H2b 9: Variational Monte Carlo

Alfred Juhlin Onbeck

December 16, 2022

Task №	Points	Avail. points
Σ		

Introduction

Implementation of the Metropolis algorithm inorder to sample the trial wave function with a parameter α and to optimize α to approximate its ground state. This technique is called the variational Monte Carlo method and is based on the variational method from quantum mechanics and the Metropolis algorithm. The variational method is in general stated as

$$E\left[\psi_T(\alpha)\right] = \frac{\langle\psi_T(\alpha)|\,\mathcal{H}\,|\psi_T(\alpha)\rangle}{\langle\psi_T(\alpha)|\psi_T(\alpha)\rangle} \geq E_0,$$

where $E\left[\psi_T(\alpha)\right]$ is the energy, $\psi_T(\alpha)$ is the trial wave function, α is a parameter and E_0 is the energy of the ground state. If the quantum state is made of several particles, the integral will be of a very high dimension covering all of the degrees of freedom and will be difficult to compute directly. Solving the integral using the Monte Carlo integration method with Metropolis algoritm can be helpful for such high dimensional integrals.

First we have to rewrite the integrand in factors of $E_L(\mathcal{R}), \rho(\mathcal{R})$

$$E\left[\psi_T(\alpha)\right] = \int E_L(\mathcal{R})\rho(\mathcal{R})d\mathcal{R},\ E_L = \frac{\mathcal{H}\psi_T(\mathcal{R})}{\psi_T(\mathcal{R})},$$

where E_L is the local energy and

$$\rho(\mathcal{R}) = \frac{|\psi_T(\mathcal{R})|^2}{\int |\psi_T(\mathcal{R})|^2 d\mathcal{R}},$$

is the wave function's probability distribution and $\mathcal{R}$ is combination of all the coordinates for every particle in the system[1]. Note that atomic units (shortened with a.u.) are used for all of the energy values presented here and corresponds to $1\,\text{a.u.} = 1\,\text{E}_\text{h} \approx 27.2114\,\text{eV}$.

Problem 1

Here we will consider the helium atom, and using the ansatz of the trial wave function for the two electrons defined as

$$\psi_T\left(\boldsymbol{r}_1, \boldsymbol{r}_2\right) = \exp\left[-2(r_1 + r_2) + \frac{r_{12}}{2\left(1 + \alpha r_{12}\right)}\right],$$

where $r_1 = |\boldsymbol{r}_1|,\ r_2 = |\boldsymbol{r}_2|,\ r_{12} = |\boldsymbol{r}_1 - \boldsymbol{r}_1|$ and α is a parameter that can be optimized later. From this definition of ψ_T we get the local energy as

$$E_L\left(\boldsymbol{r}_1, \boldsymbol{r}_2\right) = -4 + \frac{\left(\hat{\boldsymbol{r}}_1 - \hat{\boldsymbol{r}}_2\right)\cdot\left(\boldsymbol{r}_1 - \boldsymbol{r}_2\right)}{r_{12}\left(1 + \alpha r_{12}\right)^2}$$
$$-\frac{1}{r_{12}\left(1 + \alpha r_{12}\right)^3} - \frac{1}{4\left(1 + \alpha r_{12}\right)^4} + \frac{1}{r_{12}},$$

where $\hat{\boldsymbol{r}}_i = \boldsymbol{r}_i / r_i$. Before the α parameter can be optimized we will let it be $\alpha = 0.1$.

A necessary step in the Metropolis algorithm is the trail change, or in this implemenation, a new set of coordinates. This trail change was implemented through randomly choosing one of the six coordinates and displacing randomly with uniform distribution of $[-d/2, d/2]$ and its mean value being the previous coordinate. The spread d of the uniform distribution is a constant during the algorithm and was fine tuned to $d = 4$ so the acceptance ratio of algorithm was at approximately 0.35 ± 0.05. The first set of coordinates were chosen randomly and uniformly within ± 25.

After a new set of coordinates have been established, a comparison of $|\psi_T(\boldsymbol{r}_{1,new}, \boldsymbol{r}_{2,new})|^2$ and $|\psi_T(\boldsymbol{r}_{1,old}, \boldsymbol{r}_{2,old})|^2$ is done as: if

$$|\psi_T(\boldsymbol{r}_{1,new}, \boldsymbol{r}_{2,new})|^2 / |\psi_T(\boldsymbol{r}_{1,old}, \boldsymbol{r}_{2,old})|^2 > U[0, 1)$$

the new set of coordinates replaces the old and if not, the old coordinates will be used to as the base for a new set again. Here U[0, 1) is a random number in the uniform distribution from and including 0 up to, but not including, 1. Note that $|\psi_T(\boldsymbol{r}_1, \boldsymbol{r}_2)|^2$ is not normalized and does not have to be since they only their ratio is evaluated during the algorithm.

The algorithm is stopped after a great deal of trail changes and the number of samples used here was $N = 15\,000\,000$.

In fig. 1 the sampled $|\psi_T(r)|^2$ can be seen as well as $\rho(r) = Z^3 4r^2 e^{-2Zr}$ from the central field approximation, using $Z = \{2,\ 27/16\}$. There is a minor difference amongst the sampled probability and the cental field approximations. Which was expected due to $|\psi_T|^2$ is proportional to the central field approximation with $Z = 2$, if you disregard the added $r_{12}-$term accounting for electron-electron repulsion.

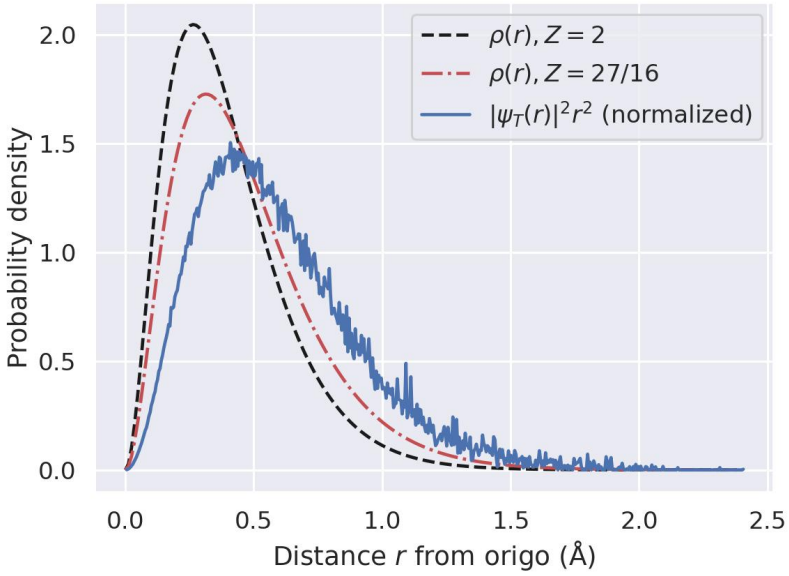

Figure 1: Probalility distribution of the sampled function and two central field approximations with $Z = 2,\ Z = 27/16$.

How correlated are the two electrons? By letting

$$x = \cos\theta = \frac{\boldsymbol{r}_1 \cdot \boldsymbol{r}_2}{|\boldsymbol{r}_1||\boldsymbol{r}_2|}$$

and observing its distribution we have some insight into their correlation. If their positions were independent of eachother, the $\cos\theta$ distribution should be uniform but as seen in the histogram in fig. 2 it is not uniform. The distribution shows that the electron are more often on opposite side of the helium atom ($\cos\theta \approx -1$) rather than on the same side ($\cos\theta \approx 1$). Due to the electron repelling eachother with the Coulomb force this is expected for small r_{12} distances, so some correlation is definitely expected.

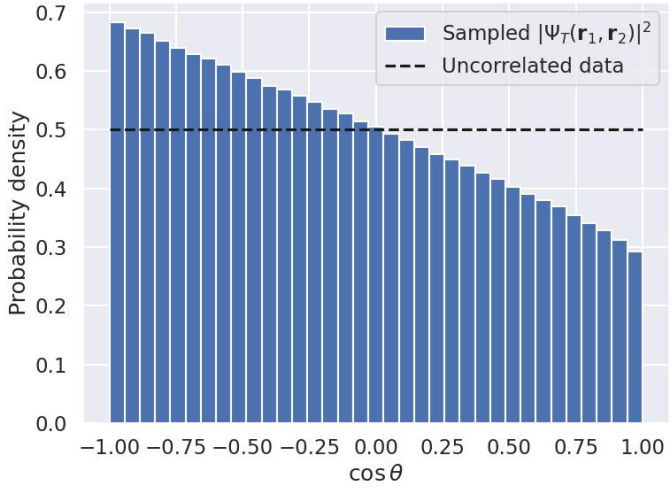

Figure 2: Correlation between the two electrons. Uncorrelated events give a uniform distribution.

Since the initial positions of the electrons can vary a great deal, it will take a certain number of iterations for the local energy to settle. This number will be denoted by N_{eq} and represents the time (steps) for the system to equilibrate. All samples during this equilibrium is mostly worthless and statistically insignificant. This is however not much of a problem due to the large amount of total steps and their relatively short computation time, for reference 10^6 steps took ≈ 1 s and

$N_{eq} \lesssim 1000$. In fig. 3 the local energy is shown for the first 3000 steps showing a drift of the mean for the first ~ 500 steps which corresponds to N_{eq}. To have a large margin for error, N_{eq} was set to the first 1 000 iterations. The ground state energy is approximated to $E_0 = \langle E_L \rangle \pm \frac{\sigma}{\sqrt{N/n_s}} = -2.8779 \pm 0.0005$ a.u., which is within reason and n_s denotes the statistical inefficiency was approximated to $n_s = 16.0$.

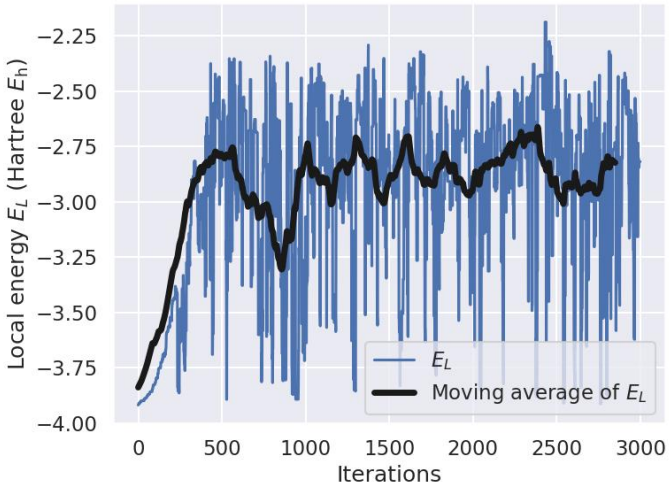

Figure 3: Local energy E_L and a moving average during the Metropolis algorithm, visualising the systems time to equilibrate (~ 500 iterations).

Due to the data from the Metropolis algorithm can be very correlated, the actual variance of the data will be from the uncorrelated samples that are independent of eachother. Let E_i denote the dataset with M measurements of a variable E, the energy and

$$I = \frac{1}{M} \sum_{i=1}^{M} E_i$$

is the mean. To determine the variance of I you need the number of uncorrelated, independent samples,

$$\text{Var[I]} = \frac{1}{M/n_s} \text{Var}[E],$$

where M/n_s is number of uncorrelated samples and n_s is the statistical inefficiency. A way to determine the statistical inefficiency is the use of the correlation function

$$\Phi_k = \frac{\langle E_i E_{i+k} \rangle - \langle E \rangle^2}{\langle E^2 \rangle - \langle E \rangle^2}.$$

For large $k, k > M_c$ it will decay to zero and if the total steps of algorithm is greater than M_c the variance of the set of energies I can be written as[1]

$$\text{Var}[I] = \text{Var}[E] \frac{1}{M} \sum_{k=-M_c}^{M_c} \Phi_k,$$

$$n_s = \sum_{k=-M_c}^{M_c} \Phi_k.$$

In fig. 4 the correlation function can be seen and there n_s is the iteration that has the correlation value that is the closest to e^{-2}, which results in $n_s = 15$. Using the sum of $\Phi(k)$ the value comes closer to $n_s \approx 16.6$.

Another way of determining the statistical inefficiency is through a method called block averaging. Block averaging is dividing the data in to blocks of size M_B until the mean of blocks becomes uncorrelated and

$$\frac{B \,\text{Var}[F]}{\text{Var}[E]} \to n_s, \text{as } B \text{ becomes large,}$$

converges towards the statistical inefficiency n_s, where B is the block size and Var$[F]$ is variance of the blocks. From this method the statistical inefficiency is $n_s \approx 15.8$ (can be seen in fig. 5), which is inbetween the values obtained using the correlation function. For approximating the variance the statistical inefficiency will be used as $n_s = 16.0$, with variance $\text{Var}[I] = \text{Var}[E]/(N/n_s)$.

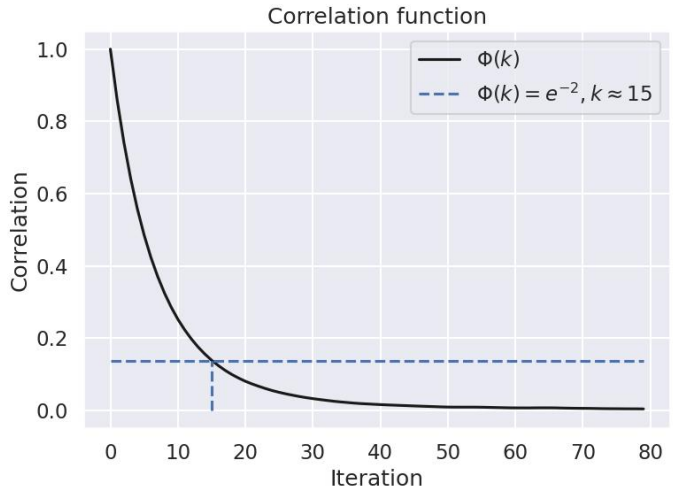

Figure 4: Correlation function to determine the statistical inefficiency. $n_s \approx 15$ steps, that is when the correlation is $e^{-2} \sim 0.135$.

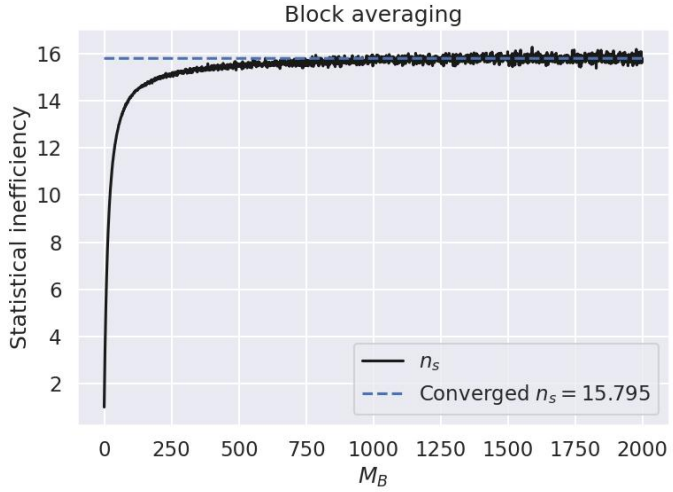

Figure 5: Block averaging, dividing the data in to blocks of size M_B until the data becomes uncorrelated and converge to the statistical inefficiency.

Now that the statistical inefficiency is determined we can visualise the local energy's mean and standard deviation for a range of different α values. This can be seen in fig. 6 and where a minimum is seen around $\alpha \approx 0.15$ with a corresponding ground state energy of $E_0 = -2.878 \pm 0.00075\,\mathrm{E}_h$. Using a quadratic fit around the observed minimum the optimal α can be approximated a bit better, with minimum at $\alpha = 0.14286$. No estimate is made here of the variance for the minimum in the quadratic fit. One possible way of estimating its variance is through using a set of different seeds and repeating the sweep for each seed. A total of 50 α-parameters were tested each using $N = 1.5 \cdot 10^7$ steps and showing errorbars for every third α-point.

While a decent approximation can be found from the sweep with a quadratic fit in fig. 6, a direct optimization can be done using a gradient descent method, this will be necessary for higher degrees of optimization where a sweep of the parameters is not feasible. The optimization will be done using dampenend gradient descent defined as

$$\boldsymbol{\alpha}_{p+1} = \boldsymbol{\alpha}_p - \gamma_p \nabla_{\boldsymbol{\alpha}} E\left(\boldsymbol{\alpha}_p\right),$$

where γ_p is

$$\gamma_p = p^{-\beta}.$$

Due to fluctuations in the local energy direct numerical differention through small finite differences is hard, but by differentiating $E(\boldsymbol{\alpha})$ analytically we get the following expression of the gradient for an easier numerical differention[2]

$$\nabla_{\boldsymbol{\alpha}} E(\boldsymbol{\alpha}) = 2\left[\langle E_L(\mathcal{R}) \nabla_{\boldsymbol{\alpha}} \ln \Psi_{\mathrm{T}}(\mathcal{R})\rangle - \langle E_L(\mathcal{R})\rangle \langle \nabla_{\boldsymbol{\alpha}} \ln \Psi_{\mathrm{T}}(\mathcal{R})\rangle\right].$$

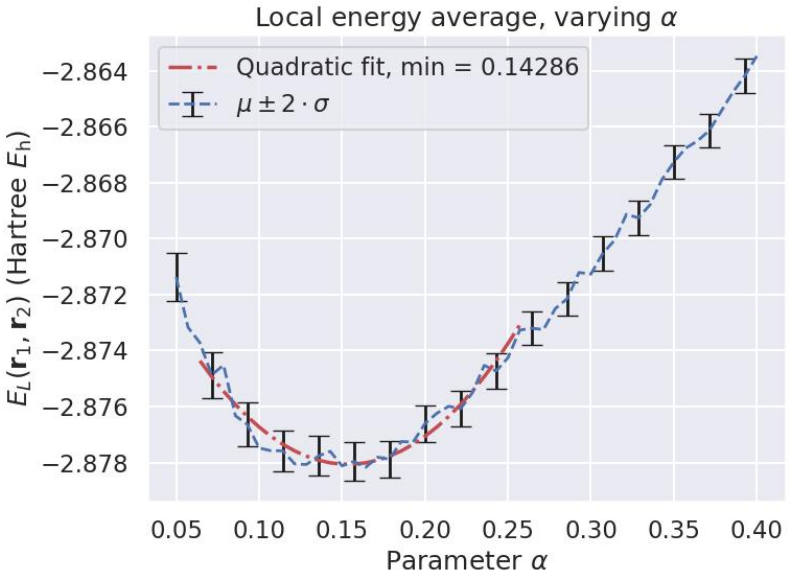

Figure 6: Mean and error bars for the local energy for a range of 50 α values $\alpha = [0.05, 0.40]$, error bars correspond to two times the standard deviation and is present for every third datapoint. A quadratic fit is also shown to approximate the minimum.

The results from the gradient descent can be seen in fig. 7, where the ground state was found at $\alpha = 0.14307 \pm 0.00007$ using $\beta = 0.75$. The error is based of the standard deviation of the last 15 α values from the gradient descent. This α value is quite close to the one obtained from the sweeping method with a quadratic fit but the gradient descent method should have a lower error. From the two methods, the gradient descent method is preferred due to it also scales up to higher dimensions of α and also has lower error. However, the sweeping method could be adequate for certain 1D optimization problems and possibly even in two dimensions.

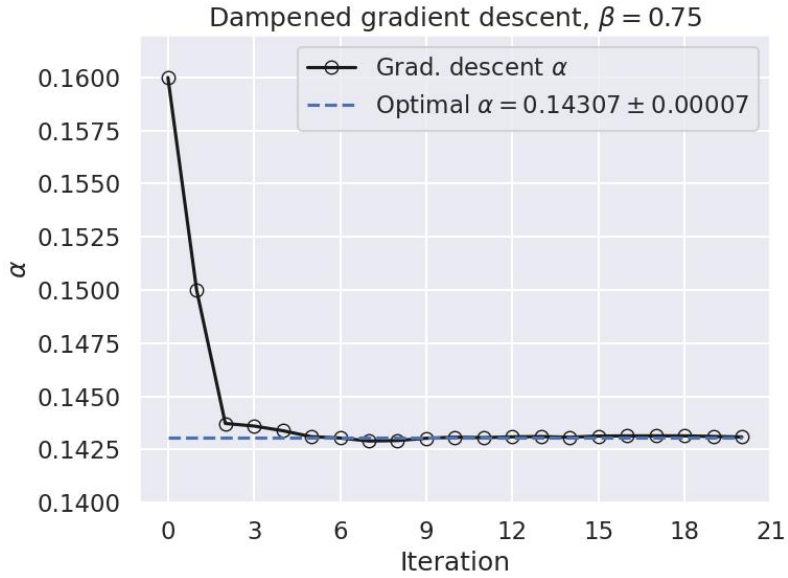

Figure 7: Dampened gradient descent to find the ground state energy.

References

[1] Göran Wahnström, (2019). Monte Carlo - Lecture notes. [Accessed 30th november 2022]

[2] Göran Wahnström, (2021). Quantum Structure - Lecture notes. [Accessed 30th november 2022]