqrt{{r_{i_x}}^2 + {r_{i_y}}^2}$.

If we set the frequency to $\hbar\omega = 1$ we would expect to get an energy of 2 atomic units (a.u.) for the non-interactive case. According to Taut, M. (1993) [9], if we include the interaction part as well the energy should be 3 a.u.

We will get back to how we will model the wave function later.

B. Boltzmann Machine

Boltzmann machines (BM) are a type of generative stochastic recurrent neural networks where the state of any node in the network is dependent on every other node, i.e all nodes are connected to each other. The relative importance or strength of the connections between the nodes is determined by the weights in the network. The distribution of nodes in a standard BM network is a set of observable nodes $\hat{x}$ and a set of unknown hidden nodes $\hat{h}$.

Due to their generative properties, BMs are often used for sampling and generating probability distributions, such as drawing new data based on the training data. They also have the property of being stackable, such that one can use a sample generated from one BM as an input to another BM. These properties make BMs popular in the field of deep learning, however due to their design and many connections, they are difficult to train and computationally inefficient. To remedy this fault the so called restricted

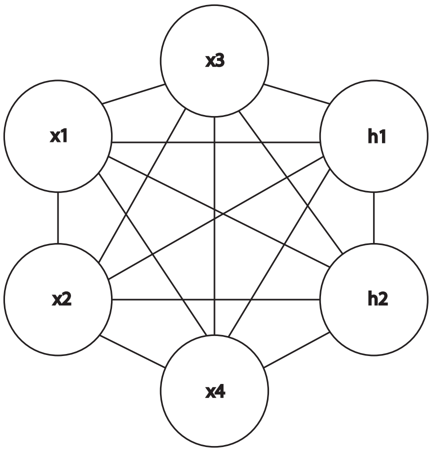

FIG. 1. Example of a Boltzmann machine with 4 observable nodes and 2 hidden nodes.

Boltzmann machine was introduced, which removes the connections between nodes of the same type and introduces a more standard network layout with a clearly defined observable and hidden layer, thereby increasing efficiency. In this project we will use such a restricted Boltzmann machine to generate a wave function that accurately describes the movement of our two-electron system.

1. Joint Probability Distribution

The restricted Boltzmann machine is described by the joint probability distribution

$$F_{rbm}(\boldsymbol{x},\boldsymbol{h}) = \frac{1}{Z} e^{-\frac{1}{T_0}E(\boldsymbol{x},\boldsymbol{h})} \tag{4}$$

where Z is the partition function

$$Z = \int\int e^{-\frac{1}{T_0}E(\boldsymbol{x},\boldsymbol{h})} d\boldsymbol{x} d\boldsymbol{h}. \tag{5}$$

$E(\boldsymbol{x},\boldsymbol{h})$ is the energy function and T_0 is a parameter commonly set equal to 1. The goal of an RBM is to optimize certain parameters in the system. These parameters are found in the so called energy function $E(\boldsymbol{x},\boldsymbol{h})$, which we will get back to. It's important to not confuse this with the local energy of the wave function. Through optimizing for these parameters we can adjust our energy function to best fit our problem. The energy function itself is a function that gives the energy of a given pair of vectors $(\boldsymbol{x},\boldsymbol{h})$. The lower the energy of a configuration is, the more likely it is, which means we are looking to minimize the energy function.

2. Gaussian-Binary

RBMs can be separated into a few types, the most common being Binary-Binary and Gaussian-Binary RBMs. The differences between them lies in the type of nodes we choose for the layers and the implementation of the energy function.

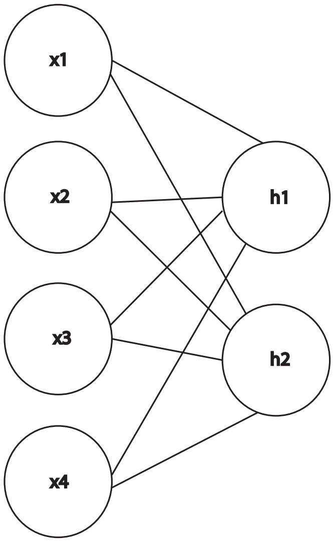

FIG. 2. Example of a restricted Boltzmann machine with 4 nodes in the visible layer and 2 nodes in the hidden layer. There is no connection between nodes in the same layer.

In a binary-binary RBM all the nodes have binary output values, like 0 and 1 or -1 to 1, while the Gaussian-binary type uses continuous values for the visible nodes and binary values for the hidden nodes. Due to the nature of the problem at hand it is natural to use continuous values for the visible nodes to represent position coordinates, whilst the hidden nodes can take values of 0 or 1. Thus, we choose to use a Gaussian-Binary RBM. We define the energy function as

$$\begin{aligned} E(\boldsymbol{x},\boldsymbol{h}) &= \sum_i^M \frac{(x_i - a_i)^2}{2\sigma_i^2} - \sum_j^N b_j h_j - \sum_{i,j}^{M,N} \frac{x_i w_{ij} h_j}{\sigma_i^2} \\ &= ||\frac{\boldsymbol{x}-\boldsymbol{a}}{2\boldsymbol{\sigma}}||^2 - \boldsymbol{b}^T\boldsymbol{h} - (\frac{\boldsymbol{x}}{\boldsymbol{\sigma}^2})^T \boldsymbol{W}\boldsymbol{h}, \end{aligned} \tag{6}$$

where we have the variance $\sigma_i^2 = \sigma^2$, the visible bias $\boldsymbol{a}$, the hidden bias $\boldsymbol{b}$ and the interaction weights $\boldsymbol{W}$. $\boldsymbol{a}$ and $\boldsymbol{b}$ are vectors with lengths M and N corresponding with the number of nodes in the visible and hidden layers respectively, and the weights $\boldsymbol{W}$ is a matrix of size $M \times N$.

C. Wave Function

A wave function (WF) is a probability amplitude that depends on observable quantities, such as positional coordinates, from which the probabilities of possible outcomes can be derived. From the RBM we have the joint probability distribution $F_{rbm}(\mathbf{x},\mathbf{h})$. From this we can define the marginal distribution

$$F_{rbm}(\mathbf{x}) = \sum_{\mathbf{h}} F_{rbm}(\mathbf{x},\mathbf{h}) = \frac{1}{Z}\sum_{\mathbf{h}} e^{-E(\mathbf{X},\mathbf{h})}, \tag{7}$$

which has similar properties to a wave function. We can therefore define our wave function as

$$\Psi(\mathbf{x}) = F_{rbm}(\mathbf{x}). \tag{8}$$

We name the expression in (8) as the neural network quantum state (NQS). Note that although the marginal distribution is similar to a wave function, they do differ in certain ways, mainly due to wave functions being probability amplitudes whilst the marginal distribution is a probability distribution. Inserting for the energy function in (8) gives us the final expression for the wave function

$$\begin{aligned}\Psi(\mathbf{x}) &= \frac{1}{Z}\sum_{\mathbf{h}} e^{-E(\mathbf{X},\mathbf{h})} \\ &= \frac{1}{Z}\sum_{\{h_j\}} e^{-\sum_i^M \frac{(X_i-a_i)^2}{2\sigma^2}+\sum_j^N b_j h_j+\sum_{i,j}^{M,N}\frac{X_i w_{ij} h_j}{\sigma^2}} \\ &= \frac{1}{Z} e^{-\sum_i^M \frac{(x_i-a_i)^2}{2\sigma^2}} \prod_j^N \left(1+e^{b_j+\sum_i^M \frac{x_i w_{ij}}{\sigma^2}}\right). \end{aligned}\tag{9}$$

1. Local Energy

With this wave function and our Hamiltonian the local energy of the system becomes

$$\begin{aligned}E_L =& \frac{1}{\Psi}\hat{\mathbf{H}}\Psi \\ =& \frac{1}{2}\sum_{m=1}^{M}\left(-\left(\frac{\partial}{\partial x_m}\ln\Psi\right)^2-\frac{\partial^2}{\partial x_m^2}\ln\Psi+\omega^2 r_m^2\right) \\ &+\sum_{i<j}\frac{1}{r_{ij}}\end{aligned}\tag{10}$$

Where we have that

$$\frac{\partial \ln\Psi}{\partial x_m} = -\frac{x_m-a_m}{\sigma^2}+\frac{1}{\sigma^2}\sum_j^N \frac{\omega_{mj}}{1+e^{-b_j-\sum_i^M \frac{X_i w_{ij}}{\sigma^2}}}, \tag{11}$$

and

$$\frac{\partial^2 \ln\Psi}{\partial x_m^2} = -\frac{1}{\sigma^2}+\frac{1}{\sigma^4}\sum_j^N \omega_{mj}^2 \frac{e^{-b_j-\sum_i^M \frac{X_i w_{ij}}{\sigma^2}}}{\left(1+e^{-b_j-\sum_i^M \frac{X_i w_{ij}}{\sigma^2}}\right)^2}. \tag{12}$$

See section A 2 for a more thorough derivation.

2. Interaction

Normally when two electrons approach each other the repulsive force between them get stronger. You might have noticed that this isn't modeled in our wave function, in fact, the only interaction between the particles is in the Hamiltonian. An interaction in the WF could be modelled using say a Jastrow factor of the form $e^{-\beta r_{ij}}$, however we won't do that here. Such a Jastrow factor has a relatively small impact on the local energy, and we want to test how well we can emulate the system without it.

Henceforth when we refer to the non-interactive or the interactive case we are referring to whether or not the repulsive interaction part of the Hamiltonian is included, unless otherwise specified.

D. Minimization

1. Nesterov Accelerated Gradient

The minimization method we will be using to optimize the variables in the RBM is called Nesterov accelerated gradient. It has the formula

$$v_t = \gamma v_{t-1}+\eta\nabla_\theta J\left(\theta-\gamma v_{t-1}\right) \tag{13}$$

$$\theta = \theta - v_t. \tag{14}$$

η is known as the the learning rate, γ is a variable between 0 and 1, θ represents all the variables we are optimizing, and J is the cost function. Nesterov accelerated gradient can be seen as a continuation of the momentum method, where one adds a guess for the next step[7].

2. Cost function

Since we are solving an optimisation problem using an gradient descent method, we use a cost function as the measure of our accuracy. For a quantum mechanical problem such as this we don't necessarily have a set of training data to define our cost function from, however, from the variational principle we know that minimizing the expectation value of the energy of a trial wave function should yield the ground state wave function. As the energy describing the system is the local energy E_L, we get the cost function

$$\begin{aligned}G_i = \frac{\partial\langle E_L\rangle}{\partial\alpha_i} &= 2\left(\left\langle E_L\frac{1}{\Psi}\frac{\partial\Psi}{\partial\alpha_i}\right\rangle-\langle E_L\rangle\left\langle\frac{1}{\Psi}\frac{\partial\Psi}{\partial\alpha_i}\right\rangle\right) \\ &= 2\left(\left\langle E_L\frac{\partial\ln\Psi}{\partial\alpha_i}\right\rangle-\langle E_L\rangle\left\langle\frac{\partial\ln\Psi}{\partial\alpha_i}\right\rangle\right),\end{aligned}\tag{15}$$

where we have

$$\alpha_i = a_1,\dots,a_M,b_1,\dots,b_N,w_{11},\dots,w_{MN}, \tag{16}$$

$$\begin{aligned}\ln\Psi = -\ln Z-\sum_i^M\frac{(X_i-a_i)^2}{2\sigma^2} \\ +\sum_j^N\ln\left(1+e^{b_j+\sum_i^M\frac{X_i w_{ij}}{\sigma^2}}\right)\end{aligned}\tag{17}$$

See section A 3 for a more thorough derivation.

III. METHOD

The code we wrote was based upon a program written by M. Hjorth-Jensen. [5]

The RBM variables were initially randomised using a Gaussian distribution, with the average being 0 and the mean being 0.5. To calculate and evaluate new positions for the particles we used importance sampling, with the time step $\Delta t = 0.05$.

Our other variables were:

- Number of particles = 2
- Number of dimensions = 2
- Number of hidden nodes = 2
- Equilibration fraction = 0.1
- Total number of iterations = 100
- $\eta = 0.05$
- $\gamma = 0.9$

We ran the code first with no interaction and then with interaction. To test the errors of our result we used the blocking method. Our blocking code was based upon a blocking python script, also written by Hjort-Jensen. [6]. Finally, we test the program for different values of η.

IV. RESULTS

A plot of the final energy each iteration with and without interaction can be seen in figure 4 and 3 respectively.

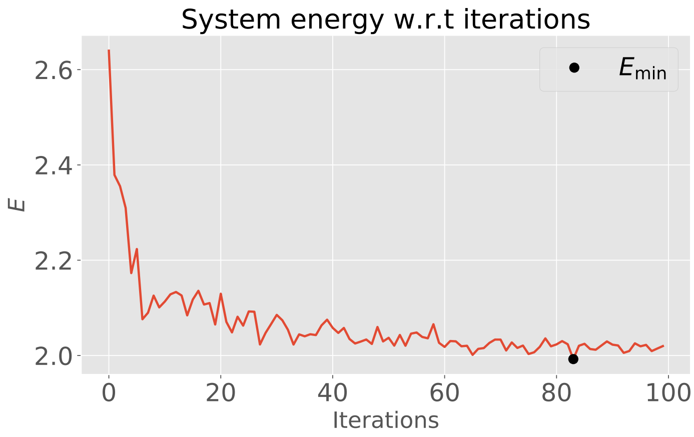

FIG. 3. The final energy each iteration with no interaction. The lowest energy was $E_{\min} = 1.99$ at iteration 83.

A. Blocking Results

A plot of the mean energy each iteration with and without interaction can be seen in figure 6 and 5 respectively.

A plot of the standard deviation of the energy each iteration with and without interaction can be seen in figure 8 and 7 respectively.

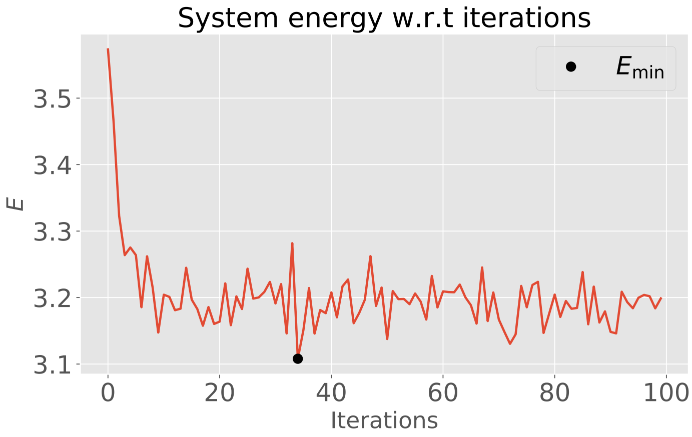

FIG. 4. The final energy each iteration with interaction. The lowest energy was $E_{\min} = 3.10$ at iteration 34.

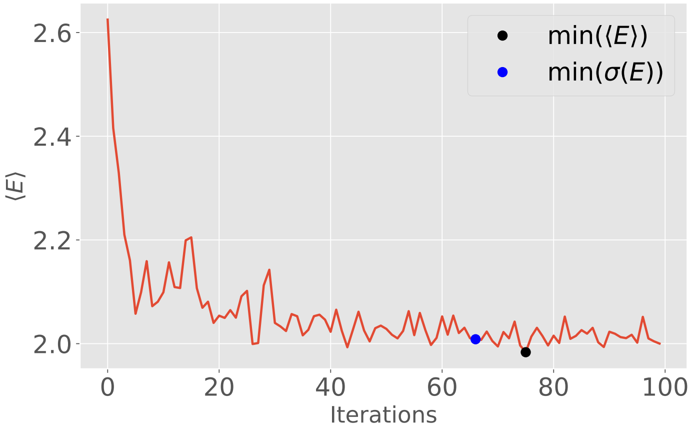

FIG. 5. The mean energy each iteration with no interaction. The lowest mean energy was $\min(\langle E\rangle) = 1.98$ at iteration 75, which had a $\sigma = 3.55 \cdot 10^{-3}$.

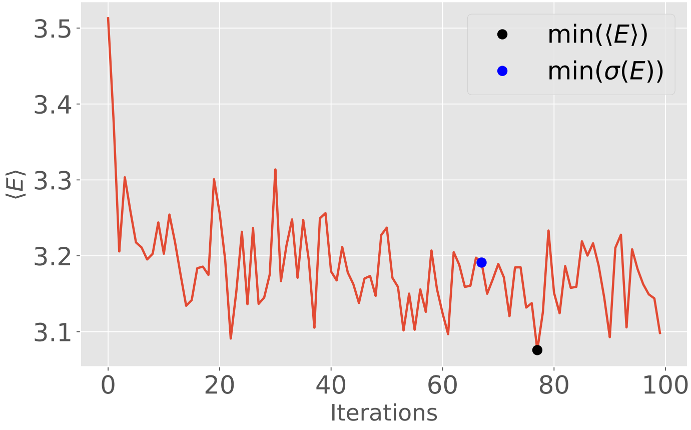

FIG. 6. The mean energy each iteration with interaction. The lowest mean energy was $\min(\langle E\rangle) = 3.07$ at iteration 77, which had a $\sigma = 2.08 \cdot 10^{-2}$.

In figure 9 we show the performance of the Nestrov ac-

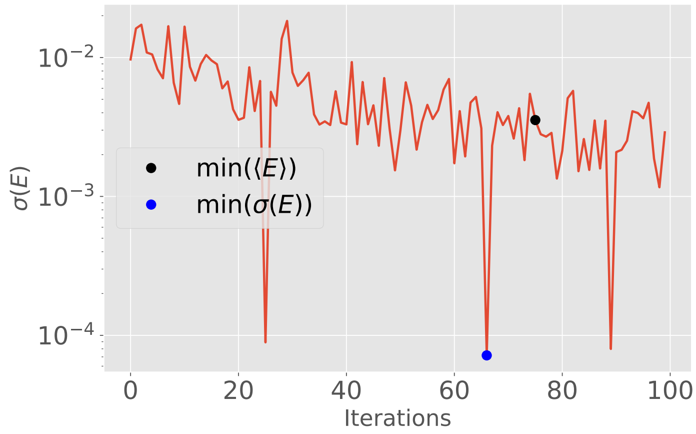

FIG. 7. The standard deviation of the energy each iteration with no interaction. The lowest σ was $\sigma = 7.17 \cdot 10^{-5}$ at iteration 66, which had a mean energy of $\langle E\rangle = 2.00$.

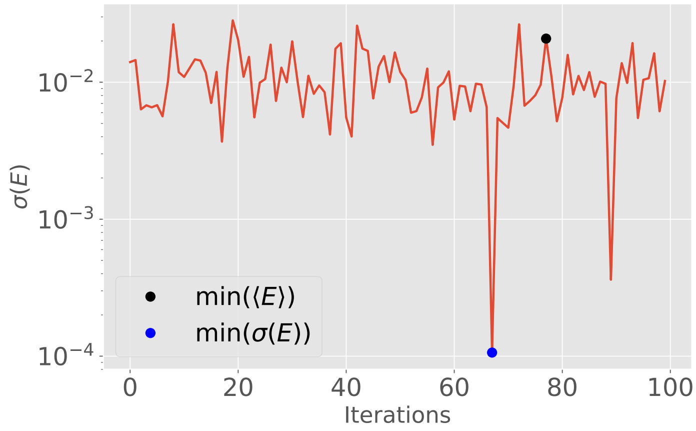

FIG. 8. The standard deviation of the energy each iteration with interaction. The lowest σ was $\sigma = 1.06 \cdot 10^{-4}$ at iteration 67, which had a mean energy of $\langle E\rangle = 3.19$.

celerated gradient descent method using variable values for the learning rate η.

V. DISCUSSION

Looking at figure 3 we see that the energy when there's no interaction starts at over 2.6 a.u, but then rapidly decreases and starts oscillating at a little over 2 a.u. This is a good sign, since our expected result was 2 a.u. The oscillation at the end is minor, and can be chucked up to statistical fluctuations. We can conclude that the RBM is an almost perfect emulation of the wave function in the non-interactive case.

From figure 4 we see that the energy in the interactive case displays a similar pattern. It starts at about 3.6 a.u, then drops to oscillate at around 3.2 a.u. At its lowest it becomes 3.10 a.u. This is close to our expected result of 3.0 a.u, especially considering our approximation of not including an interaction term in the wave function. We can conclude that the RBM is a pretty good emulation of the wave function in the interactive case.

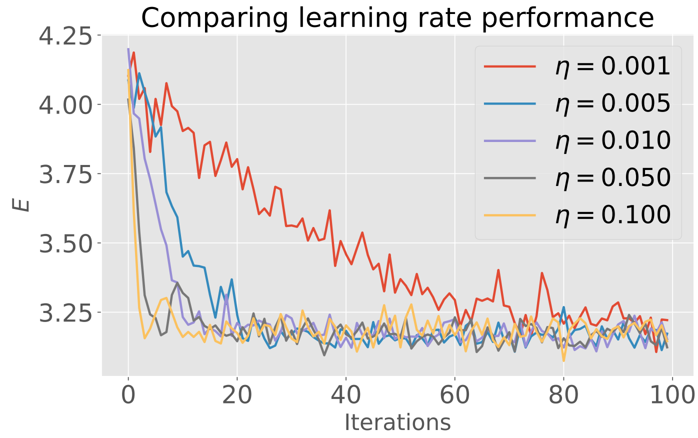

FIG. 9. Comparing the performance of the gradient descent method for differing values of η

A. Blocking

In figure 5 we see more or less the same thing as in figure 3, only the values are slightly smaller. The standard deviation for the lowest mean energy is also rather low, at $3.55{\cdot}10^{-3}$, which is a good sign.

In figure 6 we see that the mean energy is more or less the same as the final energy in figure 4. The only noticeable difference is that the mean energy fluctuates a little more. The STD of the lowest mean value is almost an order of magnitude larger than for the non-interacting case, but it is still relatively small.

From figure 7 we see that the standard deviation of the energy with no interaction starts at a little over 10^{-2} a.u, and then decreases slightly to fluctuate somewhere between 10^{-3} and 10^{-2} a.u. The only exceptions is three separate iterations where is suddenly drops to less than $1 \cdot 10^{-4}$ a.u.

From figure 8 we see that the STD generally fluctuates around 10^{-2} a.u. It also has two places where the STD suddenly drops a lot, though one drops far more than the other.

It is not entirely clear why the standard deviation falls so heavily during some iterations. We do not believe that this is a case of any wrong doing on our part, but surmise that this could simply be an issue of the random seeds that were chosen for the system weights and Metropolis step. We therefore consider these results as outliers and chose to ignore them when considering the results as a whole.

B. Learning Rate

In figure 9, we see how the choice of learning rate η did not have too much of an impact on results, as long as one lets the system run for enough iterations. Even the worst performing learning rate, which was $\eta = 0.001$, eventually converges on a similar value as the other learning rates that were tested, although more than 60 iterations were needed. The data does suggest that a learning rate of $\eta = 0.1$ or $\eta = 0.05$ are suitable choices for the problem at hand. They are small enough that we get convergences early on in the simulation with predictable results. Even though it was not tested for, it would be interesting to test how the Nesterov accelerated gradient descent method would perform had we tested both bigger and smaller values for the learning rate. Our results suggest that a $\eta = 0.001$ is too small, converging late, but values bigger than $\eta = 0.1$ might have been possible to use, without losing the predictability of the results.

VI. CONCLUSION

We have seen that a restricted Boltzman machine works as an excellent emulation of the wave function of two non-interacting electrons. It produces an end energy that is almost exactly the same as the analytical result of 2 a.u.

Furthermore, we have seen that a RBM is also a good emulation in the interactive case, even without a Jastrow factor or a similar term to represent the repulsive interaction in the wave function. We get results which rarely are more than 0.2 a.u. away from the expected result of 3 a.u. We thus see great potential in further use of RBMs to approximate the wave functions of many body systems with high accuracy.

We found through the blocking method that the produced results do not suffer from any large statistical errors, with standard deviations of $\sigma = 3.55 \cdot 10^{-3}$ and $\sigma = 2.08 \cdot 10^{-2}$ for the corresponding lowest mean energies $\min(\langle E \rangle) = 1.98$ and $\min(\langle E \rangle) = 3.07$ for the non-interactive case and the interactive case respectively.

Finally, we discover that a learning rate of $\eta = 0.05$ is just right for the problems discussed in this report, giving mostly predictable results with fast convergence, reaching the point of convergence at around 20 iterations.

Appendix A: Some math

1. Base Formulas

Following are some of the basic formulas we will use here. Hamiltonian:

$$\hat{H} = \sum_{i=1}^{N} \left(-\frac{1}{2}\nabla_i^2 + \frac{1}{2}\omega^2 r_i^2 \right) + \sum_{i<j} \frac{1}{r_{ij}} \tag{A1}$$

Wave function:

$$\begin{aligned}\Psi(\mathbf{X}) &= \frac{1}{Z} \sum_{\{h_j\}} e^{-E(\mathbf{X},\mathbf{h})} \\ &= \frac{1}{Z} \sum_{\{h_j\}} e^{-\sum_i^M \frac{(X_i - a_i)^2}{2\sigma^2} + \sum_j^N b_j h_j + \sum_{i,j}^{M,N} \frac{X_i w_{ij} h_j}{\sigma^2}} \\ &= \frac{1}{Z} e^{-\sum_i^M \frac{(x_i - a_i)^2}{2\sigma^2}} \prod_j^N \left(1 + e^{b_j + \sum_i^M \frac{x_i w_{ij}}{\sigma^2}} \right)\end{aligned} \tag{A2}$$

2. Local Energy

We are to find an expression for the local energy $E_L = \frac{1}{\Psi}\mathbf{H}\Psi$.

$$\begin{aligned}E_L &= \frac{1}{\Psi} \left[\sum_{i=1}^{N} \left(-\frac{1}{2}\nabla_i^2 + \frac{1}{2}\omega^2 r_i^2 \right) + \sum_{i<j} \frac{1}{r_{ij}} \right] \Psi \\ &= \frac{1}{\Psi} \left(-\frac{1}{2} \sum_{i=1}^{N} \nabla_i^2 \right) \Psi + \sum_{i=1}^{N} \frac{1}{2}\omega^2 r_i^2 + \sum_{i<j} \frac{1}{r_{ij}} \\ &= \chi + \sum_{i=1}^{N} \frac{1}{2}\omega^2 r_i^2 + \sum_{i<j} \frac{1}{r_{ij}}\end{aligned} \tag{A3}$$

$$\begin{aligned}\chi &= -\frac{1}{2}\frac{1}{\Psi} \sum_{i=1}^{N} \sum_{i=d}^{D} \frac{\partial^2}{\partial x_{ip}^2} \Psi \\ &= -\frac{1}{2} \sum_{i=1}^{N} \sum_{i=d}^{D} \left(\left(\frac{\partial}{\partial x_{ip}} \ln \Psi \right)^2 + \frac{\partial^2}{\partial x_{ip}^2} \ln \Psi \right) \\ &= -\frac{1}{2} \sum_{m=1}^{M} \left(\left(\frac{\partial}{\partial x_m} \ln \Psi \right)^2 + \frac{\partial^2}{\partial x_m^2} \ln \Psi \right)\end{aligned} \tag{A4}$$

$$\begin{aligned}E_L =& \frac{1}{2} \sum_{m=1}^{M} \left(-\left(\frac{\partial}{\partial x_m} \ln \Psi \right)^2 - \frac{\partial^2}{\partial x_m^2} \ln \Psi + \omega^2 r_m^2 \right) \\ &+ \sum_{i<j} \frac{1}{r_{ij}}\end{aligned} \tag{A5}$$

Where we have that:

$$\ln\Psi = -\ln Z - \sum_i^M \frac{(X_i - a_i)^2}{2\sigma^2} + \sum_j^N \ln\left(1 + e^{b_j + \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}\right) \tag{A6}$$

$$\begin{aligned}\frac{\partial \ln\Psi}{\partial x_m} &= -\frac{x_m - a_m}{\sigma^2} + \sum_j^N \frac{e^{b_j + \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}}{1 + e^{b_j + \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}} \frac{\omega_{mj}}{\sigma^2} \\ &= -\frac{x_m - a_m}{\sigma^2} + \frac{1}{\sigma^2} \sum_j^N \frac{\omega_{mj}}{1 + e^{-b_j - \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}}\end{aligned} \tag{A7}$$

$$\begin{aligned}\frac{\partial^2 \ln\Psi}{\partial x_m^2} &= -\frac{1}{\sigma^2} + \frac{1}{\sigma^2} \sum_j^N \omega_{mj} \frac{e^{-b_j - \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}}{\left(1 + e^{-b_j - \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}\right)^2} \frac{\omega_{mj}}{\sigma^2} \\ &= -\frac{1}{\sigma^2} + \frac{1}{\sigma^4} \sum_j^N \omega_{mj}^2 \frac{e^{-b_j - \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}}{\left(1 + e^{-b_j - \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}\right)^2}\end{aligned} \tag{A8}$$

3. Gradient

To find an expression for the gradient of the local energy, equation 15, we need to find the derivative of the logarithm of the wave function with respect to the variational parameters of the RBM,

$$\frac{\partial \ln\Psi}{\partial \alpha_i}, \tag{A9}$$

where

$$\alpha_i = a_1, \ldots, a_M, b_1, \ldots, b_N, w_{11}, \ldots, w_{MN}, \tag{A10}$$

and

$$\ln\Psi = -\ln Z - \sum_i^M \frac{(X_i - a_i)^2}{2\sigma^2} + \sum_j^N \ln\left(1 + e^{b_j + \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}\right).$$

The derivative of this will be different depending on which type of parameter we are using:

$$\frac{\partial \ln\Psi}{\partial a_i} = -\sum_i^M \frac{X_i - a_i}{\sigma^2} \tag{A11}$$

$$\begin{aligned}\frac{\partial \ln\Psi}{\partial b_i} &= \frac{1}{1 + e^{b_j + \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}} e^{b_j + \sum_i^M \frac{X_i w_{ij}}{\sigma^2}} \\ &= \frac{1}{1 + e^{-b_j - \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}}\end{aligned} \tag{A12}$$

$$\begin{aligned}\frac{\partial \ln\Psi}{\partial w_{ij}} &= \frac{1}{1 + e^{b_j + \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}} e^{b_j + \sum_i^M \frac{X_i w_{ij}}{\sigma^2}} \frac{X_i}{\sigma^2} \\ &= \frac{X_i}{\sigma^2\left(1 + e^{-b_j - \sum_i^M \frac{X_i w_{ij}}{\sigma^2}}\right)}\end{aligned} \tag{A13}$$

[1] Project 2 fys4411. URL: https://github.com/bendkok/Project_2-FYS4411.

[2] Giuseppe Carleo and Matthias Troyer. Solving the quantum many-body problem with artificial neural networks. *Science*, 355(6325):602–606, 2017. URL: https://science.sciencemag.org/content/355/6325/602, doi:10.1126/science.aag2302.

[3] Adam Coates, Honglak Lee, and Andrew Y. Ng. An analysis of single-layer networks in unsupervised feature learning. 2011. URL: http://proceedings.mlr.press/v15/coates11a/coates11a.pdf.

[4] Bendik Steinsvåg Dalen and Annika Gjolme Eriksen. Monte carlo estimation of ground state energy for a bose gas. *Self Published*, pages 1–15, April 2017. URL: https://github.com/ageriksen/variational-monte-carlo-fys4411/blob/master/FYS4411___Project_1.pdf.

[5] Morten Hjorth-Jensen. 5.7. python version for the two non-interacting particles. URL: http://compphysics.github.io/ComputationalPhysics2/doc/LectureNotes/_build/html/boltzmannmachines.html#python-version-for-the-two-non-interacting-particles.

[6] Morten Hjorth-Jensen. block.py. URL: https://github.com/CompPhysics/ComputationalPhysics2/blob/gh-pages/doc/Programs/BoltzmannMachines/VMC/python/block.py.

[7] Johannes Kuhn. Adaptive learning rate method, Jan 2017. URL: https://wiki.tum.de/x/l7RnAQ.

[8] Y. Larochelle, H. Bengio. Classification using discriminative restricted boltzmann machines. 2008. URL: https://dl.acm.org/doi/10.1145/1390156.1390224, doi:10.1145/1390156.1390224.

[9] M. Taut. Two electrons in an external oscillator potential: Particular analytic solutions of a coulomb correlation problem. *Phys. Rev. A*, 48:3561–3566, Nov 1993. URL: https://link.aps.org/doi/10.1103/PhysRevA.48.3561, doi:10.1103/PhysRevA.48.3561.