FYS4411 - Project 2

Espen Kristian Wulff Wold

Abstract

In a previous project [1] we built a Markov Chain Monte Carlo numerical integration routine to solve the many-dimensional integrals needed to simulate interacting particles trapped in harmonic potentials.

In this project, we expand upon that previous work by decoupling the MCMC numerical integration from the modelling of the trial wave function. This allows us to change the wave function model without changing the MCMC routine. We use this freedom to investigate how well a wave function model based on a Restricted Botzmann Machine approximates the ground state by optimisation of the variational parameters using gradients produced by the MCMC numerical integration.

Our results reproduce analytical results for the non-interacting case, but consistently produce too high energies for the interacting case. This despite the fact that the one-body densities seems to indicate that the optimisation really is adjusting to the addition of a repulsive potential. We were unable to resolve this seeming inconsistency.

Note: the code for this project is available at https://github.com/EspenWold/ComputationalQuantumPhysics, and is needed to fully understand this report and be able to recreate its results. The code contains a readme file that explains how to run the code and reproduce the results.

1. Introduction

In the second project for this course, we will extend our work from project number 1 by investigating more flexible ways of encoding the trial wave function. In the previous project [1] we constructed a trial wave function that was parameterised by one or two parameters. This obviously limits the expressive range of the trial wave function, and we are extremely dependent on our guesses for the form of the wave function. Although we will never escape the need for us to bring some physics understanding to the modelling of the wave function, we will in this project investigave the opportunity of letting our trial wave functions be more richly parameterised and increasingly shaped by optimisation of machine learning routines.

Succeeding with more expressive and automatically adjusted trial wave functions will allow us to use the methods employed in projcet 1 in situations where we are less knowledgeable about the expected shape of the ground state wave function.

We will first develop useful analytical expressions for the physical system in question, and breifly explain the theory of the Restricted Boltzmann Machine (RBM). Then we shall go through how we expand upon the code from the previous project [1] to let our wave function be modelled by an RBM and how we proceed to optimise the variational parameters of said RBM. Lastly, we will present our results and discuss their interpretation before we summarise and offer our conclusions.

2. Theory

As we want to experiment with different ways of encoding the trial wave function, it will be beneficial to develop expressions for some of the elements we worked with in project 1 that are agnostic with regards to the details of the wave function. For our system, we will look at particles confined in a spherical harmonic oscillator potential with a repulsive interaction given by the Hamiltonian

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

Hence, we have for the local energy and quantum force

$$\begin{aligned}
E_L(\boldsymbol{r}) &= \frac{1}{\Psi_T}H\Psi_T \\
&= \frac{1}{2}\sum_{i=1}^{N}\left(\frac{-1}{\Psi_T}\nabla_i^2\Psi_T + \omega^2 r_i^2\right) + \sum_{i<j}\frac{1}{r_{ij}} \\
&= \frac{1}{2}\sum_{i=1}^{N}\omega^2 r_i^2 - \frac{1}{2}\sum_{i=1}^{N}\sum_{d=1}^{D}\frac{1}{\Psi_T}\frac{\partial^2\Psi_T}{\partial x_{id}^2} + \sum_{i<j}\frac{1}{r_{ij}} \qquad (2) \\
&= \frac{1}{2}\sum_{i=1}^{N}\omega^2 r_i^2 + \sum_{i<j}\frac{1}{r_{ij}} \\
&\quad - \frac{1}{2}\sum_{i=1}^{N}\sum_{d=1}^{D}\left(\frac{\partial^2}{\partial x_{id}^2}\ln\Psi_T + (\frac{\partial}{\partial x_{id}}\ln\Psi_T)^2\right)
\end{aligned}$$

$$
\begin{aligned}
F_i(\boldsymbol{r}) &= \frac{2}{\Psi_T}\nabla_i \Psi_T = \frac{2}{\Psi_T}(\frac{\partial \Psi_T}{\partial x_{i1}}, \frac{\partial \Psi_T}{\partial x_{i2}}, \frac{\partial \Psi_T}{\partial x_{i3}}) \\
&= 2(\frac{\partial \ln \Psi_T}{\partial x_{i1}}, \frac{\partial \ln \Psi_T}{\partial x_{i2}}, \frac{\partial \ln \Psi_T}{\partial x_{i3}})
\end{aligned}
\tag{3}
$$

where x_{id} denotes the d-dimension coordinate for particle i. For the sake of definiteness we have assumed three dimensions for the quantum force, though the generalisation to other dimensionalities should be obvious. In fact, we see that for all our coordinates (across all particles) the quantum force is simply twice the derivative of the log of the wave function with regards to that coordinate.

Since we want to minimise the expected value of the local energy, we'll want the gradient with regards to any parameters involved in the modelling of our trial wave function. Letting the vector $\boldsymbol{\alpha}$ represent all such parameters $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_M)$ we have (derived in project 1)

$$
\begin{aligned}
\nabla_\alpha \langle E[\boldsymbol{\alpha}] \rangle &= 2\Bigg[\langle E_L \frac{\nabla_\alpha \Psi_T}{\Psi_T}\rangle - \langle E_L \rangle \langle \frac{\nabla_\alpha \Psi_T}{\Psi_T}\rangle\Bigg] \\
&= 2\Bigg[\langle E_L \nabla_\alpha \ln \Psi_T \rangle - \langle E_L \rangle \langle \nabla_\alpha \ln \Psi_T \rangle\Bigg]
\end{aligned}
\tag{4}
$$

or for any specific parameter

$$
\big(\nabla_\alpha \langle E[\boldsymbol{\alpha}] \rangle\big)_m = 2\Bigg[\langle E_L \frac{\partial \ln \Psi_T}{\partial \alpha_m}\rangle - \langle E_L \rangle \langle \frac{\partial \ln \Psi_T}{\partial \alpha_m}\rangle\Bigg] \tag{5}
$$

In sum, we see that the only components we would have to change if we want to try a new model for the wave function are all these derivatives of the log of the wave function with regards to all coordinates and variational parameters. Hence, we could in principle build a general solution for any wave function model which only needs to be supplied with oracle functions for the wave function itself and these various log-derivatives.

One possible model for the wave function is a Restricted Boltzmann Machine (RBM) which is a way to model a physics-inspired probability distribution.

$$
F_{\text{RBM}}(\boldsymbol{X}, \boldsymbol{H}) = \frac{1}{Z} e^{-\frac{1}{T_0} E(\boldsymbol{X}, \boldsymbol{H})} \tag{6}
$$

Z is a normalisation constant produced by integrating out the variables and T_0 is a 'temperature'-variable simply set to one for our purposes. $E(\boldsymbol{X}, \boldsymbol{H})$ is called the energy function, as an anlaogy to statistical mechanics (and hence not at all connected to the energy of the system we are simulating). We choose to use a Gaussian-Binary RBM, for which the energy is

$$
E(\boldsymbol{X}, \boldsymbol{H}) = \frac{||\boldsymbol{X} - \boldsymbol{a}||^2}{2\sigma^2} - \boldsymbol{b}^T \boldsymbol{H} - \frac{\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{H}}{\sigma^2} \tag{7}
$$

We call the vector $\boldsymbol{X}$ (of length M) the visible nodes, $\boldsymbol{H}$ (of length N) the hidden nodes and $\boldsymbol{W}$ is the M by N matrix holding the weights for the connection of every visible node to every hidden node. We are free to chose the number of hidden nodes N, but the number of coordinates is given by the system we are simulating

$$
M = \text{number of particles} \times \text{number of dimensions}
$$

$\boldsymbol{a}$ and $\boldsymbol{b}$ are, repectively, the visible bias terms (length M) and the hidden bias terms (length N).

Since we are only interesten in variation with $\boldsymbol{X}$ which will represent the coordinates in our system, we'll sum over the two possible values - 1 and 0 - for the hidden nodes.

$$
\begin{aligned}
F_{\text{RBM}}(\boldsymbol{X}) &= \sum_{\{h_j\}} \frac{1}{Z} e^{-\frac{||\boldsymbol{X}-\boldsymbol{a}||^2}{2\sigma^2} + \sum_{j^N} b_j h_j + \sum_{i,j}^{M,N} \frac{X_i w_{ij} h_j}{\sigma^2}} \\
&= \frac{1}{Z} e^{-\frac{||\boldsymbol{X}-\boldsymbol{a}||^2}{2\sigma^2}} \prod_j^N \left(1 + e^{b_j + \frac{(\boldsymbol{X}^T \boldsymbol{W})_j}{\sigma^2}}\right)
\end{aligned}
\tag{8}
$$

We can either let the Boltzmann machine model the wave function directly $\Psi_T(\boldsymbol{X}) = F_{\text{RBM}}(\boldsymbol{X})$, or be inspired by the interpretation of Boltzmann machines as modelling a probability distribution and let it model the square of the wave function $\Psi_T(\boldsymbol{X}) = \sqrt{F_{\text{RBM}}(\boldsymbol{X})}$. For all the log-derivatives we need to calculate, the difference in this choice is only a factor of two and we give the results only for the former case (and will divide by 2 to get the latter case):

$$
\begin{aligned}
\frac{\partial}{\partial a_m} \ln \Psi_T &= \frac{X_m - a_m}{\sigma^2} \\
\frac{\partial}{\partial b_n} \ln \Psi_T &= \frac{1}{e^{-b_n - \frac{(\boldsymbol{X}^T \boldsymbol{W})_n}{\sigma^2}} + 1} \\
\frac{\partial}{\partial w_{mn}} \ln \Psi_T &= \frac{X_m}{\sigma^2} \frac{1}{e^{-b_n - \frac{(\boldsymbol{X}^T \boldsymbol{W})_n}{\sigma^2}} + 1}
\end{aligned}
$$

$$
\frac{\partial}{\partial x_m} \ln \Psi_T = -\frac{X_m - a_m}{\sigma^2} + \frac{1}{\sigma^2} \sum_n^N \frac{w_{mn}}{e^{-b_n - \frac{(\boldsymbol{X}^T \boldsymbol{W})_n}{\sigma^2}} + 1}
$$

$$
\frac{\partial^2}{\partial x_m^2} \ln \Psi_T = -\frac{1}{\sigma^2} + \frac{1}{\sigma^4} \sum_n^N w_{mn}^2 \frac{e^{b_n + \frac{(\boldsymbol{X}^T \boldsymbol{W})_n}{\sigma^2}}}{\left(e^{b_n + \frac{(\boldsymbol{X}^T \boldsymbol{W})_n}{\sigma^2}} + 1\right)^2}
$$

3. Method

Inspired by the observation that all that differs between different models for our system is the evaluation of the wave function and its various log-derivatives, we implement a class to hold the model's internalt state (i.e. the current set of variational parameter values) and to act as an oracle to provide these evaluations given the current parameter state.

To achieve a clean interface between the wave function model class and our MCMC routine, we let the input and output of both coordinates and parameters be a flat list. This means we are letting the model class translate back and forth between any internal structure that the model has (e.g. building a matrix from the w_{ij}s for efficient computation). When we want to make changes to the wave function model, we can then do so without making any changes to the MCMC code.

Adapting our code from project 1, we set up our MCMC-scheme to calculate local energy (2) and quantum force (3) by using the spatial log-derivatives provided by the wave function model class. Our simple gradient descent routine is adapted to be updating the state of our model class based on the parameter gradient (4):

Algorithm 1 Wave function model gradient descent

```
Initiate wave function model ($WFM$) with N randomised parameters
Set a learning rate, $\eta$
Set a number of MCMC cycles, M
$energy \leftarrow 0$
Initiate zero-valued array $gradient[N]$
for iterations = 1:100 do
    $energy, gradient \leftarrow MCMC(WFM, M)$
    $WFM.parameters \leftarrow WFM.parameters - \eta \times gradient$
return $WFM$
```

The $MCMC$ function contains all of the machinery from project 1 with the changes to local energy and quantum force calculation described above.

4. Results

We would first like to test our solution for the non-interacting case. Our model is initiated with all parameters drawn from a normal distribuition with mean zero and standard deviation 0.001. We note that initiating all paramters to exactly zero reproduces the exact ground level wave function in the non-interacting case up to a normalisation factor that is not relevant to our use (as long as $\sigma^2 = 1$ if the RBM represents our wave function and $\sigma^2 = 0.5$ if the RBM represents the square of the wave function).

As per the task description pdf for this project, we expect a ground state energy of $\frac{ND}{2}$ in natural unit for the non-interacting system. Running our experiment with 1 particle in 2 dimension, we thus expect an energy of 1.

Our results look promising, producing energy estimates between 1.0001 and 0.9999 during the gradient descent to find the best parameters, and producing a value $E_{\text{RBM}} = 1.0 \pm 0.000001$ by analysing the results from the the main simulation of 2^{19} cycles with the blocking method. The one-body density plot produces reasonable-looking gaussians, see figure 1.

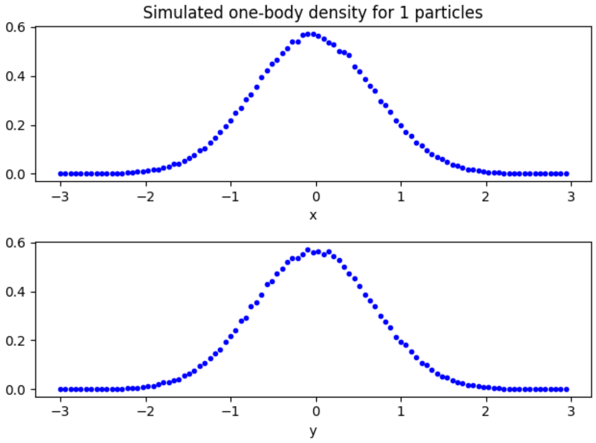

Figure 1: One-body density for one particle in two dimensions, from a run with RBM prameters initialised close to zero.

However, we note that our final variational parameters ended up with deviations from zero (i.e. the exact ground state) of similar magnitude to our initialisation. Hence, we have not really tested that our optimisation routine works.

Initialising our model with parameters further from the exact ground state wave function provides such a test. A run where we increased the standard deviation of the distribution from which we draw initial parameter values to 0.5 gave an initial energy estimate of 2.13 and then showed a steady (though not monotonic) decrease towards 1.0 over the course of a hundred iterations of gradient descent. The main simulation result was $E_{\text{RBM}} = 1.040178 \pm 0.000838$.

Increasing the number of hidden nodes from 2 to 10 seemed not to improve the model in any appreciable way. Another example run with 0.5 standard deviation for initial parameter values gave an initial energy of 14.7 which decreased towards 1.0 very similarly to the runs with 2 hidden nodes. The main simulation result was $E_{\text{RBM}} = 1.083707 \pm 0.000943$.

Lastly, as a test of our analytical expressions for the log-derivatives we implemented an automatic differentiation scheme where position and parameter derivatives are calculated by automatically differentiating the same function (for the log of the wave function) with respect to the different inputs. This proved to be vastly more time consuming but showed near perfect correspondence with the analytical expressions.

We now want to include the interaction characterised by the $\frac{1}{r_{ij}}$ term in our Hamiltonian. We will test our system with two particles in two dimensions (a configuration for which the project description pdf provides the expected energy of 3.0 in natural units). We initially use two hidden units.

We first try to initiate our variational paramters close to zero (standard deviation of 0.001), leaving the optimisation to discover what deviation from the non-interacting solution are needed. However, this leaves us with an energy varying between 3.2 and 3.3 during the optimisation

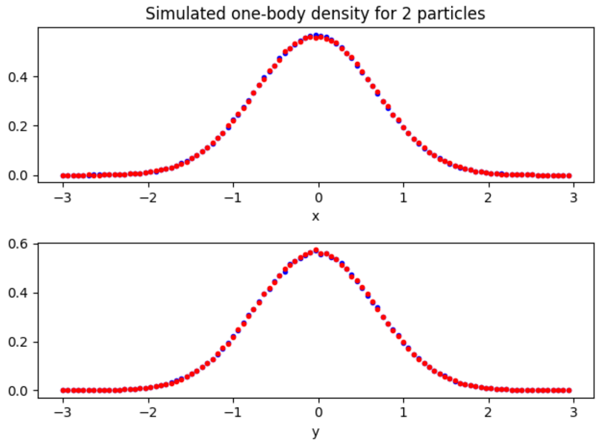

Figure 2: One-body densities with and without interactions for two particles in two dimensions. Non-interacting case plotted in red and interacting case plotted in blue. Both runs with RBM prameters initialised close to zero.

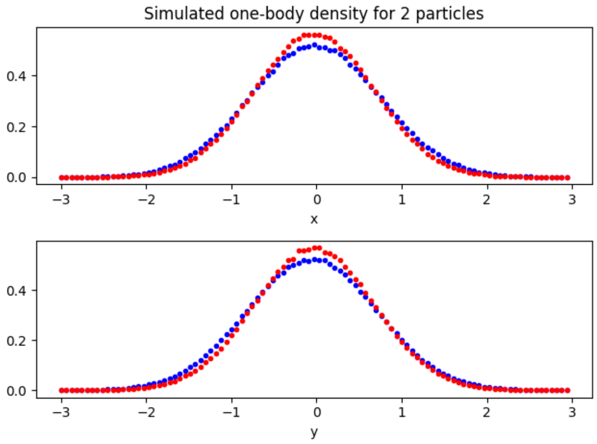

Figure 3: One-body densities with and without interactions for two particles in two dimensions. Non-interacting case plotted in red and interacting case plotted in blue. Both runs with no hidden nodes in the RBM and prameters initialised close to zero.

and never converging on 3.0. This behaviour persists when trying various learning rates. Initiating the variational parameters futher from zero causes high intial energies that dependably converge toward this range (3.2-3.3) but only rarely converge on values closer to 3.0. The closest runs converged on a range of 3.1-3.2.

We take this to indicate that our RBM has trouble modelling the correct wave function efficiently, or at least that the gradients in parameter space are not smooth enough for our optimisation routine to easily find the minimum.

We perform the experiment with a learning rate of 0.01 both with and without interactions and compare our results. The non-interacting case produced an energy $E_{\mathrm{RBM}} = 2.00$, while for the interacting case we got $E_{\mathrm{RBM}} = 3.25$. The one-body densities produced by the two simulations are plotted in figure 2.

The one-body densities are nearly identical, and we don't see any indication of the broadening we might expect from the addition of a repulsive interaction. Supecting that our model is making life difficult for the optimisation, we would like to test the effect of the interaction on a simpler model. Setting the number of hidden nodes to zero, we get a model of two identically shaped gaussians that are offset from the origin by the remaining bias parameters. This simplified system is very close to the non-interacting solution, and as we would expect, the biases are driven to zero by the optimisation in the non-interacting case. When we turn on the interaction however, we get an interesting result: the two gaussians end up drifting to opposite sides of the origin. (This result was not completely consistent, seemingly depending on the initial parameters. However, this behaviour displayed a convergence and diminishing of the gradient that we did not observe for the runs which did not exhibit this positioning of the gaussians.)

In one representative run, the biases (i.e. the position of the gaussians) were initialised as $x_1 = -0.543, y_1 = -0.493, x_2 = -0.303, y_2 = -0.713$ and over the course of optimisation changed to $x_1 = -0.289, y_1 = 0.254, x_2 = 0.303, y_2 = -0.311$. The resulting one-body density is plotted compared to an analogous non-interacting run (for which all biases were driven to zero) in figure 3.

This is clearly the kind of behaviour we would expect from a repulsive interaction, and can give us some confidence that our general setup is working. The energy in these cases does not converge any closer to 3.0 that with our whole RBM model, but that is not surprising: this severely constrained model should not be expected to be able to express something close to the correct wave function for the interacting case.

To make our model more fitting for the interacting case, we introduce another piece to the wave function $\Psi_{\mathrm{cor}} = \exp\left(\gamma \sum_{i<j} r_{ij}\right)$ that is specifically motivated by the interaction we have added. γ is a variational paramter controlling how much we 'reward' configurations where the particles are further apart. The whole wave function will then be

$$\Psi_T = F_{\mathrm{RBM}} \times \Psi_{\mathrm{cor}} \quad \text{or} \quad |\Psi_T|^2 = F_{\mathrm{RBM}} \times |\Psi_{\mathrm{cor}}|^2 \tag{9}$$

depending on which mode we are using. As for the rest of the wave function we need the log-derivatives with regards to both particle positions and variational parameters for this new piece of the wave function.

$$\begin{aligned} \nabla_k \ln \Psi_{\mathrm{cor}} &= \nabla_k \left(\gamma \sum_{i<j} r_{ij} \right) = \gamma \sum_{i \neq k} \nabla_k r_{ik} \\ &= \gamma \sum_{i \neq k} \frac{\boldsymbol{r}_k - \boldsymbol{r}_i}{r_{ik}} \end{aligned} \tag{10}$$

$$
\begin{aligned}
\nabla_k^2 \ln \Psi_{\text{cor}} &= \nabla_k \left(\gamma \sum_{i \neq k} \frac{\boldsymbol{r}_k - \boldsymbol{r}_i}{r_{ik}} \right) \\
&= \gamma \sum_{i \neq k} \left(\frac{1}{r_{ik}} - \frac{(\boldsymbol{r}_k - \boldsymbol{r}_i) \nabla_k r_{ik}}{r_{ik}^2} \right) \\
&= \gamma \sum_{i \neq k} \left(\frac{1}{r_{ik}} - \frac{(\boldsymbol{r}_k - \boldsymbol{r}_i)^2}{r_{ik}^3} \right) \\
&= \gamma \sum_{i \neq k} \left(\frac{1}{r_{ik}} - \frac{1}{r_{ik}} \right) = 0
\end{aligned}
\tag{11}
$$

$$
\frac{\partial}{\partial \gamma} \ln \Psi_{\text{cor}} = \frac{\partial}{\partial \gamma} \left(\gamma \sum_{i<j} r_{ij} \right) = \sum_{i<j} r_{ij} \tag{12}
$$

[When doing these calculations I realised that I made a sign error in Project 1 that gave me $\nabla_k \frac{(\boldsymbol{r}_k - \boldsymbol{r}_i)}{r_{ik}^2} = \frac{2}{r_{ik}} \neq 0$. I have not had time to re-do all the subsequent work to see how big an effect this mistake made on the final results in Project 1.]

Incorporating these results into the class we built to model the RBM, we again run our simulations for the case of two particles in two dimensions. We initiate γ by drawing from the same distribution as the other variational parameters (mean 0, std 0.001), which in the case of γ as well corrensponds to starting very close to the non-interacting solution.

As expected, in the non-interactive case all parameters are still driven close to zero and we get the usual energy of 2.0. With the interaction turned on the optimisation clearly siezes on the new Ψ_{cor} part of the wave function, consistently driving γ to a value of about 1. We also see a pronounced difference in the resulting one-body densities, as in the plot in figure 4 from a representative run.

Here the optimisation has very clearly found a way to accommodate the effects of a repulsive potential by a marked broadening of the wave function. Curiously, these runs with the new Ψ_{cor} term and interaction turned on produced energies of around 3.45, which is further from the expected 3.0 than what we got with a less expressive wave function. I am not sure how to interpret this, in light of the optimisation clearly 'wanting' to use the Ψ_{cor} term but at the same time increasing the energy.

This behaviour persisted when testing with large numbers of hidden nodes, varied learning rates, switching between the RBM representing Ψ_T and $|\Psi_T|^2$, and incresing the number of MCMC-cycles for each iteration of the gradient descent routine up to 50.000. As for the original model for the trial wave function we test our analytical expressions by comparing them to the gradients produced by the automatic differentiation scheme, and again the different methods result in close to identical gradients.

I am left confused by these results. Clearly there is something wrong since the energy minimisation routine drives the wave function to a higher energy when provided a more expressive model, especially since the non-interacting case shows us that the optimisation is easily capable of driving γ to zero and neglecting to use the new degree of freedom provided by the Ψ_{cor} term. On the other hand, our test of various possible sources of this error don't produce any obvious culprits. I especially think it is weird that we get such a consistent utilisation of the Ψ_{cor} term and a broadening of the wave function, both of which would make perfect sense if they were accompanied by a decrease in energy rather than an increase.

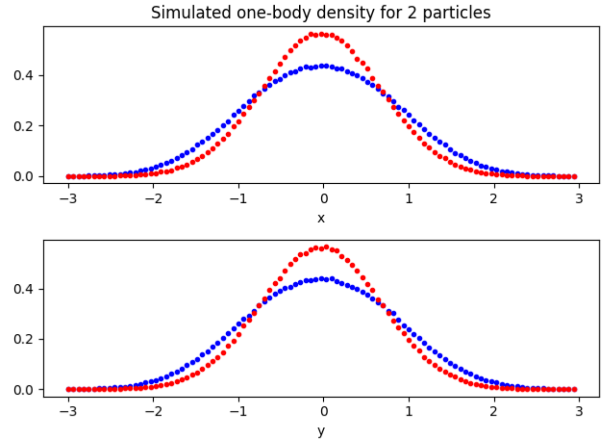

Figure 4: One-body densities with and without interactions for two particles in two dimensions. Non-interacting case plotted in red and interacting case plotted in blue. Both runs with a corrolation term added to the RBM, and prameters initialised close to zero.

Testing our system for three non-interacting particles in three dimensions and three hidden nodes we see the optimisation converge on the expected value 4.5 and setting all parameters close to zero. With interaction turned on, we end up with an energy of 7.8 while the optimisation drives most parameters close to zero except γ which is set to 0.65. The one-body density plotted is plotted in figure 5 and shows a similar broadening of the wave function with the introduction of the interaction that we saw in the two particle 2D case:

For 10 particles in 3D with 10 hidden nodes, we similarly get an energy of 15.0 and all paramterers driven close to zero in the non-interacting case and an energy of 51 and all parameters except $\gamma =$ 0.38 driven close to zero. As we might have expected, the one-body density (plotted in figure 6) shows more broadening with the interaction added than in the three particle case.

5. Conclusion

We have successfully extended our MCMC scheme from project 1 to allow more complex models for the trial wave function, and have shown that optimisations of such models can be achieved without the MCMC scheme itself needing to "know" the particulars of the model. We have also seen that the optimisation is only as good as the model it is given: it seems to struggle to make meaningful changes for the RMB-model in the interacting case, but once we introduce a term in the wave function to account for correlation this is very consistently siezed upon by the optimisation.

Despite these interesting resluts, we consistently saw energies for the interacting cases that were higher than what we expected from the analytical results alluded to in the project description. Even worse, with the optimisation choosing to use the added correlation term it actively drove the energy higher that it did without the correlation term present. We were unable to find the cause of this bahaviour.

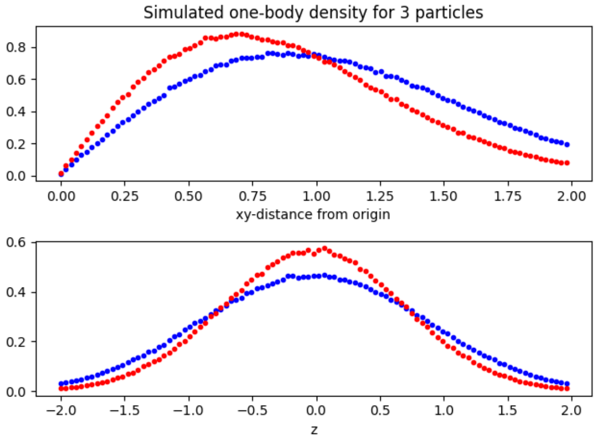

Figure 5: One-body densities with and without interactions for three particles in three dimensions. Non-interacting case plotted in red and interacting case plotted in blue. Both runs with a corrolation term added to the RBM, and prameters initialised close to zero.

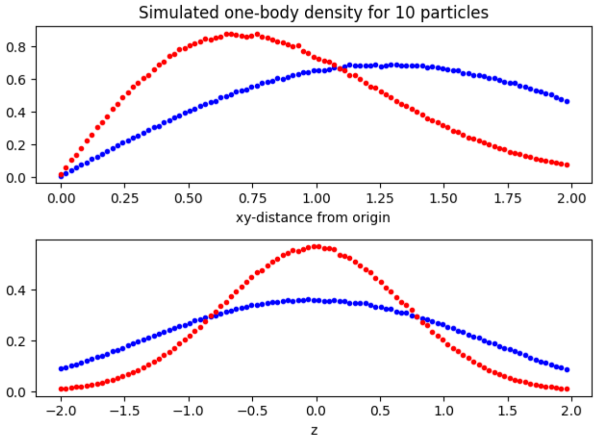

Figure 6: One-body densities with and without interactions for ten particles in three dimensions. Non-interacting case plotted in red and interacting case plotted in blue. Both runs with a corrolation term added to the RBM, and prameters initialised close to zero.

References

[1] Espen K. W. Wold, *Project 1 in the course FYS4411 at the University of Oslo*, PDF report available at https://github.com/EspenWold/ComputationalQuantumPhysics, (2022).