

# Jacobi's Method Applied to a Buckling Beam and the Three-Dimensional Harmonic Oscillator

Håkon Fossheim

Github repository

## Abstract

The buckling beam and the three-dimensional harmonic oscillator have been studied. The eigenmodes of the buckling beam were found, and the ground-state for the three dimensional harmonic oscillator was studied in a single-electron system and a two-electron configuration where Coulomb forces are present. This was done numerically by applying Jacobi's method.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Theory</b>	<b>2</b>
2.1	Buckling Beam . . . . .	2
2.2	Jacobi's Method . . . . .	4
2.3	Preservation of Orthogonality . . . . .	6
2.4	Harmonic oscillator, one electron . . . . .	7
2.5	Two electrons . . . . .	7
<b>3</b>	<b>implementation</b>	<b>8</b>
<b>4</b>	<b>Results</b>	<b>8</b>
<b>5</b>	<b>Discussion</b>	<b>12</b>
<b>6</b>	<b>Conclusion</b>	<b>12</b>

# 1 Introduction

Through re-scaling of equations, the algorithm for solving one particular problem may be applied to a completely separate problem with only minor adjustments, as long as they share similar differential equations. The differential equation that describes how a beam fastened at both ends buckles under stress is [3]

$$\frac{d^2u(\rho)}{d\rho^2} = -\lambda u(\rho) \tag{1}$$

The same algorithm for solving this problem can be applied to solving the Schödinger equation for an electron in a three-dimensional harmonic oscillator potential. This was done first for one electron, then for two electrons subject to each others coulomb potential.

The theory section of this article will go through the Buckling beam- and the harmonic oscillator problem and present the details needed for setting up the algorithm and solving the problems. Jacobi's method will be introduced as a way of solving the problems numerically. Results will then be presented, The Jacobi algorithm will be shortly discussed, and a conclusion of the article will be made.

## 2 Theory

### 2.1 Buckling Beam

A beam is fastened at both endpoints  $x = 0$  and  $x = L$ . A force with direction towards  $x = 0$  is then applied at  $x = L$ , making the beam deviate from the horizontal.

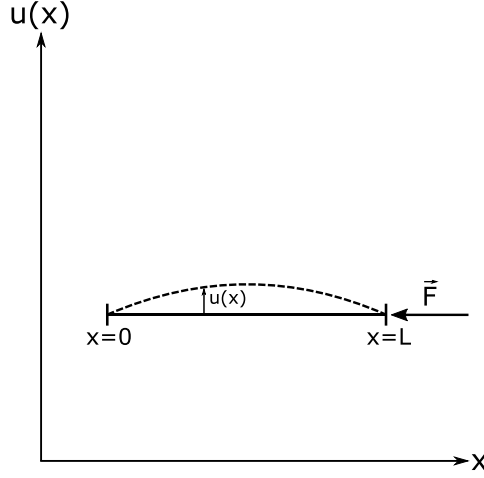


Figure 1: The figure shows a beam fastened at both endpoints. A force is applied at  $x = L$ , making the beam buckle.

This deviation at any point  $x \in [0, L]$  is described by the following differential equation

$$R \frac{d^2 u(x)}{dx^2} = -F u(x) \quad (2)$$

Where  $F$  is the mentioned force and  $R$  is a constant defined by properties like the rigidity of the beam.

To scale the equation so that  $x \in [0, L] \rightarrow \rho \in [0, 1]$ , set  $\rho = x/L$ . This gives

$$\frac{d^2 u}{dx^2} = \underbrace{\frac{d^2 \rho}{dx^2}}_0 \frac{du}{d\rho} + \left( \frac{d\rho}{dx} \right)^2 \frac{d^2 u}{d\rho^2} = \frac{1}{L^2} \frac{d^2 u}{d\rho^2} \quad (3)$$

The scaled differential equation thus becomes

$$\frac{d^2 u(\rho)}{d\rho^2} = -\frac{FL^2}{R} u(\rho) = -\lambda u(\rho) \quad (4)$$

By Maclaurin expanding  $u(x + h)$  and  $u(x - h)$  around  $x$  and ignoring fourth order terms, the resulting discretized version of equation 4 is found.

$$\frac{-u_{i+1} + 2u_i - u_{i-1}}{h^2} = \lambda u_i \quad (5)$$

Where  $u_{i+1} = u(x + h)$ ,  $u_i = u(x)$  and  $u_{i-1} = u(x - h)$ . The step size is given by  $h = \frac{\rho_{max} - \rho_{min}}{N}$  where  $N$  is the number of grid-points.

When written in matrix form, this set of equations transforms into the following eigenvalue problem

$$\mathbf{A}\vec{u} = \lambda\vec{u} \quad (6)$$

$$\begin{bmatrix} d & e & 0 & \dots & \dots & 0 \\ e & d & e & 0 & \dots & \dots \\ 0 & e & d & e & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & e & d & e \\ 0 & \dots & \dots & 0 & e & d \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \dots \\ u_{N-2} \\ u_{N-1} \end{bmatrix} = \lambda \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \dots \\ u_{N-2} \\ u_{N-1} \end{bmatrix} \quad (7)$$

Where  $d = 2/h^2$  and  $e = -1/h^2$ .

## 2.2 Jacobi's Method

To find the eigenvalues of equation 6, a series of similarity transforms are performed. This is done by multiplying  $\mathbf{S}^T$  on the left of both sides of equation 6 and inserting  $\mathbf{S}^T\mathbf{S} = \mathbf{I}$  between  $\mathbf{A}$  and  $\vec{u}$

$$(\mathbf{S}\mathbf{A}\mathbf{S}^T)(\mathbf{S}\vec{x}) = \lambda(\mathbf{S}^T\vec{x}) \quad (8)$$

The matrices  $\mathbf{A}$  and  $\mathbf{S}\mathbf{A}\mathbf{S}^T$  are similar. The importance of similarity lies in the fact that the resulting matrix has the same eigenvalues. Therefore, if  $\mathbf{S}$  is picked in a way that reduces the off-diagonal values, this procedure can be repeated until

$$(\mathbf{S}_N^T \dots \mathbf{S}_1^T \mathbf{A} \mathbf{S}_1 \dots \mathbf{S}_N)(\mathbf{S}_N^T \dots \mathbf{S}_1^T \vec{u}) = \lambda(\mathbf{S}_N^T \dots \mathbf{S}_1^T \vec{u}) \quad (9)$$

$$\mathbf{D}\vec{v} = \lambda\vec{v} \quad (10)$$

The eigenvalues are given by the diagonal values of  $\mathbf{D}$ . These may be distinct or not. To find the eigenvectors, set  $\mathbf{S}_N^T \dots \mathbf{S}_1^T = \tilde{\mathbf{S}}^T$  and  $\mathbf{S}_1 \dots \mathbf{S}_N = \tilde{\mathbf{S}}$ . It can then be shown that [1]

$$\mathbf{A}\tilde{\mathbf{S}}(:,j) = \lambda_j\tilde{\mathbf{S}}(:,j) \quad (11)$$

Thus, the eigenvectors are given by the columns of  $\tilde{\mathbf{S}}$ .

Let's first consider a 2x2 example. The transformation matrix  $S$  rotates the plane around an axis by an angle  $\theta$ . It is given by

$$\mathbf{S} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} = \begin{pmatrix} c & s \\ -c & s \end{pmatrix} \quad (12)$$

The matrix  $\mathbf{B}$  resulting from the transformation

$$\mathbf{B} = \mathbf{S}^T \mathbf{A} \mathbf{S} \quad (13)$$

$$\begin{pmatrix} b_{kk} & b_{kl} \\ b_{lk} & b_{ll} \end{pmatrix} = \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \begin{pmatrix} a_{kk} & a_{kl} \\ a_{lk} & a_{ll} \end{pmatrix} \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \quad (14)$$

Will then have elements

$$\begin{aligned} b_{kk} &= a_{kk}c^2 - 2a_{kl}cs + a_{ll}s^2 \\ b_{ll} &= a_{ll}c^2 + 2a_{kl}cs + a_{kk}s^2 \\ b_{kl} &= b_{lk} = a_{kl}(c^2 - s^2) + (a_{kk} - a_{ll})cs \end{aligned}$$

The goal is for the off-diagonals of  $\mathbf{B}$  to be equal to zero. Asserting this yields

$$\underbrace{\frac{a_{ll} - a_{kk}}{a_{kl}}}_{2\tau} cs = c^2 - s^2 \quad (15)$$

Defining  $t = \frac{s}{c}$  and manipulating equation 15 gives

$$t = -\tau \pm \sqrt{1 + \tau^2} \quad (16)$$

$$c = \frac{1}{\sqrt{1 + t^2}} \quad (17)$$

$$s = tc = \frac{t}{\sqrt{1 + t^2}} \quad (18)$$

By consistently picking the smallest of the two roots in equation 16, the matrix is rotated by a tiny angle, and the off-diagonals of  $\mathbf{B}$  are ensured to get smaller.

This algorithm can be extended to a general  $N \times N$  case. The rotation matrix is then called a Given's matrix and performs a rotation around an axis in a hyperplane. It's then given by [2]

$$\mathbf{S} = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & 0 & \dots \\ 0 & 0 & \dots & \cos \theta & 0 & \dots & 0 & \sin \theta \\ 0 & 0 & \dots & 0 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & 0 & \dots \\ 0 & 0 & \dots & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & \dots & -\sin \theta & \dots & \dots & 0 & \cos \theta \end{pmatrix} \quad (19)$$

Where

$$s_{kk} = s_{ll} = \cos \theta, s_{kl} = -s_{lk} = -\sin \theta \quad (20)$$

and  $k < l$

The matrix  $\mathbf{B}$  resulting from a similarity transform given by equation 13 will have elements

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

### 2.3 Preservation of Orthogonality

Consider a basis of vectors  $\vec{v}_i$

$$\vec{v}_i = \begin{bmatrix} v_{i1} \\ \cdots \\ \cdots \\ v_{in} \end{bmatrix} \quad (21)$$

and assume that the basis is orthogonal

$$\vec{v}_j^T \vec{v}_i = \delta_{ij} \quad (22)$$

Performing an orthogonal transformation on both vectors:

$$\vec{w}_j = \mathbf{S}\vec{v}_j, \vec{w}_i = \mathbf{S}\vec{v}_i \quad (23)$$

and so

$$\vec{w}_j^T \vec{w}_i = (\mathbf{S}\vec{v}_j)^T (\mathbf{S}\vec{v}_i) = \vec{v}_j^T \mathbf{S}^T \mathbf{S} \vec{v}_i \quad (24)$$

Since this is an orthogonal transformation ( $\mathbf{S}^T \mathbf{S} = \mathbf{I}$ ), equation 24 gives

$$\vec{w}_j^T \vec{w}_i = \vec{v}_j^T \vec{v}_i = \delta_{ij} \quad (25)$$

And orthogonality is therefore preserved.

Note that

$$\vec{v}_j^T \vec{v}_i = \vec{v}_j \cdot \vec{v}_i = \vec{w}_j \cdot \vec{w}_i \quad (26)$$

Thus, the dot product is also conserved after an orthogonal transformation.

## 2.4 Harmonic oscillator, one electron

Consider an electron moving in a three-dimensional harmonic oscillator potential. The potential is spherically symmetric, so only the radial part of Schrödinger's equation is of interest. It reads [3]

$$-\frac{\hbar}{2m} \left( \frac{1}{r^2} \frac{d}{dr} r^2 \frac{d}{dr} - \frac{l(l+1)}{r^2} \right) R(r) + V(r)R(r) = ER(r) \quad (27)$$

Where  $V(r) = \frac{1}{2}kr^2$  is the potential and  $k = m\omega^2$  where  $\omega$  is the angular frequency of the oscillation. The energies are given by

$$E_{nl} = \hbar\omega \left( 2n + l + \frac{2}{3} \right) \quad (28)$$

With  $n = 0, 1, 2, \dots$  and  $l = 0, 1, 2, \dots$ . In this project, the angular momentum  $l$  is set to zero. After re-scaling equation 28 and performing some algebraic manipulations [3], one arrives at the following equation

$$-\frac{d^2}{d\rho^2} u(\rho) + \rho^2 u(\rho) = \lambda u(\rho) \quad (29)$$

Which has boundary conditions  $u(0) = 0$  and  $u(\infty) = 0$ .  $\rho$  is dimensionless and is given by  $\rho = (r/\alpha)$  where  $\alpha$  is a constant with dimension length. The first few eigenvalues are  $\lambda_0 = 3$ ,  $\lambda_1 = 7$ ,  $\lambda_2 = 11$  and  $\lambda_3 = 15$ . The minimum value of  $\rho$  is  $\rho_{min} = 0$ . Since  $\rho_{max}$  cannot be set equal to infinity, it has to be approximated by a different number. Reasonable results were achieved using  $\rho_{max} = 5$ .

Equation 29 shows that the potential term is given by  $V = \rho^2$ . Equation 29 is very similar to that of the differential equation for a buckling beam. The algorithm for the buckling beam can indeed be applied here, but minor changes have to be made, namely adding the potential term to the diagonals. The step size also needs to be changed to  $h = \frac{\rho_{max} - \rho_{min}}{N}$  where  $N$  is the number of grid points.

## 2.5 Two electrons

For two electrons with no repulsive Coulomb interaction, the following Schrödinger equation holds [3]

$$\left( -\frac{\hbar^2}{2m} \frac{d^2}{dr_1^2} - \frac{\hbar^2}{2m} \frac{d^2}{dr_2^2} + \frac{1}{2}kr_1^2 + \frac{1}{2}kr_2^2 \right) u(r_1, r_2) = E^{(2)}u(r_1, r_2) \quad (30)$$

Note that  $u(r_1, r_2)$  is the two-electron wave function and  $E^{(2)}$  is the two-electron energy. By introducing the relative coordinate  $\vec{r} = \vec{r}_1 - \vec{r}_2$  and the center-of-mass coordinate  $\vec{R} = \frac{1}{2(\vec{r}_1 + \vec{r}_2)}$ , equation 30 becomes

$$\left(-\frac{\hbar^2}{m} \frac{d^2}{dr^2} - \frac{\hbar^2}{4m} \frac{d^2}{dR^2} + \frac{1}{4}kr^2 + kR^2\right) u(r, R) = E^{(2)}u(r, R) \quad (31)$$

Where  $E^{(2)} = E_r + E_R$ . Using the ansatz  $u(r, R) = \psi(r)\phi(R)$  and adding the repulsive Coulomb interaction between the electrons

$$v(r_1, r_2) = \frac{\beta e^2}{|\vec{r}_1 - \vec{r}_2|} \quad (32)$$

where  $\beta e^2 = 1.44\text{eV nm}$  gives

$$\left(-\frac{\hbar^2}{m} \frac{d^2}{dr^2} + \frac{1}{4}kr^2 + \frac{\beta e^2}{r}\right) \psi(r) = E_r \psi(r) \quad (33)$$

By introducing the dimensionless variable  $\rho/\alpha$  and manipulating equation 33, the Schrödinger equation takes the form

$$-\frac{d^2}{d\rho^2} \psi(\rho) + \omega_r^2 \rho^2 \psi(\rho) + \frac{1}{\rho} = \lambda \psi(\rho) \quad (34)$$

Here  $\omega_r = \frac{1}{4} \frac{mk}{\hbar^2} \alpha^4$ ,  $\alpha = \frac{\hbar^2}{m\beta e^2}$  reflects the strength of the potential, and  $\lambda = \frac{m\alpha^2}{\hbar^2} E$  are the eigenvalues.

Again, by altering the values of  $\rho_{min}$  and  $\rho_{max}$  and setting the diagonal elements to  $d = \omega_r^2 \rho^2 + \frac{1}{\rho}$  (which is the potential), the algorithm used for the buckling beam can be applied.

### 3 implementation

Catch2 was used to perform unit tests in this project. The user needs to add "catch.hpp" to the project and include this header file in the main.

The Armadillo library is used throughout the code.

The function "jacobiMethod" performs  $n^3$  similarity transforms, where  $n$  is the number of grid-points, and stores the eigenvectors in the matrix  $R$ . The eigenvalues are stored in the matrix  $A$  as the diagonal elements. These eigenvectors can then be written to file using the function "writeToFile".

### 4 Results

To determine how many transforms are needed for the off-diagonal values to be sufficiently small, the maximum off-diagonal value was found for  $n$ ,  $n^2$  and  $n^3$  iterations for different values of  $n$ . The results are shown in table 1



Maximum off-diagonal values			
Integration points, n	$n$ iterations	$n^2$ iterations	$n^3$ iterations
10	60.5	0.0655786	$6.12792 \cdot 10^{-9}$
50	1300.5	0.453996	$9.55781 \cdot 10^{-9}$
100	5100.5	0.724792	$9.94187 \cdot 10^{-9}$
200	20200.5	1.27345	$9.99204 \cdot 10^{-9}$

Table 1: The three rightmost columns show the maximum off-diagonal element after a certain number of rotations.

As seen in table 1, the maximum off-diagonal values for  $n$  iterations are far too large. For  $n^2$  iterations, the values become more reasonable, but are still too large to be approximated by zero. For  $n^3$  iterations, the maximum off-diagonal values are vanishingly small, and the matrix can be approximated as being diagonal.

Run times for Jacobi's method and Armadillo's eigen solver		
Integration points, n	Jacobi	Armadillo
10	0.12326 ms	1.04552 ms
50	2.03342 ms	19.2883 ms
100	17.7089 ms	126.641 ms
200	99.1053 ms	902.498 ms

Table 2: Run times for Jacobi's method vs Armadillo's eigen solver for the buckling beam problem.

Table 2 show the run times for Jacobi's method compared with that of Armadillo's inbuilt function for finding eigenvalues and eigenvectors. As seen, Jacobi's method is about 10 times faster, regardless of the number of grid-points.

Eigenvalues for three-dimensional harmonic oscillator				
Integration points, n	$\lambda_0$	$\lambda_1$	$\lambda_2$	$\lambda_3$
10	2.9339	6.6597	10.1415	13.3454
50	2.9969	6.9849	10.9634	14.9374
100	2.9992	6.9961	10.9908	14.9885
200	2.9998	6.9990	10.9978	15.0015
500	2.9999	6.9998	10.9998	15.0051

Table 3: Eigenvalues for the one-electron harmonic oscillator. The analytical eigenvalues are  $\lambda_0 = 3, \lambda_1 = 7, \lambda_2 = 11$  and  $\lambda_3 = 15$

As seen in table 3, the numerical eigenvalues approach the analytical ones as  $n$  increases. It is noteworthy that  $\lambda_3 - 15$  is larger for  $n = 500$  than it is for  $n = 200$ . This might be due to round-off errors as  $h$  gets small.

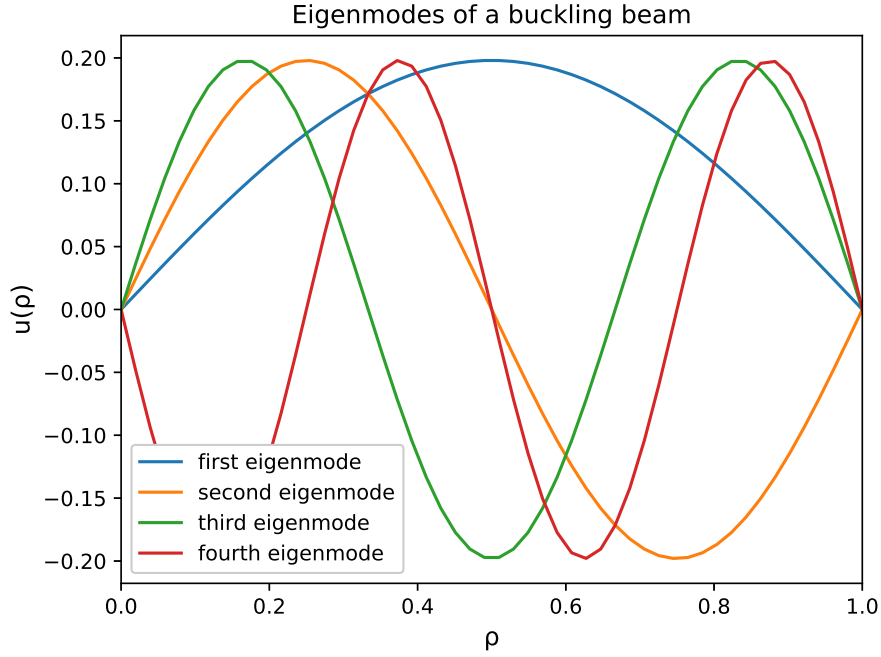


Figure 2: Analytical solution vs Gaussian elimination with 100 grid points

Figure 2 shows the ways a beam fastened at both ends might buckle under a force applied to one of the endpoints. The eigenmodes seen in the figure are the same as the standing waves for a string fastened at both ends.

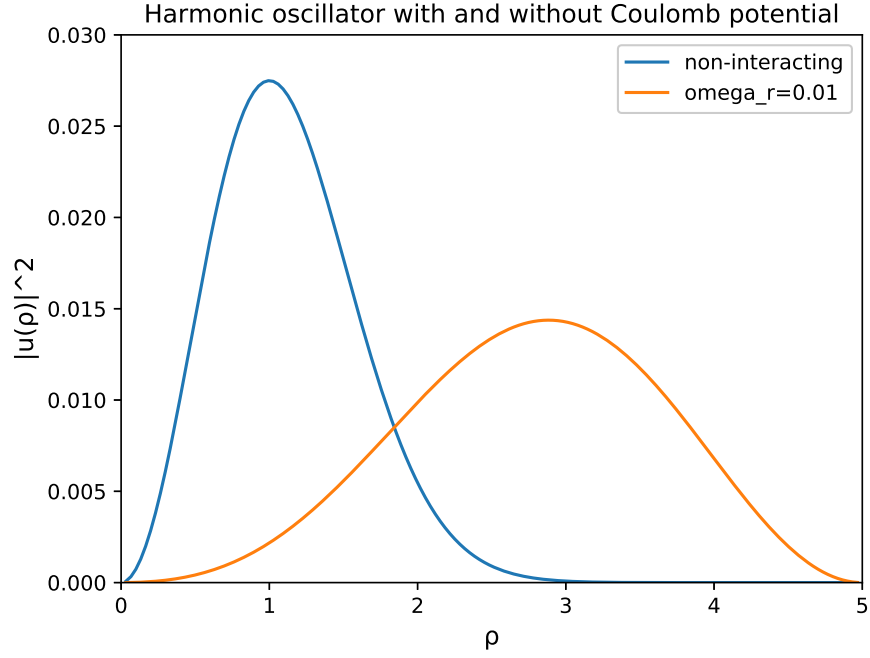


Figure 3: Analytical solution vs Gaussian elimination with 100 grid points

The probability density  $|u(\rho)|^2$  for the one- and two electron harmonic oscillator is shown in figure 3. The plots makes sense if one imagines the electrons are fastened to strings. When two electrons are fastened to both ends for a string, they will be more likely to be found further from the center of the string, since the Coulomb potential will repel them from each other. Therefore, the peak of the probability density should be shifted to the left, as seen in figure 3.

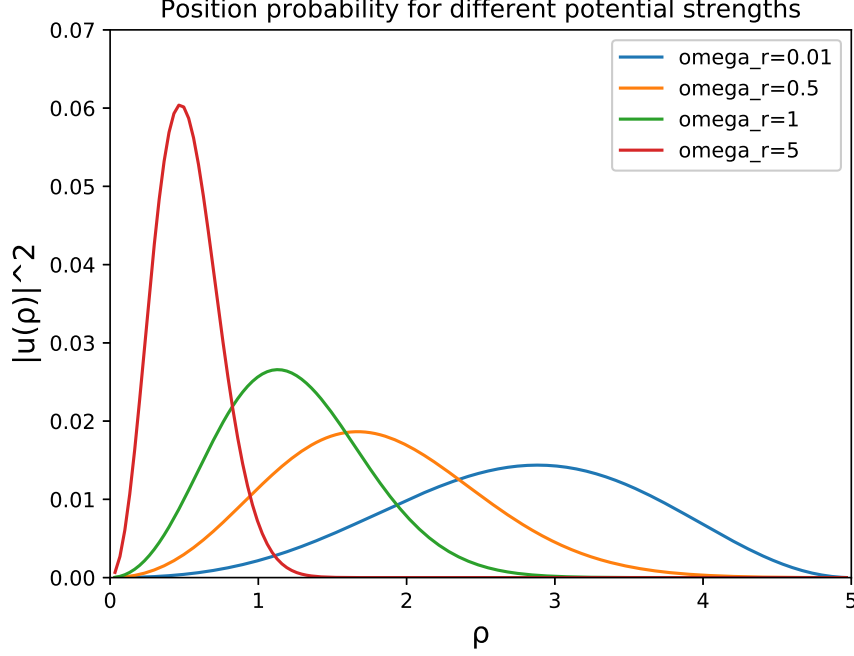


Figure 4: Analytical solution vs Gaussian elimination with 100 grid points

Figure 4 shows the ground state probability density for different values of  $\omega_r$ . As  $\omega_r$  increases, the electrons get forced closer together, and the position probability peak gets shifted to the left.

## 5 Discussion

The strength of Jacobi's method is that it is easy to parallelize. The convergence rate however, is somewhat poor. In this project,  $n^3$  rotations was used to eliminate the off-diagonal elements. This may be a bit much, but let's assume this many rotations are necessary. Every rotation requires  $4n$  operations, resulting in  $4n^4$  operations to diagonalize the matrix. The reason so many rotations are required is that off-diagonal elements that were previously zero may become non-zero after a new rotation.

## 6 Conclusion

By re-scaling the differential equations for different problems, the same algorithm for solving one problem may be applied (with minor adjustments) to solving another problem that shares a similar differential equation. In this case, the algorithm for the buckling beam could be applied to the har-

monic oscillator problem by simply changing the diagonal elements of the matrix to be rotated.

The way the beam buckles under pressure was found to have distinct eigenmodes similar to the standing waves of a string fastened at both ends.

Also, the probability function was found to shift further out from the origin when a coulomb force is present in the harmonic oscillator.

## References

- [1] Golub Gene H. and Van Loan Charles F. *Matrix Computations. Third Edition.* The Johns Hopkins University Press, 1996.
- [2] M. Hjorth-Jensen. *Computational Physics Lecture Notes Fall 2015.* Aug. 2015. URL: <https://github.com/CompPhysics/ComputationalPhysics/tree/master/doc/Lectures>.
- [3] M. Hjorth-Jensen. *Project 2, Computational Physics I FYS3150/FYS4150.* Sept. 2018. URL: <https://github.com/CompPhysics/ComputationalPhysics/blob/master/doc/Projects/2018/Project2/pdf/Project2.pdf>.