VARIATIONAL MONTE CARLO METHOD APPLIED TO A SYSTEM OF INTERACTING BOSONS IN A ELLIPTICAL POTENTIAL

UNIVERSITY OF OSLO - FYS4411 COURSE
COMPUTATIONAL PHYSICS II: QUANTUM MECHANICAL SYSTEMS - PROJECT 1

Emiliano Staffoli, Matteo Zortea, Alexander Ferraro

March 26, 2021

Abstract

In this project we implemented a Variational Monte Carlo (VMC) code to reproduce the properties of a Bose-Einstein condensate (BEC) [1], at first considering the bosons as independent and later adding a pair interaction represented by a hard-sphere potential. Both the brute-force Metropolis [2] and the Metropolis-Hastings algorithms [3] were tested while considering the non-interacting case, revealing the better performance provided by the latter in terms of higher acceptance ratio and better sampling properties. The energy of the system was evaluated on varying the number of particles, the dimensionality and the variational parameter α, revealing the accordance with analytical solutions available for this simple case. A statistical analysis of the results was performed through the blocking technique [4], which provided us with estimate of the degree of correlation between data generated within a VMC simulation.
The interaction between the particles was then taken into account, along with a modification of the trap shape. The research performed through the gradient descent method revealed that the value of α minimizing the energy of the system decreases when the number of particles grows. This shows that adding elements into the apparatus moves it further and further away from the ideal condition described by the non-interacting model. This fact was supported by both an increased average energy per particle and an increased average distance from the origin when more particles are included in the system. Finally, the evaluation of the one-body density revealed also a reduction in the displacement along the direction in which the trap has been made steeper.

1 INTRODUCTION

A Bose-Einstein condensate (BEC) is a state of matter that we can observe in a system of bosons confined and cooled down to temperatures close to the absolute zero. Since this physical concept was first proposed by Einstein and Bose [1] in the 1920s, a lot of years have passed until the first experimental observation of this phenomenon, which was achieved by C. E. Wieman & E. A. Cornell [5] and W. Ketterle [6] only in 1995. Working independently, these two groups were able to observe a system of bosons constituted respectively by Rubidium atoms and Sodium atoms trapped at extremely low temperatures.

Nowadays these systems are still intensively studied both from a experimental and theoretical perspective. In the former case, a BEC is produced in the laboratory exploiting laser cooling techniques and magnetic traps to confine the bosons [7]. A theoretical analysis can instead be performed adopting a statistical approach, especially when the density of the system is sufficiently high to prevent from a proper description through the Gross-Pitaevskii equation [8][9]. The problem can then be faced with the implementation of a Variational Monte Carlo (VMC) code, which allows to get pieces of information about a system using a statistical approach instead of trying to derive analytical solutions. Exploiting the concept of Markov chains [10] and algorithms based on pseudo-random numbers generation, we are allowed to explore all the possible states of a system and thus obtain numerical quantities related to it as averages over the possible configurations. The Monte Carlo method becomes particularly effective and time-saving when we have to deal with multidimensional integrals whose analytical evaluation would be impossible and the usage of traditional numerical methods would require a great effort.

In this project we will analyzed the properties of a system constituted by ^{87}Rb atoms through the implementation of a VMC code. We considered a non-interacting Hamiltonian and then switched to the interacting case, adopting for both the configurations a properly built trial wavefunction. The traps for the bosons were described through a spherical or an elliptical potential. In our analysis we included also the search for the value of the variational parameter α corresponding to the minimum energy of the system. The VMC steps were performed exploiting different variants of the Metropolis algorithm [2] [3] and a statistical analysis of the obtained results was provided by the blocking technique [4]. The parallelization of the code through `OpenMP` contributed to a reduction of the execution time.

1.1 OVERVIEW OF THE WORKFLOW

In the following section we will introduce the physical aspects behind the considered problem, together with the mathematical instruments that we adopted for the description of the system in terms of Hamiltonians, wavefunctions and one-body densities. A first mention to the error analysis is also reported, with particular attention to the problem of correlated data. In Section 3 we will switch to a description of the methods adopted to treat

the problem, with particular attention to the different variant of the Metropolis algorithm, the gradient descent technique and the blocking method for the error analysis. The code structure and implementation is then treated in Section 4, followed by the presentation of the obtained results in Section 5 and the subsequent discussion in Section 6. We will proceed specifying possible further improvements to enhance code performance and some more detailed analysis that could be included (Section 7), concluding then with the final remarks about the project (Section 8). Some additional results and calculations can be found in the Appendix.

2 THEORY AND MATHEMATICAL INSTRUMENTS

When considering a system of particles at a high temperature and in the limit of low particle density, the Maxwell-Boltzmann distribution provides a very good description of the statistical features of the system. However, when the temperature decreases, approaching the absolute zero, a classical description is no more sufficient, giving way to a quantum mechanical treatment. In this context, the bosonic or fermionic nature of the particles comes into play in determining the statistical behaviour of the apparatus. Fermions are driven by the Pauli exclusion principle, so each quantum state must be at most single-populated and the population will obey the Fermi-Dirac statistics. In case of a non-interacting system, the wavefunction describing the ground state would be provided by a Slater determinant, accounting for the anti-symmetry required by the Pauli exclusion principle. On the contrary, for bosons there is no limit on the occupational number for each single quantum state. In addition, in the non-interacting case at $T = 0$, the system is described by the product of the single-particle ground state wavefunctions [11].

Focusing our attention to a system constituted by bosons, it is known that when the temperature gets below a certain critical value T_c, the ground state starts being densely populated and a transition to the BEC phase occurs. At this point the Bose-Einstein statistics takes the lead in the description of the system, accounting for purely quantum effects that are not predictable with a classical approach [11].

Alkali atoms are particularly suitable to produce BECs: in fact, due to their ground state electronic structure and scattering properties, they are relatively easy to be laser-cooled and trapped [11]. The so-formed condensates are characterized by a high-diluteness condition, that is the size of the trap in which the atoms are inserted and the interatomic spacing are both large compared to the typical atomic dimension. In this project we treated a gas composed by ^{87}Rb atoms, whose s-wave scattering length (representing the interaction between the atoms) is usually chosen to be $a_{Rb} = 100a_0$, where a_0 is the Bohr radius. Supposing to describe the trap as a harmonic potential, a typical choice for its size is $a_{Rb}/a_{ho} = 0.0043$ with $a_{ho} = (\hbar/m\omega_{ho})^{1/2}$. Combining these data with an atomic density of $n \simeq 10^{12} - 10^{14}$ atoms/cm^3, one gets an average interatomic spacing of $l \simeq 10^4$Å, and thus the conditions cited above for the diluteness of the system are satisfied [12]. A useful quantity related to the local density of the system $n(\boldsymbol{r})$ is the gas parameter $x(\boldsymbol{r}) = n(\boldsymbol{r})a^3$, which allows to access the diluteness of the considered apparatus. A BEC characterized by an average gas parameter $x_{av}(\boldsymbol{r}) \leq 10^{-3}$ can be properly described by the Gross-Pitaevskii equation; however other methods as VMC simulations become a valuable choice for describing the apparatus when the density increases [13].

2.1 HAMILTONIAN AND TRIAL WAVEFUNCTION

The hamiltonian describing a population of N ^{87}Rb atoms is given by

$$\hat{H} = \sum_{i=1}^{N} \left(-\frac{\hbar^2}{2m}\nabla_i^2 + V_{ext}(\mathbf{r}_i) \right) + \sum_{i<j}^{N} V_{int}(\mathbf{r}_i, \mathbf{r}_j) \quad (1)$$

where we have used the shorthand notation for

$$\sum_{i<j}^{N} V_{int}(\mathbf{r}_i, \mathbf{r}_j) = \sum_{i=1}^{N} \sum_{j=i+1}^{N} V_{int}(\mathbf{r}_i, \mathbf{r}_j)$$

The external potential describing the trap for the atoms will be modelled in two different ways, namely as a spherical (S) or elliptical (E) harmonic trap.

$$V_{ext}(\mathbf{r}) = \frac{1}{2}m\omega_{ho}^2(x^2 + y^2 + z^2) \qquad (S) \qquad (2)$$

$$V_{ext}(\mathbf{r}) = \frac{1}{2}m[\omega_{ho}^2(x^2 + y^2) + \omega_z^2 z^2] \qquad (E) \qquad (3)$$

For the sake of generality here we have shown the three-dimensional version of the spherical potential, but notice that this form will be adopted also to describe systems with lower dimensionality. On the contrary, the elliptical potential is aimed to describe only 3D systems. We introduce also the parameter $\gamma = \omega_z/\omega_{ho}$, whose value will be defined succesively.

Since we are considering a dilute system, the interaction between particles is mainly described through 2-body collisions, which in this context are well represented by the s-wave scattering length a for ^{87}Rb. This repulsive interaction is modelled by a hard-core repulsion term

$$V(\boldsymbol{r}_i, \boldsymbol{r}_j) = V(|\boldsymbol{r}_i - \boldsymbol{r}_j|) = \begin{cases} \infty & |\boldsymbol{r}_i - \boldsymbol{r}_j| \leq a \\ 0 & |\boldsymbol{r}_i - \boldsymbol{r}_j| > a \end{cases}$$

For our treatment the value of a is fixed by $a/a_{ho} = 0.0043$, which, as stated in the previous paragraph, is a typical choice for relating the size of the trap to the dimension of the atoms.

The trial wavefunction used in the VMC approach is given by

$$\begin{aligned} &\Psi_T(\boldsymbol{R}, \alpha, \beta) = \Psi_T(\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N, \alpha, \beta) = \\ &= \left[\prod_{i=1}^{N} g(\alpha, \beta, \boldsymbol{r}_i)\right] \left[\prod_{j<k} f(a, |\boldsymbol{r}_j - \boldsymbol{r}_k|)\right] \\ &g(\alpha, \beta, \boldsymbol{r}_i) = \exp\left[-\alpha\left(x_i^2 + y_i^2 + \beta z_i^2\right)\right] \\ &f(a, |\boldsymbol{r}_j - \boldsymbol{r}_k|) = \begin{cases} 0 & |\boldsymbol{r}_j - \boldsymbol{r}_k| \leq a \\ 1 - \frac{a}{|\boldsymbol{r}_j - \boldsymbol{r}_k|} & |\boldsymbol{r}_j - \boldsymbol{r}_k| > a \end{cases} \end{aligned} \quad (4)$$

where $\boldsymbol{R}$ is a collective variable standing for the coordinates of all the particles in the system. In principle both α and β could be treated as variational parameters, however in this project only the former will be used with this connotation, while the latter will be set to a constant. Notice that the $\Psi_T(\boldsymbol{R}, \alpha, \beta)$ written above is the trial wavefunction corresponding to the most general case treated in the project, namely interacting

atoms trapped in an elliptical potential. In this case we will set $\beta = \gamma = 2.82843$. The analogue version for the non-interacting particles trapped in a spherical potential can be simply obtained from Eq. 4 by setting $a = 0$ and $\beta = 1$. In this last case, the number of terms in the quadrature sum appearing in $g(\alpha, \beta = 1, \boldsymbol{r}_i)$ will be determined by the dimensionality of the system.

From now on, we will refer to the introduced quantities considering them as re-scaled according to what follows

$$\hbar = m = \omega_{ho} = 1$$

Thus $a_{ho} = 1$ and $\gamma = \omega_z$. Energy values will thus be expressed in units of $\hbar\omega_{ho}$, wile lengths in units of a_{ho}.

2.2 ONE-BODY DENSITY

Another interesting quantity to explore related to our system is the one-body density $\rho(r)$, which is defined as

$$\rho(\boldsymbol{r}) = \frac{1}{\int d\boldsymbol{R}|\Psi_T(\boldsymbol{R})|^2} \int d\boldsymbol{r}_2 \dots d\boldsymbol{r}_N |\Psi_T(\boldsymbol{R})|^2$$

It acts as a PDF describing the spatial distribution of a particle inserted in a many-body system. For the apparatus considered in this project, the above integral can be easily analytically solved in the simple non-interacting case, while its evaluation becomes much more challenging when the interaction between particles is taken into account. We limited the one-body density analysis to this second case, evaluating the integral through a VMC simulation involving the system in its ground state (namely using the α value obtained after the energy minimization process).

2.3 ERROR ANALYSIS

In Monte Carlo experiments and also more in general, we are looking for expectation values and an estimate of how accurate they are. This kind of simulation has to deal with two possible sources for errors: systematical ones and statistical ones. The former are given by factors that drive the observation away from its true value in a predictable way, e.g. repeated measurements could be shifted by a constant value away from the correct result and the accuracy is obviously compromised. This sort of error is difficult to detect and every system needs a different treatment to investigate possible sources, but once that they are eventually found, we can reasonably get rid of them. On the other hand, statistical errors can not be avoided and they affect the precision of the measurement by inserting some random fluctuations in the observation process. This kind of error reveals itself when repeated observations of the same quantity return different values reciprocally shifted by a non-predictable amount. Random errors can be estimated using standard tools from statistics.

When the final result of an experiment consists in a mean value over several independent observations of the same quantity, the estimation of the error is provided by the Central Limit Theorem (CLT). We consider a set of discrete measurements $\{x_1, x_2, \dots, x_N\}$ with sample mean μ_s and sample variance σ_s^2 defined as

$$\mu_s = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\sigma_s^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_s)^2$$

We assume them to be independent and identically distributed (iid), that is these observations are sampled from the same distribution with ideal mean μ and variance σ^2. The CLT states that the PDF describing the distribution of μ_s has a limiting form obtained for $N \to \infty$ described by a gaussian centered at μ with variance $\sigma_\mu^2 = \sigma_s^2/N$. This occurs independently of the distribution type from which the samples x_i are driven. For a finite but sufficiently high value of N, it is thus possible to estimate variance of the distribution of μ_s as $\sigma_\mu^2 \approx \sigma_s^2/N$.

While performing a VMC simulation, the usage of pseudo-random number generators combined with Markov Chains causes the hypothesis of iid data to decay, since sequentially generated observations will be statistically correlated. In this more general case, the estimation of σ_μ^2 from the discrete sampled values $\{x_i\}$ must include a correlation term. It is possible to demonstrate that the following estimate holds

$$\sigma_\mu \approx \frac{\sigma_s^2}{N} + \frac{2}{N^2}\sum_{k<l}(x_k - \mu_s)(x_l - \mu_s) \tag{5}$$

where the sum accounts for the correlation between sampled data. By introducing the iid hypothesis, for sufficiently large N the sum decays to zero and the CLT is restored. However, for the VMC simulations analyzed in this project, since correlation between data is present, we must include this sum (or an estimate of it) while providing the error on a specific measurement.

3 METHODS

3.1 VARIATIONAL MONTE CARLO

While trying to access the properties of a system, one often has to deal with coupled differential equations or with multidimensional integrals. In both these cases finding a solution is absolutely non trivial: an analytical expression could even not exist and using the traditional numeric methods can be very time consuming, especially for systems with a high number of degrees of freedom.

A possible alternative to face with these kind of problems is provided by the VMC method: this technique is based on a statistical approach and on the exploration of the possible configurations for a system through the generation of random numbers. At each step of a VMC simulation, a new proposal for a possible configuration is created and properly implemented algorithms relying on pseudo-random numbers discriminate between the acceptance or rejection of such move. These algorithms are built to ensure that the frequency of the states, during the walk in the configurations' space, reproduces a given probability distribution for the system of being in a given state. After a sufficiently large number of steps necessary for the system to thermalize to the desired probability distribution, the quantities of interest for the considered apparatus (e.g. energy) are provided as averages evaluated on the configurations explored during the simulation.

The success of this technique is mainly due to its extreme versatility and to the fact that, as previously mentioned, it allows to avoid the direct treatment of integrals and differential equations. This methods reveals to be much effective in providing us with a result without needing to spend much effort in analytical or numerical evaluations. However, as it will be shown in the next sections, the results computed through this technique strongly depend on the knowledge of the function describing the probability for the system of being in a

certain state.

3.2 LOCAL ENERGY

One of the most relevant quantities in which we are interested for the characterization of our system is its ground state energy. The variational principle states that the expectation value of the Hamiltonian evaluated on any wavefunction that can be written as a linear combination of the system's eigenstates is an upper bound for the true ground state energy E_0. For our specific case, this reads

$$E_0 \leq E[\hat{H}](\alpha) = \langle \Psi_T(\boldsymbol{R},\alpha)|\hat{H}|\Psi_T(\boldsymbol{R},\alpha)\rangle = \frac{\int d\boldsymbol{R}\Psi_T^*(\boldsymbol{R},\alpha)\hat{H}\Psi_T(\boldsymbol{R},\alpha)}{\int d\boldsymbol{R}\Psi_T^*(\boldsymbol{R},\alpha)\Psi_T(\boldsymbol{R},\alpha)}$$

Our aim is then to find the value of α that minimizes the integral reported above. A Variational Monte Carlo simulation allows to face the problem of its evaluation. First one can define a probability density function $P(\boldsymbol{R},\boldsymbol{\alpha})$ as

$$P(\boldsymbol{R},\boldsymbol{\alpha}) = \frac{|\Psi_T(\boldsymbol{R},\boldsymbol{\alpha})|^2}{\int d\boldsymbol{R}\Psi_T^*(\boldsymbol{R},\boldsymbol{\alpha})\Psi_T(\boldsymbol{R},\boldsymbol{\alpha})}$$

which simply describes the probability to find the system in a given state in which particles' positions are described by the collective variable $\boldsymbol{R}$. Defining then the local energy $E_L(\boldsymbol{R},\boldsymbol{\alpha})$ as

$$E_L(\boldsymbol{R},\boldsymbol{\alpha}) = \frac{1}{\Psi_T(\boldsymbol{R},\boldsymbol{\alpha})}\hat{H}\Psi_T(\boldsymbol{R},\boldsymbol{\alpha}) \tag{6}$$

one can finally rewrite the expected energy value as

$$E[\hat{H}](\alpha) = \int d\boldsymbol{R}\ P(\boldsymbol{R},\boldsymbol{\alpha})E_L(\boldsymbol{R},\boldsymbol{\alpha}) \tag{7}$$

At this point one can introduce a Monte Carlo estimator

$$E[\hat{H}](\alpha) \simeq \frac{1}{N_s}\sum_{i=1}^{N_s} E_L(\boldsymbol{R}_i,\boldsymbol{\alpha}) \tag{8}$$

where N_s is the number of steps performed within a Monte Carlo simulation and $\boldsymbol{R}_1,\ldots,\boldsymbol{R}_{N_s}$ are sets of positions sampled from the distribution $P(\boldsymbol{R},\alpha)$. From a statistical point of view the law of large numbers guarantees that evaluating integral in Eq. 7 is equivalent to evaluate the sum in Eq. 8 if $N_s \to \infty$, but since this is obviously an unattainable condition one searches for a compromise between precision and computational cost. However, even with a finite N_s steps, better results in terms of both these aspects can be obtained with a previous knowledge of the analytical form of the local energy function.
For the system considered in this project it was possible to obtain an analytical expression for the energy of the system as a function α in the simple case of N non-interacting particles in a D-dimensional spherical potential. This result will be adopted for comparisons with the estimates produced by the VMC code.

$$\langle E_L\rangle(\alpha) = ND\left(\frac{\alpha}{2}+\frac{1}{8\alpha}\right) \tag{9}$$

3.2.1 ANALYTICAL FORM

For the explicit calculations that lead to the results reported below, see Appendix A.

In the most complex case treated in the project, namely a $3D$ system of interacting particles and elliptical potential, the result is

$$E_L(\boldsymbol{R},\alpha) = \alpha(2+\beta)N + \sum_i^N\left[(x_i^2+y_i^2)\left(\frac{1}{2}-2\alpha^2\right)+ + z_i^2\left(\frac{1}{2}\omega_z^2-2\alpha^2\beta^2\right)\right] - \frac{1}{2}\sum_i^N\sum_{j\neq i}\frac{a}{r_{ij}^2(r_{ij}-a)}\times \times\left\{-4\alpha\left(x_i,y_i,\beta z_i\right)\cdot\mathbf{r}_{ij}+2+\frac{a-2r_{ij}}{r_{ij}-a}+ + \mathbf{r}_{ij}\cdot\sum_{m\neq i}\frac{\mathbf{r}_{im}}{r_{im}}\frac{a}{r_{im}\left(r_{km}-a\right)}\right\}\right\} \tag{10}$$

The analytic expression for the local energy in the case of a D-dimensional system of N non-interacting particles subject to a spherical potential ($a=0$, $\beta=\omega_z=1$) results to be

$$E_L(\boldsymbol{R},\alpha) = DN\alpha + \left(\frac{1}{2}-2\alpha^2\right)\sum_{j=1}^N r_j^2 \tag{11}$$

3.2.2 NUMERICAL EVALUATION OF THE SECOND DERIVATIVE

The local energy for the considered system can be estimated recurring to the numerical evaluation of the laplacian appearing in the Hamiltonian. In general, the second derivative of a function $f(x)$ in $x=x_0$ can be numerically approximated as

$$\frac{\partial^2 f(x_0)}{\partial x^2} \approx \frac{f(x_0+h)+f(x_0-h)-2f(x_0)}{h^2}$$

with an asymptotic error of $O(h^4)$. This method allows to avoid all the calculations cited above in order to find the analytical expression for the local energy, but provides less accurate results and implies a much greater computational effort. In fact, one can observe that, according to Eq. 1 and Eq. 6, in order to estimate all the terms descending from the Laplacian appearing in the Hamiltonian of a D-dimensional system constituted by N particles, one would need to evaluate the trial wavefunction $3ND$ times, and this procedure would have to be repeated for each step of the VMC simulation. The analytical approach described above is thus preferable, however this numerical alternative will be implemented for a comparison.

3.3 METROPOLIS ALGORITHM

A very simple and effective algorithm usually implemented in the framework of a VMC simulation to discriminate between the acceptance and the rejection of a proposed move is the Metropolis one. This is intimately linked with the concept of Markov chains [10], which constitute the theoretical basis for all the possible implementation of a Monte Carlo simulation. In this framework, $P_i^{(n)}$ represents the probability of finding the system in a state i at time step n (assuming to have discrete time) and the transition probability matrix $W(j\to i) = W_{ij}$ gives the probability of a transition from state j to state i. The elements W_{ij} are modelled as products between an acceptance probability and a transition probability

$$W(j\to i) = A(j\to i)T(j\to i)$$

and this is completely legitimate, as in general the form of W is unknown. For both the matrices A and T the following normalization condition applies

$$\sum_j A(j \to i) = \sum_j T(j \to i) = 1$$

The evolution of the Markov chain is then described by

$$P_i^{(n+1)} = \sum_j \Big\{ A(j \to i) T(j \to i) P_j^{(n)} + \\ + \left[1 - A(i \to j)\right] T(i \to j) P_i^{(n)} \Big\} \tag{12}$$

which expresses the fact that at each step of the simulation the state i can be reached after a transition from a state j or after the rejection of a proposed move, being the system already in state i. Ideally, for a converging Markov chain the steady state distribution P is reached after an infinite number of steps, then

$$P = WP \qquad , \qquad P = \lim_{n \to \infty} P^{(n)}$$

Inserting this condition into Eq. 12, we obtain the detailed balance condition

$$\frac{A(j \to i)}{A(i \to j)} = \frac{T(i \to j) P_i}{T(j \to i) P_j} \tag{13}$$

This is the driving equation for the Metropolis algorithm that we are going to use for the VMC code implemented in this project. The algorithm is built in such a way that, starting from the system being in a state j, it will propose a transition to a state i with a given probability $T(j \to i)$, but then the move itself will be actually finalized with a probability $A(j \to i)$, otherwise the system will remain in its current state i. This last function is modelled by the Metropolis algorithm as

$$A(j \to i) = \min\left\{1, \frac{T(i \to j) P_i}{T(j \to i) P_j}\right\} \tag{14}$$

With this we impose that the system will surely move to a new proposed state if the ratio on the right side of of Eq. 13 is greater than 1, otherwise the move will be accepted only if a generated random number is smaller than the ratio itself. This choice for the acceptance probability does not affect the ergodic hypothesis for the Markov process, since every possible state of the system can in principle be explored. Moreover, since we are considering the ratio P_i/P_j, any problem due to the evaluation of multidimensional integrals related to the normalization of such PDF totally disappears.

Possible models for the transition probability matrix $T(j \to i)$ and for the generation of the new configurations for the system are discussed below.

3.3.1 BRUTE-FORCE METROPOLIS ALGORITHM

A suitable choice for the transition probability matrix consists in imposing a symmetry condition, namely $T(i \to j) = T(j \to i)$. For each step of the VMC simulation the algorithm acts as follows

- considering a randomly selected particle, each component $k = \{x, y, z\}$ of its position vector is modified according to

  $$(\boldsymbol{r}')_k = (\boldsymbol{r})_k + r_{step} * \eta$$

  where $\eta \in (-1, 1)$ is a random number generated from a uniform distribution and r_{step} is a step length chosen by the user;

- this move gets actually accepted if $\eta' < A(j \to i)$, with $\eta' \in (0, 1)$ being another random number generated from a uniform distribution and

  $$A(j \to i) = \min\left\{1, \frac{|\Psi_T(\boldsymbol{R}_i, \alpha)|^2}{|\Psi_T(\boldsymbol{R}_j, \alpha)|^2}\right\} \tag{15}$$

We notice that in order to let the algorithm work correctly it is necessary to choose a proper step r_{step}: a too small value implies that a higher number of Monte Carlo step is needed to reach the convergence of the Markov chain and to explore a significant number of configurations of the system. On the contrary, a too large step would produce a very low fraction of accepted moves during the computation, especially if the wavefunction is particularly peaked in a specific region of the space. Generally the step length used for the so-called brute-force Metropolis algorithm [2] is set by a comparison with the typical length which characterises the system. In our specific case, the step was chosen to be equal to the typical trap size for ^{87}Rb atoms, namely $a_{ho} = (\hbar/m\omega_{ho})^{1/2}$, which according to the chosen units becomes $r_{step} = a_{ho} = 1$. As a rule of thumb we could state that a properly chosen step length should provide an acceptance ratio around 0.5.

3.3.2 IMPORTANCE SAMPLING

In the brute-force Metropolis algorithm the generation of the new possible state for the system is entirely based on a uniformly distributed random variable and the information provided by the wavefunction comes to play only at the moment of the evaluation of the acceptance probability. On the contrary, one could think of exploiting the knowledge of the PDF already at the moment of the generation of the proposed transition, including a drift force whose role it to drive the system towards high-probability configurations. This leads to an increased acceptance ratio, meaning a more efficient simulation since less steps are wasted with the rejection of the proposed move. This is the core of the Importance sampling method [3] in the context of Metropolis algorithm.

The idea just illustrated can be implemented starting from the Fokker-Plank equation [14], which describes the behaviour of a time-dependent PDF $P(\boldsymbol{r}, t)$ associated to a diffusive process characterized by a diffusion coefficient D. The equation reads

$$\frac{\partial P(\boldsymbol{r}, t)}{\partial t} = D \sum_i \frac{\partial}{\partial \boldsymbol{r}_i} \left(\frac{\partial}{\partial \boldsymbol{r}_i} - \boldsymbol{F}_i \right) P(\boldsymbol{r}, t)$$

where i is an index running on the components of vector $\boldsymbol{r}$. The function $\boldsymbol{F}$ represents the aforementioned drift force term that we want to introduce in order to guide the system towards high-probability regions of the configurations space. The form of this drift force can be deduced by the assumption that for $t \to \infty$ the PDF will converge to a time-independent steady-state solution $P(\boldsymbol{r})$. Imposing this condition into the last equation, one reaches that $\boldsymbol{F}$ assumes the following form

$$\boldsymbol{F}(\boldsymbol{r}) = \frac{1}{P(\boldsymbol{r})} \nabla P(\boldsymbol{r})$$

The fundamental role played by this drift force in the problem becomes clearer when we consider the Langevin equation, which describes the diffusive motion of a particle under the action of a drift force $\boldsymbol{F}(x(t))$ and a random

contribution $\boldsymbol{\eta}$

$$\frac{\partial \boldsymbol{r}(t)}{\partial t} = D\boldsymbol{F}(\boldsymbol{r}(t)) + \boldsymbol{\eta}$$

By solving this differential equation assuming a discrete time step δt, we get a new algorithm for the generation of the proposed move for each component k of the position vector of a randomly chosen particle, namely

$$(\boldsymbol{r}')_k = (\boldsymbol{r})_k + D\delta t(\boldsymbol{F}(\boldsymbol{r}))_k + \xi\sqrt{\delta t} \tag{16}$$

Here ξ is a gaussian random variable with null mean value and unitary standard deviation. We can clearly appreciate the effect of the drift force: in fact it will introduce a tendency for the particles to generate high-probability configurations by pushing them towards regions where the PDF is flat and high-valued. A single move is then more likely to be accepted. We notice that even with this new way of generating the propose for the new position of the particles the ergodic hypothesis on the Markov chain is still preserved, since every state can possibly be reached within a sufficiently high number of steps.

Finally, by solving the Fokker-Plank equation it is possible to derive an expression form for the transition probability appearing in Eq. 14, which results to be in the form of the Green's function [14], namely

$$G(\boldsymbol{r}', \boldsymbol{r}, t) = \frac{1}{(4\pi D\delta t)^{3N/2}} \exp\left[-\frac{(\boldsymbol{r}' - \boldsymbol{r} - D\delta t\boldsymbol{F}(t))^2}{4D\delta t}\right]$$

The importance sampling algorithm will then act as follows:

- considering a randomly selected particle, each component $k = \{x, y, z\}$ of its position vector is modified according to

  $$(\boldsymbol{r}')_k = (\boldsymbol{r})_k + D\delta t\boldsymbol{F}_k + \xi\sqrt{\delta t}$$

  where ξ is a gaussian random variable with $\mu = 0$ and $\sigma = 1$, while δt is the time step chosen by the user;

- this move gets actually accepted if $\eta < A(j \to i)$, with $\eta \in (0, 1)$ is a random number generated from a uniform distribution and

  $$A(j \to i) = \min\left\{1, \frac{|\Psi_T(\boldsymbol{R}_i, \alpha)|^2}{|\Psi_T(\boldsymbol{R}_j, \alpha)|^2}\frac{G(\boldsymbol{r}', \boldsymbol{r}, \delta t)}{G(\boldsymbol{r}, \boldsymbol{r}', \delta t)}\right\} \tag{17}$$

The expressions for the drift force obtained in the interacting and non-interacting case are reported together with the corresponding calculations in Appendix B.

3.4 GRADIENT DESCENT METHOD FOR ENERGY MINIMIZATION

Since we are dealing with a one-variational parameter problem, the search for the minimum energy of the system is reduced to find the value α_{GS} which satisfies

$$\left.\frac{d\langle E_L(\boldsymbol{R}, \alpha)\rangle}{d\alpha}\right|_{\alpha_{GS}} = 0 \tag{18}$$

If the function to be minimized is convex, the condition expressed here guarantees that α_{GS} actually corresponds to a global minimum of that function.

For a shorter notation, in this section the dependence of E_L and Ψ_T on $\boldsymbol{R}$ will be omitted. The search for α_{GS} is performed through the standard gradient descent method [15]: after choosing an initial guess for α, at each iteration a new value for this parameter is generated according to

$$\alpha_{k+1} = \alpha_k - \gamma\left.\frac{d\langle E_L(\alpha)\rangle}{d\alpha}\right|_{\alpha_k} \tag{19}$$

The choice of the new proposed value is thus driven by the derivative of the function that has to be minimized and by a parameter γ chosen by the user. A too large step size may lead to a failure of the research, while a too small one to high CPU time for reaching α_{GS}. Thus one searches again for a compromise. The method is arrested when the evaluation of the derivative of the considered function returns a value smaller than a chosen tolerance ε, resembling thus the content of Eq. 18.

In this minimization problem we must therefore provide the algorithm with the derivative appearing in Eq. 19. Starting from the content of Eq. 6 and Eq. 7, exploiting the Hermiticity of $\hat{H}$ one gets (see Appendix C)

$$\frac{d\langle E_L(\alpha)\rangle}{d\alpha} = 2\left[\left\langle E_L(\alpha)\frac{\overline{\Psi}_T(\alpha)}{\Psi_T(\alpha)}\right\rangle - \langle E_L(\alpha)\rangle\left\langle\frac{\overline{\Psi}_T(\alpha)}{\Psi_T(\alpha)}\right\rangle\right] \tag{20}$$

with

$$\overline{\Psi}_T(\alpha) = \frac{\partial\Psi_T(\alpha)}{\partial\alpha}$$

For each α_k the averages appearing in the last equation are evaluated through a Monte Carlo simulation performed with a relatively small amount of steps. The main aim is to find the best approximation to α_{GS} and once that the convergence has been reached or k has exceeded an upper limit set by the user, then a larger simulation is performed and the result will be considered as the most accurate estimate of the ground state energy for the system.

When applying the standard gradient descent method, a fundamental role is played by the initial guess for the parameter that initializes the chain of Eq. 19. As a matter of fact, a bad choice could lead to a wrong convergence of the method and the parameters provided by the algorithm could correspond to local minima or saddle points instead of global minima of the considered function. Nevertheless, these possible issues may apply to complicated systems in which a high number of parameters is involved. On the contrary, in the simple framework of non-interacting particles in a spherical potential treated in this project it is easy to prove the convexity of $\langle E_L(\alpha)\rangle$, which is actually minimized by $\alpha = 0.5$ (see Eq. 9). This result will be exploited for a comparison with the numerical ones provided by the algorithm. Regarding the interacting case, a possible analytical proof of the convexity of $\langle E_L(\alpha)\rangle$ would have been quite demanding, but the shape of the curve plotted as a function of α (see Section 5) did not showed any unpredictable behaviour, making us still confident in the application of the gradient descent method.

3.5 ONE-BODY DENSITY EVALUATION

The main aim in evaluating the one-body density consists in observing the spatial distribution of a particle with respect to the origin of the reference system. In this project, we opted for the evaluation of ρ as a function of the distance from the origin: at every Monte Carlo step we evaluate the modulus of the position vector for each particle, increasing then the count in the k-th bin when the obtained distance falls between r_k and r_{k+1}. The extrema of each bin are selected by evenly spac-

ing the interval $(0, r_{max})$ in N_{bins} cells, where both r_{max} and N_{bins} are selected by the user. The so-built one-body density has been evaluated in the interacting case both imposing $a = 0$ and $a \neq 0$ in order to compare the modifications introduced by the hard-sphere potential. Using the same method, also the average displacement of the particles along some specific directions (e.g. x and z) have been evaluated, in this case to show the effect brought by the interaction between the particles and by the switch between sperical and elliptical potential.

3.6 BLOCKING METHOD FOR VARIANCE ANALYSIS

As stated in Section 2.3, the correlation between data generated in a VMC simulation must be taken into account while estimating the error on an average quantity provided by the code. However, in our specific case the correlation term appearing into Eq. 5 requires much computational power to be evaluated and a better solution consists in estimating it, avoiding a precise calculation of its value. For this reason, one efficient strategy is provided by the blocking method, which is a iterative procedure performed by repeating blocking transformation on the given data set. At each iteration we form a new bunch of data by taking the mean of every pair of subsequent elements of the array obtained at the previous step. As the number of iterations increases, the evaluation of the variance on the obtained bunch of data will include more and more of the correlation contribution of Eq. 5, providing thus a better estimate of the error on the desired quantity.

In a more detailed way, the method proceed as follows: first, consider a large data sample X with cardinality $n = 2^d$ with d integer greater than 1. Since we have to provide a correct estimation for the error on the average energy coming from a VMC simulation, the elements of the sample X in our case are the value of the local energy at every Monte Carlo step. We take a set X_k of measurements, starting with $k = 0$ (no blocking transformation has been applied to original data set), then we create a new set of data X_{k+1} where the elements x'_i of this new set are the mean of subsequent pair of elements from X_k. At each blocking transformation the sample size is halved.

$$\{X_k\} = \{x_1, x_2, x_3 \dots x_n\}$$
$$\{X_{k+1}\} = \left\{ x'_1 = \frac{(x_1 + x_2)}{2}, \dots, x'_{\frac{n}{2}} = \frac{(x_{n-1} + x_n)}{2} \right\}$$

Then at each transformation we compute σ_k^2 and $\gamma_k(h) = \mathrm{cov}(\{X_k\}_i, \{X_k\}_j) = \mathrm{cov}(\{X_k\}_i, \{X_k\}_{i+h})$ with $h = |i - j|$. These blocking transformations proceed until we end up with a sample of size of 2. The aforementioned quantities σ_k^2 and $\gamma_k(h)$ enter in the following equation

$$\sigma_\mu^2 = \frac{\sigma_k^2}{n_k} + \frac{2}{n_k} \sum_{h=1}^{n_k - 1} \left(1 - \frac{h}{n_k}\right) \gamma_k(h) = \frac{\sigma_k^2}{n_k} + e_k \quad (21)$$

which is simply a rewriting of Eq. 5. Here it is expressed the fact that the variance on an average value does not simply correspond to the sample variance divided by the amount of points in the set, but it is also affected by an error e_k accounting for the correlation. For each k we will thus obtain a new estimate for σ_μ, which will be closer and closer to the true value as k increases. In fact, it has been proved [4] that e_k it can be made negligible by performing a sufficiently high number of blocking transformations. Furthermore, a method to achieve the minimum k that makes e_k negligible before σ_k starts having an erratic behaviour is still provided in [4].

4 CODE STRUCTURE

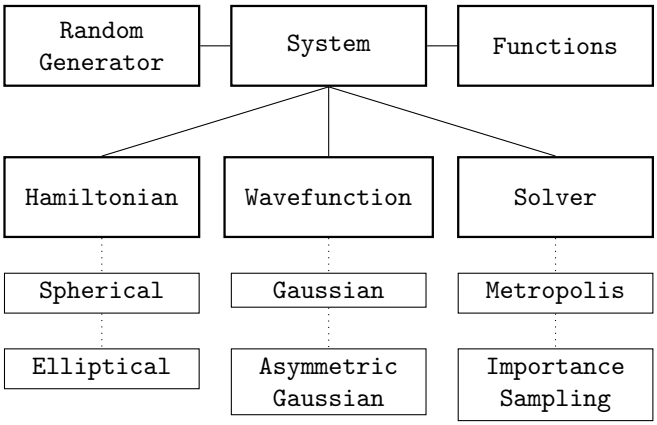

Figure 1: Classes diagram and hierarchy. Elliptical and spherical are Hamiltonian's subclasses, Gaussian and AsymmetricGaussian are Wavefunction's sublclasses, Metropolis and ImportanceSampling are Solver's sublasses.

We chose C++ as the main programming language to implement the aforementioned methods for this project. This allowed us to keep a good compromise between computational speed and abstraction. Some python algorithms were adopted for post-analysis operations and plots construction. The C++ algorithm has been fully object oriented making the code modular, in the sense one can in principle reuse the code for other purposes by only replacing some pieces and keeping the general structure. All the implemented classes can be categorized as follows:

- `System`: contains information on the system as a whole (includes for example the collection of particles) and provides a reference point for communication between the other classes;
- `Solvers`: provide different tools to perform the actual VMC run and evaluate the energy of the system for a chosen Hamiltonian and a selected wavefunction on different complexity levels, depending on the needs;
- `Wavefunctions`: the wavefunction describing the whole system is implemented here, together with the possibility of adjusting the parameters entering in its definition;
- `Hamiltonians`: this includes functions for the evaluation of the local energy;
- `Functions`: this includes some ad-hoc implemented functions that combine elements of the previously cited classes and perform specific tasks requested for the project;
- `RandomGenerator`: it contains functions for the generation of uniform and gaussian distributed random numbers.

The code has been documented via Doxygen and the documentation, together with the code itself, can be found at the GitHub link provided at the end of the report, hence hereby we discuss only some of the peculiar points

of the implementation. The tests conducted to verify the solidity and the correctness of the results provided by the code will be described in the next section.

For the sake of clarity, the below section of pseudo-code describes the structure that we gave to our implementation.

```
#include <defined_classes>

/* Select between parallelized and standard
execution */
bool parallel;

// CREATE OBJECTS
System system(int dimension, int N_particles);

Hamiltonian hamiltonian(&system);
//spherical or elliptical

Wavefunction wavefunction(&systems);
// gaussian or antisymmetric gaussian

Solver solver(&system);
// metropolis or importance

Functions functions(&system);
RandomGenerator randomgenerator(&system);

// set the attributes of the system
system.setHamiltonian(&hamiltonian);
system.setWavefunction(&wavefunction);
system.setSolver(&solver);
system.setRandomGenerator(&randomgenerator);

// perform tasks
functions.task();
```

4.1 SINGLE-PARTICLE WAVEFUNCTION EVALUATION

According to what previously discussed, for every VMC step we are in need to perform an acceptance test for the proposed move by evaluating a ratio of wavefunctions. Eq. 4 together with the fact that for each VMC step only one particle is moved, one infers that the acceptance test can be carried out by evaluating only those components in the wavefunction that include information on the moved particle. More in detail, in the non-interacting case if the wavefunction at the step k of the simulation is

$$\Psi_T^k \equiv \Psi(\boldsymbol{R}(t_k)) = \prod_{i=1}^{N} g(\alpha, \boldsymbol{r}_i(t_k)) \equiv \prod_{i=1}^{N} g_i^{(k)}$$

then, if at the step $k+1$ we modify the position of the particle n, the wavefunction becomes

$$\Psi_T^{k+1} = g_n^{(k+1)} \prod_{\substack{i=1 \\ i \neq n}} g_i^k$$

Thus the ratio between the wavefunctions appearing in Eq. 15 and Eq. 17 reduces to $|g_n^{(k+1)}|^2/|g_n^{(k)}|^2$, allowing a reduction in the computational time.

4.2 RELATIVE POSITION BETWEEN PARTICLES IN THE INTERACTING CASE

The interacting case leads to way more complicated expressions for the wavefunction and the local energy; therefore, finding a clever way to reduce as much as possible the computational effort, is not so trivial as in the non-interacting case. However we noticed the repetitive demand for the same quantities, like the relative positions and relative distances between particles, in different expressions. Hence, we stored those values in proper matrices in order to avoid repetitive re-evaluations of those high time-demanding operations once that the position of the particles is set after a simulation step. The ij-th element of the distance matrix D_{ij} is the distance between the particle i and particle j: here one can note that this matrix is symmetrical, hence only one half of the off-diagonal values must be computed. In a similar way the ij-th element of the relative position matrix $R_{ij} = \boldsymbol{r}_i - \boldsymbol{r}_j$ is the relative position, in cartesian coordinates, of the particles i with respect to to the particle j: this matrix is antisymmetrical, hence, also in this case, only half of the off-diagonal elements can be computed. The matrices are built and initialised at the beginning of the program, and after one particle is moved, only the corresponding rows and columns in the matrices are updated.

4.3 CODE PARALLELIZATION

The code has been parallelized via OpenMP. The strategy we adopted consisted in creating a copy of the system for every thread, each one with the same settings, and the selected number of steps for a single MC run was equally split into the threads. Each parallel simulation was thus carried with a reduced number of steps and data generated from each thread was saved into different files, successively collected together in a python environment and the analysis was carried on the whole dataset. The computational time improvement was analyzed on a 4-cores computer, observing a reduction factor in the simulation time which laid between 3.0 and 3.5.

A slightly different approach regards the gradient descent process, for which an independent research was launched on every thread, each of them providing us at the end with a different estimate of α_{GS} for the same configuration of the system.

5 RESULTS

The analysis of the results produced by our code passed through a series of intermediate and progressive steps, continuously checking the outcomes obtained via the simulation with the analytical formulas, fortunately available for the non-interacting case. We will now proceed with the presentation of the results derived within the context of our project, enlightening the features of any performed simulation and the methods to validate the solidity and efficiency of the code. First we will focus on the runs carried out in non-interacting case both with the brute-force Metropolis algorithm and the importance sampling, proceeding then towards a more detailed statistical analysis through the blocking method. Then the interaction between particles will be introduced, together with the gradient descent method for the research of the α which minimizes the energy and then concluding with the one-body density evaluation.

5.1 NON-INTERACTING SYMMETRICAL CASE

5.1.1 BRUTE-FORCE METROPOLIS ALGORITHM

At first we chose a spherical potential trap with a simple gaussian trial wavefunction, these obtained by setting $a = 0$ and $\beta = 1$ into Eq. 4 and adopting the potential described in Eq. 2. In order to obtain a numerical estimation of the ground state energy of the system we first adopted the so called Brute-Force Metropolis algorithm, previously introduced. We performed different simulations varying many parameters of the system and before each of these runs the system was submitted to 10^5 thermalization steps. This choice has been preserved for each simulation presented from now on in the report. In principle we could have reduced the number of thermalization steps for the most simple configurations, involving for example 1 particle in a 1-dimensional system. However, since the most time-demanding operation is the evaluation of the local energy (not performed for the thermalization steps), the computational time for the initialization is considerably low, even with such an amount of steps and a large number of particles. We notice also that with the highest number of particles adopted for the system ($N = 500$), within a 10^5-steps thermalization process each of them will on average be moved 200 times, which we retained sufficient for the considered Markov chain to converge to the stationary PDF.

Table 1 reports the results of the analysis of the system's dependence on the number of particles and dimension. The value of α was kept fixed at $\alpha = 0.5$. The corresponding results obtained using the numerical method to evaluate the second derivative are reported in Table 2. One can immediately notice a reduction in the precision and a significant increase in the simulation time with the numerical approach, hence leaving the only purpose of this second method to perform preliminary tests and qualitative studies. Proceeding, Figure 2, shows the value of the ground state energy of the system as a function of the parameter α for a particular configuration of the simulation settings, that is $N = 10$ particles, 3 dimensions, 2^{21} steps and Metropolis step-size $r_{step} = 1.0$. One can immediately notice that the function is convex, which guarantees the existence of one unique minimum point, in accordance to what discussed in the context of the gradient descent method. In this simple case, the convexity of the energy curve as a function of α was expected, as it can be derived from the analytical form available in Eq. 9. Some selected results for different α values are reported also in Table 5, referring to simulations performed again for a $3D$ system with $N = 10$ particles and $N_{steps} = 2^{21}$. This table contains also a comparison between the variance calculated as $\sigma_E^2 = (\langle E_L^2\rangle - \langle E_L\rangle^2)/N_{steps}$ and the variance calculated via the blocking method as explained in Section 3.6.

N_{part}	N_{dim}	$\langle E_L\rangle$	σ_E	t_{CPU} $[s]$	acc.ratio
1	1	0.50000	0.00000	0.250	0.7293
1	2	1.0000	0.00000	0.277	0.5958
1	3	1.5000	0.00000	0.293	0.5053
10	1	5.0000	0.00000	0.397	0.7290
10	2	10.000	0.00000	0.426	0.5956
10	3	15.000	0.00000	0.453	0.5047
100	1	50.000	0.00000	1.967	0.7287
100	2	100.00	0.00000	2.084	0.5950
100	3	150.00	0.00000	2.388	0.5043
500	1	250.00	0.00000	20.695	0.7293
500	2	500.00	0.00000	22.008	0.5961
500	3	750.00	0.00000	21.776	0.5051

Table 1: The table reports the results of the simulations performed in the non-interacting case using a simple gaussian trial wavefunction and a brute-force Metropolis sampling method with $N_{therm} = 10^5$, $N_{steps} = 2^{21}$, step-size set to $r_{step} = 1.0$ and constant value for $\alpha = 0.5$. The local energy is evaluated exploiting the analytical formula and σ_E is the error on $\langle E_L\rangle$ estimated from the standard deviation of of the E_L samples generated during the run.

N_{part}	N_{dim}	$\langle E_L\rangle$	σ_E	t_{CPU}	acc.ratio
1	1	0.5000000000	7e-10	0.497	0.7290
1	2	1.000000000	1e-9	0.631	0.5949
1	3	1.500000000	2e-9	0.799	0.5050
10	1	5.000000006	2e-9	2.217	0.7296
10	2	10.000000000	3e-9	2.975	0.5960
10	3	15.000000010	5e-9	4.035	0.5051
100	1	49.999999913	7e-9	20.71	0.7287
100	2	99.99999991	1e-8	28.49	0.5957
100	3	149.99999965	2e-8	37.02	0.5057
500	1	250.99999950	1e-8	237.05	0.7298
500	2	500.00000067	2e-8	347.24	0.5952
500	3	749.99999921	3e-8	413.98	0.5045

Table 2: The table reports the results of the simulations in the non-interacting case using a simple gaussian trial wavefunction and a brute-force Metropolis sampling method with $N_{therm} = 10^5$, $N_{steps} = 2^{21}$, step-size set to $r_{step} = 1.0$ and constant value for $\alpha = 0.5$. The local energy is evaluated exploiting the numerical second derivative and σ_E is the error on $\langle E_L\rangle$ estimated from the standard deviation of the E_L samples generated during the run.

5.1.2 IMPORTANCE SAMPLING

The immediate improvement of this simple model was the inclusion of the importance sampling. This allowed us to bias the random walk in the configuration space according to the probability distribution, leading thus to a higher number of accepted steps in the Monte Carlo run. As for the Brute Force Metropolis, results for fixed $\alpha = 0.5$ obtained with the analytical and numerical evaluation of the local energy are reported respectively in Table 3 and Table 4. The estimated energy as a function of α for a $3D$ system populated by 10 particles is reported in Figure 3 with some selected results in Table 5. All the simulations cited above have been carried out with 2^{21} Monte Carlo steps and $\delta t = 0.1$.

Even if simple and primitive, this model allowed us to test many of the fundamental parts of the code and made us understand how to fine-tune some parameters involved in the problem. For example, we analysed the effects of the time-step length δt in the importance sampling case, concluding that a good choice for the parameter could be $\delta t = 0.1$. Our choice is justified by the fact that this value of δt provides us both with a relatively high speed in the exploration of the space of configurations and a high acceptance ratio. In fact, it make large steps more likely to be proposed for the particles (according to Eq. 16), still having number of accepted moves close to the actual number of performed steps. Both these aspects can be appreciated in Figure 4 and Figure 5: the former shows the acceptance ratio as a function of δt, the latter demonstrates that a smaller value for the time-step introduces more correlation between data. For the realization of this second graph the parameter α was set to 0.4 in order to introduce some statistical fluctuations in the data, otherwise absent in the case of $\alpha = 0.5$.

N_{part}	N_{dim}	$\langle E_L \rangle$	σ_E	t_{CPU}	acc. ratio
1	1	0.50000	0.00000	0.373	0.9928
1	2	1.0000	0.00000	0.385	0.98877
1	3	1.5000	0.00000	0.432	0.9857
10	1	5.0000	0.00000	0.714	0.9929
10	2	10.000	0.00000	0.792	0.9889
10	3	15.000	0.00000	0.793	0.9858
100	1	50.000	0.00000	4.197	0.9929
100	2	100.00	0.00000	4.378	0.9887
100	3	150.00	0.00000	4.339	0.9856
500	1	250.00	0.00000	36.624	0.9929
500	2	500.00	0.00000	39.380	0.9888
500	3	750.00	0.00000	39.965	0.9858

Table 3: The table reports the results of the simulations in the non-interacting case using a simple gaussian trial wavefunction and the importance sampling method with $N_{therm} = 10^5$, $N_{steps} = 2^{21}$, $\delta t = 0.1$ and constant value for $\alpha = 0.5$. The local energy is evaluated exploiting the analytical formula and σ_E is the error on $\langle E_L \rangle$ estimated from the standard deviation of the E_L samples generated during the run.

N_{part}	N_{dim}	$\langle E_L \rangle$	σ_E	t_{CPU}	acc.ratio
1	1	0.5000000027	7e-10	0.761	0.9928
1	2	1.000000000	1e-9	0.961	0.9888
1	3	1.500000004	2e-9	1.332	0.9856
10	1	4.999999937	2e-9	3.311	0.9928
10	2	10.000000039	3e-9	5.772	0.9887
10	3	15.000000053	5e-9	8.097	0.9857
100	1	49.999999990	7e-9	31.444	0.9929
100	2	99.99999966	1e-8	55.850	0.9888
100	3	150.00000048	2e-8	85.201	0.9857
500	1	250.00000078	1e-8	323.70	0.9927
500	2	500.00000208	2e-8	540.58	0.9886
500	3	749.99999858	3e-8	877.82	0.9859

Table 4: The table reports the results of the simulations in the non-interacting case using a simple gaussian trial wavefunction and the importance sampling method with $N_{therm} = 10^5$, $N_{steps} = 2^{21}$, $\delta t = 0.1$ and a constant value of $\alpha = 0.5$. The local energy is calculated exploiting the numerical evaluation of the second derivative and σ_E is the error on $\langle E_L \rangle$ estimated from the standard deviation of the E_L samples generated during the run.

5.1.3 STATISTICAL ANALYSIS THROUGH BLOCKING METHOD

As previously stated, Table 5 contains estimations of the error affecting $\langle E_L(\alpha) \rangle$ obtained directly from the variance provided by the VMC simulations and the one obtained with the blocking technique. The python script that provided us with these latter results is based on the theoretical background described in Section 3.6. The consistency of the mentioned script was also tested, in particular for what concerns the convergence of the estimate for σ_μ^2 (see Eq. 21) to a finite value after a sufficiently high number of blocking iterations. In Figure 6 we present the values of σ_k after each iteration of the mentioned method: the results are derived from a simulation performed with 2^{23} steps on a $3D$ system containing $N = 10$ non-interacting particles inserted in a spherical potential. The sampling was performed through the Metropolis-Hastings algorithm using $\delta t = 0.1$ and the variational parameter was again set to $\alpha = 0.4$. As expected, we can observe that σ_k reaches a plateau for sufficiently high values of k, and then an erratic behaviour

Metropolis				
α	$\langle E_L \rangle$	σ_E	σ_B	acc.ratio
0.3	17.000	0.001	0.02	0.60313
0.4	15.3768	0.0006	0.007	0.54918
0.5	15.0000	0.0000	0.0000	0.50533
0.6	15.2492	0.0005	0.006	0.46724
0.7	15.8461	0.0009	0.01	0.43341
Importance Sampling				
α	$\langle E_L \rangle$	σ_E	σ_B	acc.ratio
0.3	16.9924	0.001	0.02	0.99343
0.4	15.3826	0.0006	0.007	0.98982
0.5	15.0000	0.0000	0.0000	0.98587
0.6	15.2471	0.0005	0.005	0.98123
0.7	15.8468	0.0009	0.008	0.97630

Table 5: Results of the simulations for different values of the variational parameter α. Here σ_E is the error on $\langle E_L \rangle$ estimated from the standard deviation of the E_L samples generated during the run, while σ_B comes from the blocking analysis. For both the brute-force Metropolis case and the importance sampling case we used 10 particles in 3 dimensions, $N_{therm} = 10^5$ and $N_{steps} = 2^{21}$. The chosen step-size for the Metropolis algorithm was $r_{step} = 1.0$ and the time step-size for the Importance Sampling was $\delta t = 0.1$.

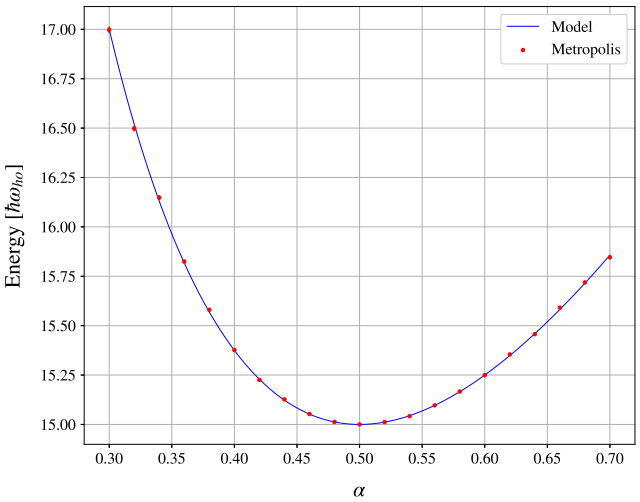

Figure 2: The figure reports the energy estimates as a function of α generated by various VMC simulations exploiting the brute-force Metropolis algorithm with $N_{therm} = 10^5$, $N_{steps} = 2^{21}$ and $r_{step} = 1.0$. The curve is referred to a $3D$ system with $N = 10$ non-interacting particles. For some of the points the errorbar is too small to be seen.

enters into play, this due to the progressive reduction of the amount of points on which the variance is estimated.

5.2 INTERACTING CASE

Subsequently we moved on to the interacting and asymmetrical case with trial wavefunction given by Eq. 4 and Hamiltonian described in Eq. 1 and Eq. 3. At first we compared results with those obtained from the symmetrical case, again to test the solidity of our code: this could be done by setting $\beta = 1$, $w_z = w_{ho} = 1$ and the s-wave scattering length a to 0. We verified that the results were coincident with those obtained in the non-interacting case, using both the Brute-force

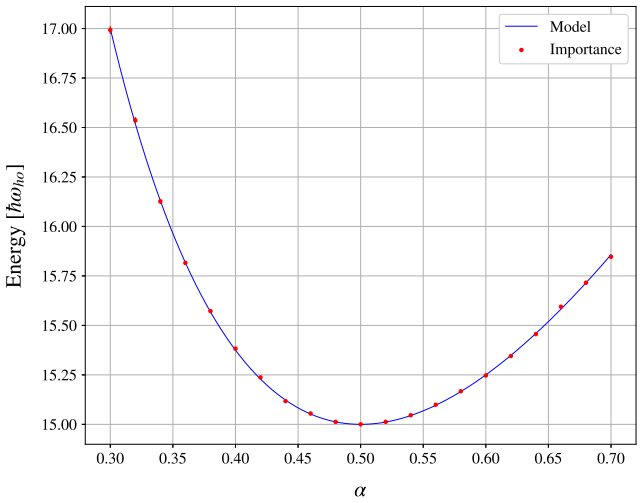

Figure 3: The figure reports the energy estimates as a function of α generated by various VMC simulations exploiting the importance sampling with $N_{therm} = 10^5$, $N_{steps} = 2^{21}$, $\delta t = 0.1$. The curve is referred to a $3D$ system with $N = 10$ non-interacting particles. For some of the points the errorbar is too small to be seen.

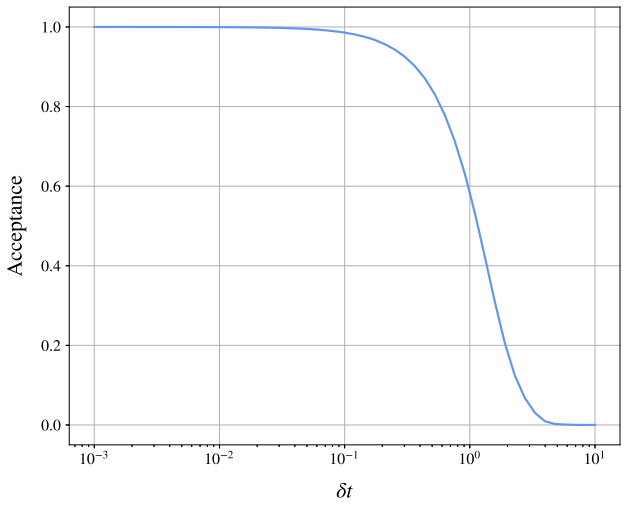

Figure 4: Acceptance ratio as a function of the time-step δt in the context of VMC simulation performed with the importance sampling technique. Each run is conducted with $N_{therm} = 10^5$, $N_{steps} = 2^{21}$ and involves a system of 10 non-interacting particles in 3-dimensions with $\alpha = 0.5$.

Metropolis algorithm and the importance sampling (see Appendix D). From now on we will omit to specify the sampling algorithm used for each simulation, since all the next results were derived exploiting the Metripolis-Hastings one: this choice was driven by the fact that the this technique provides the combination of a higher acceptance ratio and better statistical sampling of the data. Furthermore, the variance on the energy value provided by each VMC was estimated through the blocking analysis. From now on, this information will be omitted too.

After the aforementioned consistency check we set the parameters' values $\omega_z = \beta = 2.82843$ and $a = 0.0043$ and ran the simulations for $N = 10, 50, 100$ particles while varying α. The simulations for 10 and 50 particles

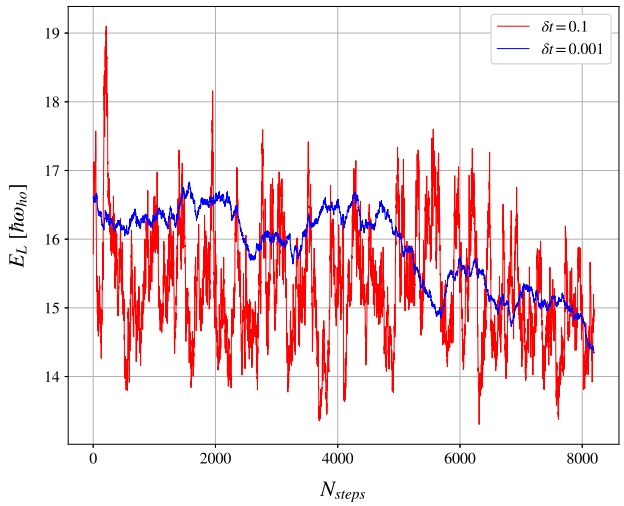

Figure 5: Local energy evaluated in each step of a VMC simulation after $N_{therm} = 10^5$. A $3D$ system populated by 10 non-interacting particles inserted in a spherical potential is considered here. The image shows two curves for the chosen parameter $\delta t = 0.001$ and $\delta t = 0.1$ used for the importance sampling method. The variational parameter α is set as constant to 0.4 to introduce some statistical fluctuations in the data, absent for $\alpha = 0.5$.

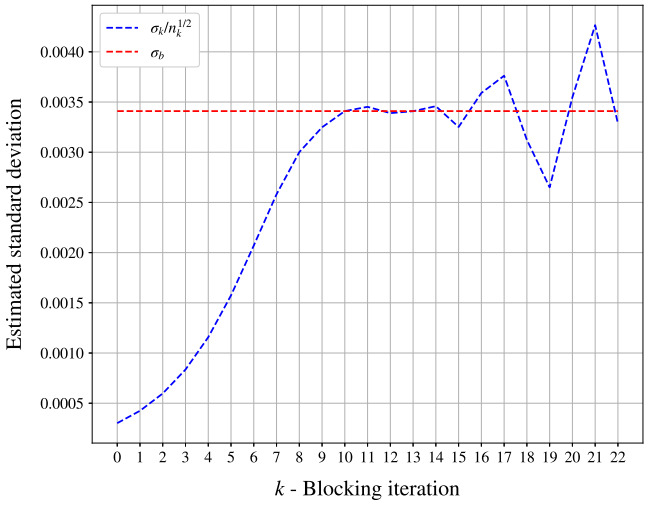

Figure 6: The figure shows the behaviour of the variance estimation provided by the blocking technique implemented in a python script. The simulation was performed with $N_{therm} = 10^5$ and $N_{steps} = 2^{23}$ on a non-interacting $3D$ system characterized by $\alpha = 0.4$ and populated by 10 particles. The Metropolis-Hastings algorithm with $\delta t = 0.1$ was adopted for the sampling.

were carried out using 2^{21} steps, while in the case of 100 particles the high computational time imposed us a reduction of the number of steps to 2^{19}. The parameter $\delta t = 0.1$ was the same for every run and the corresponding results are reported in Table 6. Furthermore, as an example the results obtained for 10 particles are reported in Figure 7, compared with their homologous obtained in the non-interacting symmetrical case. Figure 8 reports a comparison between the energy per particle as a function of the variational parameter α obtained for the three different values of N.

$N = 10$			
α	$\langle E_L \rangle$	σ_B	acc.ratio
0.3	27.59	0.02	0.98185
0.4	24.969	0.008	0.97230
0.5	24.3991	0.0003	0.96178
0.6	24.822	0.006	0.94964
0.7	25.83	0.01	0.93758
$N = 50$			
α	$\langle E_L \rangle$	σ_B	acc. ratio
0.3	142.8	0.1	0.97840
0.4	129.81	0.04	0.96866
0.5	127.287	0.006	0.95759
0.6	129.90	0.03	0.94583
0.7	135.54	0.06	0.93361
$N = 100$			
α	$\langle E_L \rangle$	σ_B	acc. ratio
0.3	296.8	0.4	0.97610
0.4	271.2	0.1	0.96336
0.5	266.51	0.07	0.93323
0.6	272.8	0.2	0.94392
0.7	285.1	0.2	0.92906

Table 6: The table reports the results of the simulations for different values of the variational parameter α for a $3D$ system constituted by $N = 10,\ 50,\ 100$ particles. The simulations were performed exploiting the Metropolis-Hastings algorithm with $\delta t = 0.1$, $N_{therm} = 10^5$ and $N_{steps} = 2^{21}$ for $N = 10,\ 50$, while and $N_{steps} = 2^{19}$ for $N = 100$.

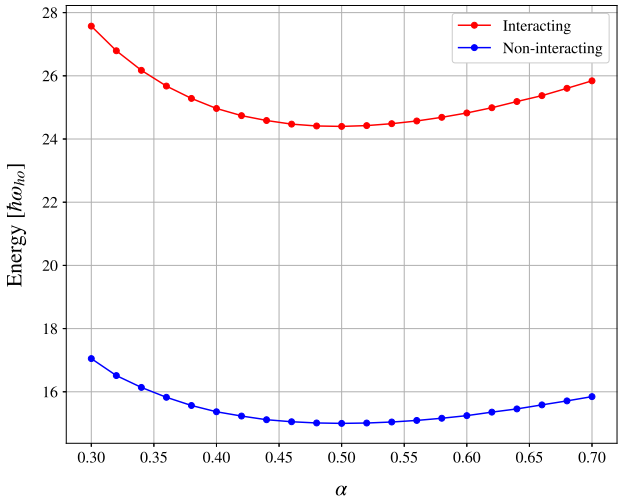

Figure 7: The figure reports the energy estimates as a function of α generated by various VMC simulations exploiting the importance sampling algorithm with $N_{therm} = 10^5$, $N_{steps} = 2^{21}$, $\delta t = 0.1$. The curves are referred to a $3D$ system populated respectively by 10 non-interacting particles in a spherical potential and 10 interacting particles in an elliptical potential. The errorbars are too small to be seen.

5.3 GRADIENT DESCENT

When considering the interaction between particles, a possible proof of the convexity of the function describing $\langle E_L \rangle$ as a function of α is non-trivial, thus in principle we can not state with absolute confidence that the curve

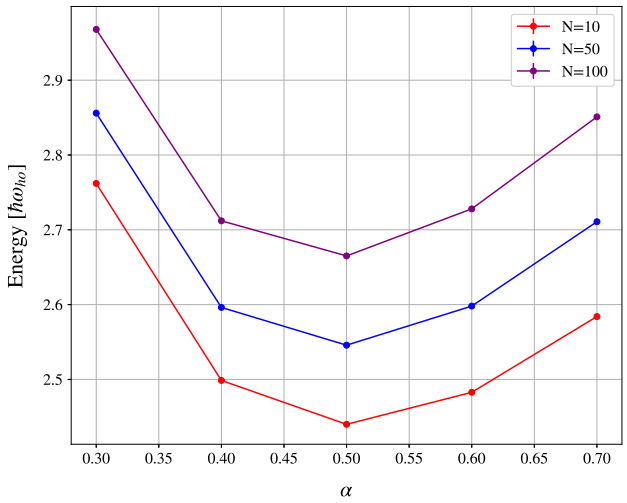

Figure 8: The figure reports the average energy per particle as a function of α generated by various VMC simulations exploiting the importance sampling algorithm with $N_{therm} = 10^5$ and $\delta t = 0.1$. The curves are referred to a $3D$ system populated respectively by 10, 50 and 100 interacting particles in an elliptical potential. The number of employed steps was 2^{21} for $N = 10$ and $N = 50$, while 2^{19} for $N = 100$.

is convex. However, as we can see from the latest figures, the behaviour of $\langle E_L \rangle$ as a function of α still encourages us to exploit the gradient descent method to research for α_{GS} which minimizes the energy of the system. Before proceeding with the actual investigation in the interacting case, we tested the correct functioning of our code by applying it to a non-interacting system. In this simple case, it is well known that the optimal value of the variational parameter is $\alpha_{GS} = 0.5$ and this result is independent of the number of particles and dimensionality. Therefore we expect to reach a convergence to this value whatever the adopted configuration. In particular, when considering particles without interaction we have at our disposal the analytical expression for the total energy of the system (see Eq. 9) and the convexity of it is easy to prove. This ensures us that, if the method is capable in reaching a convergence for a sufficiently small value of the so-called learning rate γ appearing in Eq. 19, the gradient descent will lead to a proper estimation of α_{GS}. Table 7 reports the results obtained with the application of the gradient descent method to a 3-dimensional system of 1, 10 and 100 non-interacting particles. Each VMC run was performed with $\delta t = 0.1$ and 10^5 thermalization steps, followed by another 10^5 for the actual evaluation of the quantities reported in Eq. 20. The learning rate was properly chosen for each configuration and the tolerance on the derivative was fixed to $\varepsilon = 10^{-8}$ with no limit on the maximum number of iterations. The stability of the algorithm was also tested by performing the descent starting from two different guesses for the variational parameter, namely $\alpha_0 = 0.4$ and $\alpha_0 = 0.6$.

Switching now to consider the interaction between particles, in this case the problem was less trivial, since the optimal value of the variational parameter depends also on the chosen number of particles and on the starting value of α. Moreover, contrarily to the previous case, the exact value of α_{GS} is unknown and thus our research must be even more rigorous. However, the major limitation to precision here is constituted by the evaluation of the derivative of $\langle E_L \rangle$ with respect to α. In fact, running a simple gradient descent with 10 particles, $\alpha_0 = 0.4$ and $\gamma = 10^{-3}$ we observed that after some steps the derivative

N	γ	α_0	α_{best}	N_{conv}
1	1e-2	0.4	0.500	285
1	1e-2	0.6	0.500	295
10	1e-2	0.4	0.500	21
10	1e-2	0.6	0.500	23
100	1e-3	0.4	0.500	24
100	1e-3	0.6	0.500	26

Table 7: Convergence analysis for the energy minimization process performed on a $3D$ non-interacting system of 1, 10, 100 particles in a spherical potential. The simulations were performed with 10^5 steps for both thermalization and actual run, adopting the importance sampling with $\delta t = 0.1$, adjustable γ and tolerance $\varepsilon = 10^{-8}$. N_{conv} refers to the number of gradient descent steps to get to convergence and α_0 is the initial guess selected for the variational parameter.

N	α_0	α_{best}	γ_{min}	$N_{steps-max}$	N_{conv}
10	0.4	0.49751	1e-3	1e5	74
10	0.6	0.49751	1e-3	1e5	74
50	0.4	0.48881	1e-5	2e5	21
50	0.6	0.48939	1e-5	2e5	21
100	0.4	0.481	1e-5	5e5	16
100	0.6	0.483	1e-5	5e5	11

Table 8: Convergence analysis for the energy minimization process performed on a interacting system of 10, 50 and 100 particles in a elliptical potential. A detailed description of the algorithm implemented for reaching these results is described in Paragraph 5.3. Here γ_{min} and $N_{steps-max}$ are the minimum learning rate and maximum number of VMC steps used to perform a single run, while N_{conv} is the iteration number at which the algorithm was stopped. Each run was performed employing the importance sampling with $\delta t = 0.1$ and $N_{therm} = 10^5$.

N	α^{GS}	E	σ_B	acceptance
10	0.49751	24.39843	0.00003	0.96234
50	0.4891	127.48	0.07	0.94926
100	0.482	266.38	0.04	0.95036

Table 9: Results obtained for the ground state energy of a system of interacting particles in an elliptical potential. The final estimates were derived through VMC runs using the importance sampling method with $\delta t = 0.1$ and 10^5 thermalization steps. We selected $N_{steps} = 2^{27}$, $N_{steps} = 2^{25}$ and $N_{steps} = 2^{23}$ respectively for the systems in each of the 3 configurations with 10, 50 and 100 particles. The simulations were carried out exploiting the optimal variational parameter found after the gradient descent minimization and reported in Table 8.

started oscillating between positive and negative values, always similar in magnitude between each others. The stochastic evaluation of this derivative enters here: in fact, even increasing the number of steps for each simulation or reducing the learning rate, we did not observe a further stabilization of the derivative. Its values were always oscillating around zero, without decreasing in magnitude. We therefore retained to implement the method as follows: in the parallelized version of the code, each thread hosted a full gradient descent and for each number of particles we performed a first descent by adopting $\gamma = 10^{-3}$ for a rapid convergence to a proximity of the result we aim at. When the aforementioned derivative started showing its oscillatory behaviour, a smaller γ was selected and combined with a higher number of steps for each VMC simulation, this with the hope of increasing the precision in the evaluation of the derivative. If with these two improvements the algorithm still provided us with the above described oscillatory behaviour, the descent was interrupted and the selected α_{GS} is the average between the last values provided by each of the cores. Again the initial bunch of single runs were performed for $N = 10$, 50, 100 using 10^5 VMC steps and $\delta t = 0.1$, increasing then the number of steps as described above. For each N the research of the optimal variational parameter was performed starting from 2 initial guesses. Our analysis is resumed in Table 8.

Once that the values of α_{GS} has been achieved for any selected configuration, we proceeded with a final VMC run to estimate the ground state energy for each apparatus. The results reported in Table 9 are referred to simulations performed with $\delta t = 0.1$ and with a number Monte Carlo steps varying with the number of particles involved in the system (this choice was caused by the huge amount of time required for the simulations involving a highly populated system). We selected 2^{27} MC steps for $N = 10$, 2^{25} MC steps for $N = 50$ and 2^{23} MC steps for $N = 100$.

5.4 ONE-BODY DENSITY

With the energy-minimizing variational parameter at our disposal, we proceeded with the last step of our analysis, that is the previously introduced evaluation of the one-body density. Figure 9 shows the results obtained with the procedure described in section 3.5 applied to a system populated by 10 particles. For the construction of the curves reported below, the trial wavefunction of Eq. 4 was considered both with and without the Jastrow factor, varying the value of the parameter a to amplify the effects introduced by the interaction term. In Figure 10 we represent also the spatial distribution of the particles along the x and z directions, in order to show the effects introduced by the modifications of the trap shape. The curves are referred to a system populated respectively by 10, 50 and 100 particles. Each of the simulations performed for this section were carried out with $\delta t = 0.1$ and 2^{23} steps, setting the α value to those reported in Table 9.

6 DISCUSSION ABOUT THE RESULTS

6.1 NON-INTERACTING CASE

We start now the discussion about the results obtained for systems of non-interacting particles in a spherical potential. This simple case is analytically solvable, in the sense that the analytical expressions for all the quantities involved in this project can be found. Thus in principle a VMC approach could be avoided, but we still adopted this system for the sake of testing the implemented algorithms that would have been applied in a second moment to the interacting case.

BRUTE-FORCE METROPOLIS ALGORITHM

It is well known that the value of the variational parameter minimizing the energy of a non-interacting system considered in this project is $\alpha_{GS} = 0.5$, independently on the degrees of freedom involved in the system. This is furthermore validated by Table 1 and Table 2, reporting data produced with the brute-force Metropolis algorithm. As previously stated, the simulations exploiting the numerical approach were performed just for a comparison in precision and CPU time with those performed

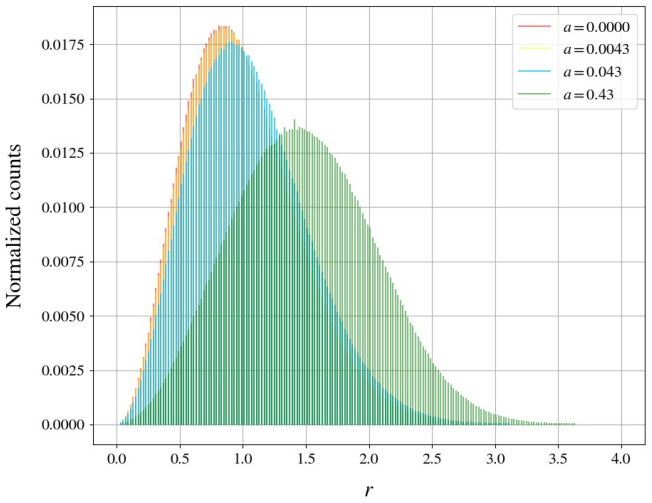

Figure 9: One-body density as a function of the radial distance from the center of the chosen reference frame for a $3D$ system with $N = 10$ particles described by an asymmetric gaussian wavefunction and inserted in a elliptical potential. The histograms are built for different values of a in order to increase the effects introduced by the Jastrow factor in the wavefunction. The simulations were carried our exploiting the importance sampling with $N_{therm} = 10^5$, $N_{steps} = 2^{23}$, $\delta t = 0.1$ and the α value deriving from the minimization of the energy (reported in Table 9). Notice that the curves for $a = 0$ and $a = 0.0043$ are almost perfectly overlapped.

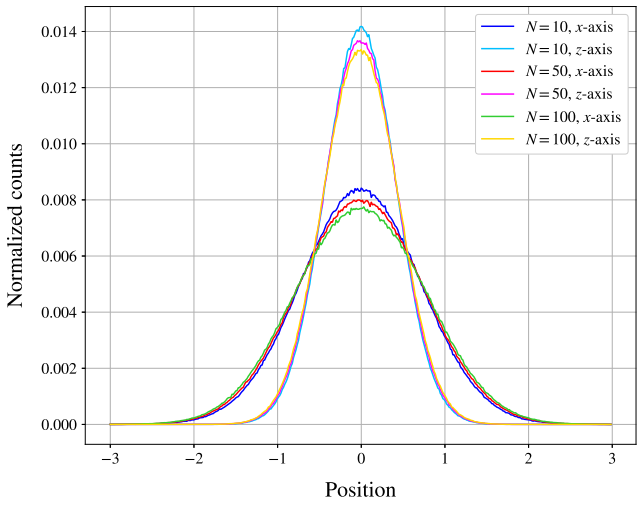

Figure 10: Particles' spatial distribution along the coordinates x and z for a system populated respectively by 10, 50 and 100 particles described by an asymmetric gaussian wavefunction and inserted in a elliptical potential. The simulations were carried our exploiting the importance sampling with $N_{therm} = 10^5$, $N_{steps} = 2^{23}$ steps and $\delta t = 0.1$, setting α to the values reported in Table 9.

using analytical formulas. One can notice that the adoption of the numerical approach for the evaluation of the second derivative appearing in Eq. 6 produces less precise results, namely σ_E is non-null, on the contrary of what happens when the local energy is evaluated analytically. Moreover, the CPU time is also drastically increased when the numerical approach is adopted.

ACCEPTANCE RATIO

A more interesting aspect to notice is the behaviour of the acceptance ratio as a function of the dimensionality of the system. While using the brute-force Metropolis algorithm, the step size for generating the new proposal for each coordinate of the particles' position vector was always kept constant for all the possible tested configurations. This reflects on the acceptance ratio, which diminishes when the dimension D of the apparatus grows. In fact, with the just mentioned choice for r_{step} we are implicitly allowing a particle to move at each step up to a distance $r_{step} * \sqrt{D}$ apart from its previous position. As D increases, particles can undergo larger movements and thus they are also more likely to end more distant from the origin, with a higher rejection probability for that move.

IMPORTANCE SAMPLING

The considerations about precision and CPU time apply also to the simulations for $\alpha = 0.5$ performed using the importance sampling algorithm, whose results are reported in Table 3 and Table 4. The computational time spent on the single VMC runs performed here is slightly higher than the corresponding one needed for the analogous simulations employing the brute-force approach. This was expected too, since the importance sampling requires also the Greens' functions to be evaluated apart from the wavefunctions in order to discriminate between acceptance and rejection of a single move. Despite this little inconvenient, the advantages brought by this second algorithm legitimate its implementation. The bias introduced by the drift force term in the proposal for a new move drives the system to high-probability states, leading thus to a higher acceptance ratio and to a better sampling in the high-probability regions of the space of configurations, which are those that more matter for statistics.

STUDY OF α-DEPENDENCE

Once that the most basic features of our code were tested, we switched to a more information-providing case, that is $\alpha \neq 0.5$. Plots appearing respectively in Figure 2 and Figure 3 show the almost perfect overlap between the experimental points and the theoretical curve, remarking again the efficacy of VMC simulations in reproducing the expected behaviour. Furthermore, adopting $\alpha \neq 0.5$ brought us away from the trivial case of null error on the estimated energy of the system. In fact, substituting $\alpha = 0.5$ into Eq. 11, one completely eliminates the dependence on the position of the particles. Thus, in this case E_L sampled along a VMC simulation assumes always the same value. On the contrary, with $\alpha \neq 0.5$ the dependence of the local energy from $\boldsymbol{R}$ is reintroduced, bringing up some more meaningful statistics in the acquired data. This was also the occasion for a first encounter with the error estimation through the blocking method, which allowed us also to access the impact of correlation in the acquired measurements.

CHOICE OF TIME STEP δt

We observed that the choice of the parameter δt impacts not only on the acceptance ratio, but also on the correlation between samples. Combining data from Figure 4 with the column referred to σ_B in Table 5 helped us in tuning the parameter δt for the successive VMC runs. We noticed that choosing a too small δt introduced more correlation in the generated data, as

can be clearly seen in Figure 5. Here we see that for smaller δt particles are less likely to be moved far away from their current position and thus more VMC cycles are needed to reach a sufficient degree of uncorrelation. For these reasons, $\delta t = 0.1$ was chosen for the successive simulations, making large steps more suitable for the particles, still keeping a high acceptance ratio as typical for the importance sampling. The content of Figure 6 is reported to testify the correct behaviour of the python script used for the blocking analysis and is also a good instrument for a better comprehension of its functioning. As expected, the provided error estimate increases at each blocking iteration since always a larger fraction of the correlation term in Eq. 5 is taken into account. After a certain number of iterations, a plateau is reached and then an erratic behaviour comes into play, due to the lack of data on which the variance is being evaluated.

6.2 INTERACTING CASE

The analysis of the non-interacting case acted as a springboard for the introduction of the interaction term between the particles. This new framework involved a system of 1, 50, 100 particles inserted in a elliptical potential and described by an asymmetric gaussian wavefunction multiplied by a Jastrow factor, as illustrated Eq. 4. The investigations in this case became much less trivial, since an analytic solution for the energy of the system as a function of α is not available for any comparison. The simulations became also much more time demanding, due both to the complexity of the analytical formulas included in the code (e.g. Eq. 10) and to the introduction of new computational elements which played a fundamental role in the evaluation of the needed quantities. These factors imposed a much more careful costs/benefits analysis before launching each simulation, especially a compromise had to be found between computational time and precision achieved for the results.

STUDY OF α-DEPENDENCE

Table 6 reports the evaluation of the energy of the system for a bunch of α values. The simulations involving a group of 100 particles were clearly the most demanding, thus looking at the problem from a time-saving perspective we choose to employ less steps for this kind of run. Though, the selected number of steps allowed us to keep a sufficiently low relative error on the energy estimation as a function of α. In any case, data contained in the mentioned table provide a lot of pieces of information for what concerns the features of the new systems. In particular, one can see that the modifications added to both the hamiltonian and the wavefunction obviously changed the α-dependence for the expected energy value. Still, the new α_{GS} which minimizes the energy of the system is once more included in the interval $(0.4, 0.6)$ for every analyzed N. This fact is proved for the case of $N = 10$ by the graph of Figure 7: here we can see the great impact introduced by the modifications applied to the system with respect to the non-interacting case. The two curves show a similar shape, with the clear presence of a minimum around $\alpha = 0.5$, but the energy values related to our new configurations are always higher than those corresponding to a bunch of independent particles. This behaviour is completely reasonable, since now the trap is steeper in the z direction and simultaneously the interparticles distance must be higher than the typical s-wave scattering length a.

AVERAGE ENERGY PER PARTICLE

A graphical representation of the content of Table 6 is presented in Figure 8. Another interesting fact is highlighted here: in the case of non-interacting particles, we experienced that the energy of the system resulted to be simply proportional to the number of particles and the dimensionality of the system, as suggested also in Eq. 9. Here instead the aforementioned plot suggests that the interaction term appeared in the Hamiltonian has introduced a stronger dependence in the energy on the number of particles populating the system. This tendency is also shown in [12]. Unfortunately the article focuses on systems populated by a higher number of particles than those that we employed, thus a more precise comparison was not affordable. Back to our case, if the interaction between the particles is switched off (i.e. one sets $a = 0$) the three curves of the mentioned figure would be almost perfectly overlapping (modulo stochastic fluctuations deriving from the VMC run), while when the interaction is considered they appear as clearly distinguished. This fact is again extremely reasonable, since, considered the repulsive interaction between the particles, increasing the population of the system will lead to a larger average distance from the origin and thus to a larger average energy per particle.

IMPACT OF NUMBER OF PARTICLES ON α_{GS}

After all these preliminary considerations on the interacting system, we passed to the research of the α value which minimizes the energy. Results shown in Table 7 come from the application of the gradient descent method to a non-interacting system with different number of particles: this kind of experiment was used to test the correctness of the code implementation, exploiting again the knowledge of the true value of α_{GS}. For every case presented in the table, the descent proceeded without any issue or strange behaviour up to convergence. Once that the program was launched, the values of $\langle E_L \rangle$ described a monotonic decreasing function towards the minimum, with the corresponding derivatives with respect to α always decreasing in modulus down to the fixed tolerance. A much more interesting and also more demanding task is the research of this parameter in the interacting case, for which the only preliminary knowledge regards the range in which to search for it, namely $\alpha \in (0.4, 0.6)$, as suggested by Figure 8. Contrarily to what happens for the non-interacting case, here we do not have at disposal a straightforward way to possibly demonstrate the convexity of $\langle E_L \rangle$ as a function of α. However, again the shape of the curves reported in Figure 8 encourages us to apply the gradient descent method.

In this much more complicated case, we had to face with a problem arising from the stochastic evaluation of the energy derivative with respect to α. Our initial intent was to make a first descent to reach the proximity of the ground state energy and then proceed by choosing a smaller value of the parameter γ appearing in Eq. 19 in order to perform a finer search. However, the reduction of the learning rate revealed to be insufficient to guarantee the correct continuation of the descent, since the fluctuations in the derivative values were too large. Two different single VMC runs performed in the same conditions and with the same value for α could produce two estimates for $d\langle E_L \rangle / d\alpha$ close to zero, but maybe differing also in the sign. A possible solution to get rid of this inconvenient could consist in increasing the number of steps employed for each simulation along the descent, hoping that this could lead to a better estimation of the derivative. However, again we had to find a compromise between computational time and precision in the results,

so we decided to proceed by implementing the gradient descent as already described in Section 5.3. The results derived from these procedures appear in Table 8: one can notice that the number of gradient descent iterations that were necessary to reach the reported results decreases as the number of particles in the system grows. This is due to the fact that for larger N the magnitude of the fluctuations in the derivative increased, thus limiting the descent to a few steps.

For all the simulations performed in the context of gradient descent, we observed the extreme sensitivity of the method to the initial guess for α, the value of the learning rate and the number of steps performed in each MC simulation.

Table 9 shows that the estimated values of α_{GS} decrease as N increases, confirming again that adding particles to the configuration contributes to move away the system from the non-interacting approximation treated before. The simulations with the final estimates of α_{GS} for 10, 50 and 100 particles populating the systems were finally performed and the corresponding results are still in Table 9. For this kind of simulations the number of steps was higher than the average amount used for the other runs treated in the project, since we aimed to reach an accurate estimation of the ground state energy of the system.

Using a even higher number of particles would have lead to a non-affordable computational time for our devices, but a result for $N = 500$ was found in [13]. In this case the value found was $\alpha_{GS} = 0.475$, confirming the tendency just described for α_{GS}.

ONE-BODY DENSITY

As a final remark, the one-body density plotted in Figure 9 for a bunch of a values tells us about the role played by the repulsive interactions in determining their radial distribution in the system. As the minimum allowed distance between the particles increases, we can see a clear tendency to occupy positions further from the origin of the system. Here we find another evidence of what discussed above about the role played by the repulsive interaction: in principle the particles tend to occupy positions close to the origin of the system, in order to minimize its energy, but here they are forced to have reciprocal distance larger than a. Two of them are thus forbidden to be simultaneously close to the origin. The figure shows that an increase in the value of a leads to an increase in the average distance from the origin, despite the particles have to face with a steeper potential.

The introduction of the elliptical trap caused the particles to be more confined along the z direction, as shown in Figure 10. In addition, this graph constitutes a further confirmation of the fact that adding more and more particles to the system makes it always less comparable to a non-interacting one. As a matter of fact, in this simpler case the average dispersion of the particles along a specific direction wouldn't be influenced by the number of elements appearing in the system, contrarily on what happens when the interaction in taken into account. Here we see again that a larger population leads to a greater dispersion of the particles with respect to the origin of the system. The same statements are reported also in [12], but here the observations are again focused on systems with thousands of particles in them.

7 POSSIBLE FURTHER IMPROVEMENTS

The code has been built since the beginning as a fully object-oriented program, defining thus a clear and well organized structure suitable also for further applications to other systems. New sub-classes can in principle be added to the primary ones, allowing the user to approach other problems in the context of Quantum Mechanics. The speed of the code can also be enhanced by the adoption of libraries pointed towards high performance computing (e.g. `<armadillo>`), but this would probably require heavy modifications to the already implemented structure.

A finer research of the α value which minimizes the energy of the system could be achieved substituting the very rough standard gradient descent with an improved version of it (e.g. stochastic gradient or conjugate gradient), maybe also exploiting some built-in python functions to approximate the Hessian matrix needed for the implementation.

8 CONCLUSIONS

In the first part of the present project we tested the implementation of the code by applying it to describe a system of non-interacting bosons in a spherical trap. Both the VMC simulations based on the analytical and numerical evaluation of the local energy for $\alpha = 0.5$ provided the expected results, the latter suffering a small lack of precision and requiring a much higher CPU time. Moreover, the algorithm exploiting the analytical formula for the local energy provided excellent results also for $\alpha \neq 0.5$, following the expected behaviour described by the theoretical curve of $\langle E_L \rangle$ as a function of α. The comparison of the results coming from the brute-force Metropolis algorithm and the Metropolis-Hastings one lead us to prefer the latter because of the higher acceptance ratio and the better sampling properties provided by it. The time-step was set to $\delta t = 0.1$, which allows for having a high acceptance ratio combined with a larger average step length for the particles. This leads also to a faster exploration of the possible configurations and in a reduction of the correlation between consecutive steps.

Switching to the interacting case, we verified that by setting the parameters ω_z, β and a in order to lead back to the non-interacting system, we achieved a last successful comparison with the known analytical results. The application of the gradient descent provided us with three values for the α minimizing the energy of the system, respectively $\alpha_{10} = 0.49751$, $\alpha_{50} = 0.4891$ and $\alpha_{100} = 0.482$, showing a decrease as the number of particles N increases (this tendency was also confirmed in [13]). The fluctuations in the estimate for the α-derivative of $\langle E_L \rangle$ came into play after a few iterations of the gradient descent, preventing us to reach a higher precision in the estimate of α_{GS}. This precision actually decreases with increasing N, since the magnitude of the fluctuations increases too. The MC simulations performed with the estimated α_{GS} provided us with $\langle E_L \rangle_{10} = 24.39843 \pm 0.00003$, $\langle E_L \rangle_{50} = 127.48 \pm 0.07$ and $\langle E_L \rangle_{100} = 266.38 \pm 0.04$, with the uncertainties estimated through the blocking method. The error affecting the result for 100 particles is surely underestimated: this may be attributable to the smaller amount of steps employed for the final simulation, not allowing for a proper estimation of the error through the blocking method. The modifications to the Hamiltonian and the introduction of the Jastrow factor in the wavefunction influenced also the behaviour of the average energy per particle introducing a dependence on N, contrarily to what observed for independent particles in the non-interacting case. The

hard-sphere potential lead also to an increase of the average distance between the particles and the origin, this fact appearing as a little modification in the shape of the one-body density. Also in this case, the modifications became more and more evident as the population of the apparatus increased. These evidences are confirmed by the results reported in [12], where larger populations are analyzed.

The combination of all these observations manifests the fact that when the interaction between particles is considered, a higher N will make a possible approximation to the non-interacting case always less and less precise.

LINK TO GITHUB REPOSITORY

https://github.com/Matteo294/FYS4411

APPENDIX

A LOCAL ENERGY: ANALYTICAL FORM

This section is devoted to show the detailed analytical derivation of the local energy formula in the general case. All the specific cases can be gotten from the final result reported below by setting properly the parameters of the system. For the sake of notation, we remind that we set $m = \hbar = \omega_{ho} = 1$ and we introduce the variable $\boldsymbol{r}$ as a cumulative variable for $\{\boldsymbol{R}, \alpha, \beta\}$. The trial wavefunction describing the most general configuration for the system is reported in Eq. 4, however to better face the analytical derivation we rewrite it here as:

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

with

$$u(r_{ik}) = \ln(f_{ik}) = \ln\left(1 - \frac{a}{r_{ik}}\right) \tag{22}$$

and

$$g(\alpha, \beta, \mathbf{r}_i) = \exp\left[-\alpha(x_i^2 + y_i^2 + \beta z_i^2)\right] = \phi(\mathbf{r}_i) \tag{23}$$

The Hamiltonian of the system acquires the following form:

$$\hat{H} = \frac{1}{2}\sum_i^N \left(-\nabla_i^2 + x_i^2 + y_i^2 + \omega_z^2 z_i^2\right) + \sum_{i<j}^N V_{int}(\mathbf{r}_i, \mathbf{r}_j)$$

We consider the case of $r_{ij} > a$, $\forall i, j$, then $V_{int}(\mathbf{r}_i, \mathbf{r}_j) = 0$. According to the definition of the local energy, we get

$$\begin{aligned} E_L(\mathbf{r}) &= \frac{1}{\Psi_T(\mathbf{r})}\hat{H}\Psi_T(\mathbf{r}) \\ &= -\frac{1}{2}\sum_k^N \frac{\nabla_k^2 \Psi_T(\mathbf{r})}{\Psi_T(\mathbf{r})} + \sum_k^N \frac{1}{2}\left(x_k^2 + y_k^2 + \omega_z^2 z_k^2\right) \end{aligned} \tag{24}$$

One of the most nasty part of the derivation of the local energy is the evaluation of the kinetic term, which starts from the derivative of the trial wavefunction with respect to the position of the k-th particle:

$$\begin{aligned} \nabla_k \Psi_T(\mathbf{r}) &= \nabla_k \left\{ \left[\prod_i \phi(\mathbf{r}_i)\right] \exp\left(\sum_{j<m} u(r_{jm})\right) \right\} \\ &= [\nabla_k \phi(\mathbf{r}_k)] \left[\prod_{i\neq k} \phi(\mathbf{r}_i)\right] \exp\left(\sum_{j<m} u(r_{jm})\right) + \\ &+ \left[\prod_i \phi(\mathbf{r}_i)\right] \exp\left(\sum_{j<m} u(r_{jm})\right) \nabla_k \left[\sum_{j<m} u(r_{jm})\right] \\ &= [\nabla_k \phi(\mathbf{r}_k)] \left[\prod_{i\neq k} \phi(\mathbf{r}_i)\right] \exp\left(\sum_{j<m} u(r_{jm})\right) + \\ &+ \left[\prod_i \phi(\mathbf{r}_i)\right] \exp\left(\sum_{j<m} u(r_{jm})\right) \left[\sum_{j\neq k} \nabla_k u(r_{jk})\right] \end{aligned} \tag{25}$$

In our specific case this yields to:

$$\begin{aligned} &\nabla_k \Psi_T(\mathbf{r}) = \left[\nabla_k e^{-\alpha(x_k^2 + y_k^2 + \beta z_k^2)}\right] \left[\prod_{i\neq k} e^{\left[-\alpha(x_i^2 + y_i^2 + \beta z_i^2)\right]}\right] \times \\ &\times \left[\prod_{j<m}\left(1 - \frac{a}{r_{jm}}\right)\right] + \left[\prod_i e^{-\alpha(x_i^2 + y_i^2 + \beta z_i^2)}\right] \times \\ &\times \left[\prod_{j<m}\left(1 - \frac{a}{r_{jm}}\right)\right] \left[\sum_{j\neq k} \nabla_k \ln\left(1 - \frac{a}{r_{jk}}\right)\right] \\ &= -2\alpha(x_k, y_k, \beta z_k)\Psi_T(\mathbf{r}) + \sum_{j\neq k} \frac{1}{\left(1 - \frac{a}{r_{jk}}\right)} \frac{a\mathbf{r}_{kj}}{r_{jk}^3} \Psi_T(\mathbf{r}) \\ &= \Psi_T(\mathbf{r}) \left[-2\alpha(x_k, y_k, \beta z_k) + \sum_{j\neq k} \frac{a}{(r_{jk} - a)\, r_{jk}^2} \mathbf{r}_{kj}\right] \end{aligned} \tag{26}$$

To evaluate the second derivative we restart from Eq. 25. Proceeding we get:

$$\begin{aligned} &\nabla_k^2 \Psi_T(\mathbf{r}) = \left[\nabla_k^2 \phi(\mathbf{r}_k)\right] \left[\prod_{i\neq k} \phi(\mathbf{r}_i)\right] \exp\left(\sum_{j<m} u(r_{jm})\right) \\ &+ [\nabla_k \phi(\mathbf{r}_k)] \left[\prod_{i\neq k} \phi(\mathbf{r}_i)\right] \exp\left(\sum_{j<m} u(r_{jm})\right) \times \\ &\times \left[\sum_{j\neq k} \nabla_k u(r_{jk})\right] + [\nabla_k \phi(\mathbf{r}_k)] \left[\prod_{i\neq k} \phi(\mathbf{r}_i)\right] \times \\ &\times \exp\left(\sum_{j<m} u(r_{jm})\right) \left[\sum_{j\neq k} \nabla_k u(r_{jk})\right] + \\ &+ \left[\prod_i \phi(\mathbf{r}_i)\right] \exp\left(\sum_{j<m} u(r_{jm})\right) \left[\sum_{j\neq k} \nabla_k^2 u(r_{jk})\right] \end{aligned}$$

Then dividing by $\Psi_T(\mathbf{r})$ we get:

$$\begin{aligned} &\frac{\nabla_k^2 \Psi_T(\mathbf{r})}{\Psi_T(\mathbf{r})} = \frac{\nabla_k^2 \phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)} + 2\frac{\nabla_k \phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)} \left[\sum_{j\neq k} \nabla_k u(r_{jk})\right] + \\ &+ \left[\sum_{j\neq k} \nabla_k u(r_{jk})\right]^2 + \left[\sum_{j\neq k} \nabla_k^2 u(r_{jk})\right] \\ &= \frac{\nabla_k^2 \phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)} + 2\frac{\nabla_k \phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)} \left[\sum_{j\neq k} \nabla_k u(r_{jk})\right] \\ &+ \left[\sum_{j\neq k} \sum_{m\neq k} \nabla_k u(r_{jk}) \nabla_k u(r_{mk})\right] + \left[\sum_{j\neq k} \nabla_k^2 u(r_{jk})\right] \end{aligned}$$

At this point we rewrite the gradient in spherical coordinates

$$\nabla f = \frac{\partial f}{\partial r}\frac{\mathbf{r}}{r} + \frac{1}{r}\frac{\partial f}{\partial \theta}\hat{\theta} + \frac{1}{\sin\theta}\frac{\partial f}{\partial \phi}\hat{\phi}$$

This choice simplifies the calculations, since the dependence of u is limited to the relative distance between two particles. Applying then the gradient to

$u(r_{jk})$, we get

$$\nabla_k u(r_{jk}) = \nabla_k(\mathbf{r}_k - \mathbf{r}_j)\nabla_{kj} u(r_{kj}) = \frac{\mathbf{r}_{kj}}{r_{kj}}\frac{du(r_{kj})}{dr_{kj}}$$

With the same reasoning, one can get also the expression for the Laplacian of $u(r_{jk})$ starting from

$$\nabla^2 f = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial f}{\partial r}\right) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial f}{\partial\theta}\right) + + \frac{1}{r^2\sin^2\theta}\frac{\partial^2 f}{\partial\phi^2}$$

which in our case reduces to

$$\nabla_k^2 u(r_{jk}) = \frac{1}{r_{jk}^2}\frac{\partial}{\partial r_{jk}}\left(r_{jk}^2\frac{\partial u}{\partial r_{jk}}\right) = \frac{1}{r_{jk}^2}\left(2r_{jk}\frac{\partial u}{\partial r_{jk}} + r_{jk}^2\frac{\partial^2 u}{\partial r_{jk}^2}\right) = \frac{2}{r_{jk}}\frac{\partial u}{\partial r_{jk}} + \frac{\partial^2 u}{\partial r_{jk}^2}$$

Joining all the terms evaluated up to now, we derive the expression for the kinetic energy term appearing in the expression for $E_L(\boldsymbol{r})$, namely

$$\frac{\nabla_k^2\Psi_T(\mathbf{r})}{\Psi_T(\mathbf{r})} = \underbrace{\frac{\nabla_k^2\phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)}}_{\text{Term 1}} + \underbrace{2\frac{\nabla_k\phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)}\left[\sum_{j\neq k}\frac{\mathbf{r}_{kj}}{r_{kj}}\frac{du(r_{kj})}{dr_{kj}}\right]}_{\text{Term 2}} + \underbrace{\left[\sum_{j\neq k}\sum_{m\neq k}\frac{\mathbf{r}_{kj}}{r_{kj}}\frac{du(r_{kj})}{dr_{kj}}\frac{\mathbf{r}_{km}}{r_{km}}\frac{du(r_{km})}{dr_{km}}\right]}_{\text{Term 3}} + \underbrace{\left[\sum_{j\neq k}\frac{2}{r_{jk}}\frac{\partial u(r_{jk})}{\partial r_{jk}} + \frac{\partial^2 u(r_{jk})}{\partial r_{jk}^2}\right]}_{\text{Term 4}} \tag{27}$$

Substituting the Eq.s 22 and 23 into Eq. 27 we can evaluate each term for the general case.

Term 1

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

Term 2

$$2\frac{\nabla_k\phi(\mathbf{r}_k)}{\phi(\mathbf{r}_k)}\left[\sum_{j\neq k}\frac{\mathbf{r}_{kj}}{r_{kj}}\frac{du(r_{kj})}{dr_{kj}}\right] = = 2\left(-2\alpha\left(x_k, y_k, \beta z_k\right)\right)\sum_{j\neq k}\frac{\mathbf{r}_{kj}}{r_{kj}}\frac{a}{r_{kj}\left(r_{kj}-a\right)} = -4\alpha\left(x_k, y_k, \beta z_k\right)\cdot\sum_{j\neq k}\frac{\mathbf{r}_{kj}}{r_{kj}}\frac{a}{r_{kj}\left(r_{kj}-a\right)}$$

Term 3

$$\sum_{j\neq k}\sum_{m\neq k}\frac{\mathbf{r}_{kj}}{r_{kj}}\frac{du(r_{kj})}{dr_{kj}}\frac{\mathbf{r}_{km}}{r_{km}}\frac{du(r_{km})}{dr_{km}} = = \sum_{j\neq k}\sum_{m\neq k}\frac{\mathbf{r}_{kj}}{r_{kj}}\cdot\frac{\mathbf{r}_{km}}{r_{km}}\frac{a}{r_{kj}\left(r_{kj}-a\right)}\frac{a}{r_{km}\left(r_{km}-a\right)}$$

Term 4

$$\sum_{j\neq k}\left[\frac{2}{r_{jk}}\frac{\partial u(r_{jk})}{\partial r_{jk}} + \frac{\partial^2 u(r_{jk})}{\partial r_{jk}^2}\right] = = \sum_{j\neq k}\left[\frac{2}{r_{jk}}\frac{a}{r_{kj}\left(r_{kj}-a\right)} + \frac{a\left(a-2r_{kj}\right)}{r_{kj}^2\left(r_{kj}-a\right)^2}\right]$$

Now, we have all the tools to evaluate analytically the local energy in Eq. 24:

$$\begin{aligned}
E_L(\mathbf{r}) &= -\frac{1}{2}\sum_i^N\Bigg\{2\alpha\left[2\alpha(x_i^2+y_i^2+\beta^2 z_i^2) - 2 - \beta\right] \\
&+ \sum_{j\neq i}\frac{a}{r_{ij}^2(r_{ij}-a)}\Bigg\{-4\alpha\left(x_i, y_i, \beta z_i\right)\cdot\mathbf{r}_{ij} + 2 + \\
&+ \frac{a-2r_{ij}}{r_{ij}-a} + \mathbf{r}_{ij}\cdot\sum_{m\neq i}\frac{\mathbf{r}_{im}}{r_{im}}\frac{a}{r_{im}\left(r_{km}-a\right)}\Bigg\}\Bigg\} + \\
&+ \frac{1}{2}\sum_i^N(x_i^2+y_i^2+\omega_z^2 z_i^2) \\
&= \alpha(2+\beta)N - 2\alpha^2\sum_i^N(x_i^2+y_i^2+\beta^2 z_i^2) - \\
&- \frac{1}{2}\sum_i^N\sum_{j\neq i}\frac{a}{r_{ij}^2(r_{ij}-a)}\Bigg\{-4\alpha\left(x_i, y_i, \beta z_i\right)\cdot\mathbf{r}_{ij} + 2 \\
&+ \frac{a-2r_{ij}}{r_{ij}-a} + \mathbf{r}_{ij}\cdot\sum_{m\neq i}\frac{\mathbf{r}_{im}}{r_{im}^2}\frac{a}{(r_{km}-a)}\Bigg\}\Bigg\} \\
&+ \frac{1}{2}\sum_i^N(x_i^2+y_i^2+\omega_z^2 z_i^2) \\
&= \alpha(2+\beta)N + \sum_i^N\Bigg[(x_i^2+y_i^2)\left(\frac{1}{2} - 2\alpha^2\right) + \\
&+ z_i^2\left(\frac{1}{2}\omega_z^2 - 2\alpha^2\beta^2\right)\Bigg] - \frac{1}{2}\sum_i^N\sum_{j\neq i}\frac{a}{r_{ij}^2(r_{ij}-a)}\times \\
&\times\Bigg\{-4\alpha\left(x_i, y_i, \beta z_i\right)\cdot\mathbf{r}_{ij} + 2 + \frac{a-2r_{ij}}{r_{ij}-a} \\
&+ \mathbf{r}_{ij}\cdot\sum_{m\neq i}\frac{\mathbf{r}_{im}}{r_{im}^2}\frac{a}{(r_{km}-a)}\Bigg\}\Bigg\}
\end{aligned}$$

which is the result reported in Eq. 6. As previously stated, this is the most general result for the local energy in the context of this project, namely it corresponds to the the case of N particles in an elliptical potential. However, by simply imposing conditions on a, β and ω_z and possibly reducing the dimensionality of the system, every other treated case can be restored.

B GENERAL EXPRESSION FOR THE DRIFT FORCE

The Drift Force term for particle k appearing in the importance sampling context is defined as:

$$\boldsymbol{F}_k = \frac{2\nabla_k\Psi_T(\mathbf{r})}{\Psi_T(\mathbf{r})}$$

Using the result of Eq. 26 in Appendix A, it yields:

$$\begin{aligned}\boldsymbol{F}_k &= \frac{2}{\Psi_T(\mathbf{r})}\Psi_T(\mathbf{r})\left[-2\alpha(x_k,y_k,\beta z_k)+\sum_{j\neq k}\frac{a}{(r_{jk}-a)\,r_{jk}^2}\mathbf{r}_{kj}\right]\\ &= -4\alpha(x_k,y_k,\beta z_k)+2\sum_{j\neq k}\frac{a}{(r_{jk}-a)\,r_{jk}^2}\mathbf{r}_{kj}\end{aligned}$$

C LOCAL ENERGY DERIVATIVE WITH RESPECT TO ALPHA

Here we aim to evaluate the derivative of the expected value of the local energy with respect to the variational parameter α. This expression enters in the generation of a new value α_k in the context of the gradient descent method for the research of the minimum energy of a given system. Starting from the content of Eq. 6 and Eq. 7, we proceed as follows

$$\begin{aligned}\frac{\partial\langle E_L\rangle}{\partial\alpha} &= \frac{\partial}{\partial\alpha}\left[\int d\boldsymbol{R}P(\boldsymbol{R},\boldsymbol{\alpha})E_L(\boldsymbol{R},\boldsymbol{\alpha})\right]\\ &= \frac{\partial}{\partial\alpha}\left[\frac{\int d\boldsymbol{R}\Psi_T^*\hat{H}\Psi_T}{\int d\boldsymbol{R}\Psi_T^*\Psi_T}\right]\end{aligned}$$

For the cases treated in the problem the trial wavefunction is real, leading to a simplification in the calculations. Furthermore, for the sake of reducing the notation, in the following steps we omit the dependences of Ψ_T, thus

$$\begin{aligned}\frac{\partial\langle E_L\rangle}{\partial\alpha} &= \frac{\partial}{\partial\alpha}\left[\frac{\int d\boldsymbol{R}\Psi_T\hat{H}\Psi_T}{\int d\boldsymbol{R}\Psi_T^2}\right]\\ &= \frac{1}{\left[\int d\boldsymbol{R}\Psi_T^2\right]^2}\left\{\left[\int d\boldsymbol{R}\overline{\Psi}_T\hat{H}\Psi_T+\int d\boldsymbol{R}\Psi_T\hat{H}\overline{\Psi}_T\right]\times\right.\\ &\left.\times\left[\int d\boldsymbol{R}\Psi_T^2\right]-\left[\int d\boldsymbol{R}\Psi_T\hat{H}\Psi_T\right]\left[\int d\boldsymbol{R}2\Psi_T\overline{\Psi}_T\right]\right\}\end{aligned}$$

where $\overline{\Psi}_T$ is the derivative of Ψ_T with respect to alpha. Using now the Hermiticity of $\hat{H}$, we get

$$\int d\boldsymbol{R}\Psi_T\hat{H}\overline{\Psi}_T=\langle\Psi_T|\hat{H}|\overline{\Psi}_T\rangle=\langle\hat{H}\Psi_T|\overline{\Psi}_T\rangle=\int d\boldsymbol{R}\overline{\Psi}_T\hat{H}\Psi_T$$

Thus we reduce to

$$\begin{aligned}\frac{\partial\langle E_L(\alpha)\rangle}{\partial\alpha} &= \frac{2}{\left[\int d\boldsymbol{R}\Psi_T^2\right]^2}\left[\int d\boldsymbol{R}\frac{\overline{\Psi}_T}{\Psi_T}\frac{\hat{H}\Psi_T}{\Psi_T}\Psi_T^2\right]\cancel{\int d\boldsymbol{R}\Psi_T^2}-\\ &-\frac{2}{\left[\int d\boldsymbol{R}\Psi_T^2\right]^2}\left[\int d\boldsymbol{R}\frac{\hat{H}\Psi_T}{\Psi_T}\Psi_T^2\right]\left[\int d\boldsymbol{R}\frac{\overline{\Psi}_T}{\Psi_T}\Psi_T^2\right]\\ &= 2\left[\left\langle\frac{\overline{\Psi}_T}{\Psi_T}E_L\right\rangle-\left\langle\frac{\overline{\Psi}_T}{\Psi_T}\right\rangle\langle E_L\rangle\right]\end{aligned}$$

which is the result reported in Eq. 20.

D REDUCTION TO THE NON-INTERACTING CASE FROM THE INTERACTING ONE

In our code implementation, the non-interacting case has been treated separately from the interacting one, namely the user can choose to adopt a spherical or elliptical hamiltonian, corresponding to a gaussian or asymmetric gaussian as wavefunction. A brief test was conducted in order to find some possible mistakes in the implementation of the functions for the interacting case. We report some simulations performed with the elliptical Hamiltonian (see Eq. 1 and Eq. 3) and the full wavefunction (see Eq. 4) with the parameters set as in the non-interacting case, namely $\beta=\omega_z=1$ and $a=0$. We report here the results for $\langle E_L(\alpha)$ estimated within a VMC simulation based on the importance sampling with $\delta t=0.1$ and 2^{21} steps for a 3D system of 10 particles. From the following table is possible to notice that the results obtained with the properly tuned interacting hamiltonian are compatible with those obtained in the non-interacting case.

α	$\langle E_L\rangle_{NI}$	σ_{NI}	$\langle E_L\rangle_I$	σ_I
0.3	17.02	0.02	16.98	0.02
0.4	15.368	0.007	15.370	0.007
0.5	15.000	0.000	15.000	0.000
0.6	15.250	0.005	15.250	0.005
0.7	15.860	0.008	15.855	0.008

Table 10: Comparison between the results obtained with a gaussian wavefunction combined with a non-interacting Hamiltonian (results labelled as NI) and a asymmetric gaussian wavefunction combined with an elliptical Hamiltonian with $\beta=\omega_{ho}=1$ and $a=0$ (results labelled as I). The simulations were performed with the importance sampling algorithm with $\delta t=0.1$, $N_{therm}=10^5$, $N_{steps}=2^{21}$ for a 10 particles $3D$ system. The uncertainty comes from the blocking method estimation.

References

[1] Albert Einstein. "Quantentheorie des einatomigen idealen Gases". In: *Königliche Preußische Akademie der Wissenschaften, Sitzungsberichte, Berlin, Physikalisch-mathematische Klasse 1924, Gesamtsitzung vom 10. Juli 1924* (1924), pp. 261–267.

[2] Nicholas Metropolis et al. "Equation of State Calculations by Fast Computing Machines". In: *The Journal of Chemical Physics* 21.6 (1953), pp. 1087–1092. DOI: 10.1063/1.1699114.

[3] W. K. Hastings. "Monte Carlo sampling methods using Markov chains and their applications". In: *Biometrika* 57.1 (Apr. 1970), pp. 97–109. DOI: 10.1093/biomet/57.1.97.

[4] Marius Jonsson. "Standard error estimation by an automated blocking method". In: *Physical Review E* 98 (Oct. 2018). DOI: 10.1103/PhysRevE.98.043304.

[5] M. H. Anderson et al. "Observation of Bose-Einstein Condensation in a Dilute Atomic Vapor". In: *Science* 269 (5221 1995), pp. 198–201. DOI: 10.1126/science.269.5221.198.

[6] K. B. Davis et al. "Bose-Einstein Condensation in a Gas of Sodium Atoms". In: *Phys. Rev. Lett.* 75 (22 Nov. 1995), pp. 3969–3973. DOI: 10.1103/PhysRevLett.75.3969. URL: https://link.aps.org/doi/10.1103/PhysRevLett.75.3969.

[7] Michèle Leduc, Julien Dugué, and Juliette Simonet. "Laser Cooling, Trapping, and Bose-Einstein Condensation of Atoms and Molecules". In: *AIP Conference Proceedings* 1119.1 (2009), pp. 37–42. DOI: doi:10.1063/1.3137904.

[8] Gross E.P. "Structure of a quantized vortex in boson systems". In: *Nuovo Cimento* 20 (1961), pp. 454–477. DOI: doi:10.1063/1.3137904.

[9] L.P Pitaevskii. "Vortex lines in an imperfect Bose gas". In: *Sov. Phys. JETP.* 13 (1961), pp. 451–454.

[10] Gamerman and Hedibert. *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*. Text in Statistical Science. Chapman and Hall, 2006. ISBN: 9781584885870.

[11] Franco Dalfovo et al. "Theory of Bose-Einstein condensation in trapped gases". In: *Rev. Mod. Phys.* 71 (3 Apr. 1999), pp. 463–512. DOI: 10.1103/RevModPhys.71.463.

[12] J. L. DuBois and H. R. Glyde. "Bose-Einstein condensation in trapped bosons: A variational Monte Carlo analysis". In: *Physical Review A* 63.2 (Jan. 2001). ISSN: 1094-1622. DOI: 10.1103/physreva.63.023602.

[13] J. K. Nilsen et al. "Vortices in atomic Bose-Einstein condensates in the large-gas-parameter region". In: *Physical Review A* 71.5 (May 2005). ISSN: 1094-1622. DOI: 10.1103/physreva.71.053610.

[14] M. Hjorth-Jensen. *Computational Physics - Lecture Notes Fall 2015*. 2015. URL: https://github.com/CompPhysics/ComputationalPhysics2/blob/gh-pages/doc/Literature/lectures2015.pdf.

[15] Jonathan Richard Shewchuk. *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain*. 1994.