

# Project 2, Schrödinger's equation for two electrons in a three-dimensional harmonic oscillator well

Andreas Ellewsen, Peder Forfang

## 1. Introduction

The aim of this project is to solve Schrödinger's equation for two electrons in a harmonic oscillator potential, with, and without a repulsive Coulomb interaction. This is done using the Jacobi rotation method. First for one electron in a harmonic oscillator potential to test the precision of the method, and then for both variations with two electrons.

## 2. Discretizing the wave equation

To solve this using the Jacobi method, we first have to discretize the equation and rewrite it to a square matrix. The radial part of the wave equation for a single electron in a harmonic oscillator potential reads

$$-\frac{\hbar^2}{2m} \left( \frac{1}{r^2} \frac{d}{dr} r^2 \frac{d}{dr} - \frac{l(l+1)}{r^2} \right) R(r) + V(r)R(r) = ER(r). \quad (1)$$

In our case  $V = 1/2kr^2$ , with  $k = m\omega^2$ , and

$$E_{nl} = \hbar\omega \left( 2n + l + \frac{3}{2} \right). \quad (2)$$

Here  $E_{nl}$  is the energy of the harmonic oscillator in three dimensions, and  $n = 0, 1, 2, \dots, l = 0, 1, 2, \dots$ . In this project we will use  $l = 0$ . We introduce the dimensionless variable  $\rho = r/\alpha$  where  $\alpha$  is a constant with dimension length. Rewriting equation 1 we end up with

$$-\frac{d^2}{d\rho^2} u(\rho) + \rho^2 u(\rho) = \lambda u(\rho). \quad (3)$$

where

$$\lambda = \frac{2m\alpha^2}{\hbar^2} E, \quad (4)$$

and

$$\alpha = \left( \frac{\hbar^2}{mk} \right)^{1/4}. \quad (5)$$

If we now use that the second derivative of a function  $u$  can be written

$$u'' = \frac{u(\rho + h) - 2u(\rho) + u(\rho - h)}{h^2} + O(h^2), \quad (6)$$

where  $h$  is our step length, defined as

$$h = \frac{\rho_{\max} - \rho_{\min}}{n_{\text{step}}} \quad (7)$$

giving us

$$\rho_i = \rho_{\min} + ih, \quad i = 0, 1, 2, \dots, n_{\text{step}}. \quad (8)$$

Using this we can rewrite equation 3 to the discretized form

$$-\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + V_i u_i = \lambda u_i \quad (9)$$

which is the one we will be using. Solving this numerically, we will define a tridiagonal matrix where

$$d_i = \frac{2}{h^2} + V_i \quad (10)$$

is the diagonal elements, and

$$e_i = -\frac{1}{h^2} \quad (11)$$

is the non-diagonal elements. Finally we can rewrite the Schrödinger equation to

$$d_i u_i + e_{i-1} u_{i-1} + e_{i+1} u_{i+1} = \lambda u_i. \quad (12)$$

This is the equation we will be using Jacobi's method on.

## 3. Solving the discretized wave equation with Jacobi's method

To solve this we make a program in c++ implementing Jacobi's method. The program can be seen in the github repository linked in the references.

We compare our result with the analytical solutions of the eigenvalues. The lowest solutions are 3, 7 and 11. To get a 4 digit precision of the lowest three, we need  $n_{\text{step}} = 200$ . The eigenvalues depend on the value of  $\rho_{\max}$ , which is expected as  $\rho_{\max}$  is the maximum distance from the potential well. In our computation we find that the solutions with  $\rho_{\max} = 5$  fits well with the analytical solutions. The number of similarity transformations needed before the non-diagonal elements in the matrix are essentially zero depend on how many  $n_{\text{step}}$  we choose. With the 4 digit precision we reach a number of 71168 iterations. Trial and error for different values of  $n$  reveals that the number of iterations is proportional to  $n^2$ . More precisely the number of iterations needed is approximately  $7n^2/4$ .

Comparing our algorithm to the one in the armadillo library reveals that the two methods produce exactly the same results. However the armadillo solver is a lot faster than our method. For  $n = 100$ , our method uses 1.20562 seconds, while armadillo uses 0.004314 seconds.

Next we will study two electrons in a harmonic oscillator well which also interact via a repulsive Coulomb interaction. Setting up the equation for this results in

$$\left( -\frac{\hbar^2}{2m} \frac{d^2}{dr_1^2} - \frac{\hbar^2}{2m} \frac{d^2}{dr_2^2} + \frac{1}{2} k r_1^2 + \frac{1}{2} k r_2^2 \right) u(r_1, r_2) = E^{(2)} u(r_1, r_2). \quad (13)$$

State	$\omega_r = 0.01$	$\omega_r = 0.5$	$\omega_r = 1$	$\omega_r = 5$
1	0.105775	2.23009	4.05784	17.4484
2	0.141508	4.13433	7.90952	37.0704
3	0.177961	6.0736	11.8188	56.8505
$\rho_{max}$	60	20	10	6

**Table 1.** Excitation energy of the three lowest states in eV.

State	$\omega_r = 0.01$	$\omega_r = 0.5$	$\omega_r = 1$	$\omega_r = 5$
1	0.105041	2.05658	3.62193	14.1863
2	0.139682	3.78863	7.08603	31.5068
3	0.174323	5.52068	10.5501	48.8273

**Table 2.** Analytical approximation of excitation energy of the three lowest states in eV.

To make this mess look nicer we change coordinates to the center of mass coordinate  $R$ , and make the equation dimensionless by multiplying with some constants, arriving at the final equation for the system

$$-\frac{d^2}{d\rho^2}\psi(\rho) + \omega_r \rho^2 \psi(\rho) + \frac{1}{\rho} = \lambda \psi(\rho), \quad (14)$$

where

$$\omega_r = \frac{1}{4} \frac{mk}{\hbar^2} \alpha^4, \quad \alpha = \frac{\hbar^2}{m\beta e^2}, \quad \lambda = \frac{m\alpha^2}{\hbar^2} E.$$

Looking at equation 14 we see that we will be solving the same equation again, only this time the potential term takes on a different form. This means that we can reuse our code from the one electron case to this one.

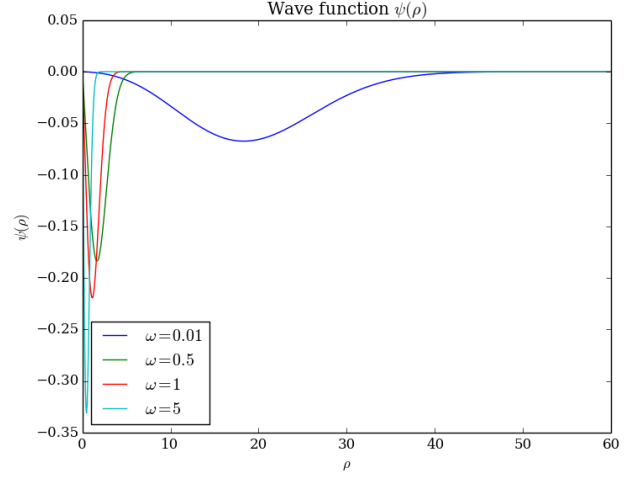
We treat  $\omega_r$  as a strength parameter for the oscillator potential, and will study our case for different values of  $\omega_r$ . The lowest energies for the different  $\omega$  values are tabulated in table 1. Studying the table shows that the excitation energies increase with higher  $\omega$ , which is logical since the potential is proportional with  $\omega$ . The potential well gets deeper with higher oscillator strength, making it harder to get out of the potential. The results in table 1 are for  $n = 1000$ . Note that  $\rho_{max}$  is varied depending on the value of  $\omega_r$  since larger  $\omega_r$  makes the potential narrower, making it unnecessary to calculate values for large  $\rho$ .

In this case there exists good approximations to the analytical solutions. It can be shown that an approximation can be written

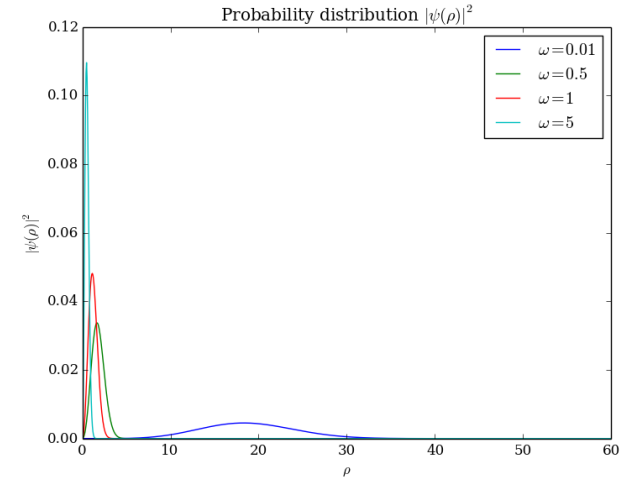
$$\epsilon_m = 3\left(\frac{\omega_r}{2}\right)^{2/3} + \sqrt{3}\omega_r \quad (15)$$

Table 2 shows the analytical approximation of the eigenvalues. Comparing this to our numerical solutions we can see that they are quite coherent. This bodes well for our selfesteem and our numerical code.

Figure 1 shows the wavefunction for different  $\omega_r$ , while figure 2 shows the probability distribution. Here  $\rho$  represents the coordinate of the mass center of the two electrons. As  $\omega_r$  increases, the probability of finding the mass center in the potential well increases. In short, bigger  $\omega$  gives a deeper well, which in turn increases the probability that the electrons are located in the potential well. Meaning that by increasing  $\omega_r$  we can constrain the distance between the two electrons to some probable range.



**Fig. 1.** Wavefunction with different  $\omega_r$ .



**Fig. 2.** Probability distribution of  $\psi$ .

## 4. Testing

As a final insurance that our code works the desired way, we will utilize some tests to make sure our code is coherent with mathematical laws and regulations. We know that if we set  $n = 2$  then the program should complete in 1 iteration. We also know that the eigenvectors we find should all be orthogonal. Mathematically this means that we can take the dot product of them all and this should be equal to zero.

Running the code with  $n = 2$  does in fact complete the calculation in 1 iteration as it should. To calculate whether the eigenvectors are orthogonal we set a tolerance of  $10^{-4}$  and say that if the dot product of two vectors is smaller than this tolerance the dot product is zero. The reason we need to do this is because we never get precisely zero when adding things together on a computer. When looking at the 3 first eigenvectors with  $n = 1000$ ,  $\rho_{max} = 50$ , and the potential corresponding to two electrons with repulsion, the code passes this test.

## 5. Conclusions

In this project we have seen that we can use the Jacobi method to calculate the wavefunction of one electron in a potential well

and two electrons in another potential well with repulsion. For the first case we saw that armadillo is much more effective than our method, and provides the same solution. Our approach came quite close to the analytical solutions, but it demanded a lot of steps leading to many iterations and thus long computation time. Nevertheless the numerical approximation was very close to the analytical solution. We also saw that the method works well for two electrons in a potential well with Coulomb repulsion between them. In this case  $\rho_{max}$  has been set to different values for different  $\omega_r$ . This needed to be done because the solutions were highly dependent on both stepsize and  $\rho_{max}$ . Looking at the graphs for the wavefunctions reveal clearly why there is no point using very high  $\rho_{max}$  for high  $\omega_r$ .

In conclusion we have seen that the jacobi rotation method works well for finding eigenvalues, and -vectors for quantum mechanical problems.

## 6. Source code

The source code for this document, the c++ project, and the python file for plotting can be found at <https://github.com/tellewsen/Project2>.

## 7. References

M. Taut, Phys. Rev. A 48, 3561 - 3566 (1993)