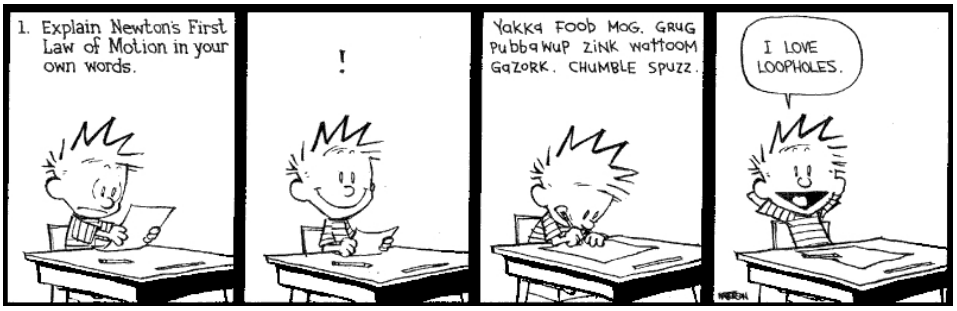
Variational Monte Carlo on bosonic systems

Winther-Larsen, Sebastian Gregorius[1] and Schøyen, Øyvind Sigmundson[1]

[1] *University of Oslo*

(Dated: March 25, 2018)

Something very abstract and clever should go here.



CONTENTS

I. INTRODUCTION

We will in this project study the Variational Monte Carlo (VMC) method, and use it to evaluate the ground state energy of a trapped, hard sphere Bose gas. [1] [2]

II. THEORY

To model the trapped bosonic gas particles we use the potential

$$V_{\text{ext}}(\mathbf{r}) = \begin{cases} \frac{1}{2}m\omega^2 r^2 & \text{(S)}, \\ \frac{1}{2}m\left[\omega^2(x^2+y^2)+\omega_z^2 z^2\right] & \text{(E)}, \end{cases} \tag{1}$$

where we can choose between a spherical (S) or an elliptical (E) harmonic trap. The two-body Hamiltonian of the system is given by

$$H = \sum_{i=1}^{N} h(\mathbf{r}_i) + \sum_{i<j}^{N} w(\mathbf{r}_i, \mathbf{r}_j), \tag{2}$$

where the single particle one body operator, h, is given by

$$h(\mathbf{r}_i) = -\frac{\hbar^2}{2m}{}^2 + V_{\text{ext}}(\mathbf{r}_i), \tag{3}$$

(we assume equal mass) and the two-body interaction operator, w, is

$$w(\mathbf{r}_i, \mathbf{r}_j) = \begin{cases} \infty & |\mathbf{r}_i - \mathbf{r}_j| \leq a, \\ 0 & |\mathbf{r}_i - \mathbf{r}_j| > a, \end{cases} \tag{4}$$

where a is the hard sphere of the particle. The trial wavefunction, $\Psi_T(\mathbf{r})$, we will be looking at is given by

$$\Psi_T(\mathbf{r}) = \Phi_T(\mathbf{r}) \prod_{j<k}^{N} f(a, \mathbf{r}_j, \mathbf{r}_k) \tag{5}$$

$$= \left(\prod_{i=1}^{N} g(\alpha, \beta, \mathbf{r}_i) \right) \prod_{j<k}^{N} f(a, \mathbf{r}_j, \mathbf{r}_k), \tag{6}$$

where α and β are variational parameters and

$$\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N, \alpha, \beta). \tag{7}$$

Here g are the single particle wavefunctions given by

$$g(\alpha, \beta, \mathbf{r}_i) = \exp\big[-\alpha(x_i^2 + y_i^2 + \beta z_i^2)\big] \equiv \phi(\mathbf{r}_i), \tag{8}$$

and $\Phi_T(\mathbf{r})$ the *Slater permanent* consisting of the N first single particle wavefunctions, and the correlation wavefunction, f, given by

$$f(a, \mathbf{r}_j, \mathbf{r}_k) = \begin{cases} 0 & |\mathbf{r}_j - \mathbf{r}_k| \leq a, \\ \left(1 - \frac{a}{|\mathbf{r}_j - \mathbf{r}_k|}\right) & |\mathbf{r}_j - \mathbf{r}_k| > a. \end{cases} \tag{9}$$

We will for brevity use the notation $\phi(\mathbf{r}_i) = \phi_i$ and $r_{jk} = |\mathbf{r}_j - \mathbf{r}_k|$.

A. Local energy

As the many-body wavefunction creates a very large configuration space, where much of the wavefunction is small, we use the Metropolis algorithm in order to move towards regions in configuration space with "sensible" values. We define the *local energy*, $E_L\mathbf{r}$, by

$$E_L(\mathbf{r}) = \frac{H\Psi_T(\mathbf{r})}{\Psi_T(\mathbf{r})}. \tag{10}$$

If $\Psi_T(\mathbf{r})$ is an exact eigenfunction of the Hamiltonian, E_L will be constant. The closer $\Psi_T(\mathbf{r})$ is to the exact wave function, the less variation in E_L as a function of $\mathbf{r}$ we get. One of the most computationally intensive parts of the VMC algorithm will be to compute E_L. We therefore find an analytical expression for E_L in terms of the trial wavefunction.

When we are performing the Monte Carlo sampling we are interested in the expected value of the local energy as this will serve as our estimate of the true energy. Mathematically this is expressed as

$$\langle E_L \rangle = \frac{\int \mathrm{d}\mathbf{r} \Psi_T^*(\mathbf{r}) H \Psi_T \mathbf{r}}{\int \mathrm{d}\mathbf{r} |\Psi_T(\mathbf{r})|^2} = \frac{\int \mathrm{d}\mathbf{r} |\Psi_T(\mathbf{r})|^2 E_L}{\int \mathrm{d}\mathbf{r} |\Psi_T(\mathbf{r})|^2}, \tag{11}$$

where we have multiplied and divided by the trial wavefunction to get the latter expression.

B. The drift force

A disadvantage in the use of the brute-force Metropolis algorithm is that we might be spending much computational resources in an uninteresting part of configuration space. To make smarter moves we will use the Metropolis-Hastings algorithm (which will be discussed in due time). This algorithm is dependent on the drift force of the system.

$$\mathbf{F}(\mathbf{r}) = \sum_{k=1}^{N} \mathbf{F}_k(\mathbf{r}) = \sum_{k=1}^{N} \frac{2\nabla_k \Psi_T(\mathbf{r})}{\Psi_T(\mathbf{r})}. \tag{12}$$

Using this expression we are able to move towards parts of configuration space where the gradient increases or decreases yielding a better choice of movements. We will mainly be interested in the drift force of a single particle k.

III. NON-INTERACTING HARMONIC OSCILLATORS

We start by looking at a simple system of non-interacting harmonic oscillators. That is, where $a = 0$ and $\beta = 1$. We thus get the trial wavefunction

$$\Psi_T(\mathbf{r}) = \Phi_T(\mathbf{r}) = \prod_{i=1}^{N} \exp\big[-\alpha |\mathbf{r}_i|^2\big], \tag{13}$$

where $|\mathbf{r}_i| = r_i$. As $a = 0$ the interaction term, $w(\mathbf{r}_i, \mathbf{r}_j)$, vanishes and the Hamiltonian is given by (in the spherical case)

$$H = \sum_{i=1}^{N} h(\mathbf{r}_i) = \sum_{i=1}^{N} \left(-\frac{\hbar^2}{2m} \nabla_i^2 + \frac{1}{2} m\omega^2 r_i^2 \right). \tag{14}$$

To find the drift force and the local energy we have to compute the gradient and the Laplacian of the trial wavefunction. The gradient is given by

$$\nabla_k \Psi_T(\mathbf{r}) = -2\alpha \mathbf{r}_k \Psi_T(\mathbf{r}), \tag{15}$$

whereas the Laplacian yields

$$\nabla_k^2 \Psi_T(\mathbf{r}) = \big(-2d\alpha + 4\alpha^2 r_k^2 \big) \Psi_T(\mathbf{r}), \tag{16}$$

where d is the dimensionality of the problem determined by $\mathbf{r}_k \in \mathbb{R}^d$. We can thus use the gradient to find an expression for the drift force for particle k.

$$\mathbf{F}_k(\mathbf{r}) = -4\alpha \mathbf{r}_k. \tag{17}$$

Using the Laplcian we can compute the kinteic term in the expression for the local energy. We get

$$E_L(\mathbf{r}) = \sum_{i=1}^{N} \left(-\frac{\hbar^2}{2m} \big[-2d\alpha + 4\alpha^2 r_i^2 \big] + \frac{1}{2} m\omega^2 r_i^2 \right). \tag{18}$$

In natural units, with $\hbar = c = m = 1$, this reduces to

$$E_L(\mathbf{r}) = \alpha dN + \left(\frac{1}{2}\omega^2 - 2\alpha^2\right)\sum_{i=1}^{N} r_i^2. \tag{19}$$

It is worth noting that for $\alpha = \frac{1}{2}\omega$ (α is required to be positive) we will find a stable value which turns out to be the exact energy minimum. This happens as the entire sum over all the random walkers disappears.

A. Exact variational energy

As the system is non-interacting and consisting of Gaussians we can find an expression for the exact energy as a function of the variational parameter α, i.e.,

$$E(\alpha) = \frac{\langle \Psi_T | \, H \, | \Psi_T \rangle}{\langle \Psi_T | \Psi_T \rangle}. \tag{20}$$

The final result for the energy is

$$E(\alpha) = \left(\frac{\hbar^2\alpha}{2m} + \frac{m\omega^2}{8\alpha}\right) dN. \tag{21}$$

By minimizing this expression, i.e., finding the derivative of the energy with respect to α and equating this to zero, yields the expected minimum of variational energy to be

$$\frac{\mathrm{d}E(\alpha)}{\mathrm{d}\alpha} = 0 \implies \alpha_0 = \frac{m\omega}{2\hbar}, \tag{22}$$

which in natural units reduces to $\alpha_0 = \frac{1}{2}\omega$. The energy at this value of α (in natural units) is then

$$E(\alpha_0) = \frac{\omega dN}{2}. \tag{23}$$

IV. INTERACTING HARD SPHERE BOSONS

Moving to the full system allowing β to vary and setting $a \neq 0$ we can write the trial wavefunction as

$$\Psi_T(\mathbf{r}) = \Phi_T(\mathbf{r}) J(\mathbf{r}), \tag{24}$$

where $|\Phi_T\rangle$ is the same Slater permanent as in Equation 6 and $J(\mathbf{r})$ is the *Jastrow factor* given by

$$J(\mathbf{r}) = \exp\left(\sum_{j<l}^{N} u(r_{jl})\right), \tag{25}$$

where $r_{jk} = |\mathbf{r}_j - \mathbf{r}_k|$ and

$$u(r_{jk}) = \ln\big[f(a, \mathbf{r}_j, \mathbf{r}_k)\big]. \tag{26}$$

To further shorten the notation we will use $u_{jk} = u(r_{jk})$. Computing the gradient of the wavefunction we get

$$\nabla_k \Psi_T(\mathbf{r}) = \Big[\nabla_k \Phi_T(\mathbf{r})\Big] J(\mathbf{r}) + \Phi_T(\mathbf{r}) \nabla_k J(\mathbf{r}). \tag{27}$$

The gradient of the Slater permament for particle k is given by

$$\nabla_k \Phi_T(\mathbf{r}) = \nabla_k \phi_k \prod_{i \neq k}^{N} \phi_i = \frac{\nabla_k \phi_k}{\phi_k} \Phi_T(\mathbf{r}). \tag{28}$$

The gradient of the Jastrow factor is given by

$$\nabla_k J(\mathbf{r}) = J(\mathbf{r}) \nabla_k \sum_{m<n}^{N} u_{mn} \tag{29}$$

$$= J(\mathbf{r}) \left(\sum_{m=1}^{k-1} \nabla_k u_{mk} \sum_{n=k+1}^{N} \nabla_k u_{kn}\right) \tag{30}$$

$$= J(\mathbf{r}) \sum_{m \neq k}^{N} \nabla_k u_{km}, \tag{31}$$

where the gradient of the interaction term splits the anti-symmetric sum into two parts. As $r_{ij} = r_{ji}$ we can combine these sums into a single sum. This in total yields the gradient

$$\nabla_k \Psi_T(\mathbf{r}) = \left(\frac{\nabla_k \phi_k}{\phi_k} + \sum_{m \neq k}^{N} \nabla_k u_{km}\right) \Psi_T(\mathbf{r}). \tag{32}$$

The Laplcian of the trial wavefunction is found by finding the divergence of Equation 32.

$$\nabla_k^2 \Psi_T(\mathbf{r}) = \left(\nabla_k \left[\frac{\nabla_k \phi_k}{\phi_k}\right] + \sum_{m \neq k}^{N} \nabla_k^2 u_{km}\right) \Psi_T(\mathbf{r}) \tag{33}$$

$$+ \left(\frac{\nabla_k \phi_k}{\phi_k} + \sum_{m \neq k}^{N} \nabla_k u_{km}\right)^2 \Psi_T(\mathbf{r}), \tag{34}$$

where the squared term came from taking the gradient of the trial wavefunction. To further simplify we divide by the trial wavefunction. This yields

$$\frac{\nabla_k^2 \Psi_T(\mathbf{r})}{\Psi_T(\mathbf{r})} = \frac{\nabla_k^2 \phi_k}{\phi_k} + 2\frac{\nabla_k \phi_k}{\phi_k} \sum_{m \neq k} \nabla_k u_{km}$$

$$+ \sum_{m \neq k}^{N} \nabla_k^2 u_{km} + \left(\sum_{m \neq k}^{N} \nabla_k u_{km}\right)^2. \tag{35}$$

To go from here we have to find the gradient and the Laplacian of the single particle functions, ϕ_k, and the interaction functions u_{km}. For the single particle functions we use Cartesian coordinates when finding the derivatives whereas we for the interaction functions will use spherical coordinates and do a variable substitution. Beginning with the gradient of the single particle functions we get

$$\nabla_k \phi_k = \nabla_k \exp\big[-\alpha(x_k^2 + y_k^2 + \beta z_k^2)\big] \tag{36}$$

$$= -2\alpha(x_k \mathbf{e}_i + y_k \mathbf{e}_j + \beta z_k \mathbf{e}_k)\phi_k. \tag{37}$$

Note that the subscripts on the unit vectors $\mathbf{e}_i$ are *not* the same as the subscripts used for its components. The Laplacian yields

$$\nabla_k^2\phi_k = \Big[-2\alpha\big(d-1+\beta\big) + 4\alpha^2\big(x_k^2+y_k^2+\beta^2 z_k^2\big)\Big]\phi_k, \tag{38}$$

with d as the dimensionality of the problem. In order to derive the interaction functions we have to do a variable substitution using $r_{km} = |\mathbf{r}_k - \mathbf{r}_m|$. We can then rewrite the ∇_k-operator as

$$\nabla_k = \nabla_k \frac{\partial r_{km}}{\partial r_{km}} = \nabla_k r_{km}\frac{\partial}{\partial r_{km}} \tag{39}$$

$$= \frac{\mathbf{r}_k - \mathbf{r}_m}{r_{km}}\frac{\partial}{\partial r_{km}}. \tag{40}$$

Applying this version of the ∇_k-operator to u_{km} yields

$$\nabla_k u_{km} = \frac{\mathbf{r}_k - \mathbf{r}_m}{r_{km}}\frac{\partial u_{km}}{\partial r_{km}}. \tag{41}$$

For the Laplacian we switch a little back and forth between the two ways of representing the ∇_k-operator. We thus get

$$\nabla_k^2 u_{km} = \frac{\nabla_k \mathbf{r}_k}{r_{km}}\frac{\partial u_{km}}{\partial r_{km}} + \left[\nabla_k \frac{1}{r_{km}}\right](\mathbf{r}_k - \mathbf{r}_m)\frac{\partial u_{km}}{\partial r_{km}} + \frac{\mathbf{r}_k - \mathbf{r}_m}{r_{km}}\nabla_k \frac{\partial u_{km}}{\partial r_{km}} \tag{42}$$

$$= \frac{d}{r_{km}}\frac{\partial u_{km}}{\partial r_{km}} - \frac{(\mathbf{r}_k - \mathbf{r}_m)^2}{r_{km}^3}\frac{\partial u_{km}}{\partial r_{km}} + \frac{(\mathbf{r}_k - \mathbf{r}_m)^2}{r_{km}^2}\frac{\partial^2 u_{km}}{\partial r_{km}^2} \tag{43}$$

$$= \frac{d-1}{r_{km}}\frac{\partial u_{km}}{\partial r_{km}} + \frac{\partial^2 u_{km}}{\partial r_{km}^2}, \tag{44}$$

where d is again the dimensionality of the problem. In total we can state an intermediate version of the Laplacian occuring in the local energy as

$$\frac{\nabla_k^2\Psi_T(\mathbf{r})}{\Psi_T(\mathbf{r})} = \frac{\nabla_k^2\phi_k}{\phi_k} + 2\frac{\nabla_k\phi_k}{\phi_k}\sum_{m\neq k}^{N}\frac{\mathbf{r}_k - \mathbf{r}_m}{r_{km}}\frac{\partial u_{km}}{\partial r_{km}} + \sum_{m\neq k}^{N}\left(\frac{d-1}{r_{km}}\frac{\partial u_{km}}{\partial r_{km}} + \frac{\partial^2 u_{km}}{\partial r_{km}^2}\right) + \sum_{m,n\neq k}^{N}\frac{\mathbf{r}_k - \mathbf{r}_m}{r_{km}}\frac{\mathbf{r}_k - \mathbf{r}_n}{r_{kn}}\frac{\partial u_{km}}{\partial r_{km}}\frac{\partial u_{kn}}{\partial r_{kn}}. \tag{45}$$

Moving on to the derivatives of the interaction terms, u_{km}, to get an explicit expression for the Laplacian.

$$\frac{\partial u_{km}}{\partial r_{km}} = \frac{a}{r_{km}(r_{km}-a)}, \tag{46}$$

$$\frac{\partial^2 u_{km}}{\partial r_{km}^2} = \frac{a^2 - 2ar_{km}}{r_{km}^2(r_{km}-a)^2}. \tag{47}$$

The local energy and the drift force can now be found by combining these expressions. For brevity, we will not write out the explicit expressions as these will be called by separated functions in our programs.

A. Scaling the system

We now introduce a scaled distance $\mathbf{r}' = \mathbf{r}/a_{\text{ho}}$ [1], where

$$a_{\text{ho}} = \sqrt{\frac{\hbar}{m\omega}}. \tag{48}$$

We can then rewrite the Hamiltonian for the elliptic potential in terms of this new distance. By doing a variable substitution for each direction in the Laplace operator we get

$$\nabla_k^2 = \frac{1}{a_{\text{ho}}^2}\nabla_k'^{\,2}. \tag{49}$$

Looking at the one-body part of the Hamiltonian we get

$$H = \sum_{k=1}^{N}\left\{-\frac{\hbar^2}{2m}\nabla_k^2 + \frac{1}{2}m\left[\omega^2(x_k^2+y_k^2)+\omega_z^2 z_k^2\right]\right\} \tag{50}$$

$$= \frac{\hbar\omega}{2}\sum_{k=1}^{N}\left\{-\nabla_k'^{\,2} + \left[x_k'^{\,2} + y_k'^{\,2} + \lambda^2 z_k'^{\,2}\right]\right\}, \tag{51}$$

where we have introduced the dimensionless frequency $\lambda = \omega_z/\omega$. The single particle functions also gets a scaling factor

$$\phi_k = \exp\left[-\frac{\alpha\hbar}{m\omega}(x_k'^{\,2} + y_k'^{\,2} + \beta^2 z_k'^{\,2})\right] \equiv \phi(\mathbf{r}_k'). \tag{52}$$

The correlation wavefunction and the interaction potential remain the same, but using the scaled hard core sphere radius $a/a_{\text{ho}} = 0.0043$. For brevity we will remove the ticks on the variables.

V. ALGORITHMS

In the project we rely on a Monte Carlo approach of random sampling to obtain numerical results. We simulate random walks over a volume in order to find optimal parameters in our trial wavefunctions. The most common of such methods, which we make use of herein, is the Metropolis-Hastings algorithm.

A. The Metropolis-Hastings Algorithm

The Metropolis-Hastings algorithm can in our particular situation be condensed down to the following steps:

1. The system is initialised by a certain number N of randomly generated positions, or particles. This allows us to evaluate the wavefunction at these points and compute the local energy E_L.

2. The initial configuration is changed by setting a new position for one of these particles. The particle is picked at random.

3. A ratio between new wavefunction density and the previous (initial) density is computed and compared to a random number. This acceptance probability decides if the particle move is rejected or accepted. The particle is only allowed to move a predetermined step length.

4. If the particle movement is accepted and the local energy E_L is computed for the new system.

5. Repeat steps until convergence and an optimum is reached.

The algorithm described above can be applied in an "exhaustive" search of the parameter space in order to find the optimal parameters. Whether a proposed move is accepted or not is determined by a transition probability and the acceptance probability. The strength of the algorithm is that the transition algorithm need not be known. For example, the simplest case is to accept the new state, i.e., the new position for the random walker, if the ratio

$$q(\mathbf{r}_{i+1}, \mathbf{r}_i) = \frac{\left|\Psi_T(\mathbf{r}_{i+1})\right|^2}{\left|\Psi_T(\mathbf{r}_i)\right|^2}, \tag{53}$$

where $\mathbf{r}_{i+1}$ are all the positions at step $i+1$, is greater than a uniform probability $p \in [0, 1)$.

1. Importance Sampling

A problem with the naïve Metropolis-Hastings sampling approach is that the sampling of position space is done with no regard for where we are likely to find a particle. This problem can be remedied through *importance sampling*. It is reasonable to assume that the particles we erratically scatter in space are prone to movement towards the peaks of the probability density as dictated by the wave function. Consider therefore the Fokker-Planck equation,

$$\frac{\partial \Psi_T}{\partial t} = D\nabla \cdot (\nabla - \mathbf{F})\,\Psi_T, \tag{54}$$

which describes the evolution in time of a probability density function. In our case this is the trial wavefunction Ψ_T. Originaly an equation that models diffusion, we have a diffusion term D and the drift force Equation 12. In our case the diffusion term D is simply $1/2$ from the kinetic energy (in natural units).

We use the Langevin equation to find the new position of the particle.

$$\frac{\partial \mathbf{r}}{\partial t} = D\mathbf{F}(\mathbf{r}) + \boldsymbol{\eta}, \tag{55}$$

where $\boldsymbol{\eta}$ is a uniformly distributed stochastic variable for each dimenion. Solving Langevin's equation by Euler's method gives a recursive relation for the subsequent new positions of a particle.

$$\mathbf{r}_{i+1} = \mathbf{r}_i + D\mathbf{F}(\mathbf{r}_i)\delta t + \boldsymbol{\xi}\sqrt{\delta t}, \tag{56}$$

for a given time step δt [1] and a normally distributed stochastic variable $\boldsymbol{\xi}$.

Now we need to change the acceptance probability of the metropolis algorithm to something that takes the new sampling method into account.

$$q(\mathbf{r}_{i+1}, \mathbf{r}_i) = \frac{G(\mathbf{r}_{i+1}, \mathbf{r}_i, \delta t)\left|\Psi_T(\mathbf{r}_{i+1})\right|^2}{G(\mathbf{r}_i, \mathbf{r}_{i+1}, \delta t)\left|\Psi_T(\mathbf{r}_i)\right|^2}, \tag{57}$$

where G is the Green's function of the Fokker-Planck equation given by

$$G(\mathbf{r}_{i+1}, \mathbf{r}_i, \delta t) = \exp\left(-\frac{\left[\mathbf{r}_{i+1} - \mathbf{r}_i - D\mathbf{F}(\mathbf{r}_i)\delta t\right]^2}{4D\delta t}\right) \times \frac{1}{(4\pi D\delta t)^{dN/2}}, \tag{58}$$

where d is the dimensionality.

---

[1] Bear in mind that Equation 55 is only valid as $\Delta t \to 0$, a property stemming from the use of Euler's method.

B. Statistical Analysis

If the results of the metropolis sampling were completely uncorrelated, it would be enough to compute the standard deviation in a familiar way,

$$\sigma = \sqrt{\frac{1}{N}(\langle E_L^2\rangle - \langle E_L\rangle^2}, \tag{59}$$

where N is the number of samples, or Monte-Carlo cycles, in the experiment. However, it is reasonable to assume that the data we are dealing with in this study is liable to suffer from *autocorrelation* and Equation 59 does not hold. The prevailing definition of autocorrelation in a data stream or signal is correlation between a delay of the signal and the original signal. One would be interested to find the delay, or lag, in the signal at which the "self-correlation" is highest. We refer to this spacing as d, and define the following correlation function,

$$f_d = \frac{1}{n-d}\sum_{k=1}^{n-d}(x_k - \bar{x}_n)(x_{k+d} - \bar{x}_n). \tag{60}$$

The keen reader would have noticed that the function f_d in Equation 60 would be equal to the sample variance for $d = 0$. We can now define the *autocorrelation function*

$$\kappa_d = \frac{f_d}{\mathrm{Var}(x)}, \tag{61}$$

which is equal to 1 if the data exhibits no autocorrealtions, equating to $d = 0$. From the autocorrelation function (61) we in turn define the *autocorrelation time*,

$$\tau = 1 + 2\sum_{d=1}^{n-1}\kappa_d, \tag{62}$$

notice that the autocorrelation time is 1 for a correlation free experiment.

We are now able to make a correction to the expression for the standard deviation improving on Equation 59 by taking correlation into account,

$$\sigma = \sqrt{\frac{1+2\tau/\Delta t}{N}(\langle E_L^2\rangle - \langle E_L\rangle^2)}, \tag{63}$$

where Δt is the time between each sample. The main problem at this point is to find τ, which we do not know for any given system and it is generally very expensive to compute. In order to find a good estimate of τ we use a procedure called *blocking*.

1. Blocking

In the method of blocking we group the samples into blocks of increasing size. If one where to compute the standard deviation for each block, one should see the variance increase with the block size. The standard deviation would only increase up to a certain point, from whence it would stay almost constant. What is happening is that we have reached a point where a particular sample from one block is no longer correlated with a corresponding sample from an adjacent block. The block size for this point of convergence now functions as an estimate for the autocorrelation time τ.

Instead of going by this "manual" method of looking at charts, we will construct a test statistic and let the computer do the necessary considerations in a more "automatic way.

The easiest way to do this is by use of the *automatic blocking scheme*, nicely illustrated with a flow chart in Figure 1. The input parameter for the method is an array of the local energies, but it can be any array of time series data, which is why it is herein referred to as $\mathbf{X}$. Any array of sequential data, where any particular data point depends on the previous data point inhibits all attributes of a time series. Our array of local energies is therefor most definitely a time series.

From $\mathbf{X}$ we compute the variance and sample covariance between the series itself and a one-point lag of the series. This covariance is usually referred to as the first order autocovariance.

$$\gamma_i = \frac{1}{n}\left(\sum_{k=1}^{n}\left((\mathbf{X}_i)_k - (\bar{\mathbf{X}}_i)_k\right)\right) \times \left(\sum_{k=0}^{n-1}\left((\mathbf{X}_i)_k - (\bar{\mathbf{X}}_i)_k\right)\right). \tag{64}$$

Now we "block"! The next step is to perform a transformation of the our time series $\mathbf{x}_i$, where we end up with an array that is half the length of the original time series,

$$(\mathbf{X}_i)_k = \frac{1}{2}\left((\mathbf{X}_i)_{2k-1} + (\mathbf{X}_i)_{2k}\right). \tag{65}$$

This process is were the term "blocking" stems from. There are a number of ways to perform the blocking - we could have picked the data points used in the new $(i+1)$ array randomly, or more orderly. Here we we have picked them sequentially. We have obtained $\frac{n}{2}$ new "primed" stochastic variables. Assuming that the reader is playing close attention, it is now needless to say that it is much easier to perform these sequential blocking steps if we have an array of length $n = 2^d$, where d is integer. Otherwise, problems would arise because one would have arrays of different lengths when the original array is "split in two". The ailment is easy to remedy by excluding an observation from the longest resulting array after splitting.

For each new time series $\mathbf{X}_{i+1}$ a new sample variance and sample covariance is computed. The blocking transformation is allowed to continue until we are we run out of data, i.e. the length of $\mathbf{X}_i < 2$. In other words, there

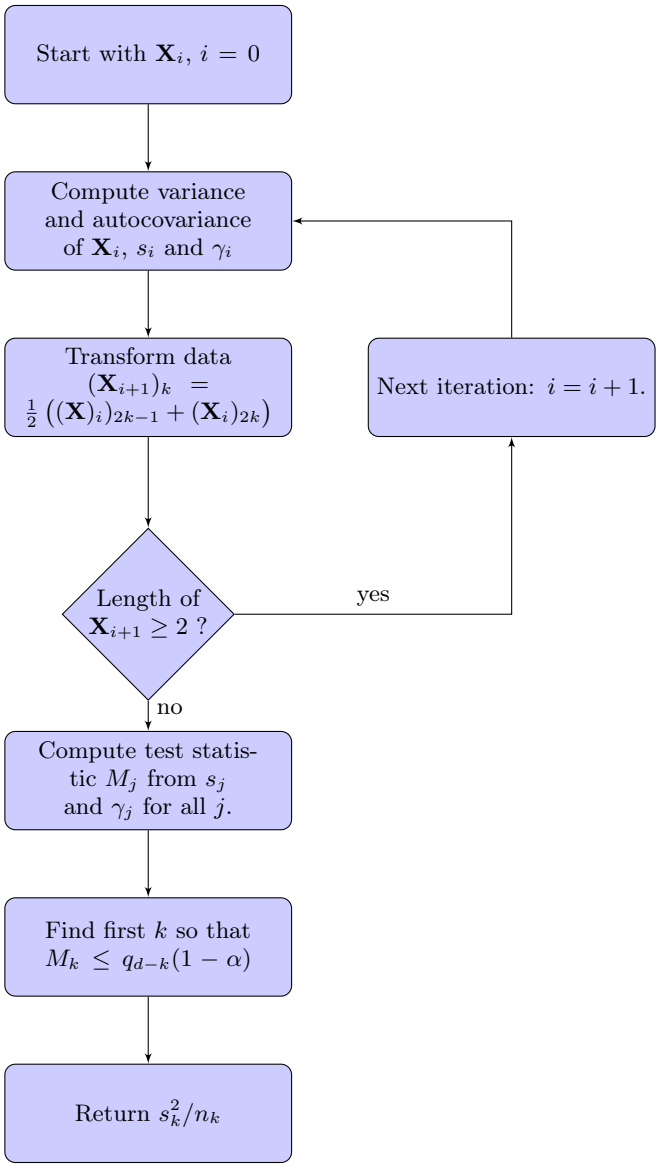


FIG. 1: Flow chart depicting the automatic blocking method for finding a better estimate for sample variance.

is only a single number left. We now compute a test statistic,

$$M_j = \sum_{k=j}^{d-1} n_k \frac{\left[(n_k - 1)\frac{s_k^2}{n_k^2} y\gamma_k(1)\right]^2}{s_k^4}. \tag{66}$$

As luck would have it, this number M is χ^2-distributed. The following step would be to look up a statistical table and find the first k, such that $M_k \leq q_{d-k}(1-\alpha)$, where q is the statistical limit and α is the significance level that your hear desires. Finally, the correct sample standard deviation is $\frac{s_k^2}{n_k}$. Again, see Figure 1 for a summary of the algorithm.

C. Gradient Descent

Scanning the entire space of variational parameters can be a somewhat despairing task. To aid us in the search we therefore implement the method of gradient descent[2]. In any optimum we generally have that the derivative should equal zero. In our case, we want to find the minimum of the expected local energy with respect to some variational parameter α, so that

$$\nabla_\alpha \left\langle E_L(\mathbf{R};\alpha)\right\rangle = 0. \tag{67}$$

The foundation for the method of gradient descent lies in the fact that some analytic [3] function $F(\alpha)$ decreases fastest in the direction of $-\nabla_\alpha F(\alpha)$. This means that we could eventually get to an ensuing minimum by trailing along the pathway of the following recursive relation,

$$\alpha_{k+1} = \alpha_k - \gamma\nabla_\alpha F(\alpha_k). \tag{68}$$

If γ is small enough we should have $f(\alpha_k) \geq F(\alpha_{k+1})$ for all k and hopefully we will experience convergence to the desired minimum.

For our special case we only have one variational parameter, α. The expression for the derivative is,

$$\nabla_\alpha \langle E_L\rangle = 2\left(\left\langle E_L \frac{1}{\Psi_T}\nabla_\alpha\Psi_T\right\rangle - \left\langle \frac{1}{\Psi_T}\nabla_\alpha\Psi_T\right\rangle \langle E_L\rangle\right). \tag{69}$$

VI. RESULTS

A. Non-Interacting Systems

Initially we are interested in seeing how our implementation performs and to what extent it functions properly. the logical thing to do would be to look at a non-interacting system with a simple Gaussian wavefunction and harmonic oscillator potential. Such a system has a simple closed-form analytic solution[4], which is further simplified by the "god-given" analytic units (Equation 19). This expression is minimised for $\alpha = \frac{1}{2}\omega$. We also set $\omega = 1$ and get an expression for the expected minimum local energy,

$$E_L(\mathbf{r}) = \frac{dN}{2}, \tag{70}$$

which will be used as the benchmark we are ultimately aiming for.

---

[2] Also known as steepest descent, easily confused with the steepest descent method for integation.
[3] We use the term analytic quite recklessly.
[4] A fact that every undergraduate physics student should be well aware of after finishing their first course in quantum physics

TABLE I: One particle in one dimension for the analytic expression with 2^{21} Monte Carlo cycles and step length of 0.5.

α	$\langle E_L \rangle$	σ	σ_b	A	t_{CPU}[s]
0.30	0.56244	0.00026	0.00129	0.89185	0.43238
0.34	0.53652	0.00019	0.00094	0.88465	0.38340
0.38	0.51863	0.00013	0.00063	0.87844	0.37200
0.42	0.50805	0.00009	0.00038	0.87161	0.37978
0.46	0.50201	0.00004	0.00017	0.86594	0.37214
0.50	0.50000	0.00000	0.00000	0.86047	0.37162
0.54	0.50183	0.00004	0.00015	0.85543	0.37959
0.58	0.50566	0.00007	0.00029	0.84994	0.37778
0.62	0.51175	0.00011	0.00041	0.84479	0.37368
0.66	0.51906	0.00014	0.00052	0.84000	0.37257
0.70	0.52883	0.00017	0.00062	0.83570	0.37342

TABLE II: One particle in one dimension for the numeric expression with 2^{21} Monte Carlo cycles and a step length of 0.5.

α	$\langle E_L \rangle$	σ	σ_b	A	t_{CPU}[s]
0.30	0.56675	0.00026	0.00135	0.89098	0.52634
0.34	0.53714	0.00019	0.00096	0.88479	0.52186
0.38	0.51836	0.00013	0.00062	0.87820	0.52351
0.42	0.50717	0.00009	0.00038	0.87180	0.52892
0.46	0.50184	0.00004	0.00017	0.86581	0.51759
0.50	0.50000	0.00001	0.00001	0.86024	0.52427
0.54	0.50133	0.00004	0.00015	0.85520	0.52052
0.58	0.50520	0.00007	0.00028	0.84971	0.52156
0.62	0.51120	0.00011	0.00039	0.84481	0.51830
0.66	0.51896	0.00014	0.00052	0.83990	0.51865
0.70	0.52794	0.00017	0.00061	0.83506	0.51255

We will start without importance sampling, that is by employing the simpler and more naïve "brute force" Monte Carlo sampling. Moreover, we will compute the value for the local energy analytically as well as with a numerical approach. In the numerical approach we employ the well-know approximation for the second derivative,

$$f''(x) \approx \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}. \tag{71}$$

Table I shows the expected local energy $\langle E_L \rangle$, standard deviations σ, σ_b, the share of accepted Monte Carlo reorientations A and cpu time for the analytic computation at different values for the variational parameter α. Table II shows the same numbers but for the numeric scheme. The first column of standard deviations σ are the "naïve" standard deviations, while σ_b marks the standard deviations computed with the blocking method. One can clearly see, in both the analytic and numeric case, that the former standard deviations are lower than the latter, meaning that autocorrelated data can lead to a lower perceived uncertainty than what is reality. The blocking method takes this fact into account and for this reason, all other standard deviations will be computed with the blocking method from here on.

TABLE III: Comparison of energy, standard deviation and the CPU time for $\alpha = 0.5$ (the correct minimum) for the analytic and numeric schemes in three dimensions and N particles. The step length is 0.5.

	Analytic			Numeric		
N	$\langle E_L \rangle$	σ_b	t_{CPU}[s]	$\langle E_L \rangle$	σ_b	t_{CPU}[s]
1	1.5	0.0	0.50	1.49997	0.00008	0.73
10	15.0	0.0	0.73	14.98278	0.00867	6.40
100	150.0	0.0	4.13	149.81453	0.81501	387.60
500	750.0	0.0	19.16	773.85435	20.69789	9595.02

The first initial observation of note is that we have found an optimum for the system in terms of a minimum energy at $\alpha = \frac{1}{2}$. Moreover, notice how the standard deviation disappears completely. This should already be apparent from Equation 19, as the term involving particle positions disappears for the optimal α.

Systems of optimum variational parameter warrants further investigation, as summed up in Table III. The analytical results are quite predictable and monotonous; the expected local energy $\langle E_l \rangle$ is perfectly proportional by a factor $\frac{3}{2}$ to the number of particles and the standard deviation σ_b is zero for systems of any number of particles. The numerical results tells a more tortuous tale. As the number of particles increase, the uncertainty also increases rapidly. For the largest system, with $N = 500$ particles, the energy misses its target critically - the true energy is more than one standard deviation away. Lastly, the cpu time is notably higher for the numeric scheme.

In the appendices, Figure 6 shows how both the analytic and numeric approach matches the exact expression for the energy as a function of α for systems of increasing size. We see also here that the uncertainty increases in the numeric scheme as the system gets larger.

1. *Introducing Importance Sampling*

Next we will introduce importance sampling, the first prediction for importance sampling compared to brute, naïve method is that we should approach an equilibrium much faster because we are now making moves in a smarter way. This prediction is confirmed in Figure 2 and Figure 3.

Figure 2 shows the standard deviation of the local energies for increasing number of Monte Carlo cycles for both the regular sampling and the importance sampling methods. We see that the standard deviation decreases for both methods for a larger number of Monte Carlo cycles, but the standard deviation is always lower for the importance sampling method. This is in accordance with our expectations.

We have also compared the expected energy with the exact energy for different number of Monte Carlo cycles. The result of this analysis is shown in Figure 3. We see that both methods will evnetually result in ex-

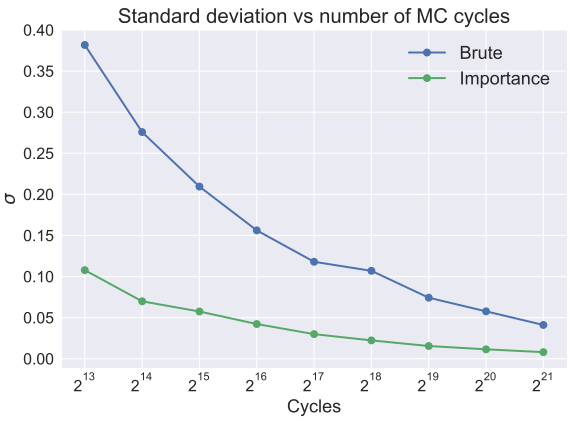


FIG. 2: As the number of Monte Carlo cycles increases the standard deviation of the local energies decrease. This effect is stronger when the importance sampling scheme is employed.

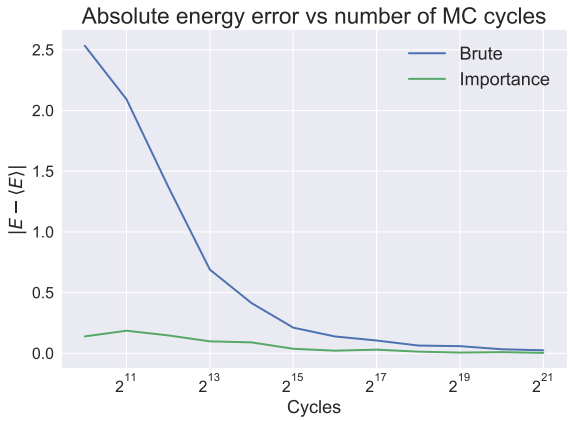


FIG. 3: The absolute difference between the expected energy and the exact energy for both brute force and importance sampling.

pected energies that are equal to the exact energies as $\left|E - \langle E\rangle\right| \to 0$. However, for the method of importance sampling the error is consistently lower.

Table IV shows the standard deviations and acceptance ratios for different time steps. The simulations in this tabel are from the same span $\alpha \in [0.3, 07]$ as in Table I and Table II. The optimum energies for $\alpha = \frac{1}{2}$ gives the same result for importance sampling as for the naïve way of sampling. Notice that by lowering the time step, the uncertainty starts to decrease and then starts to increase again. Moreover, the acceptance ratios increases as the time step decreases.

TABLE IV: Importance sampled simulations of $n = 10$ particles in $d = 3$ dimensions. Average acceptance ratios and standard deviations for eleven alpha values between 0.3 and 0.7.

δt	$\bar{\sigma}_b$	$\bar{A}$	$\langle E\rangle\left(\alpha = \frac{1}{2}\right)$
2^{+3}	0.09363	0.00531	15.000
2^{+2}	0.01549	0.04123	15.000
2^{+1}	0.00706	0.21604	15.000
$2^{\ 0}$	0.00675	0.51030	15.000
2^{-1}	0.00932	0.73779	15.000
2^{-2}	0.01578	0.86633	15.000
2^{-3}	0.02723	0.93305	15.000
2^{-4}	0.05016	0.96654	15.000
2^{-5}	0.08474	0.98316	15.000
2^{-6}	0.11555	0.99174	15.000
2^{-7}	0.17070	0.99612	15.000

2. *Test of Gradient descent*

Before moving on to the interacting, more interesting systems, it would be beneficial to test the gradient descent implementation now that we are certain that it moves towards the optimum variational parameter $\alpha = \frac{1}{2}$.

B. Interacting elliptical harmonic oscillator

We now make a change to the simple systems in the previous section in order to make everything a bit more interesting. The potential in the one-body Hamiltonian is perturbed in the z-direction according to Equation 51 by setting $\lambda = \sqrt{8} \approx 2.82843$. This corresponds to enclosing particles of the simulation in an elliptical trap instead of a spherical trap.

We also choose a new trial wavefunction with the elliptical gaussian single particle functions in Equation 52 and the Jastrow factor. The hard sphere radius is set to $a/a_{\text{ho}} = 0.0043$.

Due to similarities with the spherical harmonic oscillator system we make a guess that the true minimum of this new system should be situated close to the minimum of the previous system. Choosing seven values for $\alpha \in [0.2, 0.7]$ we run importance sampling on this system for $N = \{10, 50, 100\}$ particles in $d = 3$ dimensions using 2^{21} Monte Carlo cycles[5] and an additional 10% of the Monte Carlo cycles for thermalization of the system prior to doing any sampling. We have used $\delta t = 0.1$ yields a high acceptance ratio, but slower convergence.

In Table V the system is set to have $N = 10$ interacting particles. Notice first that the energy is higher in the

[5] This turned out to be a little bit overkill due to the fast convergence of importance sampling.

TABLE V: Simulation results for $N = 10$ bosonic interacting, elliptical, harmonic oscillators.

α	$\langle E_L \rangle$	σ_b	A	t_{CPU}[s]
0.2	35.17548	0.05399	0.990	24.5
0.3	27.62004	0.02311	0.981	13.6
0.4	24.97850	0.00827	0.972	13.5
0.5	24.39877	0.00030	0.961	13.4
0.6	24.83863	0.00604	0.950	13.3
0.7	25.82855	0.01059	0.938	13.2
0.8	27.22895	0.01436	0.924	13.1

TABLE VI: Simulation results for $N = 50$ bosonic interacting, elliptical, harmonic oscillators.

α	$\langle E_L \rangle$	σ_b	A	t_{CPU}[s]
0.2	181.69371	0.25533	0.987	199.9
0.3	142.61591	0.10782	0.978	195.8
0.4	129.88615	0.04051	0.969	190.4
0.5	127.29926	0.00595	0.957	195.1
0.6	129.97630	0.03300	0.946	193.2
0.7	135.67382	0.05785	0.933	192.0
0.8	143.23238	0.07286	0.921	192.7

interacting system than for a comparable non-interacting system (Table III). Moreover, the standard deviation gets lower as the system is close to an optimal variational parameter.

The result of simulations with a higher number of particles, $N = 50$ and $N = 100$, are shown in Table VI and Table VII respectively. We observe that the energy for systems of interacting particles increases rapidly as the size of the system increases.

1. *Finding the minimum*

Using the method of gradient descent we choose a set of six starting $\alpha_0 \in [0.2, 0.8]$ and try to locate a value for α where the gradient goes to zero. We looked at $N = 10$ with $d = 3$, $\omega = 1$ and $\beta = \lambda = \sqrt{8}$. This yields the value for the minimum α to be

$$\alpha = 0.49744 \pm 0.00002. \tag{72}$$

TABLE VII: Simulation results for $N = 100$ bosonic interacting, elliptical, harmonic oscillators.

α	$\langle E_L \rangle$	σ_b	A	t_{CPU}[s]
0.2	375.04401	0.51475	0.985	745.2
0.3	296.76879	0.20214	0.975	741.0
0.4	270.60830	0.08113	0.945	723.1
0.5	266.37263	0.02020	0.953	702.8
0.6	272.51171	0.07366	0.941	689.1
0.7	285.10285	0.11381	0.929	707.5
0.8	301.40609	0.15458	0.916	707.9

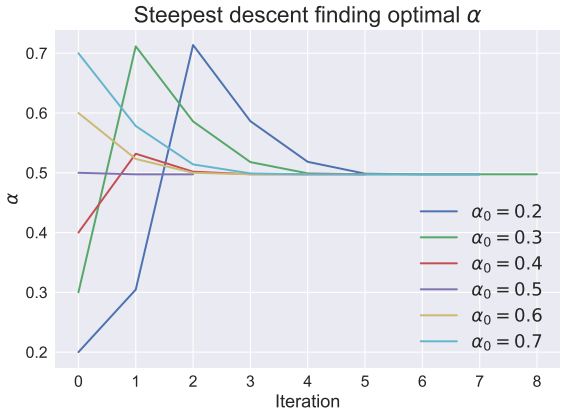


FIG. 4: In this figure we can see the convergence of different starting values for α towards the true value of α that minimizes the expected energy in the interacting case.

In Figure 4 we can see how the conjugate gradient method "moves" to find the optimal value of α.

VII. DISCUSSION

A. Energy per particle

For the non-interacting harmonic oscillator system there is no difference between a new "particle" and new dimensions [6]. This means that the inclusion of a new particle or dimension yields a linear scaling in the energy. Thus if we compute the energy per particle for systems of varying size we expect to get the same energy. This is most easily seen in Figure 6 and in the expression for the exact energy Equation 21. This, however, is *not* the case in the interacting system due to the Jastrow factor. Looking at Table V and Table VII we see that multiplying the expected energy of theformer with 10 yields a lower number than the latter. We can also see this behavior in Figure 5, where the energy per particle for $N = \{10, 50, 100\}$ have been plotted in the same figure.

Appendix A: Brute Force Metropolis-Hastings

Appendix B: Variational parameter gradient of the expectation energy

Here we show how to arrive at the expression shown in Equation 69. This requires us to restrict our view to real

[6] This is not entirely true in the case of our implementation of the Metropolis algorithms as we draw a single random particle and move all dimensions instead of iterating over all particles and all dimensions

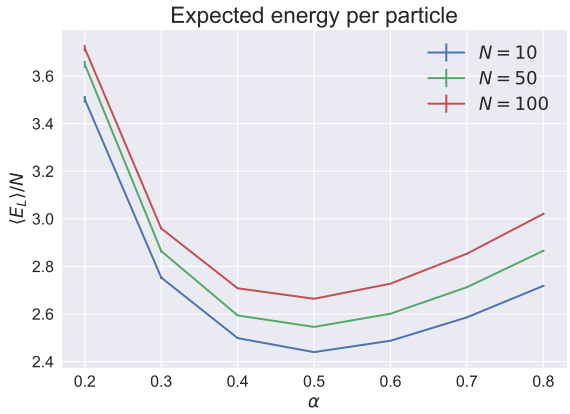


FIG. 5: Local energies as a function of variational parameter α for different number of particles.

trial wavefunctions, i.e., $\Psi_T^* = \Psi_T$, which is the case for the wavefunctions we are exploring.

$$\nabla_\alpha \langle E_L \rangle = \nabla_\alpha \left(\frac{\int d\mathbf{r} \Psi_T^* H \Psi_T}{\int d\mathbf{r} |\Psi_T|^2} \right) \tag{B1}$$

$$= -\frac{2\left(\int d\mathbf{r} \Psi_T H \Psi_T\right)}{\left(\int d\mathbf{r} \Psi_T\right)^2} \int d\mathbf{r} \Psi_T \nabla_\alpha \Psi_T \tag{B2}$$

$$+ \frac{2}{\int d\mathbf{r} \Psi_T^2} \int d\mathbf{r} \Psi_T H \left[\nabla_\alpha \Psi_T\right] \tag{B3}$$

$$= -2 \langle E_L \rangle \frac{1}{\int d\mathbf{r} \Psi_T^2} \int d\mathbf{r} \Psi_T^2 \left(\frac{\nabla_\alpha \Psi_T}{\Psi_T} \right) \tag{B4}$$

$$+ \frac{2}{\int d\mathbf{r} \Psi_T^2} \int d\mathbf{r} \Psi_T^2 E_L \left(\frac{\nabla_\alpha \Psi_T}{\Psi_T} \right). \tag{B5}$$

We now use the definition of the expectation value to group terms together. We are thus left with

$$\nabla_\alpha \langle E_L \rangle = 2 \left(\left\langle E_L \frac{1}{\Psi_T} \nabla_\alpha \Psi_T \right\rangle - \langle E_L \rangle \left\langle \frac{1}{\Psi_T} \nabla_\alpha \Psi_T \right\rangle \right), \tag{B6}$$

which is what we wanted to show.

[1] J. DuBois and H. Glyde, Physical Review A **63**, 023602 (2001).
[2] J. Nilsen, J. Mur-Petit, M. Guilleumas, M. Hjorth-Jensen, and A. Polls, Physical Review A **71**, 053610 (2005).

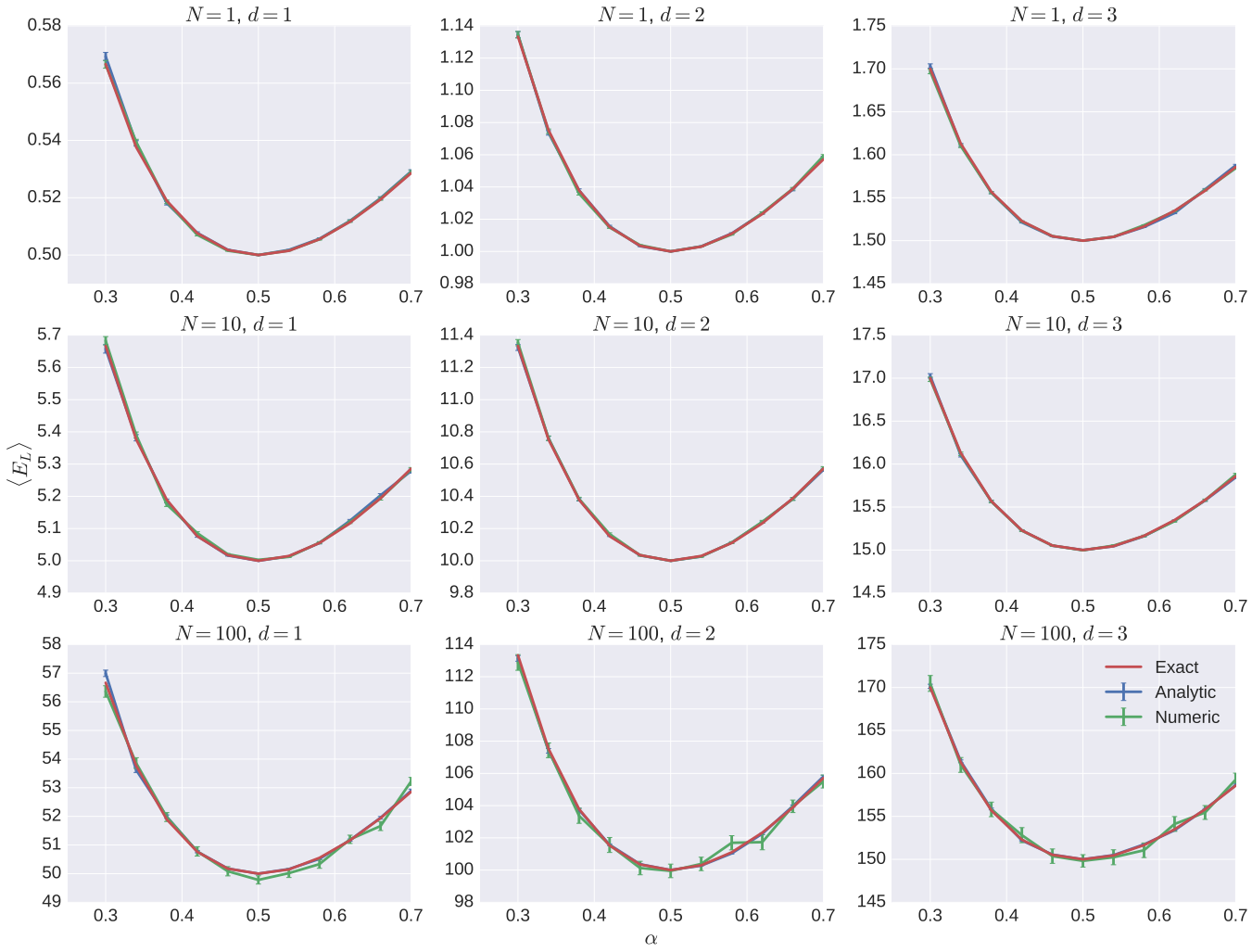


FIG. 6: Comparison of analytic and numeric results with the exact expression for the energy as a function of α. The plots show the standard deviation from the blocking method as error ticks. In the figure N is the number of particles and d the number of dimensions.