Two Electrons in a Harmonic Oscillator Potential

Sean Sweany

February 12, 2016

Abstract

This study was preformed to investigate the behavior of two electrons in a three dimensional harmonic oscillator potential with the Coulomb force included. This problem was solved numerically by discretization of the Schroedinger equation turning this into a simple eigenvalue problem. The eigenvalues and eigenvectors were then computed using Jacobi's rotation method. Initial steps for solving this problem were to first calculate the eigenvalues for just one electron in the potential, the method was then validated by several unit tests to determine if the code was working correctly. Only then moving to the case with two electrons. It was found that a higher oscillator frequency ω causes the states to be more tightly bound[2].

1 Introduction

The model for electrons trapped inside of a harmonic oscillator potential are a starting model for understanding quantum dots. Quantum dots are semiconductors which trap electrons in three dimensional spaces known as holes. These quantum dots can be manipulated to have different electronic properties, making them very hot topics in many areas of research today[1]. Some examples of the research areas involving quantum dots include quantum computing, where computational operations take place due to interactions between the quantum dots[4], and biological imaging for medical purposes [5] along with many others. Our study looks at a simple model of these quantum dots as was explained above. The numerical results that are obtained will

then be compared to an analytical solution for the two electron case of this problem.

2 Theory

2.1 One Electron Case

For the case with one electron there are two options to solving the Schroedinger Equation (possibly more but that would be hard). In Cartesian or in spherical coordinates. Here the case being investigated will be the one in spherical coordinates because due to symmetry the one electron case can be reduced from a 3 dimensional problem to a 1 dimensional one. For the one dimensional case we first have:

$$H\Psi(r,\theta,\phi) = HR(r)Y(\theta,\phi)$$

Where it was assumed that the angular and radial parts can be separated. Now the angular part will not contribute to our answer so all that is left is the radial Schroedinger equation:

$$\frac{-\hbar^2}{2m}(\frac{1}{r^2}\frac{d}{dr}r^2\frac{d}{dr} - \frac{l(l+1)}{r^2})R(r) + \frac{1}{2}m\omega^2 r^2 R(r) = ER(r)$$

Now, to simplify this set $r = \rho\alpha$ and let $R(r) = \frac{u(r)}{r}$ where α is dimensionless. Making this substitution and multiplying by $\frac{2m\alpha^2}{\hbar^2}$:

$$-\frac{d^2}{d\rho^2}u(\rho) + \frac{m^2\omega^2\alpha^4}{\hbar^2}\rho^2 u(\rho) + \frac{l(l+1)}{\rho^2}u(\rho) = \lambda u(\rho)$$

Where $\lambda = \frac{2m\alpha^2}{\hbar^2}E$ and letting $\frac{m^2\omega^2\alpha^4}{\hbar^2} = 1$ all constants fall out and it is now possible to solve for α,

$$\alpha = \sqrt{\frac{\hbar}{m\omega}}$$

The last thing to note is that l will be set to zero. With this the final form of the Schroedinger equation is finally[1]:

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

2.2 Two Electron Case

Next is the case with two electrons in the harmonic oscillator potential that experience the Coulomb interaction. Like above this case will also be solved in spherical coordinates, reducing the problem from 6 dimensions to two. Now, excluding the Coulomb potential the Schroedinger equation looks like:

$$(\frac{-\hbar^2}{2m}(\frac{d^2}{dr_1^2}+\frac{d^2}{dr_2^2})+\frac{1}{2}m\omega^2 r_1^2+\frac{1}{2}m\omega^2 r_2^2)u(r_1,r_2)=Eu(r_1,r_2)$$

This equation can now be placed into the center of mass coordinates by setting the relative separation $r = r_1 - r_2$ and the COM position as $R = \frac{1}{2}(r_1 + r_2)$, Schroedinger equation now reduces to:

$$(\frac{-\hbar^2}{m}\frac{d^2}{dr^2}+\frac{-\hbar^2}{4m}\frac{d^2}{dR^2}+\frac{1}{4}m\omega^2 r^2+m\omega^2 R^2)u(r,R)=Eu(r,R)$$

The energy for this equation can now be separated into that of the relative and center of mass energies. Where the center of mass energy is the same as was obtained in the part for the single particle in the harmonic oscillator. With the equations separated it is now easy to add in relative potentials to the relative position half. For us the Coulomb term being added has the form

$$V(r)=\frac{\beta e^2}{\rho}$$

So adding in the Coulomb potential term and doing algebraic manipulations the final form is obtained.

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

Where $\omega_r^2 = \frac{m^2\omega^2}{4\hbar^2}\alpha^4$, and now $\alpha = \frac{\hbar^2}{m\beta e^2}$ [1].

3 Numerical Methods

3.1 Jacobi's Method

Schroedinger's equation in the forms above can easily be discretized into the form:

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

Where this is just Euler's equation for solving a second order differential equation with some extra stuff added to it. Like in the last project this can in turn be transformed into a tridiagonal matrix times some vector equaling a constant times another vector,

$$\begin{pmatrix} d_1 & e_1 & 0 & 0 & \dots & 0 & 0 \\ e_1 & d_2 & e_2 & 0 & \dots & 0 & 0 \\ 0 & e_2 & d_3 & e_3 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & \dots & d_{n_{\text{step}}-2} & e_{n_{\text{step}}-1} \\ 0 & \dots & \dots & \dots & \dots & e_{n_{\text{step}}-1} & d_{n_{\text{step}}-1} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \dots \\ \dots \\ \dots \\ u_{n_{\text{step}}-1} \end{pmatrix} = \lambda \begin{pmatrix} u_1 \\ u_2 \\ \dots \\ \dots \\ \dots \\ u_{n_{\text{step}}-1} \end{pmatrix} \tag{1}$$

This is now just an eigenvalue problem where we are trying to find λ. The method that will be used to do this is called the Jacobi method (just a note all of the steps discussed here are only for symmetric matrices). The way this method works is by preforming similarity transformations on the matrix until the off diagonal elements are so small that they can be considered zero,

$$S^* A S = B$$

The eigenvalues are then just the elements along the diagonal of this matrix. The matrices used to preform the similarity transformations are rotation matrices where the position of the sin's and cos' in the matrix are such that they can be used to set the largest off diagonal element to zero. Once these largest off diagonal elements have been located there are four elements that are very important $a_{pp}, a_{qq}, a_{pq}, a_{qp}$, these off diagonal elements are the ones that we want to set to zero, and the other two are the ones on the diagonal. Once these matrices are multiplied along with other elements along the rows and columns of p and q there will be a term looking as such:

$$d_{pq} = a_{pq}(c^2 - s^2) + (a_{pp} - a_{qq})cs = 0$$

(where in the above equation c and s are sine and cosine respectively) setting this to zero is how the algorithm removes the largest off diagonal elements. Once this is done the equation can be manipulated by dividing by cs and rearranging, letting t be tangent

$$t^2 + 2t\frac{a_{pp} - a_{qq}}{2a_{pq}} - 1 = 0$$

(let the a's in the fraction be called τ, going through the math just gives that τ is $cot(2\theta)$) This is just the quadratic equation which has the solutions:

$$t = -\tau + \sqrt{1+\tau^2}, t = -\tau - \sqrt{1+\tau^2}$$

After finding t, s and c fall out naturally with $c = \frac{1}{\sqrt{1+t^2}}$ and $s = tc$. Now, with this equation there is the issue of choosing which t to use, the one with the plus or the one with the minus. The answer is you need to choose the one which gives the smallest value for t. Cleverly multiplying and dividing $\tau and \sqrt{1+\tau^2}$ with some combination of pluses and minuses you get

$$t = \frac{1}{\tau + \sqrt{1+\tau^2}}, t = \frac{-1}{-\tau + \sqrt{1+\tau^2}}$$

Now, assume that τ is less than 0. Then the first of the two values for t will be divided by $\tau + \sqrt{1+\tau^2}$, which will grow smaller, making t grow large very quickly for small changes in tau. Where as the other denominator in the second solution for t will grow and t will change very little. This will cause less error to accumulate in the code. Now, if τ is greater than zero then the opposite happens. The root with the negatives will become more unstable while the positive root grows. After going through and choosing the correct t to use and applying the similarity transformation to the rest of the matrix the next step is to go through and determine if all off diagonal elements are small enough so as to consider the matrix diagonal. Now at this point there are two options, the first is to search for the largest element and if it is below some tolerance then the code is finished. The other method that could be used is to determine the difference in the Frobenius norm of the current and previous matrix, then seeing if it is below some tolerance.

$$||\mathbf{B} - \mathbf{A}||_F^2 = 4(1-c) \sum_{i=1, i \neq p,q}^{n} (a_{ip}^2 + a_{iq}^2) + \frac{2a_{pq}^2}{c^2}.$$

The method chosen here was the former mainly because using the Frobenius method takes much more computation time where since the matrix is symmetric the first method only has to search the upper half of the matrix for the largest value[2].

3.2 Unit Tests

A good way to determine if a code is working is by implementing unit tests. These are small sections in the code which test a function or subroutine to

see if it operates properly. This is done by checking known mathematical results that the code should produce, or just checking if loops work correctly. For the Jacobi method code to work it had to pass three separate unit tests. The first is to take a simple four by four matrix where the eigenvalues were determined by Wolfram Alpha, and have the Jacobi solver function reproduce those four known eigenvalues. The second is intertwined with this, and that is knowing the largest off diagonal element in the four by four matrix can the code actually find this value. The final test that was implemented was to take two eigenvectors and after every 20 iterations multiply them together, the result should then be zero since the eigenvectors are orthogonal. After several modifications to the code all three of these requirements were held. So by unit tests the Jacobi method appears to work correctly.

4 Results

4.1 One Electron

For the case of one electron in the harmonic oscillator trap, with a matrix dimensionality of 190 and a length scale of just 5, the first three eigenvalues were found for 4 leading digits. The eigenvalues were 2.999, 6.999 and 10.99. Below is a table looking at different length scales, and matrix sizes. Now, for the length scale of 2 all of the eigenvalues were very wrong due to the fact that the wave functions for all of the solutions go to zero after 2.

Table 1: Effect of Matrix Size on Iterations and Run Time

Matrix Size	Time(s)/Iterations		
Length Scale	2	5	10
100	1/19138	1/18619	1/17798
150	8/43174	8/42264	8/40722
200	26/77004	26/75272	24/72599
250	64/120524	62/117840	63/114507
300	132/173573	130/169708	127/165167

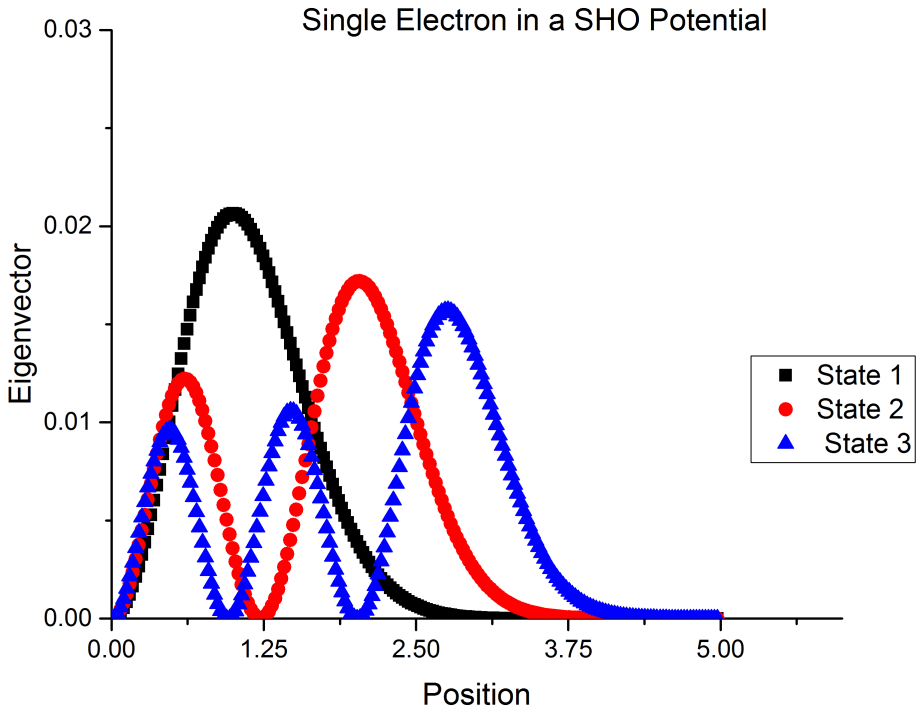


Figure 1. Eigenvectors for electron in a harmonic oscillator potential

Looking at the table it is clear that as the size of the matrix grows so do the number of iterations needed in order to find the eigenvalues. It is interesting to note that the number of iterations and time to run the calculation only weakly depend on the length scale. Actually, the only thing

Table 2: Time to compute for the tqli method

Matrix Size	Time(s)	Iterations
600	5	44721
700	8	56568
800	13	72111

the length scale seems to drastically effect is the eigenvalues. So, to get a relation between the size of the matrix and number of iterations needed to complete the calculation first the number of iterations was averaged over the three length scales. Next, these values were graphed against the size of the matrix. This graph was then fit with a polynomial and it was found that as the size of the matrix increases the iterations needed to find the eigenvalues goes as $2x^2$ where x is the size of the matrix. The above method was used for analyzing the time to run the calculation however the lowest values for the time have significant error in them since the timer in the code has a resolution of seconds, but fitting the last four points gives something like $10^{-8}x^4$.

Comparing this to the tqli solver was difficult to do because I was unable to determine how to make the program produce the number of iterations needed for this function to work. Also, measuring the time was not very useful either because the tqli method is designed for solving tridiagonal matrices and it was so fast that the time to complete a run kept being displayed was just 0 for all of the matrix sizes that I investigated for Jacobi's method. Finally, to hopefully produce at least an order of magnitude estimation of the number of iterations that the tqli method takes the following was used, taking the inverse of the two above fit equations:

$$\frac{\sqrt{0.5i}}{100t^{0.25}} = 1$$

where i is the number of iterations and t is the time. Looking at this the dependence on the matrix size is completely removed from the equation.

Comparing the size of the matrix to the iterations fitting the data gives a straight line so it appears as though for the tqli method the number of iterations increase linearly with the size of the matrix.

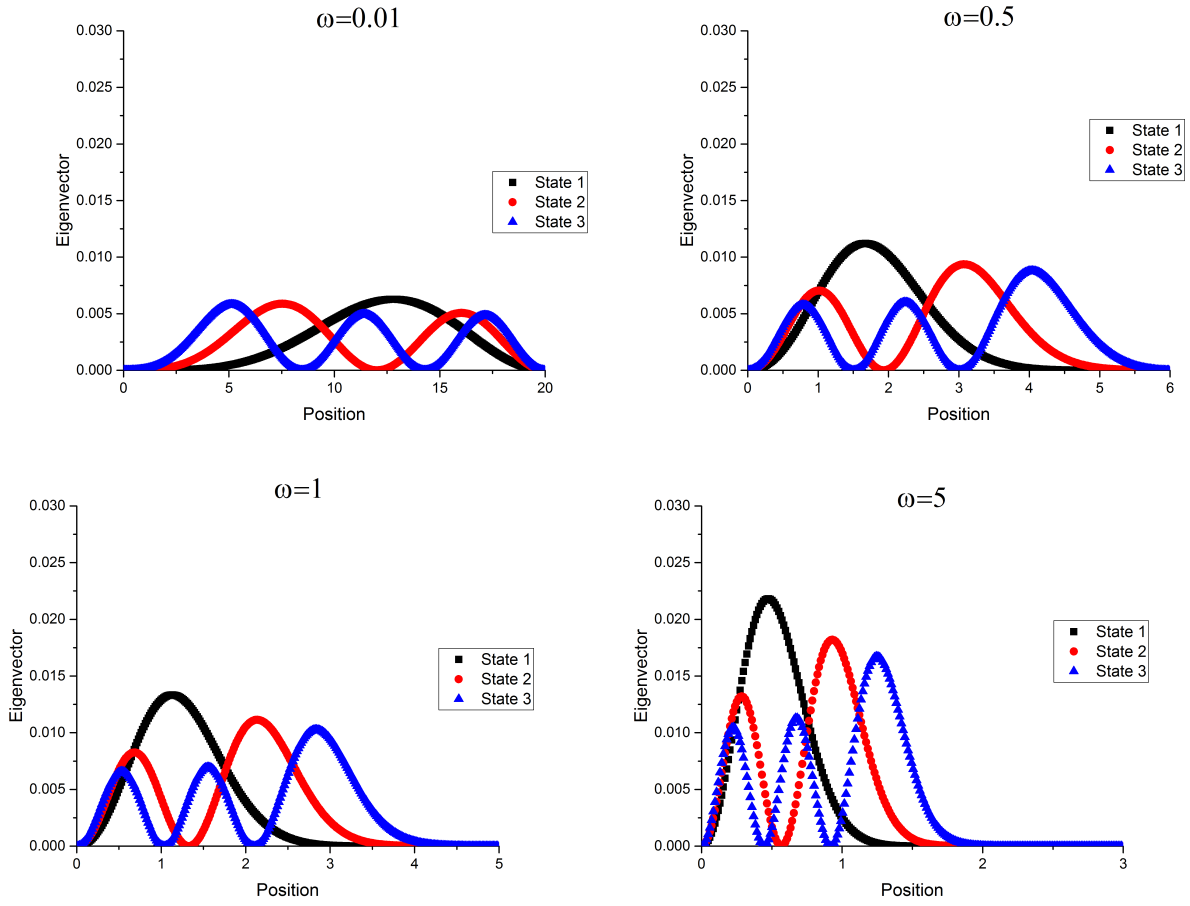
Table 3: Energies for 2 Electrons in a Harmonic Oscillator

ω_r	Eigenvalues		
0.01	0.138	0.244	0.392
0.5	2.23	4.14	6.09
1	4.05	7.91	11.81
5	17.45	37.0663	56.84

4.2 Two Electrons

For the case with two electrons I will include the eigenvalues for the three lowest states.

Looking at the values it is plain to see that as the oscillator strength decreases so do the ground state eigenvalues for the system. Comparing the ground state solution for $\omega = 0.25, \omega = 0.05$ to those in the paper by Taut[3] the values obtained match perfectly. Once ω is made smaller though it becomes harder to compute the solutions because to solve the problem we have to use longer and longer length scales, which means larger matrices are needed to solve them, this leads to more computing time. Below are plots of eigenvectors for the first three states for different values of ω. The x axis on these plots are not at the same scale, putting them at the same scale makes half of the plots look very squished so it was decided to present them in a way which makes them look pretty. Looking at the figures it is apparent that as ω_r increases the wave functions become more and more constricted, with this the energy also increases. So it appears that for a larger value of ω_r the strength of the oscillator over powers the Coulomb interaction that is being experienced by the two electrons. This can be seen clearly for $\omega_r = 0.01$, where the eigenvectors behavior seems to have actually been inverted. This indicates that there is a value for ω_r where some kind of bifurcation occurs and the system goes from being more repulsive to being more attractive.



5 Conclusion

Results for two interacting particles in a harmonic oscillator well show that there exists some barrier, where if the oscillator is strong enough the system becomes more attractive and if it is below this value then the electrons seem to be repulsive. So tuning this value of ω_r could be a way of either removing or keeping electrons trapped in the quantum dot. Also, comparing the Jacobi method to the tqli code, The Jacobi method is much slower and can take several minutes to run for a small enough step size. This in turn restricts the problems that can be solved to the length scale at which the solutions are relevant. Attempts were made to solve the same problem using the Lanczo

method however I could never get the code to work correctly.

6 References

References

[1] H, Jensen, Project 2 Introduction

[2] H, Jensen, Phys 904 Lecture Notes

[3] M. Taut, Phys Rev A, Vol 48, Num 5, Nov 1993.

[4] D. Loss, D Divincenzo, Phys Rev A, Vol 57, Num 1, Jan 1998.

[5] B, Dubertet, Science Vol 298, Nov 2009.