Variational Monte Carlo Simulations of Trapped Bosons

S. Håpnes A. Sexton
(Dated: Wednesday 25th March, 2020)

In the research presented, we study Bose-Einstein condensates of trapped bosons in an elliptical harmonic oscillator trap, by numerical approaches. We represent the systems with trial wave functions with two variational parameters. The aim is to identify and study the ground states of the systems, by performing variational Monte Carlo methods, namely the Metropolis algorithm and the Metropolis-Hastings algorithm. Lastly, we study the correlation factor in the trial wave function, by evaluating the one-body density with and without the Jastrow factor. For spherical Gaussian wave functions without correlation, we find excellent agreement with the theoretical ground state energies. We also show that the correlation, governed by the Jastrow factor, leads to a more spread out system.

I. INTRODUCTION

The theoretical study of Bose-Einstein condensates in alkali gases is a field of high interest in physics, much of it done in the framework of the Gross-Pitaevskii equation[1]. The validity of this description rests largely upon the systems having an average particle spacing which is much larger than the inter-particle interaction, which is largely governed by two-body collisions. In this project we aim to piece together the basic building blocks of a Variational Monte Carlo machinery capable of investigating such problems. We will see that there are several considerations to be made in using this method, with regards to finding certain analytic expressions wherever possible, but also in how we handle the error in our results. We also explore the added benefits of using the Fokker-Planck equation and importance sampling in our machinery, which has the effect of driving our systems towards its natural state, as opposed to standard metropolis sampling, which is of a more Brownian nature.

The use of numerical approaches is a very important field in the natural sciences. It allows us to simulate stochastic systems by the use of pseudo random numbers, and we can use algorithms to perform numerical experiments. This opens up a new world of opportunities for solving problems which could even be unsolvable analytically. In quantum mechanics, it is often impossible to find the exact solution to problems above a certain degree of complexity. When we are dealing with many particles and correlations amongst them, the only option is often either to perform overoptimistic simplifications or use numerical approaches. We have applied the numerical approaches, and it is hence important to remember that the results are based upon numerical limitations, such as the numerical precision and stability of the program, and the validity of the random number generators.

In the next section, we will present the theory on which the project is built upon. We will then go through the specific methods and algorithms used. Thereafter we will present our findings in the results section. The results will be analyzed thoroughly in the discussion section. Lastly we will summarize our work with the most important findings in the conclusion section.

The report and analyses are a cooperative effort by Alexander Harold Sexton and Simen Håpnes. The codes can be found at the following Github repository `https://github.com/alexahs/FYS4411/tree/master/project1`.

II. THEORY

There are some certain advantages in finding analytic expressions prior to implementing the code. Obtaining simpler expressions might lead to fewer floating point operations, which can significantly increase the performance of the code.

A. The Wave Function

An expression of high relevance is the trial wave function. We will adopt the following wave function for a system of N particles,

$$\Psi_T(\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N, a, \alpha, \beta) = \prod_i g(\alpha, \beta, \mathbf{r}_i) \prod_{j<k} f(a, |\mathbf{r}_j - \mathbf{r}_k|). \tag{1}$$

We will use the following convention as the indices of products and sums,

$$\sum_{j<k} = \sum_{j=1}^{N} \sum_{k=j+1}^{N}, \tag{2}$$

meaning "all combinations" of j and k exactly once.

The wave function in equation 1, will be referred to as the *fully correlated wave function*. The first part of this function is the product over the function g. This is the Gaussian part, which represents a Gaussian distribution of each particle in the system. The function g is given as

$$g(\alpha, \beta, \mathbf{r}_i) = \exp\{-\alpha(x_i^2 + y_i^2 + \beta z_i^2)\}, \tag{3}$$

where α and β are the two variational parameters. β determines the elliptical shape of the distribution (i.e. $\beta = 1$ relates to a spherical distribution). The product over f In equation 1, is the correlation wave function, which is also called *the Jastrow factor*. f is given by

$$f(a, |\mathbf{r}_j - \mathbf{r}_j|) = \begin{cases} 0 & |\mathbf{r}_j - \mathbf{r}_j| \leq a \\ \left(1 - \frac{a}{|\mathbf{r}_j - \mathbf{r}_j|}\right) & |\mathbf{r}_j - \mathbf{r}_j| > a \end{cases}. \tag{4}$$

a is the hard diameter of the particles. Note that if we set $a = 0$, then f is always equal to 1, which in turn means that there is no correlation between the particles. Additionally, one can set the elliptical parameter $\beta = 1$. We obtain a special case of the trial wave function,

$$\Psi_T(\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N, a = 0, \alpha, \beta = 1) = \prod_i \exp\left(-\alpha r_i^2\right). \quad (5)$$

Equation 5 is the special case wave function which will be referred to as the *simple Gaussian wave function.*

B. The Hamiltonian

The Hamiltonian of the system will be given as

$$H = \sum_i^N \left(\frac{-\hbar^2}{2m} \nabla_i^2 + V_{ext}(\mathbf{r}) \right) + \sum_{i<j}^N V_{int}(\mathbf{r}_i, \mathbf{r}_j) \quad (6)$$

$$V_{ext}(\mathbf{r}) = \begin{cases} \frac{1}{2} m \omega_{ho}^2 r^2 & (S) \\ \frac{1}{2}\left[\omega_{ho}^2 (x^2 + y^2) + \omega_z^2 z^2\right] & (E) \end{cases}. \quad (7)$$

Both (S) and (E) are harmonic oscillator traps, where (S) indicates a spherical trap and (E) indicates an elliptical shaped trap.

V_{int} is the potential that governs the collisions of two particles i and j, given by

$$V_{int}(|\mathbf{r}_i - \mathbf{r}_j|) = \begin{cases} \infty & |\mathbf{r}_i - \mathbf{r}_j| \leq a \\ 0 & |\mathbf{r}_i - \mathbf{r}_j| > a \end{cases}. \quad (8)$$

a represents the hard-core diameter of the bosons, and it is the same a which appears in the expression for f in equation 4. If $a = 0$, then the interacting potential $V_{int} = 0$.

C. Local Energy of the Simple Gaussian Wave Function

The local energy $E_L(\mathbf{r})$ is a quantity which will be of high relevance in the study presented. Analytically, it can be found by applying the Hamiltonian on the wave function,

$$E_L(\mathbf{r}) = \frac{1}{\Psi_T(\mathbf{r})} H \Psi_T(\mathbf{r}). \quad (9)$$

It could be very convenient to find analytic expressions for the local energy. We will do this for the *simple Gaussian wave function.* The hamiltonian in equation 6 contains the Laplacian of the wave function. This is what will be determined first.

$$\nabla_i^2 \Psi_T = \nabla_i \cdot \nabla_i \Psi_T \quad (10)$$

$$= \nabla_i \cdot \left[\frac{\partial}{\partial x_i}, \frac{\partial}{\partial y_i}, \frac{\partial}{\partial z_i} \right] \Psi_T \quad (11)$$

$$= \nabla_i \cdot \left[-2\alpha x_i \Psi_T, -2\alpha y_i \Psi_T, -2\alpha z_i \Psi_T \right] \quad (12)$$

$$= -2\alpha \Psi_T \sum_{d=x,y,z} 1 - 2\alpha d_i^2 \quad (13)$$

$$= -2\alpha \Psi_T \left(D - 2\alpha r_i^2 \right). \quad (14)$$

D represents the number of dimensions, e.g. in this case $D = 3$. Now, the local energy for this specific trial wave function is

$$E_L = \frac{1}{\Psi_T} \sum_{i=1}^N \left(\frac{-\hbar^2}{2m} \nabla_i^2 + V_{ext}(\mathbf{r}_i) \right) \Psi_T \quad (15)$$

$$= \frac{1}{\Psi_T} \sum_{i=1}^N \frac{\hbar^2 \alpha}{m} (D - 2\alpha r_i^2) \Psi_T + \frac{1}{2} m \omega_{ho}^2 r_i^2 \Psi_T \quad (16)$$

$$= \frac{ND\hbar^2\alpha}{m} - \frac{2\hbar^2\alpha^2}{m} \sum_{i=1}^N r_i^2 + \frac{1}{2} m \omega_{ho}^2 \sum_{i=1}^N r_i^2. \quad (17)$$

If we set $\hbar = m = 1$ and let r_i^2 implicitly denote the sum over r_i^2 for all particles, then the local energy is

$$E_L = ND\alpha - 2\alpha^2 r_i^2 + \frac{1}{2} \omega_{ho}^2 r_i^2. \quad (18)$$

Once again, the local energy in equation 18 is for the simple Gaussian wave function. Notice that for $\omega_{ho} = 1$ we have a simple analytic expression for the ground state energy, which occurs at $\alpha = 0.5$,

$$E_L = \frac{ND}{2}. \quad (19)$$

Another expression which we will need is the drift force on one particle. This can be found by evaluating

$$F = \frac{2\nabla_k \Psi_T}{\Psi_T} = -4\alpha \mathbf{r}_k. \quad (20)$$

This expression is found by applying the gradient of Ψ_T, and this is shown in equation 12.

D. Local Energy of the Full Wave Function

The fully correlated wave function contains more complex terms in two ways: β can now be different than 1, which means that the Gaussian distribution can be elliptically shaped. In addition, a can be larger than 0, which means that the wave function contains the Jastrow factor. It is convenient to rewrite the wave function as

$$\Psi_T(\mathbf{r}) = \left[\prod_i^N \phi(\mathbf{r}_i) \right] \exp\left(\sum_{i<j}^N u(r_{ij}) \right). \quad (21)$$

where we have defined two new functions ϕ and u by

$$\phi(\mathbf{r}_i) = \exp\left[-\alpha(x_i^2 + y_i^2 + \beta z_i^2)\right] = g(\alpha, \beta, \mathbf{r}_i), \tag{22}$$

$$u(r_{ij}) = \ln f(r_{ij}), \tag{23}$$

$$r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|. \tag{24}$$

The most cumbersome evaluation will be the *Laplacian* of Ψ_T. The full derivation for the gradient of equation 57 can be found in appendix A. The analytical expression we end up with is

$$\nabla_k \Psi_T(\mathbf{r}) = \nabla_k \phi(\mathbf{r}_k) \left[\prod_{i\neq k}^{N} \phi(\mathbf{r}_i)\right] \exp\left(\sum_{i<j}^{N} u(\mathbf{r}_{ij})\right) + \left[\prod_{i}^{N} \phi(\mathbf{r}_i)\right] \exp\left(\sum_{i<j}^{N} u(\mathbf{r}_{ij})\right) \sum_{i\neq k}^{N} \nabla_k u(\mathbf{r}_{ik}). \tag{25}$$

To find the Laplacian of the wave function we must evaluate

$$\frac{1}{\Psi_T(\mathbf{r})} \nabla_k \cdot \nabla_k \Psi_T(\mathbf{r}), \tag{26}$$

where the latter part $\nabla_k \Psi_T(\mathbf{r})$ is given in equation 65. This is a quite long calculation, and the full derivation for the result is shown in appendix A. The final expression is

$$\begin{aligned}\frac{1}{\Psi_T(\mathbf{r})} \nabla_k^2 \Psi_T(\mathbf{r}) = {} & \frac{\nabla_k^2 \phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)} + \\ & 2\frac{\nabla_k \phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)} \sum_{j\neq k} \frac{\mathbf{r}_j - \mathbf{r}_k}{r_{jk}} u'(r_{jk}) + \\ & \sum_{j\neq k} \sum_{l\neq k} \frac{\mathbf{r}_j - \mathbf{r}_k}{r_{jk}} \frac{\mathbf{r}_l - \mathbf{r}_k}{r_{lk}} u'(r_{jk}) u'(r_{lk}) + \\ & \sum_{l\neq k} \frac{2}{r_{lk}} u'(r_{lk}) + u''(r_{lk}).\end{aligned} \tag{27}$$

It is also necessary to state the following expressions

$$u'(r_{ij}) = \frac{r_{ij}}{r_{ij} - a}, \text{ for } r_{ij} > a, \tag{28}$$

$$u''(r_{kj}) = \frac{a(a - 2r_{ij})}{r_{ij}^2 (a - r_{ij})^2}, \text{ for } r_{ij} > a, \tag{29}$$

$$\frac{\nabla \phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)} = -2\alpha \begin{bmatrix} x_k^2 \\ y_k^2 \\ \beta z_k^2 \end{bmatrix}, \tag{30}$$

$$\frac{\nabla^2 \phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)} = 2\alpha \left(2\alpha[x_k^2 + y_k^2 + \beta^2 z_k^2] - 2 - \beta\right). \tag{31}$$

E. Variational Monte Carlo

The expectation value of the energy of a system is given by

$$\langle E \rangle = \frac{\int d\boldsymbol{R} \Psi_T^*(\boldsymbol{R}; \boldsymbol{\alpha}) H(\boldsymbol{R}) \Psi_T(\boldsymbol{R}; \boldsymbol{\alpha})}{\int d\boldsymbol{R} \Psi_T^*(\boldsymbol{R}; \boldsymbol{\alpha}) \Psi_T(\boldsymbol{R}; \boldsymbol{\alpha})}, \tag{32}$$

where $\boldsymbol{R}$ denotes all the positions of the particles in the system, and $\boldsymbol{\alpha}$ the variational parameters. By the variational principle, the expectation value of such a system is an upper bound for the ground state energy E_0 of the Hamiltonian H, i.e.

$$E_0 \leq \langle E \rangle \tag{33}$$

Since evaluating the above integral 32 hardly is feasible in many cases using traditional methods, we instead use Monte Carlo integration. We do this by shifting the particle configurations $\boldsymbol{R}$ according to some rule a large number of times and sample the local energy E_L (eq. 9) at each step, yielding an expectation value of the energy $\langle E \rangle$ for the Hamiltonian. By repeating this process for a set of $\boldsymbol{\alpha}$, we hopefully find the set which minimizes the energy and gives us an approximation to E_0. We use the variance of the energy

$$\sigma_E^2 = \langle E^2 \rangle - \langle E \rangle^2 \tag{34}$$

as a measure to determine if we have reached a true energy minimum, in which case $\sigma_E^2 = 0$.

F. The Metropolis Algorithm

We will apply the Metropolis algorithm. This method is a Markov chain Monte Carlo method (MCMC). The idea is to find the most likely probability distribution function (PDF). Another method which also will be applied, is the Metropolis-Hastings algorithm, which exhibits a different sampling rule.

A brief overview of the Metropolis algorithm will be given here, and more detailed information can be found in the original article [5]. Consider a system in a state i, with the probability of transitioning to state j, given by $T_{i\to j}$. The probability that this transition is accepted is given by $A_{i\to j}$.

The Metropolis choice; whether or not to accept a new move, is given by

$$A_{i\to j} = \min\left(1, \frac{p_i T_{i\to j}}{p_j T_{j\to i}}\right). \tag{35}$$

$\mathbf{P}_i^n$ is the probability of the system being in state i at step n. The variables p_i and p_j denotes $\mathbf{P}_i^{n\to\infty}$. An important assumption in the Metropolis algorithm is to assume that the transitions have equal probability $T_{i\to j} = T_{j\to i}$ so that these cancel in equation 35.

G. The Metropolis-Hastings Algorithm

A different sampling rule will also be applied, namely the Metropolis-Hastings algorithm. The goal of this sampling rule is to include the so-called drift force in order to obtain a higher ratio of accepted transitions when compared to the Metropolis algorithm. First, we must introduce the Fokker-Planck equation,

$$\frac{\partial P(x,t)}{\partial t} = D \frac{\partial}{\partial x}\left(\frac{\partial}{\partial x} - F\right) P(x,t). \tag{36}$$

D is a diffusion coefficient, F is the drift term and P is the probability density. We also have the Langevin equation

$$\frac{\partial x(t)}{\partial t} = DF(x(t)) + \eta, \tag{37}$$

where η is a random variable. When simulating the system, new positions must be suggested in a random direction for a random particle. This must be done by using a random number generator (RNG). In the ordinary Metropolis algorithm, we have a so-called brute-force sampling method, and far from all suggestions will actually be accepted (typically less than 50%). By using the Langevin equation, this ratio of accepted moves can be increased. A suggested move will still be done on a random particle, but the direction will be biased in the direction given by the Langevin equation,

$$y = x + DF(x)\Delta t + \eta\sqrt{\Delta t}. \tag{38}$$

Remember that y is a proposed move, and the direction will be affected by the drift force F, but there will still be some stochasticity regulated by the variable η. This sampling rule is known as *importance sampling*, and it will increase the ratio of accepted moves. This will in turn will waste less CPU cycles, in the execution of the program.

A solution of the Fokker-Planck equation, yields the transition probability. This solution is known as the Greens function,

$$G(y, x, \Delta t) = \frac{1}{(4\pi D\Delta t)^{3N/2}} \exp\left(-\frac{(y - x - D\Delta t F(x))^2}{4D\Delta t}\right). \tag{39}$$

This will now be a part of the sampling rule. In the Metropolis algorithm, we assumed that $T_{i\to j} = T_{j\to i}$. In the Metropolis-Hastings algorithm, the Greens function will substitute the transition probabilities, yielding an acceptance probability of

$$A_{i\to j} = \min\left(1, \frac{G(x, y, \Delta t)|\Psi_T(y)|^2}{G(y, x, \Delta t)|\Psi_T(x)|^2}\right). \tag{40}$$

H. Statistical Resampling Methods

We will adopt two resampling methods to estimate the errors of the energies, which in this case is just a large sample of a quantity. The method which will be used are bootstrapping and blocking.

Bootstrapping is a quite simple procedure to perform. Consider a sample set $\mathbf{x}$ with N observations. Do the following

1. Draw a sample of size N from $\mathbf{x}$, *with replacement.*
2. Compute the desired statistics on the new sample, and save these values.
3. Repeat M times.

Now, we have obtained *samples* of the statistics themselves. Consider that one the statistics of interest is say, the variance. The bootstrapping variance is then the mean of the variance sample. Boostrapping is a method which is often adopted when the sample size is small.

The other method, blocking, is more suitable for larger datasets. It was popularized by Flyvbjerg and Pedersen[4]. In this method we must use a sample which of size 2^d, where $d \geq 1$ is an integer. The procedure is as follows

1. Initial sample vector $\mathbf{x}_0$.
2. Compute the next sample $\mathbf{x}_1$ where each element is the average of each subsequent neighbor-pairs from $\mathbf{x}_0$. This vector is therefore half the length of the previous.
3. Repeat until we have $\mathbf{x}_{d-1}$.

For each of the obtained vectors, we can compute the variance by

$$V(\mathbf{x}_k) = \frac{\sigma_k^2}{n_k} + \frac{2}{n_k}\sum_{h=1}^{n_k-1}\left(1 - \frac{h}{n_k}\right)\gamma_k(h) \tag{41}$$

$$= \frac{\sigma_k^2}{n_k} + e_k, \tag{42}$$

where n_k is the size of vector $\mathbf{x}_k$, γ_k is the autocovariance, and e_k is called the truncation error. It is possible to show that

$$V(\mathbf{x}_k) = V(\mathbf{x}_j), \tag{43}$$

for all $k, j \in [\![0, d-1]\!]$. Therefore, we have that

$$V(\mathbf{x}_0) = \frac{\sigma_k^2}{n_k} + e_k. \tag{44}$$

It was also demonstrated in ref. [4] that the sequence $\{e_k\}_{k=0}^{d-1}$ is decreasing.

I. Gradient Descent

With a trial wave function dependent on several variational parameters, searching the entire space of configurations can be very expensive in terms of CPU cycles. An alternative approach is to treat the local energy of the trial wave function as a cost function in a minimization scheme such as steepest descent[3], and attempt to minimize it wrt. the variational parameters, i.e.

$$\boldsymbol{\alpha}_{i+1} = \boldsymbol{\alpha}_i - \eta\nabla\langle E_L(\boldsymbol{\alpha}_i)\rangle. \tag{45}$$

This iterative scheme moves $\boldsymbol{\alpha}$ in the direction which decreases $\langle E_L(\boldsymbol{\alpha})\rangle$ the most, and hopefully converges to a

global minimum, given a suitable choice for the step length η. To find an expression for the cost function, we define

$$\bar{E}_{\alpha_i} = \frac{\mathrm{d}\langle E_L\rangle}{\mathrm{d}\alpha_i}, \tag{46}$$

$$\bar{\Psi}_i = \frac{\mathrm{d}\Psi}{\mathrm{d}\alpha_i} \tag{47}$$

as the derivatives of the local energy (eq. 9) and trial wave function (eq. 1) wrt. a single variational parameter α_i. By the chain rule and the hermiticity of the Hamiltonian, we end up with

$$\bar{E}_{\alpha_i} = 2\left(\langle\frac{\bar{\Psi}_i}{\Psi}E_L\rangle - \langle\frac{\bar{\Psi}_i}{\Psi}\rangle\langle E_L\rangle\right). \tag{48}$$

The expectation values in the above expression are evaluated using Monte Carlo integration for each step of the iterative process in 45.

J. One-body density

The one-body density is a quantity of interest, especially when studying the correlated systems. This quantity represents the likelihood of finding *one* particle i at a certain position. The mathematical definition is to integrate over all dimensions except the particle in question, the one-dimensional case would look like this

$$\rho(x_1) = \int |\Psi(x_1, x_2, ...)|^2 dx_2 \dots dx_N, \tag{49}$$

which is an $N-1$ dimensional integral. This would be an extremely hard integral to solve, even for systems with few particles. On the other hand, the integral can be approximated by Monte Carlo integration. The algorithm for this, would be to set up a number of bins for say three spatial dimensions. Whilst simulating the system, during the Monte Carlo cycles, add likelihood to the bins where the particles were at. At the end, scale down the integral so that it in total sums up to 1, because it represents a probability.

III. METHOD

The numerical calculations were performed in C++. The post analyses and plots of the data were done in Python.

A. Variational Monte Carlo on the Simple Gaussian Wave Function

The initial part of this study performed Monte Carlo methods on the *simple Gaussian wave function*. This might seem like a trivial system to simulate, but it functions as an important test of the algorithms. There were numerous benefits of simulating this simple system: (1) comparison of using analytic and numerical approaches for the Laplacian, (2) comparison with exact solutions, and (3) studying the dependence of the number of particles and dimensions.

The Laplacian of the wave function is needed when finding the kinetic energy of the system. We want to compare the use of numerical and analytic methods for obtaining the Laplacian. This can give insight about the possible speedup of using analytic expressions, because the numerical three point formula is computationally intense. It will also give a possibility to compare the accuracy of these methods, which will be done by using various statistical methods for analysis, namely bootstrapping and blocking.

It is possible to compare the numerical results with exact solutions. We know that for a spherical harmonic oscillators, with $\omega_{ho} = 1$ and without correlation, $\alpha = 0.5$ gives the ground state of the system. We can also find the energy and variance for the ground state of these systems.

The simple Gaussian wave function was simulated for numerous different systems. The dimensions were varied between one, two and three dimensions. The number of particles were varied from 1, 10, 100 and finally 500 particles. This is essentially 12 different systems. Note that these simulations contains only one variational parameter, α. For all of these cases, we applied a brute-force method of finding α_{opt}, that is to simply try many different α's. The pseudo code for these simulation was as follows

```
for alpha = 0.2, 0.3, ... , 0.9:
    Initialize system
    Equilibrate system
    Simulate system (record E, E^2)
```

This for-loop was parallelized quite easily, because each iteration was completely independent of the others. OpenMP was used to speed up the code by using multi-threading. When not caring to parallelize, the code is normally executed on only one thread, whilst a modern computers can typically have e.g. 8 threads. The for-loop above contains 8 iterations, which now would be done simultaneously. Theoretically, the execution time would be almost as fast as the execution time of only one iteration.

The equilibration and simulation phases contained predefined fixed number of Monte Carlo cycles. The equilibration phase is very important, as we should allow the system to evolve from a random initial state to a more natural state. When we record a quantity, which in this case is the local energy E_L, we perform a so-called Monte Carlo integration. This means that we allow the system to evolve, and at the same time we record the energy to obtain a distribution of energies.

In addition to the 12 various combinations of systems as described earlier (various no. of particles and dimensions), the simulations were also performed for all 4 combinations of these *methods*

- Numerical second derivative (three point formula) and analytic second derivative (eq. 14).
- Metropolis sampling and importance sampling.

All these various systems and methods were analyzed by comparing the execution times of the codes, and by plotting the energies with variance vs. α.

B. Statistical Resampling Methods

In addition to the simulations describes so far, a study of the statistical errors were performed. The wave function was still limited to the *simple Gaussian*, and only the three dimensional systems were studied. As before, the number of particles were varied between 1, 10, 100 and 500 particles. The sampling rule was limited to only the importance sampling.

When the Metropolis-Hastings algorithm was performed, the energies were recorded all along. This gave a distribution of energies for all the various systems. These energy distributions were analyzed by statistical methods, such as bootstrapping and blocking. The goal of performing these analyses was to obtain proper statistical estimates of errors in the energy, σ_E.

C. The Correlated Wave Function

The fully correlated wave function, consisted of elliptical gaussian terms for each particle, and a term which represents the correlation amongst the particles themselves, namely the Jastrow factor. We will begin by rewriting the Hamiltonian (eq. 6), because it is convenient to use a natural energy scale for the system. There are two reasons for this scaling, (1) we want to increase numerical stability by avoiding underflow, and (2) it is more or less a convention to express energy in units of $\hbar\omega$ in quantum mechanics. The rewriting is done by trying to factor out $\hbar\omega_{ho}$,

$$\begin{aligned} H &= \frac{1}{2}\sum_i^N \left(-\frac{\hbar^2}{m}\nabla_i^2 + m\left[\omega_{ho}^2(x_i^2+y_i^2)+\omega_z^2 z_i^2\right]\right) \\ &= \frac{\hbar\omega_{ho}}{2}\sum_i^N \left(-\frac{\hbar}{m\omega_{ho}}\nabla_i^2 + \frac{m\omega_{ho}}{\hbar}\left[x_i^2+y_i^2+\frac{\omega_z^2}{\omega_{ho}^2}z_i^2\right]\right). \end{aligned}$$

Note that for simplicity, we have excluded the potential part (V_{int}) of the Hamiltonian. The factor $\hbar\omega_{ho}$ is now factored out. By inspection, the physical unit of $\hbar/m\omega_{ho}$ is $\sqrt{length}$, and therefore we introduce the natural length scale,

$$a_{ho} = \sqrt{\hbar/m\omega_{ho}}. \tag{50}$$

We also define the parameter

$$\gamma = \omega_z/\omega_{ho}. \tag{51}$$

Since we want to express the energy in *units* of $\hbar\omega_{ho}$, we divide by this factor, and hence it is canceled out. The Hamiltonian is now

$$H = \frac{1}{2}\sum_i^N -a_0^2\nabla_i^2 + a_0^{-2}\left(x_i^2+y_i^2+\gamma^2 z_i^2\right).$$

The last part, is to also rewrite the particle coordinates and the diameter of the bosons by the natural length scale a_0,

$$\boldsymbol{r}' \to \boldsymbol{r}/a_{ho}, \tag{52}$$

$$a' \to a/a_{ho}. \tag{53}$$

This means that also the ∇ operator and x_i, y_i, z_i are scaled. The result of this is that a_0 is canceled out in the expression for the Hamiltonian. The correlated part is again included, and the full expression is

$$H = \frac{1}{2}\sum_i^N -\nabla_i^2 + x_i^2+y_i^2+\gamma^2 z_i^2 + \sum_{i<j} V_{int}(|\boldsymbol{r}_i-\boldsymbol{r}_j|). \tag{54}$$

Knowing that the energy is in units of $\hbar\omega_{ho}$ and that the length is in units of a_0, the expression for the Hamiltonian is much shorter than earlier. The hard-core diameter of the bosons was fixed to $a/a_0 = 0.00433$ as done in refs. [2] and [6].

The aim of the simulations was to find the optimal set of the variational parameters. *Optimal* in this case, means those parameters which resulted in the system with lowest energy. There were now two variational parameters, α in the Gaussian part of the wave function and β which was governed by the trap. As done in refs. [2] and [6], we fix

$$\beta = \gamma = \sqrt{8} \approx 2.82843. \tag{55}$$

Two different approaches were performed, in order to find α_{opt}: a brute-force method and a gradient descent (GD) method. Both of these methods were executed for 10, 50 and 100 particles, and for three dimensions (elliptical trap). The simulations were performed with the standard Metropolis sampling rule.

The brute-force method was done by trying different value of α within a $[0.2, 0.8]$. The values of $\langle E\rangle$ obtained by the Monte Carlo integrations, were plotted vs. α. The errors σ_E were obtained by blocking.

The gradient descent method was done by using various initial value of α, and then performing small steps in the direction of $-\partial\langle E_L\rangle/\partial\alpha$. We used a variable learning schedule which decays with each iteration

$$\eta_{n+1} = \frac{\eta_n}{1+nd}, \tag{56}$$

where n is the iteration step and d is the decay parameter. The α would converge towards systems with a lower $\langle E_L\rangle$.

D. One-body Density

With the optimal parameters obtained in the previous subsection, a study of the one-body density was conducted.

The systems were initialized with the optimal α and β, and simulations were performed for wave functions both with and without the Jastrow factor. The aim of this comparison was to analyze the contribution of the Jastrow factor; how the systems would stabilize with and without this correlating factor.

IV. RESULTS

A. Non Interacting Bosons in Spherical Harmonic Oscillator

1. Standard Metropolis Sampling

Figure 1 shows the resulting expected energy for the simple Gaussian trial wave function from doing Monte Carlo integration using 2^{21} steps, for three systems, each with $N = 1$, 10 and 100 particles in one dimension. The search for the ground state energy has been done for $\alpha \in [0.2, 0.9]$, using both the analytic expression (eq. 14) and numerical differentiation in evaluating the kinetic energy of the system. We see from the figure that $\alpha = 0.5$ yields the lowest energy in each case, with $\langle E_{1\mathrm{p}} \rangle = 0.5$, $\langle E_{10\mathrm{p}} \rangle = 5.0$ and $\langle E_{100\mathrm{p}} \rangle = 50.0$, which corresponds well with the true values in eq. 19. For each value of α, we see that the resulting energy values from using both the analytic and numeric kinetic energy are relatively close, but in the numerical case the standard deviation is a lot higher, especially in the systems of more than one particle. Notice also that the analytic case gives a standard deviation of $\sigma = 0$ in all cases where $\alpha = 0.5$. Additional results from systems in higher dimensions can be found in Appendix B.

In table I we show the results from systems of $N = 1$, 10, 100 and 500 particles in three dimensions. We present here the energy, standard deviation, ratio of accepted Metropolis steps and CPU time for running the simulation. For one particle, we get the exact result $\langle E \rangle = 1.5 \pm 0.0$ with both the analytic and numerical schemes. This also holds for the larger particle systems in the analytic case. We see that the numerical scheme produces energy values which are very close to the exact ground state energy, but they get less and less accurate the larger the system is. The numerical scheme is also more costly in terms of CPU time usage, roughly a factor of 3 more than when using the analytic expressions. The table also shows that each proposed Metropolis step is accepted 73% of the time in all cases.

2. Importance Sampling

We investigate the proper time step Δt to use in the Fokker-Planck equation (36) in importance sampling, shown in figure 2. We find that a step length of $\Delta t = 0.1$ is well suited for systems of sizes $n = 1, 10, 100$ particles, with an average acceptance ratio of 98%. Systems of 500 particles looks to be very sensitive to the choice of step length, and rejects most of the proposed moves when $\Delta t > 10^{-4}$.

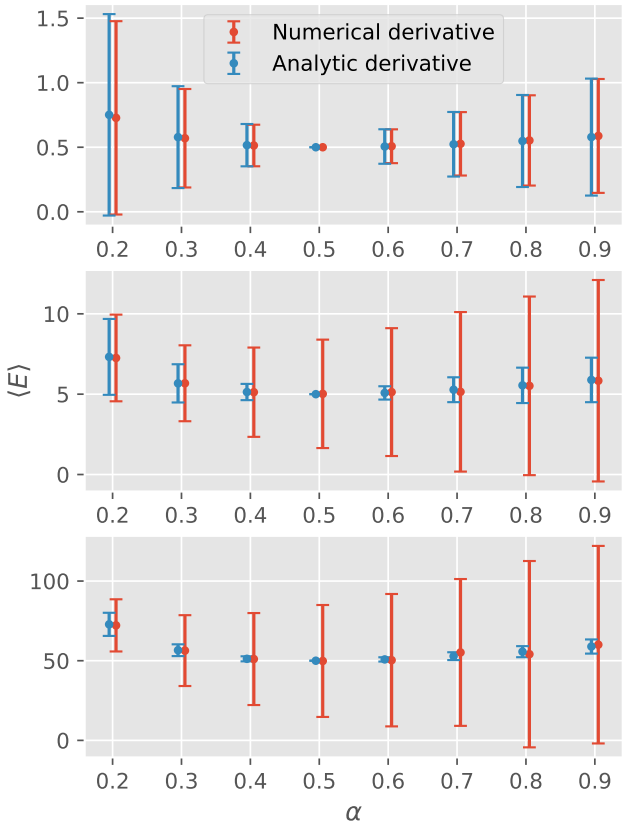

FIG. 1. Energy of systems with $N = 1$ (top), 10 (mid.) and 100 (bot.) particles in one dimension after $2^{21} \approx 2 \cdot 10^6$ Monte Carlo cycles, in search for ground state energy in $\alpha \in [0.2, 0.9]$. Both analytical expressions and numerical differentiation (step length $h = 0.001$) for the kinetic energy have been used. Error estimates represent 1σ. Values of α are slightly shifted for visibility, but they are exactly 0.2, 0.3, ..., 0.8.

TABLE I. Energies, ratio of accepted Metropolis steps and CPU run time of the integration of the simple Gaussian with $\alpha = 0.5$ and different system sizes in three dimensions, using both analytic expressions and numerical (step length $h = 0.001$) evaluation of the kinetic energy. 2^{21} Monte Carlo steps.

	Analytic				Numerical			
N	$\langle E \rangle$	σ_{mc}	ratio	t_{CPU}	$\langle E \rangle$	σ_{mc}	ratio	t_{CPU}
1	1.5	0.0	0.733	2.2	1.5000	0.0	0.734	3.4
10	15.0	0.0	0.735	3.8	15.0067	5.7848	0.735	10.1
100	150.0	0.0	0.734	23.7	149.9630	61.3759	0.734	74.2
500	750.0	0.0	0.730	113.8	744.4880	311.8724	0.731	364.5

Table II shows the results from a grid search of the ground state energy for the different values of Δt for a system of 100 particles in three dimensions, where we see that $\log_{10} \Delta t \in [-3, 0]$ yield stable results for this system.

Table III shows again the expected energies of the system

with 100 particles in three dimensions when performing a grid search for the optimal parameter α, using importance sampling. Not unexpectedly, we again find that $\alpha = 0.5$ gives the analytic result of $\langle E \rangle = 150.0 \pm 0.0$. In this table we have also included a better estimate of the error in the results, by using the bootstrap and blocking methods. We see that these estimates are considerably smaller.

In table IV we also show the results from simulations of $N = 1, 10, 100$ and 500 particles for $\alpha = 0.5$. In the case where we use analytic expressions for the kinetic energy, the resulting energy of the systems is $\langle E \rangle = \frac{ND}{2}$ with $\sigma = 0.0$ for all particle configurations as expected. When using the numerical approximation, the systems behave well for $N = 1, 10$ and 100 particles, but struggles with 500 particles with this configuration of parameters.

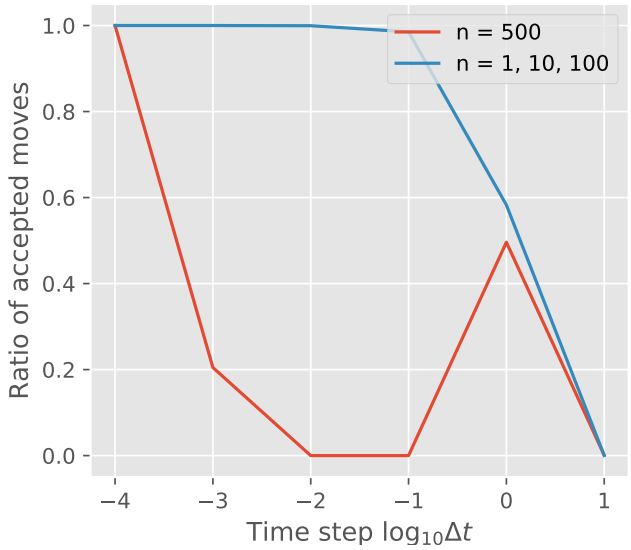

FIG. 2. Ratio of accepted moves in importance sampling. Shows average over all systems of sizes 1, 10, 100 in 3 dimensions and the 500 particle system by itself, run with 2^{21} Monte Carlo cycles.

TABLE II. Run of 100 particles in 3 dimensions with importance sampling in search for the ground state energy, for differing values of the step length Δt, showing the corresponding "optimal" value of α.

Δt	$\alpha_{E_{min}}$	$\langle E_{min} \rangle$	σ_{mc}	σ_{block}	ratio
10^{-4}	0.4	149.127	5.688	1.0	1.0
10^{-3}	0.5	150.0	0.0	0.0	0.999
10^{-2}	0.5	150.0	0.0	0.0	0.999
10^{-1}	0.5	150.0	0.0	0.0	0.985
10^{0}	0.5	150.0	0.0	0.0	0.582
10^{1}	0.4	140.646	3.061	0.754	0.001

TABLE III. Results from ground state energy search in system of 100 particles in three dimensions. 2^{21} Monte Carlo cycles with importance sampling with time step $\Delta t = 0.1$.

α	$\langle E \rangle$	σ_{mc}	σ_{boot}	σ_{block}	ratio
0.2	218.703	12.842	0.018	0.362	0.996
0.3	170.537	6.753	0.027	0.498	0.993
0.4	153.834	2.787	0.032	0.593	0.989
0.5	150.0	0.0	0.0	0.0	0.985
0.6	152.336	2.251	0.050	0.869	0.981
0.7	158.371	4.153	0.061	1.031	0.976
0.8	166.685	5.952	0.004	0.069	0.970
0.9	176.501	7.526	0.077	1.230	0.965

TABLE IV. Energies, ratio of accepted Metropolis steps and CPU run time of the integration of the simple Gaussian with $\alpha = 0.5$ and different system sizes in three dimensions, using both analytic expressions and numerical (step length $h = 0.001$) evaluation of the kinetic energy. The systems of $N = 1$, 10 and 100 particles have been run with $\Delta t = 0.1$ in the importance sampling, while in the 500 particle system we have used $\Delta t = 0.0001$. 2^{21} Monte Carlo steps.

	Analytic				Numerical			
N	$\langle E \rangle$	σ_{block}	ratio	t_{CPU}	$\langle E \rangle$	σ_{block}	ratio	t_{CPU}
1	1.5	0.0	0.985	4.335	1.5	0.0	0.985	5.593
10	15.0	0.0	0.985	6.129	14.964	0.037	0.985	12.377
100	150.0	0.0	0.985	26.178	149.799	1.254	0.985	77.446
500	750.0	0.0	1.0	114.982	799.106	20.992	1.0	366.846

B. Correlating Particles in Elliptical Harmonic Oscillator

We now run simulations for the interacting particles in the elliptical trap, according to 54. The search for the energy minimum is done with a fixed value $\omega_z/\omega_{ho} = \gamma = 2.82843$ in the Hamiltonian, as done in refs. [2] and [6]. Since the interactions of the particles are relatively weak, we suspect that the optimal α-value is close to the minimum of the previous system, and we therefore do a search for it in $\alpha \in [0.2, 0.8]$. Figure 3 shows the energy per particle $N = 10$, 50 and 100 particles, in units of $\hbar\omega_{ho}$, which we notice quickly increases with system size. The plot reveals an energy minimum for α somewhere in the range $[0.4, 0.5]$. To narrow down the search we use the gradient descent approach to search for the true minimum, as this allows us to find the optimal α-value to a much higher precision without having to do brute force simulations on a very fine search grid. We perform gradient descent with different initial values of α_0, running 2^{17} Monte Carlo steps for each iteration, and in figure 4 we show it's development of towards the minimum. We find the optimal value to be $\alpha_{opt} = 0.474 \pm 0.009$.

In table V we show the results from using this value of α in a larger Monte Carlo integration for each of the systems with $N = 10$, 50 and 100 particles, which is done to obtain a more precise value of the ground state energy for this interacting system. In order to study of the effect of the interaction between the particles, we also take a look at the one-body density of the system with 100 particles, shown in figure

5. The figure shows the density of particles in the elliptical trap, with and without the repulsive interaction, governed by the Jastrow factor. We see that the particles favour a slightly more spread out distribution with the Jastrow factor.

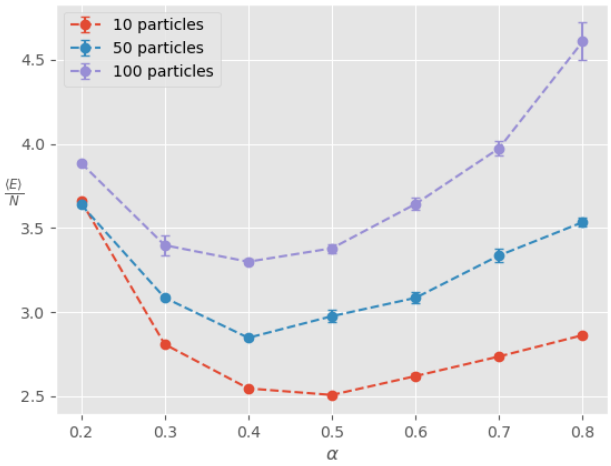

FIG. 3. The expectation value of the energy per number of particles, in units of $\hbar\omega_{ho}$. This is the brute force approach to find the optimal value of α for three different systems, which only differed in the number of particles they contained. The simulations used $2^{20} = 1,048,576$ metropolis steps for each α, and obtained the same amount of energy samples. The figure shows errorbars of $\pm\sigma_{block}$, obtained by blocking.

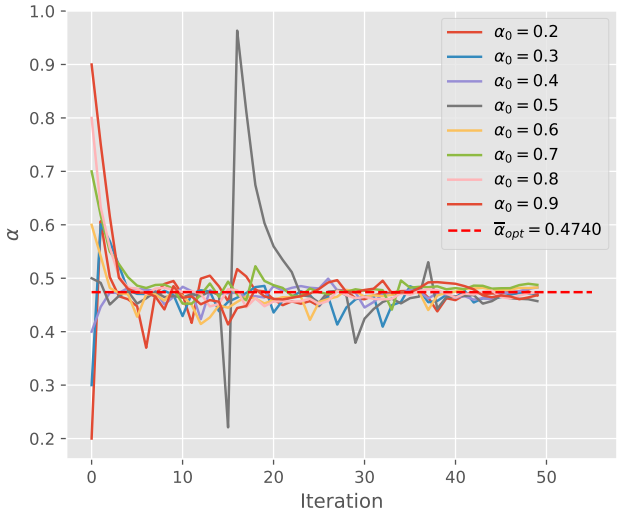

FIG. 4. Iterations in search for optimal α using Gradient descent with a learning rate of $\eta = 0.01$ and decay factor $d = 0.001$.

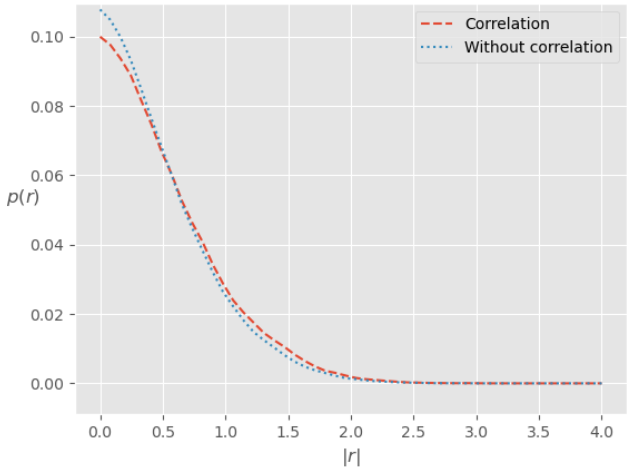

FIG. 5. One-body density in systems of 100 particles, and where $\alpha = 0.474$. The correlated system contained the Jastrow factor in the trial wave function, whereas the other system had set the hard core diameter of the bosons to $a = 0$, which corresponds to no Jastrow factor.

TABLE V. Energy with error estimates using the Blocking and Bootstrap methods, for 10, 50 and 100 particles, using the found optimal value $\alpha = 0.474$, 2^{21} Monte Carlo cycles.

N	$\langle E \rangle$ $[\hbar\omega_{ho}]$	σ_{mc}	σ_{boot}	σ_{block}	t_{CPU} $[s]$
10	25.130	13.238	0.009	0.026	5.1
50	146.160	164.509	0.109	0.792	105.9
100	339.894	640.667	0.433	3.687	413.9

V. DISCUSSION

A. The Simple Gaussian

Given that we have simple analytical expressions, independent of system size, for the non interacting particles in a harmonic oscillator, it is not as interesting to study as when we have interactions between the particles. It does, however, serve a very useful purpose in testing and benchmarking our code, and to be sure that it is working as intended. We ran our code for systems with every permutation of $N = 1$, 10, 100, 500 particles in $D = 1$, 2, 3 dimensions, which gave the exact energy with $\sigma = 0$ every time, as seen in table I for the three dimensional case. Studying this system also allows us to test out using numerical approaches to calculating the kinetic energy part of the Hamiltonian, which for more complex systems may be to cumbersome to find the analytic expression for. Although the numerical case is also accurate in terms of the energies it finds for these systems, figure 1 serves as an eye-opener with regards to the high uncertainties in the results (in all but the one particle in one dimension). In the numeric differentiation, we have used the three point rule with a step length of $h^2 = (0.001)^2$, chosen

rather arbitrarily, and looking at the high uncertainties in the results which it produces, it is clear that more thought must go in to the analysis of numerical schemes if one is to use this approach. We also see from table I that using numeric differentiation requires roughly three times the CPU usage compared to it's analytical counterpart. This fact, together with the large uncertainties in the results, should serve as a reminder that finding analytic expressions for the kinetic energy is highly desirable when ever possible.

By introducing importance sampling, we quickly see that this has beneficial effects on the number of proposed steps which are accepted by the sampling rule. This is best demonstrated by comparing tables I and IV, where we had roughly 73% accepted moves in standard Metropolis sampling and 98% when using the Metropolis-Hastings algorithm. This is expected as the particles are more likely to move in the direction of the quantum drift force. When determining a suitable choice for the time step Δt, we recorded the acceptance ratio for systems with different numbers of particles. Looking at figure 2 we found that $\Delta t \leq 0.1$ was a suitable choice for $N = 1, 10$ and 100 particles, whilst the 500 particle system required a much smaller step size of $\Delta t \leq 10^{-4}$ in order to accept proposed moves at a high rate. We also investigated how the time step affected our results in terms of finding the ground state energy of a system of 100 particles. Our findings in table II demonstrate that by using time steps $\Delta t < 10^{-3}$ and $\Delta t > 1$, the algorithm struggles to find the correct minimum. We suspect that too low a time step may have led to the system not having enough time to reach its equilibrium state within the given number of Monte Carlo steps, and too high a value may propose steps which in most cases lie outside the trap, in which case the move will be rejected by the sampling rule.

B. Interacting Particles

When introducing the Jastrow factor in the wave function, the first thing we notice in our results is that the energy of the system no longer has a linear relation to the size of the system. Due to the particles now having an additional source of potential energy, namely the particles themselves, this is not an unexpected result, and from figure 3 we can clearly see that the energy per particle increases with the system size. This figure also demonstrates the need for a new approach to finding the value α which will yield the ground state energy of the system, as we have energy minima for both $\alpha = 0.4, 0.5$, and we clearly cannot use brute force Monte Carlo sampling for finding the optimal value to several digits of precision. Using the much more CPU efficient approach of treating the local energy as a cost function in a Gradient descent scheme, we found the optimal value $\alpha_{opt} = 0.474 \pm 0.009$. This was done by sampling a system of 10 particles and a mere 2^{17} Monte Carlo steps per iteration, saving many CPU cycles compared to the alternative. We also demonstrate that it is unlikely that this is a local minimum, because we used initial values of α_0 in the range of $[0.2, 0.9]$. As a comparison between our result and previous efforts, [6] found the optimal value to be $\alpha_{opt} = 0.475$. Our results does, within its uncertainty, correspond with the previous effort. It is important to point out that the energy of the systems we are studying, look to be of a convex nature wrt. α, which is why we "get away" with using such a simple optimizing scheme. For more complex models with a trial wave functions with more variational parameters, we likely would have to make use of more sophisticated optimization methods in finding the global minimum of the local energy function.

Using the obtained optimal α-value, we performed Monte Carlo sampling on the three systems of 10, 50 and 100 particles in order to obtain a more precise estimate of the ground state energy, as seen in table V. Here we also included more rigorous estimates of the error in our results, with both the bootstrapped and blocked standard deviations of the energy. This table clearly demonstrates that using $\sigma_E^2 = \langle E^2 \rangle - \langle E \rangle^2$ in the Monte Carlo integration is a very poor estimate of the error. This can be seen by the huge errors it yields. On the other hand, when using the more sophisticated statistical analyses: bootstrapping and blocking, the errors are orders of magnitudes lower. Bootstrapping is more widely used in obtaining better error estimates of smaller data sets, whereas blocking is more suitable for larger data sets. Our data set of energy samples is very large with 2^{21} data points, and therefore we have the most confidence in the errors estimated by the blocking method.

Comparing these results with the non-interacting case in table I, we see that for instance in the 100 particle case, the energy of the system has more than doubled, which is rather surprising, given the minuscule hard-core diameters of the bosons. Notice also the difference in CPU run times of these two system in the tables, where the non-interacting particles had a run time of 26 seconds and the interacting ran for six times this amount. From a computational point of view, running Monte Carlo sampling on larger systems would be very demanding the way our code is set up, due to every particles interaction with every other particle. If we were to run Monte Carlo calculations for larger systems, a solution to this would be to define a cut-off length where the interaction potential is negligible, and only consider particles within this distance when calculating the Hamiltonian for a given particle.

By finding the one-body densities with and without the Jastrow factor, as shown in figure 5, we can get an idea of how "spread out" the particles in the system are. The figure show two systems of 100 particles each, and we can see that the correlated particles are less concentrated in the center of the system, and also have a wider distribution. This behaviour is expected, because the correlation, governed by the Jastrow factor, is a repulsive force.

C. Further Improvements

This section will summarize our suggestions for further improvements of the study presented. When finding the optimal α for the correlated system, we applied the standard gradient descent. This method is however not resistant in avoiding local minima. We defend our choice with the fact

that we additionally used a brute-force search, and we did also use a range of various initial values in the GD itself. On the other hand, there are more sophisticated algorithms that could be applied, such as conjugate gradient methods and stochastic gradient descent with mini batches. Although it was not the intention of this project, it is also possible to vary the other variational parameter β. There are also other parameters that could be varied, such as the boson diameter a, the expression for the trial wave function, the Hamiltonian, the shape or characteristics of the harmonic oscillator trap. Lastly, the code could be extended to study fermions instead of bosons. In this case, one would also have to account for spin, and Fermi's exclusion principle. As a final remark, there are many possibilities in studying quantum mechanical systems with the use of MCMC methods.

VI. CONCLUSION

In this project we use the Variational Monte Carlo method to study systems of Bose-Einstein condensates of bosons in an elliptical trap. The aim has been to use a specific trial wave function in determining the ground state energy of the systems, and investigate how the Jastrow factor affects the particle density compared to systems without the inter-particle potential. In the development of our code, we have made use of ideal non-interacting systems, to which there exists analytic solutions. We have also investigated both analytic and numerical approaches to calculating the kinetic energy of the Hamiltonian, as well as the use of the Fokker-Planck and Langevin equations for importance sampling in the Monte Carlo machinery. With the use of gradient descent minimization, we found that the optimal value to the variational parameter $\alpha = 0.474$ yielded the lowest energy state of the interacting systems. By evaluating the one-body density with the optimal set of variational parameters, we identify that the Jastrow factor accounts for a repulsive force amongst the particles. This can be seen as the particle density in the interacting systems was more spread out than in the systems without the Jastrow factor.

REFERENCES

[1] A. Fabrocini, A. P. (1999). Beyond the gross-pitaevskii approximation: Local density versus correlated basis approach for trapped bosons. *Physical Review A*, 60:2319.

[2] Dubois, J. and Glyde, H. (2000). Bose-einstein condensation in trapped bosons: A variational monte carlo analysis. *Physical Review A*, 63.

[3] Faul, A. C. (2016). *A Concise Introduction to Numerical Analysis*, chapter 2.14. CRC Press.

[4] Flyvbjerg, H. and Petersen, H. G. (1989). Error estimates on averages of correlated data. *The Journal of Chemical Physics*, 91(1):461–466.

[5] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092.

[6] Nilsen, J., Mur-Petit, J., Guilleumas, M., Hjorth-Jensen, M., and Polls, A. (2005). Vortices in atomic bose-einstein condensates in the large gas parameter region. *Physical Review A*, 71:053610.

APPENDIX A: GRADIENT AND LAPLACIAN OF THE GENERAL TRIAL WAVE FUNCTION

A. Gradient of the Trial Wave Function

For the full wave function with a non-trivial correlated part, finding it's derivatives is a bit more tricky. For convenience we rewrite the wave function in equation 1 as

$$\Psi_T(\mathbf{r}) = \left[\prod_i^N g(\alpha, \beta\mathbf{r}_i)\right] \exp\left(\sum_{i<j}^N u(r_{ij})\right), \tag{57}$$

where we have defined $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ and

$$f(r_{ij}) = \exp\left(\sum_{j<k}^N u(r_{jk})\right), \tag{58}$$

with $u(r_{ij}) = \ln f(r_{ij})$, and denote the one-body part of the trial wave function in equation 3 as

$$g(\alpha, \beta, \mathbf{r}_i) = \phi(\mathbf{r}_i). \tag{59}$$

By the product rule, the first derivative of particle k is

$$\nabla_k \Psi_T(\mathbf{r}) = \Psi_C \nabla_k \Psi_{OB} + \Psi_{OB} \nabla_k \Psi_C, \tag{60}$$

where Ψ_{OB} and Ψ_C are the one-body and correlated parts of the wave function respectively,

$$\Psi_{OB} = \prod_i^N \phi(\mathbf{r}_i), \tag{61}$$

$$\Psi_C = \exp\left(\sum_{i<k}^N u(r_{ik})\right). \tag{62}$$

Since Ψ_{OB} is nothing but the product of the individual one-body wave functions, only one of the factors are dependent on $\mathbf{r}_k$, and it's first derivative is therefore

$$\nabla_k \Psi_{OB} = \left[\prod_{i\neq k}^N \phi(\mathbf{r}_i)\right] \nabla_k \phi(\mathbf{r}_k). \tag{63}$$

Similarly for Ψ_C, the $\mathbf{r}_k$ dependent terms are the pairs $\sum_{i\neq k} u(\mathbf{r}_{ik})$, which when taking the derivative gives

$$\nabla_k \Psi_C = \exp\left(\sum_{i<j}^N u(r_{ij})\right) \sum_{i\neq k}^N \nabla_k u(\mathbf{r}_{ik}) \tag{64}$$

Bringing it together and inserting equations 63 and 64 into equation 60, we end up with

$$\begin{aligned}\nabla_k \Psi_T(\mathbf{r}) = \nabla_k \phi(\mathbf{r}_k) \left[\prod_{i\neq k}^N \phi(\mathbf{r}_i)\right] \exp\left(\sum_{i<j}^N u(\mathbf{r}_{ij})\right) \\ + \left[\prod_i^N \phi(\mathbf{r}_i)\right] \exp\left(\sum_{i<j}^N u(\mathbf{r}_{ij})\right) \sum_{i\neq k}^N \nabla_k u(\mathbf{r}_{ik}).\end{aligned} \tag{65}$$

B. Laplacian of the Trial Wave Function

To find the laplacian of the wave function we must evaluate

$$\frac{1}{\Psi_T(\mathbf{r})} \nabla_k \cdot \nabla_k \Psi_T(\mathbf{r}), \tag{66}$$

where the latter part $\nabla_k \Psi_T(\mathbf{r})$ is given in equation 65. To start out, we omit the $1/\Psi_T(\mathbf{r})$ factor and focus solely on the Laplacian.

$$\begin{aligned}\nabla_k \cdot \Bigg[\nabla_k \phi(\mathbf{r}_k) \prod_{i\neq k} \phi(\mathbf{r}_i) \exp\Big(\sum_{j<m} u(r_{jm})\Big) + \\ \prod_i \phi(\mathbf{r}_i) \exp\Big(\sum_{j<m} u(r_{jm})\Big) \sum_{l\neq k} \nabla_k u(r_{kl})\Bigg]\end{aligned} \tag{67}$$

$$\begin{aligned}= \prod_{i\neq k} \Bigg[\nabla_k^2 \phi(\mathbf{r}_k) \exp\Big(\sum_{j<m} u(r_{jm})\Big) + \\ \nabla_k \phi(\mathbf{r}_k) \cdot \nabla_k \exp\Big(\sum_{j<m} u(r_{jm})\Big)\Bigg] + \\ \nabla_k \prod_i \phi(\mathbf{r}_i) \exp\Big(\sum_{j<m} u(r_{jm})\Big) \sum_{l\neq k} \nabla_k u(r_{kl}) + \\ \nabla_k \exp\Big(\sum_{j<m} u(r_{jm})\Big) \prod_i \phi(\mathbf{r}_i) \sum_{l\neq k} \nabla_k u(r_{kl}) + \\ \nabla_k \sum_{l\neq k} \nabla_k u(r_{kl}) \prod_i \phi(\mathbf{r}_i) \exp\Big(\sum_{j<m} u(r_{jm})\Big).\end{aligned} \tag{68}$$

Before we proceed, some of the gradients that are to be found in equation 67 should be clarified her. There are three main gradients which occur, and some of them results in terms that could be canceled by $\Psi_T(\mathbf{r})$ later on.

$$\nabla_k \exp\Big(\sum_{j<m} u(r_{jm})\Big) = \exp\Big(\sum_{j<m} u(r_{jm})\Big) \sum_{l\neq k} \nabla_k u(r_{kl}), \tag{69}$$

$$\nabla_k \prod_i \phi(\mathbf{r}_i) = \prod_{i\neq k} \phi(\mathbf{r}_i) \nabla_k \phi(\mathbf{r}_k), \tag{70}$$

$$\nabla_k \sum_{l\neq k} \nabla_k u(r_{kl}) = \sum_{l\neq k} \nabla_k \cdot \left(\frac{\mathbf{r}_l - \mathbf{r}_k}{r_{lk}} u'(r_{lk})\right), \tag{71}$$

The latter equation, namely equation 72 is not evaluated completely yet. The product rule must be applied.

$$\begin{aligned}\nabla_k \sum_{l\neq k} \nabla_k u(r_{kl}) = \\ \sum_{l\neq k} \Big[\nabla_k \cdot \frac{\mathbf{r}_l - \mathbf{r}_k}{r_{lk}} u'(r_{kl}) + \frac{\mathbf{r}_l - \mathbf{r}_k}{r_{lk}} \cdot \nabla_k u'(r_{lk})\Big] \\ = \sum_{l\neq k} \frac{2}{r_{lk}} u'(r_{lk}) + u''(r_{lk})\end{aligned} \tag{72}$$

What happened in the final transition in equation 72, were firstly a rule that the gradient of a unit vector is 2 divided by the magnitude of the vector itself. Secondly, the unit vector dotted with the gradient of u', disappears because these are parallel vectors, and the result is simply the scalar second derivative of u. This would be a good time to reintroduce the Laplacian of the wave function, and this time include the factor $1/\Psi_T(\mathbf{r})$, which will cancel out may of the terms.

$$\frac{1}{\Psi_T(\mathbf{r})}\nabla_k^2\Psi_T(\mathbf{r}) = \frac{\prod_{i\neq k}\phi(\mathbf{r}_i)}{\prod_i\phi(\mathbf{r}_i)}\left[\nabla_k^2\phi(\mathbf{r}_k)+\nabla_k\phi(\mathbf{r}_k)\sum_{l\neq k}\nabla_k u(r_{kl})\right]+ \frac{\nabla_k\phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)}\sum_{l\neq k}\nabla_k u(r_{kl})+\sum_{l\neq k}\nabla_k u(r_{kl})\sum_{j\neq k}\nabla_k u(r_{kj})+ \nabla_k\sum_{l\neq k}\nabla_k u(r_{kl}). \tag{73}$$

Notice how the second and third expression are identical. In addition, the term $\nabla_k u(r_{kl})$ is simply the unit vector multiplied with the scalar derivative. The fifth and last expression is shown in equation 72. Hence our final expression is

$$\frac{1}{\Psi_T(\mathbf{r})}\nabla_k^2\Psi_T(\mathbf{r}) = \frac{\nabla_k^2\phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)}+ 2\frac{\nabla_k\phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)}\sum_{j\neq k}\frac{\mathbf{r}_j-\mathbf{r}_k}{r_{jk}}u'(r_{jk})+ \sum_{j\neq k}\sum_{l\neq k}\frac{\mathbf{r}_j-\mathbf{r}_k}{r_{jk}}\frac{\mathbf{r}_l-\mathbf{r}_k}{r_{lk}}u'(r_{jk})u'(r_{lk})+ \sum_{l\neq k}\frac{2}{r_{lk}}u'(r_{lk})+u''(r_{lk}). \tag{74}$$

APPENDIX B: ADDITIONAL RESULTS

Non Interacting Bosons in Shperical Harmonic Oscillator Trap

Standard Metropolis Sampling

Figure 6 shows additional energies vs. α for 2 and 3 dimensional systems, where the Laplacian is evaluated by both using analytic expressions and the numerical three point formula. The rather large uncertainties (especially for the numerical scheme) are obtained by MC integration, and does not reflect the better estimates of the uncertainty which can be obtained by blocking (see table IV).

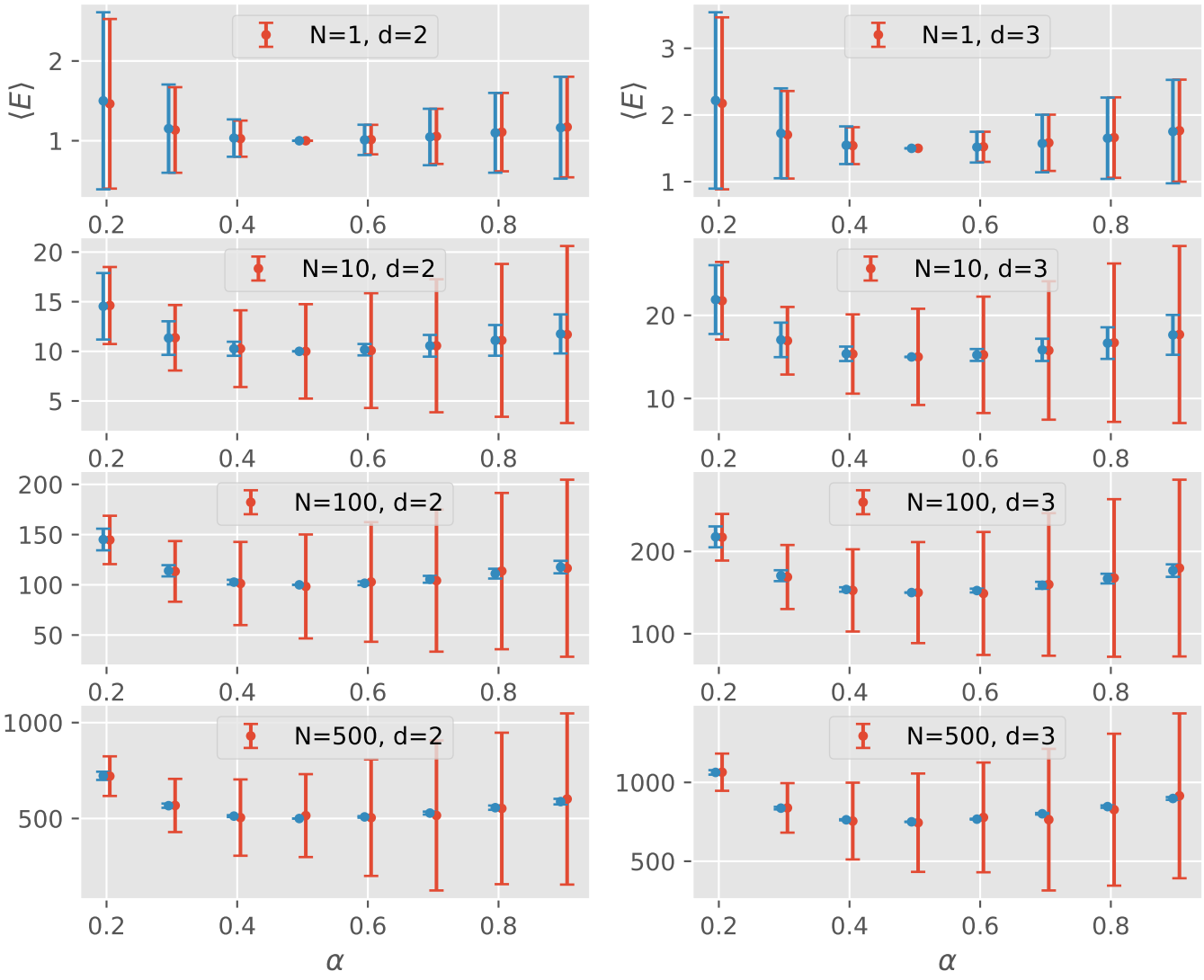

FIG. 6. Energy of systems with $N = 1$, 10, 100 and 500 particles in two and three dimensions after 2^{21} Monte Carlo cycles, in search for ground state energy in $\alpha \in [0.2, 0.9]$. Both analytical expressions (blue/left) and numerical differentiation (red/right) (step length $h = 0.001$) for the kinetic energy have been used. Error estimates represent 1σ obtained by Monte Carlo integration. Values of α are slightly shifted for visibility.