PHY 905 Project 2: Electrons in a harmonic oscillator potential

Thomas Redpath[1]

[1] *Department of Physics, Michigan State University*

We present our results from solving the Schrödinger equation for two s-wave electrons confined by a harmonic oscillator potential. We first consider the case of non-interacting electrons then proceed to include the Coulomb interaction. We use Jacobi's method to solve Schrödinger's equation numerically. Our results agree with the well known analytic eigenenergies (with $l = 0$) $2E_{n0} = 3, 7, 11, \ldots$ for the non-interacting case. With the Coulomb interaction, our results agree well with analytic solutions that exist for specific oscillator frequencies. Finally, we explore the energies and wavefunctions for interacting electrons with oscillator frequencies $\omega = 0.01, 0.5, 1.0, 5.0$ and observe how the relative distance between the electrons increases with increasing oscillator strength. We interpret this as an increasing influence of the Coulomb repulsion as the confining potential becomes more restrictive.

INTRODUCTION

The spatial confinement of two electrons is a topic of great interest in solid state physics and materials science. In this report we address the simple problem of two electrons confined in a harmonic oscillator potential. We solve Schröedinger's equation for two electrons in a spherically symmetric 3D harmonic oscillator (HO) potential first without then including the Coulomb interaction. We proceed by discretizing the radial Schröedinger equation to cast it as an eigenvalue problem. We then employ Jacobi's method to diagonalize and obtain the eigenvalues and eigenvectors. In the first section of this report, we present the theoretical framework required to solve this problem. In the Methods section we discuss how we implemented the Jacobi algorithm in a C++ program. In the Results and Discussions section we present the ground state eigenenergies and wavefunctions calculated using our code for different oscillator strengths both with and without the Coulomb interaction.

THEORY

The derivation given here is summarized from Hjorth-Jensen [1]. First, consider the radial part of Schröedinger's equation for a single electron with a HO potential given by $V(r) = 1/2kr^2$ where $k = m\omega^2$

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

The energy eigenvalues are given by

$$E_{nl} = \hbar\omega\left(2n + l + \frac{3}{2}\right), \tag{1}$$

with $n = 0, 1, 2, \ldots$ and $l = 0, 1, 2, \ldots$, where n and l are the principle angular momentum quantum numbers respectively.

With the substitution $R(r) = (1/r)u(r)$, the dimensionless variable $\rho = (1/\alpha)r$ and taking $l = 0$ we obtain

$$-\frac{d^2}{d\rho^2}u(\rho) + \frac{mk}{\hbar^2}\alpha^4\rho^2u(\rho) = \frac{2m\alpha^2}{\hbar^2}Eu(\rho).$$

where α is a constant with dimension length.

Fixing α such that

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

and defining

$$\lambda = \frac{2m\alpha^2}{\hbar^2}E,$$

we can rewrite Schröedinger's equation as

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

In 3D with $l = 0$, the eigenvalues are $\lambda_0 = 3, \lambda_1 = 7, \lambda_2 = 11, \ldots$.

It is now straightforward to see how this equation can be solved numerically using the standard numerical second derivative

$$u'' = \frac{u(\rho+h) - 2u(\rho) + u(\rho-h)}{h^2} + O(h^2), \tag{3}$$

with N grid points and the step size $h = (\rho_N - \rho_0)/N$ where ρ_0, ρ_N are the maximum and minimum values chosen for the grid. The value of ρ at the i^{th} grid point is then $\rho_i = \rho_0 + ih; i = 1, 2, \ldots, N$.

In the compact "discretized notation," Schödinger's equation becomes

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

Representing the diagonal matrix elements as

$$d_i = \frac{2}{\hbar^2} + V_i,$$

and the non-diagonal matrix element

$$e_i = -\frac{1}{\hbar^2}.$$

we arrive at a matrix eigenvalue equation representation of the problem

$$\begin{bmatrix} d_0 & e_0 & 0 & 0 & \dots & 0 & 0 \\ e_1 & d_1 & e_1 & 0 & \dots & 0 & 0 \\ 0 & e_2 & d_2 & e_2 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & \dots e_{N-1} & d_{N-1} & e_{N-1} \\ 0 & \dots & \dots & \dots & \dots & e_N & d_N \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \dots \\ \dots \\ \dots \\ u_N \end{bmatrix} = \lambda \begin{bmatrix} u_0 \\ u_1 \\ \dots \\ \dots \\ \dots \\ u_N \end{bmatrix}.$$

To account for the Coulomb interaction between two electons, it is convenient to introduce a change of coordinates (eq. 4) that allows us to separates the Schrödinger equation into two parts, one that describes the center of mass (R) of the two-body system and one that describes the interaction of the two electrons (r).

$$\begin{aligned} r &= r_1 - r_2 \\ R &= \frac{1}{2} r_1 + r_2 \end{aligned} \tag{4}$$

With this change of coordinates, we may re-write the two-body Schrödinger equation

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

as

$$\left(-\frac{\hbar^2}{m}\frac{d^2}{dr^2} - \frac{\hbar^2}{4m}\frac{d^2}{dR^2} + \frac{1}{4}kr^2 + kR^2 \right) u(r, R) = E^{(2)} u(r, R)$$

(see Appendix) and the ansatz for the wavefunction $\Psi(r, R) = \psi(r)\phi(R)$ allows us to separate the wavefunction such that the two-body eigenenergies are a sum of the center of mass and interaction energies $E^{(2)} = E_R + E_r$. With the form of the Coulomb interaction between two electrons

$$V(r_1, r_2) = \frac{\beta e^2}{|\mathbf{r}_1 - \mathbf{r}_2|} = \frac{\beta e^2}{r},$$

$\beta e^2 = 1.44$ eVnm, and the dimensionless variable $\rho = r/\alpha$ the Schrödinger equation becomes

$$-\frac{d^2}{d\rho^2}\psi(\rho) + \frac{1}{4}\frac{mk}{\hbar^2}\alpha^4\rho^2\psi(\rho) + \frac{m\alpha\beta e^2}{\rho\hbar^2}\psi(\rho) = \frac{m\alpha^2}{\hbar^2}E_r\psi(\rho).$$

By defining an "effective" oscillator frequency

$$\omega_r^2 = \frac{1}{4}\frac{mk}{\hbar^2}\alpha^4,$$

fixing α such that

$$\frac{m\alpha\beta e^2}{\hbar^2} = 1 \Rightarrow \alpha = \frac{\hbar^2}{m\beta e^2}$$

and letting

$$\lambda = \frac{m\alpha^2}{\hbar^2}E,$$

the Schrödinger equation becomes

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

which has a similar form to eq. 2. The same discretization treatment described above can be applied to this equation to arrive at the same matrix eigenvalue equation as before, just with a different potential. Namely, the V_i term in the diagonal matrix elements becomes $\omega_r^2\rho + 1/\rho$ instead of ρ^2.

Unitary transformations preserve orthogonality

Consider a set of vectors $\mathbf{v}_i$ that form an orthonormal basis,

$$\mathbf{v}_i = \begin{bmatrix} v_{i1} \\ \dots \\ \dots \\ v_{in} \end{bmatrix}$$

We assume that the basis is orthogonal, that is

$$\mathbf{v}_j^T \mathbf{v}_i = \delta_{ij}.$$

Now consider a set of vectors $\mathbf{w}_i$ derived from applying an orthogonal transformation to the original vectors $\mathbf{v}_i$

$$\mathbf{w}_i = \mathbf{S}\mathbf{v}_i,$$

The dot product between two transformed vectors is preserved due to the property $\mathbf{S}^T\mathbf{S} = \mathbf{I}$ of orthogonal matrices or $\mathbf{U}^\dagger\mathbf{U} = \mathbf{I}$ of unitary matrices. So, the transformed

vectors are orthogonal like the original vectors.

$$\begin{aligned}\mathbf{w}_j^T\mathbf{w}_i &= (\mathbf{S}\mathbf{v}_j)^T(\mathbf{S}\mathbf{v}_i)\\ &= \mathbf{v}_j^T\mathbf{S}^T\mathbf{S}\mathbf{v}_i\\ &= \mathbf{v}_j^T\mathbf{v}_i\\ &= \delta_{ij}.\end{aligned}$$

ALGORITHMS AND METHODS

To solve the matrix eigenvalue problem, we applied Jacobi's method within a C++ program to diagonalize the $N \times N$ matrix. The source code developed for this project is available at https://github.com/redpath11/phy905_thr. The directory `projects/project2/` inside that repository contains the source code described below. All file paths given here are relative to that directory of the git repository.

Jacobi's Method

Jacobi's method involves applying a series of similarity transformations to bring the original symmetric tridiagonal matrix to a diagonal one. At the end of this process, we obtain the eigenvalues and eigenvectors. The similarity transformation can be interpreted geometrically as a rotation about an angle θ in the n-dimensional Euclidean space. We denote such a transformation of the matrix $\mathbf{A}$ into the matrix $\mathbf{B}$ as

$$\mathbf{B} = \mathbf{S}^T\mathbf{A}\mathbf{S}.$$

where $\mathbf{S}$ has the form

$$\mathbf{S} = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \cos\theta & 0 & \dots & 0 & \sin\theta \\ 0 & 0 & \dots & 0 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0\dots & 1 & 0 & \\ 0 & 0 & \dots & -\sin\theta & \dots & \dots & 0 & \cos\theta \end{bmatrix}.$$

The resulting matrix elements are

$$\begin{aligned} b_{ii} &= a_{ii}, i \neq k, i \neq l\\ b_{ik} &= a_{ik}\cos\theta - a_{il}\sin\theta, i \neq k, i \neq l\\ b_{il} &= a_{il}\cos\theta + a_{ik}\sin\theta, i \neq k, i \neq l\\ b_{kk} &= a_{kk}\cos^2\theta - 2a_{kl}\cos\theta\sin\theta + a_{ll}\sin^2\theta\\ b_{ll} &= a_{ll}\cos^2\theta + 2a_{al}\cos\theta\sin\theta + a_{kk}\sin^2\theta\\ b_{kl} &= (a_{kk} - a_{ll})\cos\theta\sin\theta + a_{kl}(\cos^2\theta - \sin^2\theta)\end{aligned}$$

where θ for each transformation is chosen such that the off diagonal b_{kl} is zero. We re-write this condition with the abbreviations $t = \tan\theta$ and the definition

$$\tau = \frac{a_{ll} - a_{kk}}{2a_{kl}}$$

as

$$t^2 + 2\tau t - 1 = 0$$

which has solutions

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

Now, from $\cos^2\theta + \sin^2\theta = 1$ we can write

$$\cos\theta = \frac{1}{\sqrt{1+t^2}}$$

from which we can get $\sin\theta = t\cos\theta$.

With this procedure for zeroing an off-diagonal matrix element, we can generate a recipe for iteratively diagonalizing any symmetric matrix:

1. Set some numerical limit that determines when the matrix elements are small enough to be effectively zero (e.g. 10^{-8})
2. Find the largest matrix element
3. Compute $\tau, t, \cos\theta, \sin\theta$ that will zero this matrix element
4. Apply the similarity transformation defined in the previous step to the matrix to obtain a transformed matrix
5. Repeat the previous three steps until all matrix elements are less than the limit set in the first step

Implementing Jacobi's method

We implemented Jacobi's method in a C++ function that performs the steps listed in the previous section. This function sets the limit from step one in the previous section to 10^{-8} as well as a limit on the maximum number of iterations (N^3, where N is the dimensionality of the matrix) in case the off-diagonal limit isn't satisfied. We then, loop until one of these conditions is met. For each iteration, the function `FindMaxOffDiag` searches the current version of the matrix for the largest off diagonal matrix and `Rotate` computes θ then applies similarity transformation. The similarity transformation of each iteration is also applied to a second $N \times N$ matrix that was initialized to the N-dimensional identity

matrix. This allows us to obtain the eigenvectors. After the loop is complete, the original tridiagonal matrix has been transformed into a diagonal matrix with the eigenvalues along the main diagonal. One final loop over the main diagonal is performed to find the lowest eigenvalue which is printed out along with its index. We use the index to extract the corresponding eigenvector from the eigenvector matrix.

Unit Tests

During development of our code, we implemented a series of unit tests to verify that we were coding the algorithm correctly. The first test was to ensure that we setup the tridiagonal matrix correctly. The function `SetMatrix` generates a matrix of the specified dimension and $\rho_{\max}$. We wrote this matrix out to a file `Benchmark/test0.out` that was used as input to a Python script that employs the `numpy.linalg.eigvals` solver to find the eigenvalues. In this way, we verified that the non-interacting matrix was being properly set up by checking that the lowest eigenvalues were approximately the analytic HO eigenvalues (3,7,11,15,...). This test was implemented via the `test0` function in `src/proj2.cc`.

The second test checked that the search for the largest off-diagonal matrix element works properly. In this simple test, we set up a 4×4 matrix and initialize a few of the off-diagonal elements to random values. We then run the `FindMaxOffDiag` function on this matrix and ensure that it returns the proper element. Furthermore, we checked that the larget off-diagonal element was chosen based on the absolute value of the element. This test was coded in the `test1` function in `src/proj2.cc`.

Our third unit test checked the Jacobi solver results against the output from `numpy.linalg.eigvals` for simple 3×3 and 4×4 matrices stored in the files `Benchmark/test2_3.in` and `Benchmark/test2_4.in` respectively. The function `test2` in `src/proj2.cc` reads in one of these matrices, applys the Jacobi algorithm and checks that the eigenvectors are orthogonal.

The next test we performed checked that our code calculates eigenfunctions that make sense physically. We ran the Jacobi algorithm with 50 gridpoints and $\rho_{\max} = 5$ to check the radial behavior of the first three eigenfuctions - see FIG. 1. There are n nodes in the radial wavefunctions (see eq. 1) and the wavefunctions extend farther from 0 for higher eigenenergies. These trends are clearly visible in FIG. 1.

A final test we implemented checks the orthogonality of the eigenfunctions resulting from the diagonalization. We showed in a preceeding section that the orthogonality of a basis set is maintained under similarity transformations. Since Jacobi's method is essentially a series of similarity transformations and since the same similarity transformations are applied to a set of orthonormal unit vectors to generate the eigenvectors (see preceeding section), the resulting eigenvectors should be orthogonal. To check this, we compute the dot product between the resulting eigenvectors and write out a warning if it is greater than 10^{-8} .

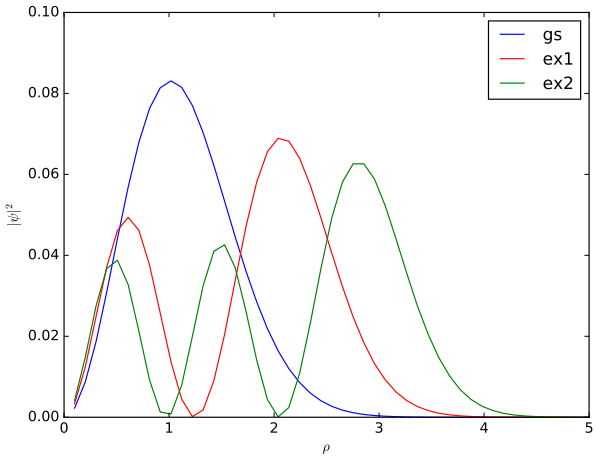

FIG. 1. The square modulus of the ground state (blue) and first two excited state (red and green) wavefunctions with computed with $N = 50, \rho_{\max} = 5$.

RESULTS AND DISCUSSIONS

The non-interacting case

We tested how many grid points and what values for $\rho_{\max}$ give the correct results for the lowest three eigenvalues to four decimal places. The results are summarized in Table I. A $\rho_{\max} = 5$ appears to be large enough to give correct predictions for the first three eigenvalues. This can also be seen in FIG. 1 where all three wavefunctions go to zero before $\rho_{\max}$ is reached. In order to attain the desired four-decimal place precision, we increased the number of grid points to 500. In general, both $\rho_{\max}$ and the number of grid points contribute to the accuracy of the numerical eigenvalues since together they determine the how close the calculation approaches the continuous case (how fine-grained the grid is). To accurately reproduce the higher eigenenergies, we need a larger $\rho_{\max}$ to ensure that the boundary condition $u(\rho_{\max}) = 0$ is satisfied for the more extended wavefunctions. If we only care about the the lowest eigenenergies, we can keep $\rho_{\max}$ small for the same number of grid points and achieve a better approximation to the continuous case.

In TABLE II we list the simulation parameters $(N, \rho_{\max})$ and the ground state eigenenergy for different oscillator strengths (ω). In general, the smaller (ω) the shallower the oscillator potential which shifts the

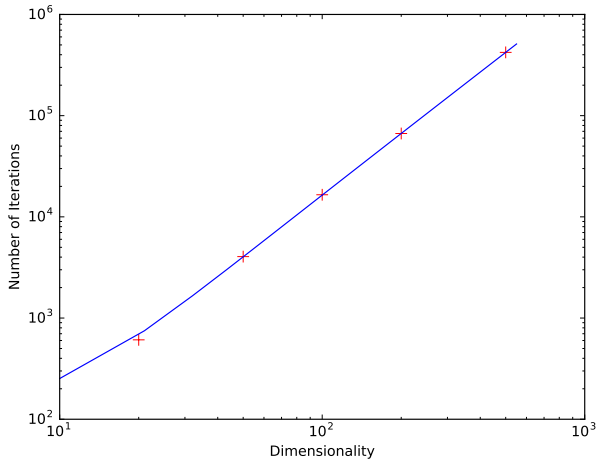

FIG. 2. The number of similarity transformations needed to diagonalize the matrix is plotted against the matrix dimensionality. A quadratic fit gives a $2N^2 - 8N$ dependence of the number of similarity transformations on the dimensionality N.

eigenenergies lower. This also makes the wavefunctions more diffuse and requires a larger $\rho_{\max}$ to ensure that the upper boundary condition is satisfied. We varied N and $\rho_{\max}$ until the ground state energy converged. TABLE II shows that for the shallowest oscillator potential ($\omega = 0.01$) a large $\rho_{\max}$ is needed to capture the ground state wavefunction. For the other three cases, $\rho_{\max} = 5$ was sufficient. The ground state wavefunctions with different ω are plotted in FIG. 3. Note that the extent of the ground state wavefunction increases with decreasing ω.

		$2E_i = 4i + 3$			
		3	7	11	
Numerical Results					
N	$\rho_{\max}$	$2E_1$	$2E_2$	$2E_3$	# iterations
20	5	2.980329	6.900902	10.752631	609
50	5	2.996871	6.984340	10.961832	4047
100	5	2.999219	6.996093	10.990595	16532
200	5	2.999805	6.999025	10.997780	66618
500	5	2.999969	6.999846	10.999802	421903
50	10	2.987443	6.936917	10.845289	3681
100	10	2.996871	6.984339	10.961741	15522
200	10	2.999219	6.996092	10.990460	64026
50	15	2.971582	6.856332	10.645120	2718
100	15	2.992951	6.964661	10.913531	14312
200	15	2.998241	6.991200	10.978512	61378

TABLE I. Lowest three eigenvalues for different values of N and $\rho_{\max}$. The first two lines give the analytic formula and the corresponding values for the first three eigenenergies of the harmonic oscillator potential.

ω	N	$\rho_{\max}$	E_0
0.01	100	10	0.0996
	150	15	0.0496
	200	20	0.0351
	250	25	0.0399
	250	50	0.0299
	250	100	0.0299
0.5	100	10	1.4992
	150	15	1.4992
	50	5	1.4994
	250	5	1.5002
1.0	100	10	2.9969
5.0	100	10	14.9214
	100	5	14.9884
	250	5	14.9969

TABLE II. Ground state eigenenergies for different combinations of $(N, \rho_{\max})$ and different oscillator strengths (ω) to check for convergence of E_0 for the different ω (non-interacting case).

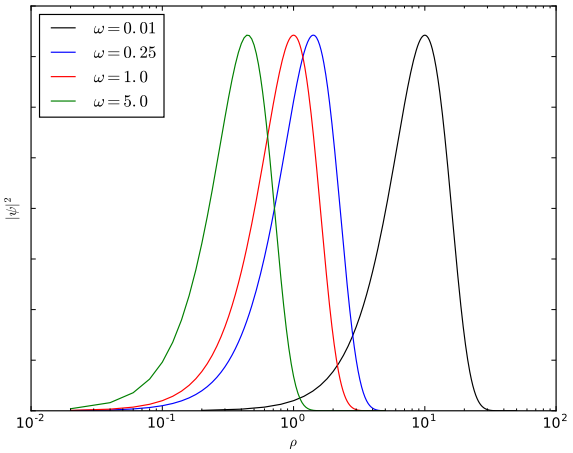

FIG. 3. Ground state wavefunctions for different oscillator strengths and no interaction between the electrons. The weakest potential ($\omega = 0.01$) is shown in black and the strongest potential ($\omega = 5$) is plotted in green. The wavefunctions have been scaled to the $\omega = 5$ wavefunction.

The Coulomb interaction

We changed the potential to include the Coulomb interaction between the two electrons as described above. This problem is more difficult to solve analytically but it has been done for certain oscillator frequencies - see Taut [2]. We compare our results to these analytic results in TABLE III. We then ran our simulation for $\omega_r = 0.01, 0.5, 1.0, 5.0$. The simulation parameters and ground state eigenenergies are listed in TABLE IV and the wavefunctions are plotted in FIG. 4.

We find that the ground state eigenenergies are higher than the non-interacting case, this is due to the additional Coulomb interaction energy. Furthermore, FIG. 4

ω_r	N	$\rho_{\max}$	E_0 (this work)	E_0 from [2]	# iterations
0.25	200	5	1.3066	1.25	67116
	200	10	1.2499		65222
	200	20	1.2498		62322
	400	20	1.2499		253817
0.05	200	10	0.3950	0.35	65508
	200	20	0.3499		63684
	400	20	0.3499		257392

TABLE III. Numerical estimates for the eigenenergies compared to the analytic results for two specific oscillator frequencies. The simulation parameters N and $\rho_{\max}$ were changed until the eigenenergies converged to the analytic values.

ω_r	N	$\rho_{\max}$	E_0	# iterations
0.01	400	200	0.1058	220550
	800	100	0.1058	970300
	400	100	0.1058	238187
	400	50	0.1058	248626
	50	50	0.1058	3655
	50	10	0.3034	3655
	50	5	0.8159	4110
0.5	400	50	2.2289	196727
	800	100	2.2289	416090
1.0	400	50	4.0527	111368
	800	100	4.0527	171271
5.0	400	50	17.3195	16217
	800	100	17.3195	22457

TABLE IV. Simulation parameters and ground state eigenenergies for two electrons in a harmonic oscillator potential.

shows that the relative distance between the two electrons increases as the oscillator strength is increased. This is somewhat counter-intuitive since the oscillator strength may be loosely interpreted as the degree of confinement for the two-electron system. However, taking into account the fact that it is the relative separation being plotted (the oscillator strength is divided out in the definition of α) we interpret this effect as the increasing influence of the Coulomb repulsion as the electrons become more tightly confined.

CONCLUSIONS

We have applied Jacobi's method to solve the two-body Schrödinger equation for electrons in a harmonic oscillator potential. We tested the C++ code developed to model this system and found it in good agreement with analytic results for both the non-interacting and Coulomb interacting cases.

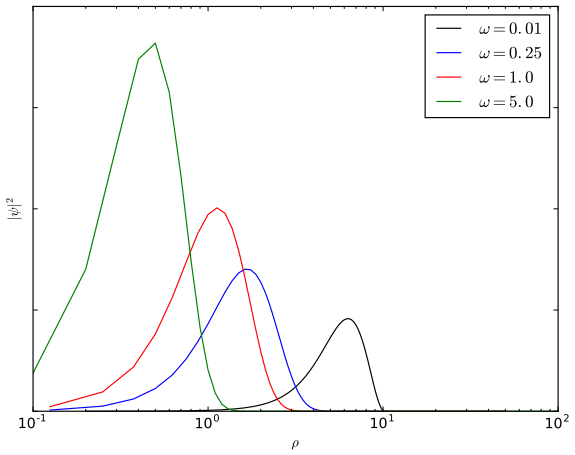

FIG. 4. Ground state wavefunctions vs. separation distance for two Coulomb-interacting electrons in a harmonic oscillator potential.

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

[2] M. Taut. Two electrons in an external oscillator potential: Particular analytic solutions of a coulomb correlation problem. *Phys. Rev. A*, 48:3561–3566, Nov 1993. doi: 10.1103/PhysRevA.48.3561. URL http://link.aps.org/doi/10.1103/PhysRevA.48.3561.