PHY905 Project 2 - Jacobi Algorithm

Alaina Ross
(Dated: March 3, 2017)

Background: Perhaps the most common potential used in quantum mechanics is that of the three dimensional harmonic oscillator. This is because there are analytic solutions which can easily be calculated. However, the introduction of an additional particle and a Coulomb interaction quickly makes the problem more difficult.
Purpose: The goal of this work is to solve numerically the aforementioned problem. We aim to study the numerical accuracy and performance of the algorithm as well as determine the effect of different oscillator shapes on the final wave functions.
Method: We approximate the derivatives in the Schrödinger equation in order to cast the equations as a matrix eigenvalue problem and use the Jacobi rotation algorithm to calculate the eigenvalues and eigenvectors. In addition, we use a more sophisticated algorithm to compare computation time.
Results: We find our calculated eigenvalues are consistent with analytical solutions for matrix sizes around $N = 400$. In addition, we find the Jacobi algorithm is very inefficient, especially for the large matrix sizes needed for our desired accuracy.
Conclusions: Our results demonstrate that the effect of including a Coulomb repulsion interaction is a more diffuse wave function as the electrons are repelled from one another. On the other hand, our results alse demonstrate that as the oscillator potential becomes stronger the electrons are forced closer to one another.

I. INTRODUCTION

There are a limited number of problems in advanced physics that can be solved analytically. For this reason, one of the most common problems encountered in undergraduate and graduate level quantum mechanics is that of a particle in a three dimensional harmonic oscillator potential, as it has analytic solutions. While the addition of another charged particle and a Coulomb interaction makes the problem more difficult, it can also be solved analytically with some considerable effort. It is the goal of this work to solve this problem numerically with high precision.

For a single electron in a harmonic oscillator potential, the radial Schrödinger equation is given by:

$$\frac{\hbar}{2m}\left(\frac{1}{r^2}\frac{d}{dr}r^2\frac{d}{dr}-\frac{\ell(\ell+1)}{r^2}\right)R_n(r) + \frac{1}{2}m\omega^2 r^2 R_n(r) = E_{n\ell}R_n(r) \tag{1}$$

where $R_n(r)$ are the radial wave functions, ω is the oscillator frequency, and $E_{n\ell}$ is the energy for the given quantum numbers:

$$E_{n\ell} = \hbar\omega\left(2n+\ell+\frac{3}{2}\right) \quad n,\ell = 0,1,2... \tag{2}$$

In this work we will focus on $\ell = 0$. Then making the substitution $u_n(r) = rR_n(r)$ and $\rho = \alpha/r$ where $\alpha^4 = \hbar^2/mk$ and $k = m\omega^2$ gives:

$$-\frac{d^2}{d\rho^2}u_n(\rho) + \rho^2 u_n(\rho) = \frac{2m\alpha}{\hbar^2}E_n u_n(\rho). \tag{3}$$

Simplifying further, we can introduce the variable λ such that:

$$-\frac{d^2}{d\rho^2}u_n(\rho) + \rho^2 u_n(\rho) = \lambda u_n(\rho). \tag{4}$$

This equation can be solved analytically to give the eigenvalues $\lambda_0 = 3$, $\lambda_0 = 7$, and $\lambda_2 = 11$.

If rather than one electron in the harmonic oscillator well there are two which are non-interacting, then Equation 1 becomes:

$$\left(-\frac{\hbar^2}{2m}\frac{d^2}{dr_1^2}-\frac{\hbar^2}{2m}\frac{d^2}{dr_2^2}+\frac{1}{2}kr_1^2+\frac{1}{2}kr_2^2\right)u_n(r_1,r_2) = E_n^{(2)}u_n(r_1,r_2) \tag{5}$$

where $u_n(r_1, r_2)$ is a two-body wave function and $E_n^{(2)}$ is the two-electron energy. We introduce new coordinates, $\mathbf{r}$ and $\mathbf{R}$, defined by:

$$\mathbf{r} = \mathbf{r_1} - \mathbf{r_2} \tag{6}$$

$$\mathbf{R} = \frac{1}{2}(\mathbf{r_1} + \mathbf{r_2}) \tag{7}$$

where $\mathbf{r}$ is the relative coordinate and $\mathbf{R}$ is the relative center-of-mass coordinate. This simplifies Equation 5 to:

$$\left(-\frac{\hbar^2}{m}\frac{d^2}{dr^2}-\frac{h^2}{4m}\frac{d^2}{dR^2}+\frac{1}{4}kr^2+kR^2\right)u_n(r,R) = E_n^{(2)}u_n(r,R) \tag{8}$$

and allows for the both the wave function and the energy to be written in terms of a relative wave function and energy, $\psi(r)$ and E_r, and a center-of-mass wave function and energy, $\phi(R)$ and E_R:

$$u(r,R) = \psi(r)\phi(R) \tag{9}$$

$$E^{(2)} = E_r + E_R. \tag{10}$$

This change of variables proves even more useful though when a repulsive Coulomb interaction is included as:

$$V(r_2,r_2) = \frac{\beta e^2}{|\mathbf{r}_1 - \mathbf{r}_2|} = \frac{\beta e^2}{r} \tag{11}$$

where $\beta e^2 = 1.44$ eVnm. As this potential only depends on the relative coordinate, we can look exclusively at the Schrödinger equation for the relative wave function:

$$\left(-\frac{\hbar^2}{m}\frac{d^2}{dr^2} + \frac{1}{4}kr^2 + \frac{\beta e^2}{r}\right)\psi(r) = E_r\psi(r). \tag{12}$$

Again we can make the substitution $\rho = r/\alpha$ where now $\alpha = \hbar^2/m\beta e^2$. In addition, we introduce the frequency $\omega_r = mk\alpha^4/4\hbar^2$, which reflects the strength of the oscillator potential, and the variable $\lambda = m\alpha^2 E_r/\hbar^2$. Equation 12 then becomes:

$$-\frac{d^2}{d\rho^2}\psi(\rho) + w_r^2\rho^2\psi(\rho) + \frac{1}{\rho} = \lambda\psi(\rho). \tag{13}$$

This equation has been solved analytically for some values of ω_r in [1]. Of specific interest in this work is the case $\omega_r = 0.05$, where the ground state eigenvalue is given by $\lambda = 0.35$.

In this work we implement the Jacobi algorithm to solve the eigenvalue problems described above and study the performance characteristics and numerical accuracy for different grid and step sizes. In addition, we investigate the influence of the frequency w_r on the shape of the wave functions in the interacting case. In Section II, the implementation of the algorithm is described. In Section III the performance and accuracy of the code are analyzed. Finally, in Section IV we give a summary and our conclusions.

II. METHODS

We first start by discretizing the functions in Equations 4 and 13 and approximating the derivatives as:

$$u'' = \frac{u(\rho+h) - 2u(\rho) + u(\rho-h)}{h^2} + O(h^2) \tag{14}$$

where h is the step size. Realistically, the wave functions should be defined from $\rho = [0, \infty)$, however, numerically we define a minimum value, $\rho_0 = 0$, and a maximum value, ρ_{max}, which will determine the step size for a given number of mesh points, N, as:

$$h = \frac{\rho_{max} - \rho_0}{N} \tag{15}$$

Using Equation 14, Equations 4 and 13 can be written as:

$$\left(\frac{2}{h^2} + V_i\right)u_i - \frac{1}{h^2}u_{i-1} - \frac{1}{h^2}u_{i+1} = \lambda u_i \tag{16}$$

where V_i is either ρ^2 for the non-interacting case, or $\omega_r^2\rho^2 + 1/\rho$ for the interacting case. The above equation is an eigenvalue problem that can be written in matrix form as:

$$\begin{bmatrix} \frac{2}{h^2}+V_1 & -\frac{1}{h^2} & 0 \\ -\frac{1}{h^2} & \frac{2}{h^2}+V_2 & -\frac{1}{h^2} \\ 0 & -\frac{1}{h^2} & \frac{2}{h^2}+V_N \end{bmatrix}\begin{bmatrix} u_0 \\ u_1 \\ u_N \end{bmatrix} = \lambda\begin{bmatrix} u_0 \\ u_1 \\ u_N \end{bmatrix} \tag{17}$$

As an N dimensional matrix will have N eigenvalues and eigenvectors, the grid size necessarily determines the number of eigenvalues that will be produced. As we are only interested in the ground state and first two excited states, the minimum matrix size we can use is N = 3.

To begin the Jacobi algorithm we implement the vectors, v_i, and matrix, $\hat{a}$, using the Armadillo library [2]. The main premise is to diagonalize the matrix via repeated unitary transformations, given by $\hat{a}' = \hat{S}^T\hat{a}\hat{S}$, which take the largest non-diagonal matrix element to zero each iteration. This process continues until all off diagonal elements are less than the specified tolerance and thus are essentially zero. Once the diagonalization is complete, the eigenvalues are simply the values left on the diagonal.

For the eigenvectors we choose as a starting point unit vectors given by:

$$v_0 = \begin{bmatrix}1\\0\\0\end{bmatrix} \quad v_1 = \begin{bmatrix}0\\1\\0\end{bmatrix} \quad v_N = \begin{bmatrix}0\\0\\1\end{bmatrix}. \tag{18}$$

Then for each iteration in the diagonalization scheme, the eigenvectors will undergo the same unitary transformation as the matrix, namely $v_i' = \hat{S}v_i$. As the transformation is unitary, the orthogonality of the eigenvectors will be preserved.

If the coordinates of the matrix element to be sent to zero are given by (k, l), the nonzero matrix elements of the rotation matrix, $\hat{S}$, are given by:

$$\begin{aligned} S_{(k,l)} &= -S_{(l,k)} = -sin(\theta) \\ S_{(k,k)} &= S_{(l,l)} = cos(\theta) \\ S_{(i,i)} &= 1 \quad i \neq k, i \neq l. \end{aligned} \tag{19}$$

After the transformation, the updated matrix elements of $\hat{a}'$ will be given by:

$$\begin{aligned} a'_{(k,l)} &= \left(a_{(k,k)} - a_{(l,l)}\right)cos(\theta)sin(\theta) + a_{(k,l)}\left(cos^2(\theta) - sin^2(\theta)\right) \\ a'_{(k,k)} &= a_{(k,k)}cos^2(\theta) - 2a_{(k,l)}cos(\theta)sin(\theta) + a_{(l,l)}sin^2(\theta) \\ a'_{(l,l)} &= a_{(l,l)}cos^2(\theta) + 2a_{(k,l)}cos(\theta)sin(\theta) + a_{(k,k)}sin^2(\theta) \\ a'_{(i,k)} &= a_{(i,k)}cos(\theta) - a_{(i,l)}sin(\theta) \quad i \neq k, l \\ a'_{(i,l)} &= a_{(i,l)}cos(\theta) + a_{(i,k)}sin(\theta) \quad i \neq k, l \end{aligned} \tag{20}$$

Since $a'_{(k,l)} \equiv 0$, $sin(\theta)$ and $cos(\theta)$ are determined via the expression above as:

$$cot(2\theta) \equiv \tau = \frac{a_{(l,l)} - a_{(k,k)}}{2a_{(k,l)}} \tag{21}$$

$$tan(\theta) = -\tau \pm \sqrt{1+\tau^2} \tag{22}$$

$$cos(\theta) = \frac{1}{\sqrt{1+tan^2(\theta)}} \tag{23}$$

$$sin(\theta) = tan(\theta)cos(\theta) \tag{24}$$

and the matrix elements are updated accordingly.

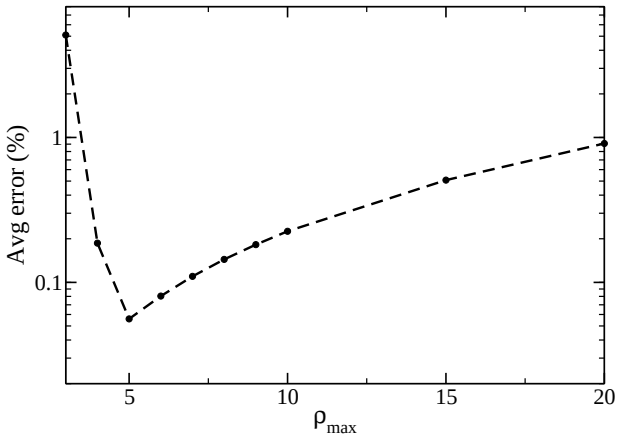

FIG. 1: Relative average error in λ_n as a function of ρ_{max}

To determine the accuracy of our solutions, we define the average error as the average of the relative differences of the first three eigenvalues calculated via the Jacobi algorithm, λ_n^j, as compared to their exact values, λ_n:

$$\epsilon_{avg} = \frac{1}{3}\left(\frac{|\lambda_0^j - \lambda_0|}{\lambda_0} + \frac{|\lambda_1^j - \lambda_1|}{\lambda_1} + \frac{|\lambda_2^j - \lambda_2|}{\lambda_2}\right) \quad (25)$$

III. RESULTS

In the following subsections we describe the results obtained both with and without the Coulomb interaction. We discuss the accuracy and performance of the methods and show the resulting probability distributions.

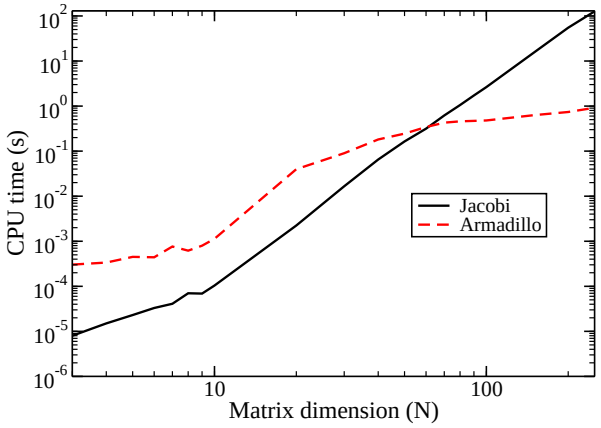

FIG. 2: Computation time as a function of matrix dimension.

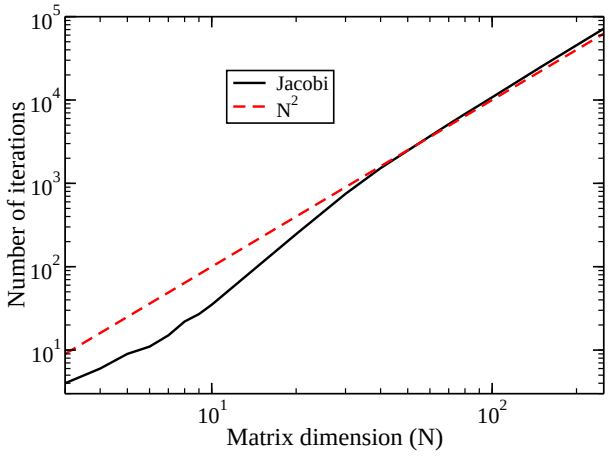

FIG. 3: Number of Jacobi rotations necessary for convergence as a function of matrix dimension.

A. Non-interacting Case

We first implement the Jacobi algorithm with only the harmonic oscillator potential. For a matrix dimension of N = 100, we tested the accuracy of the calculated eigenvalues for varying ρ_{max}, and show the relative percent error in Figure 1. From this we see the most accurate solution is for $\rho_{max} = 5$. The resulting eigenvalues are summarized in Table I. While the error is small for these parameters, the desired level of accuracy is to have agreement to four digits past the decimal. This is not achieved until a matrix size of $N = 400$ is used.

In addition, we test the performance of the code via the time to solution for varying matrix dimension. In Figure 2, we compare the performance of our Jacobi algorithm (black solid curve) to that of a more sophisticated algorithm (red dashed curve) from the Armadillo library [2]. In general, one would expect the Armadillo algorithm to be faster regardless of matrix size; however, for very small matrix dimension, the Jacobi algorithm is faster. This is likely because the value of ρ_{max} was kept constant and only the matrix dimension was changed. This means that for the small matrix sizes, the step size was larger, which means the off-diagonal matrix elements (which go like $1/h^2$) are smaller.

This point is further illustrated in Figure 3, which shows the number of rotations needed for the off diagonal elements to be less than 0.001. The black solid line corresponds to the results from the Jacobi algorithm, and the red dashed line corresponds to the predicted N^2 behavior which represents the number of matrix elements. For large matrices the results agree well with the prediction, but for small matrices the Jacobi algorithm needs less iterations than expected. Again, this is likely due to the small size of the off diagonal matrix elements for small matrix sizes.

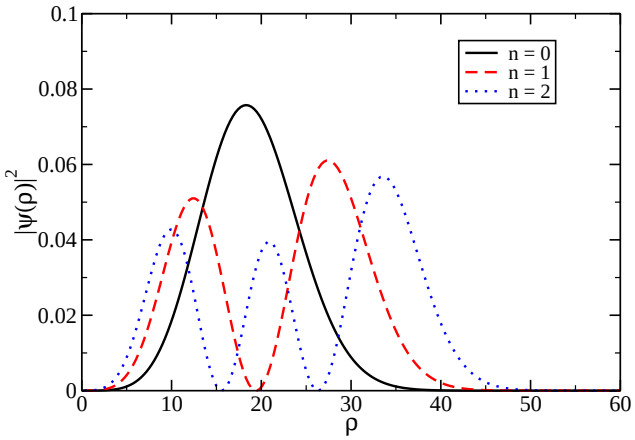

FIG. 4: $|\psi(\rho)|^2$ for $w_r = 0.01$

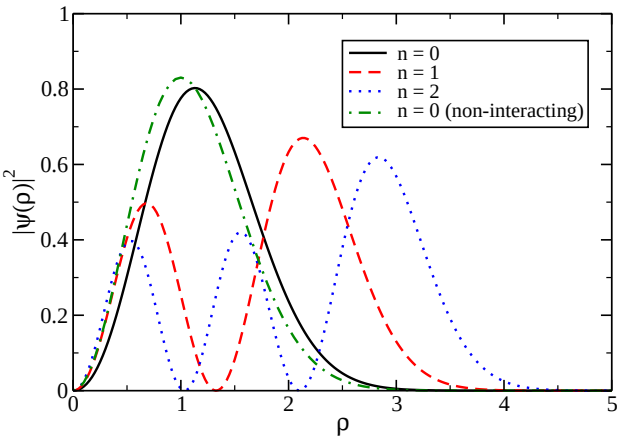

FIG. 6: $|\psi_n(\rho)|^2$ for $w_r = 1.0$

B. Interacting Case

We now implement the Jacobi algorithm with the harmonic oscillator potential and a Coulomb interaction potential. First, we use the oscillator frequency of $\omega_r = 0.05$ from [1] and compare their eigenvalues to those calculated with the Jacobi algorithm in Table I. Again, there is good agreement with the analytic solution.

In addition, we compare the probability distribution of the wave functions, $|\psi_n(\rho)|^2$, for varying ω_r in Figures 4-7. In all figures the black solid line corresponds to the ground state, and the red dashed (blue dotted) line corresponds to the first (second) excited state. Figure 4 shows the distribution for $\omega_r = 0.01$, Figure 5 for $\omega_r = 0.05$, Figure 6 for $\omega_r = 1.0$, and Figure 7 for $\omega_r = 5.0$.

As ω_r represents the strength of the harmonic oscillator potential, as ω_r increases, there should be less space for the two electrons and thus the probability distribution should be peaked at smaller values of ρ. This is indeed the same behavior seen in Figures 4-7 as the peak for the ground state probability distribution moves from roughly $\rho = 20$ for $\omega_r = 0.01$ to under 0.5 for $\omega_r = 5.0$

Finally, in Figure 6 we show the ground state wave function probability distribution with no coulomb interaction (green dot-dashed line) for comparison. Here we can see that the inclusion of the Coulomb repulsion causes the probability distribution to be shifter to larger ρ and more diffuse. This is to be expected as the Coulomb repulsion causes the the two electrons to be pushed away from one another.

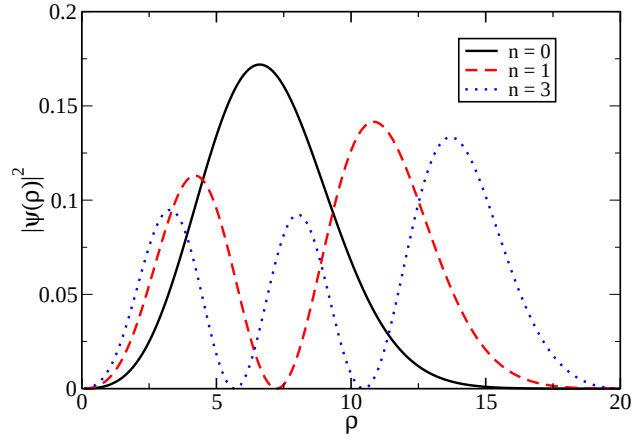

FIG. 5: $|\psi_n(\rho)|^2$ for $w_r = 0.05$

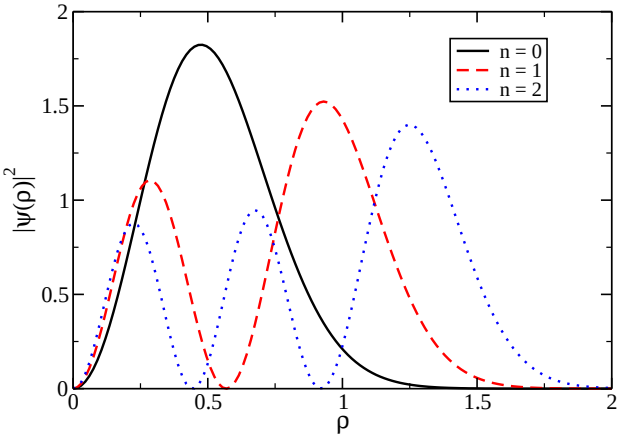

FIG. 7: $|\psi_n(\rho)|^2$ for $w_r = 5.0$

λ_n	Jacobi	Analytic	Jacobi	Analytic
λ_0	2.9992	3	0.3499	0.35
λ_1	6.9961	7	0.5325	N/A
λ_2	10.9906	11	0.7210	N/A

TABLE I: Comparison of eigenvalues for the analytic methods and the Jacobi method. For the non-interacting case N = 100 and ρ_{max}=5. For the interacting case N = $200\rho_{max} = 20$ and $\omega_r = 0.05$. Analytic values for the interacting case are from [1].

IV. CONCLUSIONS

In summary, the goal of this work was to calculate numerically the energies and wave functions corresponding to two electrons in a harmonic oscillator potential with a repulsive Coulomb interaction. The second derivative in the Schrödinger equation is approximated using a three point formula, which turns the problem into a matrix eigenvalue problem that is solved via the Jacobi algorithm.

The Jacobi algorithm is first analyzed without the Coulomb interaction to determine the numerical accuracy and performance. We found that in order to have agreement with analytic solutions for the first three eigenvalues up to four decimal places required a matrix size of roughly 400. However, we find that the Jacobi algorithm does not perform nearly as well as other more sophisticated algorithms, especially for the matrix sizes needed for the desired level of accuracy.

Next, we include the Coulomb interaction and find that the resulting eigenvalues have good agreement with the analytic solutions given in [1]. In addition, we plot the probability distributions for various harmonic oscillator strengths and find that as the potential is increased the electrons are forced closer together. Finally, we find that as expected, the addition of the Coulomb repulsion term causes the electrons to be forced further apart.

There are a number of ways that our code can be improved upon, for example there are more sophisticated algorithms to calculate eigenvalues (such as Lanczo's method [3]) which would likely prove to be more efficient. In addition, the Jacobi algorithm doesn't require that our matrix is already tridiagonal, so there are likely modifications or alternatives that could increase performance by taking advantage of the symmetries of the problem. Alternatively, while the Jacobi algorithm is not the most efficient, the use of parallelizations such as OpenMP as done in [4] could greatly improve the performance. Finally, we have constrained this work to $\ell = 0$ and have disregarded the center of mass wave function in the analysis, both of which could add important insights to the problem.

Overall, we have confirmed our physical intuition of the role of the Coulomb interaction and solved a difficult analytic problem numerically with good precision but poor performance. These results illustrate the power of using numerical methods but also the importance of choosing the correct method for the given problem.

[1] M. Taut, Phys. Rev. A **48**, 3561 (1993).

[2] C. Sanderson and R. Curtin, Journal of Open Source Software **1**, 26 (2016).

[3] C. Lanczos, Journal of Research of the National Bureau of Standards **45**, 2133 (1950).

[4] G. Hager and G. Wellein, *Introduction to High Performance Computing for Scientists and Engineers* (CRC Press, 2010).