UiO : Universitetet i Oslo

FYS3150 - Computational Physics

Project 3 - Numerical integration, calculating the correlation energy of two electrons in a helium atom.

Björn Áki Jósteinsson

October 21, 2019

The correlation energy between to two electrons in a helium atom was calculated using both Gaussian Quadrature and Monte Carlo methods. By using a combination of Legendre polynomials and Laguerre polynomials with Gaussian Quadrature and solving the correlation energy integral on a spherical form better results were obtained rather than by using only Legendre polynomials in Cartesian coordinates. Utilizing importance sampling in Monte Carlo calculations and solving the integral in spherical coordinates gives a better result than using brute force Monte Carlo for the same number of iterations. By using importance sampling and sufficient number of iterations a result closest to the known analytical value out of the two methods in question is obtained. When calculating Monte Carlo integration a significant speed increase is achieved by using parallelization.

Introduction and theory:

Integration is a vital part of physics and often integrals have to be evaluated numerically when an analytical result cannot be obtained. There are many different methods for numerical integration with the simplest being the trapezoidal, Simpson's methed etc. By using some clever properties of orthogonal polynomials numerical integration can be improved using Gaussian-Quadrature. These methods mentioned above are often referred to as main stream methods. They become slow and inefficient when evaluating multidimensional integrals which often arise in physics and many other fields. This is where Monte Carlo methods come to play. They use statistical methods to evaluate integrals. Though they can be used for low dimensional integrals their strength becomes apparent when used for solving multidimensional integrals.[1]
Many quantum mechanical problems are not solvable analytically which motivates the use of numerical methods. Here we will look at a problem which is to calculate the correlation energy between two electrons in a helium atom. Each electron is approximated by the wave function of an electron in the hydrogen atom. For an electron in the state 1s where $\mathbf{r_i}$ is the location vector[2],

$$\mathbf{r}_i = x_i\mathbf{e}_x + y_i\mathbf{e}_y + z_i\mathbf{e}_z \tag{1}$$

we can write the wave function as,[2]

$$\psi_{1s}(\mathbf{r}_i) = e^{-\alpha r_i} \tag{2}$$

where α is a parameter corresponding to the charge of the helium atom, $Z = 2$. Thus here $\alpha = 2$. This means the two electron wave function is a product of the two 1s electrons,[2]

$$\Psi(\mathbf{r}_1, \mathbf{r}_2) = e^{-\alpha(r_1+r_2)} \tag{3}$$

The integral to solve is then the correlation energy of the two electrons which stems from the classical Coulomb interaction of the two electrons. The integral is,

$$\langle \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} \rangle = \int_{-\infty}^{\infty} d\mathbf{r}_1 d\mathbf{r}_2 e^{-2\alpha(r_1+r_2)} \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|}. \tag{4}$$

For all of this project normalization is not essential to the methods being used and thus not taken into account.

A closed form expression can be found for this integral, the value of which is $\frac{5\pi^2}{16^2}$[2].
To solve this integral two types of numerical integration methods will be utilized. First Gaussian-Quadrature both with Legendre and Laguerre polynomials. A six dimensional integral of this sort is on the verge where main stream methods such as Gaussian-Quadrature become inefficient. After this threshold Monte Carlo methods show their strength. Thus Monte Carlo methods are used for solving the integral as well. Under the results section calculation of the integral 4 for the two methods and two instances for each method are found. All code and measurements can be found on the following address: `https://github.com/bjornaki/CompPhysics`.

Gaussian Quadrature:

Gaussian Quadrature relies on the properties of orthogonal polynomials. As with other methods the integral is approximated by choosing a set of integration points and weights. With Gaussian Quadrature however these integration points and weights are determined by these orthonormal polynomials and their roots. This means the integration points are not evenly spaced. A more detailed explanation and introduction to Gaussian Quadradature using different orthogonal polynomials can be found here[1].
A few different types of orthonormal polynomials can be used in accordance to the integration interval. On $[-1, 1]$ we have Legendre polynomials which can be expanded onto an arbitrary interval $[a, b]$ using a mapping,[1]

$$t = \frac{b-a}{2}x + \frac{b+a}{2}, \qquad x \in [-1, 1] \tag{5}$$

On $[0, \infty]$ we have Laguerre polynomials.
With Gaussian Quadrature one can solve integral 4 in Cartesian coordinates. The single electron wave function e^{-4r_i} dies sufficiently fast out so that we can approximate infinity as a constant, here taken to be $\lambda = 3$. This can be seen on figure 1 where the single electron wave function amplitude is plotted as a function of x-coordinate with y- and z-coordinates taken to be zero. This is true for all six variables thus for each of those infinity is approximated as $\lambda = 3$.

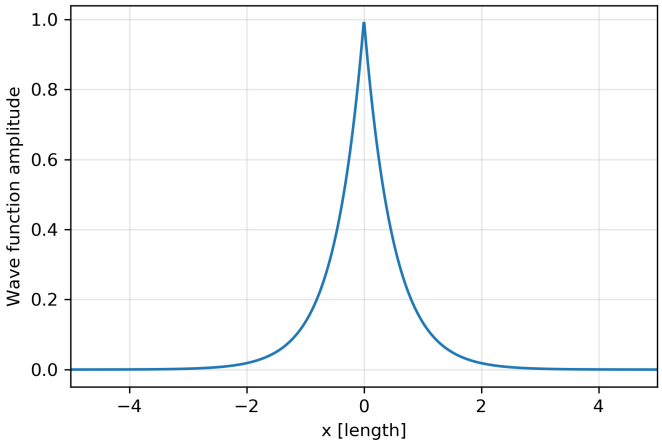

Figure 1: Amplitude of a single electron wave function plotted as a function of x-coordinate with y- and z-coordinate taken to be zero for plotting purposes.

Then using Gaussian Quadrature and Legendre polynomials in conjunction with the mapping 5 one can calculate the integral.

Improved Gaussian Quadrature

Approximating infinity with a constant gives a reason to think if other polynomials are better suited to numerically calculate the integral. By switching to spherical coordinates using a change of variables the integral becomes,

$$I = \int_0^\infty r_1^2 dr_1 \int_0^\infty r_2^2 dr_2 \int_0^{2\pi} d\phi_1 \int_0^{2\pi} d\phi_2 \int_0^\pi sin(\theta_1) d\theta_1 \int_0^\pi sin(\theta_2) d\theta_2 \frac{e^{-2\alpha(r_1+r_2)}}{r_{12}} \tag{6}$$

where[2]

$$\frac{1}{r_{12}} = \frac{1}{\sqrt{r_1^2 + r_2^2 - 2r_1 r_2 cos(\beta)}} \tag{7}$$

and[2]

$$cos(\beta) = cos(\theta_1)cos(\theta_2) + sin(\theta_1)sin(\theta_2)cos(\phi_1 - \phi_2)) \tag{8}$$

For integration limits that go from zero to infinity it is more natural to use GQ in conjunction with aforementioned Laguerre polynomials which are defined on $[0, \infty)$.
The usage of Laguerre polynomials requires the integral on the form, where $W(x) = x^\gamma e^{-x}$ is the weight function[1]. This can therefore be applied to the radial parts of the integral in spherical coordinates, e.g. for r_1

$$\int_0^\infty r_1^2 e^{-4r_1} \frac{1}{r_{12}} dr_1 = \int_0^\infty W(r_1) e^{-3r_1} \frac{1}{r_{12}} dr_1 \tag{9}$$

where $W(r_1)$ is the weight function. So by using a combination of Gauss-Legendre for the angles and Gauss-Laguerre for the radial part one would expect better results.

Furthermore other orthogonal polynomials can be used for Gauss Quadrature according to different kind of integration limits and weight functions[1].

Monte Carlo:

Monte Carlo methods rely on a statistical approach to solve integrals such as 4. This is done by sampling random variables calling a random number generator and evaluating the integral a large number of times and calculating both the average value and the standard deviation. This gives an estimate for the integral value. If we have an integral of a function $f(x)$ on the interval $[0,1]$ we have that the average of the function $f(x)$ for a given probability density function, PDF, is[1]

$$\langle f \rangle = \frac{1}{N} \sum_{i=1}^{N} f(x_i) p(x_i) \tag{10}$$

When $p(x_i)$ is the uniform distribution, that is $p(x) = 1$ when $x \in [0,1]$ and zero for all other values of x we have that the value of the integral is the average of $f(x)$ over the interval $[0,1]$, that is[1]

$$\int_0^1 f(x)dx \approx \langle f \rangle \tag{11}$$

This can then be extended to an arbitrary interval $[a,b]$ with a change of variables[1]

$$z = a + (b-a)x \tag{12}$$

or on the domain $[0,\infty)$ using the change of variables[1]

$$y(x) = -ln(1-x) \tag{13}$$

Monte Carlo using importance sampling:

The change of variables according to 13 gives rise to an important aspect of Monte Carlo methods called importance sampling. This gives us a new random variable y in the domain $[0,\infty)$ from the random variable $x \in [0,1]$.
This means that if we have an integral on the form where we can factor out an exponential, e^{-y},[1]

$$\int_0^\infty F(y)dy = \int_0^\infty e^{-y} G(y)dy \tag{14}$$

We can write the integral as[1]

$$\int_0^\infty e^{-y} G(y)dy = \int_0^1 G(y(x))dx \approx \frac{1}{N} \sum_{i=1}^{N} G(y(x_i)) \tag{15}$$

where x_i is a random number in the interval $[0,1]$
This means that if we recognize a certain PDF such as the exponential distribution as is the fact in the integral 6 we can sample according to this PDF using a random number generator and the change of variables in equation 13. This results in a generally more accurate result with smaller variance.

As Monte Carlo methods are statistical in their nature a statistical interpretation is essential. This means calculating the variance and representing either the variance or the standard deviation. The variance for an integral of $f(x)$ on the interval $[0, 1]$ as[1]

$$\sigma_f^2 = \frac{1}{N}\sum_{i=1}^{N}(f(x_i) - \langle f \rangle)^2 p(x_i) \tag{16}$$

and with the uniform distribution inserted for $p(x_i)$[1]

$$\sigma_f^2 = \frac{1}{N}\sum_{i=1}^{N} f(x_i)^2 - \left(\frac{1}{N}\sum_{i=1}^{N}\right)^2 \tag{17}$$

The standard deviation is defined as

$$\sigma_f = \sqrt{\sigma_f^2} \tag{18}$$

Using both Gaussian Quadrature and Monte Carlo methods the integral 6 was calculated. Worth noting is the possibility of parallelizing Monte Carlo methods. As each iteration only requires the generation of a random number and no dependency on other iteration means the code can easily be parallelized. A thing to watch out for is that each seed (generation) for a random number is different between each thread. If care is not taken one might end up seeding each thread the same so that instead of getting different sequence of random numbers they are all the same.

Results:

Gaussian Quadrature:

In figure 2 the results for both GQ using Legendre polynomials and improved GQ using both Legendre and Laguerre polynomials can be seen. For reference the analytical value is plotted. In figure 3 the absolute value error of both methods is plotted. That is the absolute value of the difference between the analytical value and the value obtained from GQ and improved GQ. In table 1 these value can be seen in more detail and for how many integration points they were measured.

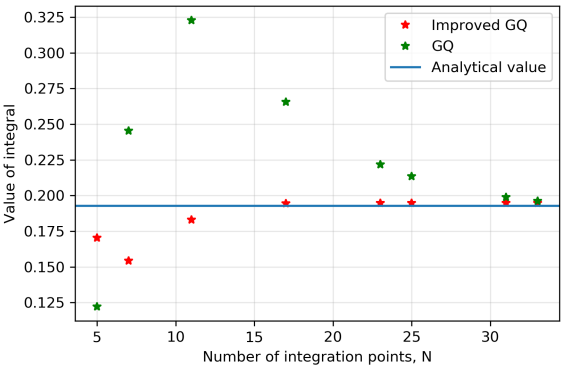

Figure 2: Results of both GQ using only Legendre polynomials and improved GQ utilizing both Legendre and Laguerre polynomials. The analytical value can be seen for reference.

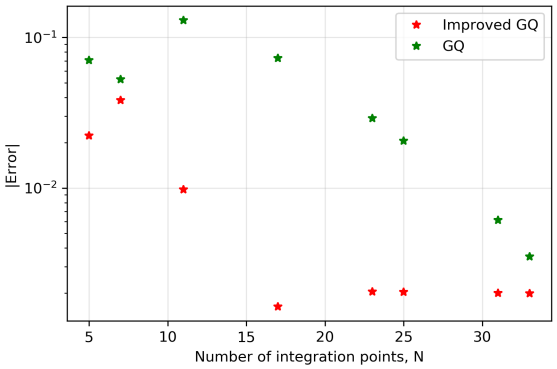

Figure 3: Absolute error for both GQ using Legendre polynomials and improved GQ using both Legendre and Laguerre polynomials. Log-y scale.

Table 1: Results for the measurements of different number of integration points. Both for GQ and improved GQ. The absolute value of the error is also shown.

N:	GQ	|Error GQ|	Improved GQ	|Error improved GQ|
5	0.12220851	0.0705572	0.17049197	0.02227374
7	0.24536729	0.05260158	0.15442207	0.03834364
11	0.32274001	0.1299743	0.18302187	0.00974385
17	0.26564817	0.07288246	0.19439573	0.00163002
23	0.22177484	0.02900913	0.19481241	0.0020467
25	0.21336473	0.02059902	0.19480424	0.00203852
31	0.19887355	0.00610784	0.19477176	0.00200605
33	0.19627595	0.00351024	0.19475479	0.00198908

Monte Carlo Integration:

In figure 4 and figure 5 one can see the results for brute force Monte Carlo method and Monte Carlo method with importance sampling. The analytical value is plotted for comparison. Table 2 shows in detail for how many iterations measurements where taken as well as giving the standard deviation.

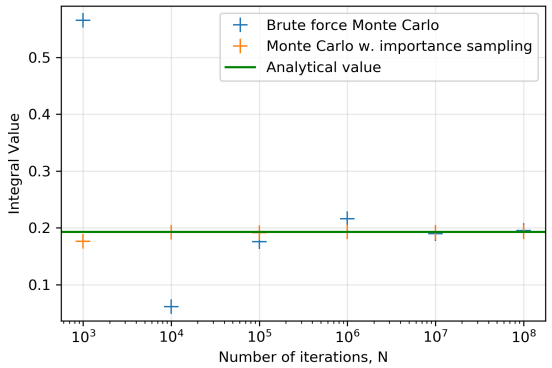

Figure 4: Results of MC integration both with and without importance sampling. The analytical value is plotted for comparison. Log-log scale

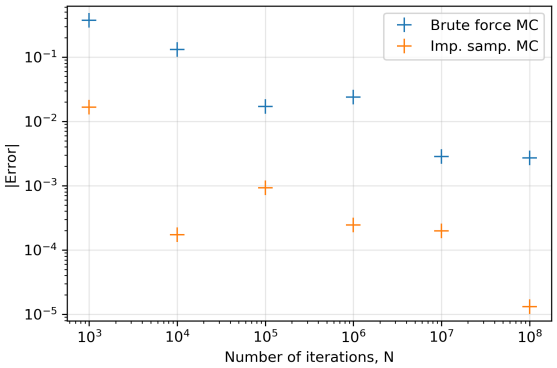

Figure 5: Absolute error for both the brute force MC and MC with importance sampling. Log-log scale

Table 2: Value of Monte Carlo calculations for different number of iterations, N. Both for brute force method and with importance sampling. The standard deviation is also given.

N:	MC	σ(MC)	MC importance sampling	σ(MC imp. samp.)
10^3	0.56497764	0.32011648	0.17618565	0.01275696
10^4	0.06170277	0.01571785	0.19259224	0.00445128
10^5	0.1758335	0.05947085	0.19184081	0.00142621
10^6	0.21646774	0.03018525	0.19300942	0.00045608
10^7	0.18991891	0.00809075	0.19256837	0.00014205
10^8	0.19546554	0.00392256	0.19275255	0.00004514

The Monte Carlo method can be easily parallelized. By comparing the time it takes to run the code we can compare the Monte Carlo method with importance sampling both with and without parallelization. This comparison can be seen in figure 6.

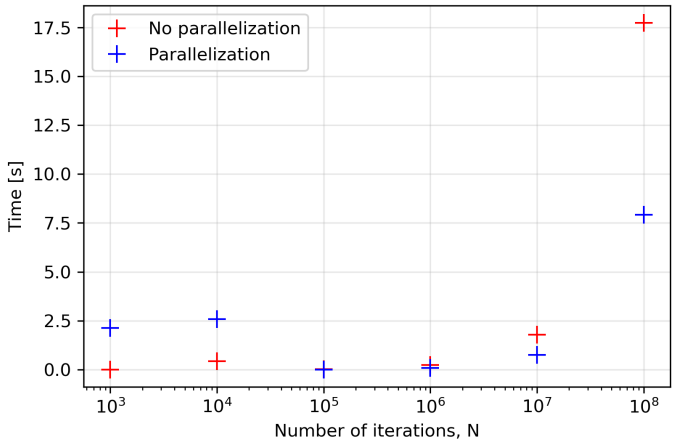

Figure 6: Timing of MC integration with importance sampling. Log-x plot. Measurements both for parallelized code and not parallelized. Time saving becomes apparent for large N.

Discussion:

Gaussian Quadrature:

In figure 2 and 3 we see that for a lower number of integration points using Gaussian Quadrature with a blend of Legendre and Laguerre polynomials is better then using only Legendre polynomials. Using Legendre in Cartesian coordinates requires the approximation of infinity. Here it was approximated as $\lambda = 3$, meaning the integration limits for all six coordinates were $[-3, 3]$.
Changing to spherical coordinates means that one obtains a closed interval for the four different angles and an interval $[0, \infty)$ for the radial part. This change of variables introduces an exponent which can be used for the weight function and Laguerre polynomials which are defined on $[0, \infty)$. A change to spherical coordinates gives the chance to use Legendre polynomials for the angles and Laguerre polynomials for the radial part. This encompasses the problem better resulting in a lower error for improved GQ versus standard GQ for the same number of integration points.

Monte Carlo Integration:

For this problem brute force MC could be implemented by using the same approximation for ∞ as earlier in Cartesian coordinates. For a low number of iterations this gives a very unsatisfactory results. This is apparent in figures 4 and 5 as well as table 2. Even for a high number of iterations the error is still substantial. This motivates the use of importance sampling. In spherical coordinates we recognize that we have an exponential term and limits that go from $[0, \infty)$. By generating random numbers from the exponential distribution it better represents the integrand and we obtain a much better calculation for the integral. This can be clearly seen in figures 4 and 5. When the number of iterations is increased to 10^8 the MC with importance sampling converges within four leading digits compared to the analytical value.

Parallelization of Monte Carlo:

The fact that each iteration is not dependent on any other iterations makes it easy to parallelize the code for MC with importance sampling. This gives a theoretical speed up dependent on the set up being used for the numerical calculations. Here a 2-core processor with hyper-threading was used, resulting in 4 virtual cores. A speed up is not apparent until for a large number of iterations, here approximately 10^7 and larger. For lower numbers it is the reverse, resulting in the parallelized code being slower. This is due to the fact that the process has to be split up between individual cores and then joined once the job is finished. For a low number of iterations this means that more time is spent on this part of the code rather then the calculations itself.

Comparison of Gaussian Quadrature and Monte Carlo methods:

For a six dimensional integral such as the one being calculated here classical method such as Gaussian Quadrature are on the verge of coming inefficient and not optimal. This is where Monte Carlo methods come to their strength. By casting the problem onto spherical coordinates a better result is obtained with GQ using Legendre and Laguerre polynomials, a result which is better than brute force MC. However if we do the same for MC, cast it onto spherical coordinates and use importance sampling, that is generate random numbers from the exponential PDF, we obtain a result which is closest to the analytical value out of all methods used here for a sufficiently large number of iterations.

Conclusions:

Here two different methods for calculating integrals numerically were introduced and applied to the problem of calculating the correlation energy of two electrons in a helium atom. These methods are Gaussian Quadrature and Monte Carlo. In both cases the methods give a better result by casting the integral 4 onto spherical coordinates. In the case of GQ this prompts the use of a combination of Legendre polynomials for the angles and Laguerre polynomials for the radial part. Moreover this rules out the necessity for approximating infinity.
Similarly for MC if we cast the integral onto spherical coordinates we can generate random numbers for the angles according to the uniform PDF and for the radial part according to exponential PDF. This represents the problem more naturally and gives a result closer to the analytical one for a lower number of iterations.
When dealing with higher dimensional integrals MC methods show their strength. Here the result closest to the analytical one was for MC with importance sampling and number of iterations $N = 10^8$. This result converged within four leading digits compared to the analytical value.
By parallelizing MC with importance sampling a significant speed up is achieved for $N = 10^8$ iterations. For a low number of iterations this is the reverse with the parallelized code taking longer to run. This is due to the fact that it takes more time to split up and join the tasks between different cores rather than doing the calculation itself.
Future work would be to further investigate the results for a large number of integration points using the Gaussian Quadrature and the optimization of MC with importance sampling using parallelization. In addition to calculating the six dimensional integral calculated here higher dimensional

integrals could be calculated for further investigation of both GQ and MC methods and seeing how MC methods become more efficient.

References

[1] Morten Hjorth-Jensen. Computational physics, lecture notes fall 2015, 2015.

[2] Morten Hjorth-Jensen. Project 3, numerical integration. https://github.com/CompPhysics/ComputationalPhysics/blob/master/doc/Projects/2019/Project3/pdf/Project3.pdf, 2019.