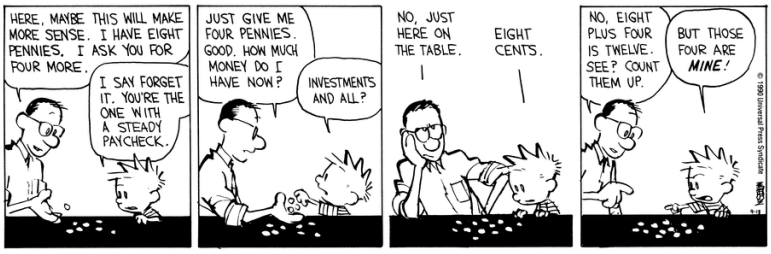
ELECTRONS IN A HELIUM ATOM

IVAR HAUGALØKKEN STANGEBY

Abstract. In this project we wish to compute the expected value of the ground state correlation energy between two electrons in a helium atom. In doing so, we also take the opportunity to discuss various methods of numerical integration. Namely Gaussian quadrature with both Laguerre and Legendre polynomials; brute force Monte Carlo integration, and using importance sampling.
All source code can be found on my GitHub page:

github.com/qTipTip



Contents

1. Introduction

We define the position of electron i as $\mathbf{r}_i = x_i\mathbf{e}_x + y_i\mathbf{e}_y + z_i\mathbf{e}_z$, and define $r_i = \sqrt{x_i^2 + y_i^2 + z_i^2}$. The *ground state correlation energy* of two electrons — and more specifically its expectation value is given by the integral

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

In this project we fix $\alpha = 2$ which corresponds to the charge of the helium atom. It can be shown that this specific integral has a closed form solution of $5\pi^2/16^2$

which we will use to assess the numerical stability and effectiveness of our numerical methods.

We will later, when we apply the Gauss-Laguerre quadrature, need our integral rewritten in spherical coordinates. The differentials are then given by

$$d\mathbf{r}_1 d\mathbf{r}_2 = r_1^2 dr_1 r_2^2 dr_2 \sin\theta_1 d\theta_1 \sin\theta_2 d\theta_2 d\phi_1 d\phi_2, \tag{2}$$

with a scale factor of

$$\gamma(\beta) = \frac{1}{r_{12}} = \left(r_1^2 + r_2^2 - 2r_1 r_2 \cos\beta\right)^{-1/2} \tag{3}$$

where

$$\cos\beta = \cos\theta_1 \cos\theta_2 + \sin\theta_1 \sin\theta_2 \cos(\phi_1 - \phi_2).$$

2. Gaussian Quadrature

In Gaussian Quadrature, the main idea is to rewrite an integral as a weighted sum of function evaluations,

$$I = \int_a^b f(x)\, dx = \sum_{i=1}^{N} \omega_i f(x_i)$$

however, we can do use various orthogonal polynomials which arise from different differential equations in the natural sciences. We will make use of both Legendre and Laguerre polynomials. The only downfall of these polynomials are that their orthogonality relation is only satisfied on certain intervals, hence we need to pick our polynomials carefully.

2.1. Legendre.

The Legendre polynomials are defined on the interval $[-1, 1]$ with a weight function $W(x) = 1$. This would lead us to believe that we can only perform integrals over the interval $[-1, 1]$, but a simple change of variables lets us use any arbitrary interval. However, one must note that this can significantly degrade the numerical accuracy. The Gauss-Legendre quadrature gives rise to the following relation:

$$I = \int_{-1}^{1} f(x)\, dx = \int_{-1}^{1} W(x) g(x)\, dx = \int_{-1}^{1} g(x)\, dx = \sum_{i=1}^{N} \omega_i g(x_i).$$

It turns out that the associated integration points x_i are the roots of the Legendre polynomials. We can rewrite eq. (1) in the Legendre form using a simple change of variables.

2.2. Laguerre.

The Laguerre polynomials are orthogonal on $[0, \infty)$ and have an associated weight function $W(x) = x^\alpha e^{-x}$. Again, the orthogonality property gives us the relation

$$I = \int_0^\infty f(x)\, dx = \int_0^\infty W(x) g(x)\, dx = \int_0^\infty x^\alpha e^{-x} g(x)\, dx = \sum_{i=1}^{N} \omega_i g(x_i).$$

In order to use the Gauss-Laguerre quadrature, we need to rewrite eq. (1) to spherical coordinates. We do this using eq. (2) and eq. (3). This gives us

$$I = \int_{-\infty}^{\infty} d\mathbf{r}_1 d\mathbf{r}_2 e^{-4(r_1,r_2)} \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|}.$$

$$= \int_0^{\infty}\int_0^{\infty}\int_0^{\pi}\int_0^{\pi}\int_0^{2\pi}\int_0^{2\pi} r_1^2 r_2^2 e^{-4r_1} e^{-4r_2} \gamma(\beta) \sin\theta_1 \sin\theta_2 dr_1 dr_2 d\theta_1 d\theta_2 d\phi_1 d\phi_2$$

Substituting $u_i = 4r_i$, $i = 1, 2$ yields

$$= \frac{1}{1024} \int_0^{\infty}\int_0^{\infty}\int_0^{\pi}\int_0^{\pi}\int_0^{2\pi}\int_0^{2\pi} u_1^2 u_2^2 e^{-u_1} e^{-u_2} \frac{\sin\theta_1 \sin\theta_2}{\sqrt{u_1^2 + u_2^2 - 2u_1 r_2 \cos\beta}} du_1 du_2 d\theta_1 d\theta_2 d\phi_1 d\phi_2$$

$$= \frac{1}{1024} \int_0^{\infty}\int_0^{\infty}\int_0^{\pi}\int_0^{\pi}\int_0^{2\pi}\int_0^{2\pi} W(u_1) W(u_2) g(u, \theta, \phi)\, du_1 du_2 d\theta_1 d\theta_2 d\phi_1 d\phi_2.$$

Based on these integral limits it can be wise to apply Gauss-Laguerre only for the first two integrals, and use Gauss-Legendre for the other four.

3. Monte Carlo Integration

In Monte Carlo integration we use a *non-deterministic* approach to computing a definite integral. We sample integration points at random. The naive approach is to just pick numbers at random in the interval $[0, 1]$, which is what we will do in the brute force Monte Carlo method however, as we will see, another method is to use so-called *importance-sampling* to find a probability distribution function (PDF) that closely resembles the function we wish to integrate. Since these methods are non-deterministic, there is always an error associated with the random aspect of the computations. We therefore, in addition to compute the integral itself, compute the standard deviation to get a grip on the uncertainty in our results. One of the effects of importance sampling is to reduce the standard deviation.

3.1. Brute force Monte Carlo.

The probability distribution function associated with the brute force Monte Carlo method is the *uniform distribution* given by

$$p(x) = \begin{cases} 1, & x \in [0, 1] \\ 0, & x \in [0, 1]^c. \end{cases}$$

Since $p(x)$ satisfy $\int_{-\infty}^{\infty} p(x)\, dx = 1$ we have that

$$\langle f \rangle = \int_{-\infty}^{\infty} p(x) f(x)\, dx = \int_0^1 p(x) f(x)\, dx \approx \sum_{i=1}^{N} f(x_i).$$

In order to apply this for more general integration limits a, b we simply note that by conservation of probability $p(y)dy = dx$, hence standard change of variables will suffice: $y(x) = a + (b - a)x$.

3.2. **Importance Sampling Monte Carlo.** We now wish to find a probability distribution function $p(y)$ that closely mimics the behavior of $f(x)$ over our chosen integration interval. For our integral, it is suitable to use the exponential distribution $p(y) = e^{-y}$, thus $p(r) = e^{-4r}$. However, in order to apply this, we must have the normalization criteria satisfied, hence

$$p(r) = Ae^{-4r}$$

is a suitable probability distribution function. Note that we wish to express our integral in spherical coordinates in order to apply this specific probability distribution.

In order to make this work for arbitrary intervals, we introduce the change of variable

$$y(x) = -\frac{\ln(1-x)}{2\alpha}$$

which will give us values $y \in [0, \infty)$.

4. Implementation

In this project we employ a lot of *black-box* functions, so there aren't that many implementation details to talk about. However, I have included some thoughts on selected methods.

4.1. **Gauss-Legendre.** In order to evaluate the integral using Gaussian Quadrature with Legendre polynomials, we need to find suitable integration limits. Clearly, we can't integrate from $-\infty$ to ∞ numerically. We observe that the function e^{-4r_i} is *essentially zero* for $r_i \approx 3$. This can be verified by the following calculations. Say we want $e^{-4r_i} = 10^{-5}$:

$$e^{-4r_i} = 10^{-5} \implies r_i = -\ln(10^{-5})/4$$

which gives us that $r_i \approx 2.88$. We can therefore chose to neglect any errors that arise from incorrect integration limits by integrating from -3 to 3 for instance.

4.2. **Gauss-Laguerre.** For the integral in spherical coordinates we chose to use Laguerre polynomials only for the two radial integrals, and leave the integrals with respect to θ and ϕ for Legendre polynomials which is more suited for such narrow integration limits.

5. Results

5.1. **Gauss-Legendre.** In table 1 we see that the convergence rate of the Gauss-Legendre method is quite poor and the numerical method can be seen as quite unstable. Increasing the number of integration steps does not necessarily translate directly into a more accurate approximation, as can be seen for the $N = 25$ to $N = 30$ jump. We have in these simulations used the integration interval $[-3, 3]$.

5.2. **Gauss-Laguerre.** As seen in table 2, converting to spherical coordinates and applying Gauss-Laguerre for the two integrals with limits 0 and ∞ dramatically increases the numerical approximation. A strong convergence towards the exact value is observed. A slight increase in elapsed time is seen, however only $N = 15$ integration steps is required for a result that matches the best achieved using only Gauss-Legendre.

5.3. **Brute Force Monte Carlo.** In table 3 we see immediately that the Monte Carlo methods require a very large number of integration steps. We also see that the results converge quickly, and the standard deviation – although fluctuating a fair bit – also decreases with increasing N. The time elapsed for this brute force method mimics the time elapsed for the brute force Gauss-Legendre method.

TABLE 1. In this table we present the computed integral, the absolute error in the calculations as well as the time spent for N integration steps for the Gauss-Legendre quadrature. The time complexity of the integral itself is $\mathcal{O}(N^6)$ which is pretty immense, and can clearly be seen based on the increase in time elapsed for increasing N.

N	Result	Absolute error	Time [sec]
5	0.2642	0.0714	0.0004
10	0.0719	0.1207	0.0267
15	0.2390	0.0463	0.3184
20	0.1561	0.0366	1.7411
25	0.1958	0.0030	6.6549
30	0.1772	0.0154	19.6983
35	0.1899	0.0028	55.4347
40	0.1844	0.0083	117.377

TABLE 2. Presented is the computed integral, the absolute error in calculations as well as time elapsed for N integration steps. The time complexity of the integral itself is again $\mathcal{O}(N^6)$ however the numerical method is converging properly as opposed to the Legendre quadrature.

N	Result	Absolute error	Time [sec]
5	0.1734	0.0193	0.0011
10	0.1864	0.0063	0.0675
15	0.1897	0.0030	0.8190
20	0.1910	0.0016	4.3892
25	0.1917	0.0010	16.8339
30	0.1921	0.0006	50.1823
35	0.1923	0.0004	128.2480
40	0.1924	0.002	291.6470

TABLE 3. Using brute force Monte Carlo integration we see that the results converge to the exact fairly quickly. The computed standard deviation however fluctuates a fair bit. None the less, the brute force Monte Carlo method is have similar performance to the brute force Gauss-Legendre integral.

N	Result	Absolute error	Standard Deviation	Time [sec]
10^3	0.0235	0.1691	0.0089	0.00007
10^4	0.0473	0.1453	0.0200	0.00073
10^5	0.1732	0.0195	0.0497	0.00632
10^6	0.1759	0.0168	0.0159	0.06715
10^7	0.1829	0.0097	0.0080	0.64801
10^8	0.1981	0.0053	0.0030	6.46146
10^9	0.1938	0.0010	0.0010	64.5177