Implementation of the Jacobi eigenvalue method with classical and quantum mechanical applications

Carl Martin Fevang, Håkon Kvernmoen

September 30, 2020

Abstract

We have implemented Jacobi's eigenvalue algorithm in C++. By applying similarity transforms using Givens rotational matrices we can reduce the original symmetric matrix into a diagonal matrix where the eigenvalues are contained as the diagonal matrix elements. Trough a numerical test we found that the number of rotations grew as N^2 with different $N \times N$ matrices and comparing with other eigenvalue solvers such as those from the popular armadillo-library we found that Jacobi's method is not feasible for larger matrices. Applying this to the classical one dimensional buckling beam problem we found that the numerical eigenvectors closely resemble the analytical ones. We also applied our solver to a quantum mechanical harmonic oscillator potential, first with one electron and later with two. We found that we could reproduce the analytical results for the eigenvalues of the energy of this system rather accurately. In the case with two electrons interacting with through the Coulomb force, the results were more complex, yet we could make some observations on the behaviour of this system also, using the our numerical solver.

1 Introduction

In this report we will investigate numerical solutions to eigenvalue problems, and relate it to three physical systems. Eigenvalue problems arise in many physical problems, and generally appear when discretizing differential equations, which are central solving classical as well as quantum mechanical systems. We will implement Jacobi's method for finding the eigenvalues and eigenvectors of symmetric matrices. The speed at which we get reasonable solutions is an important factor. Unlike Jacobi, who sent one of his students to do all the computations by hand in his initial paper on the method named after him,[1] we can use computers to do the calculations. However, the efficiency of the algorithm does still matter to us, as there are limits to our modern patience. We will benchmark our implementation of the Jacobi method against the popular C++ library armadillo's method for finding the eigenvalues and eigenvectors of symmetric matrices.

Discretization of the second derivative for our physical systems leads to tridiagonal Toeplitz-matrices where we will apply our implementation of Jacobi's method. First we will apply it to the classical problem of the buckling beam, where a force applied to a straight horizontal beam will result in deformations. This is an important concept in engineering and is thought through for most major building projects. Secondly we will look at a quantum mechanical problem, where one electron is placed in a harmonic oscillator potential. Third we add another electron to the system and look at the effect of the repulsive Coulomb force between the electrons. This is a very active research field in nanotechnology, where electrons confined in small areas (quantom dots) can act as semiconductors having both optical and electronic properties that differ from large particles due

[1] C. G. J. Jacobi. "Über ein leichtes Verfahren die in der Theorie der Säcularstöorungen vorkommenden Gleichungen numerisch aufzulösen." In: *de Gruyter* 30.4 (1864), p. 51.

to quantum mechanical effects.[2]

2 Theory

2.1 Matrix eigenvalue problems

For the matrix $\mathbf{A}$, the vector $\mathbf{x}_i$ is said to be an eigenvector for $\mathbf{A}$ if the transformation $\mathbf{A}$ only "stretches" or "shrinks" $\mathbf{x}_i$. Mathematically this is represented by

$$\mathbf{A}\mathbf{x}_i = \lambda_i \mathbf{x}_i$$

Where λ_i is the eigenvalue corresponding the eigenvector $\mathbf{x}_i$. The eigenvalue is a scalar and represents how much the eigenvector "stretches" or "shrinks". Note that we do not include solutions where $\mathbf{x_i}$ is the null vector as this would hold for any matrix, but $\lambda_i = 0$ is allowed. If $\mathbf{A}$ is a $n \times n$ symmetric matrix with real entries it has n orthogonal eigenvectors with real eigenvalues.[3] Since $\mathbf{A}$ is a real symmetric matrix there exists a real orthogonal matrix $\mathbf{S}$ such that

$$\mathbf{S}^T\mathbf{A}\mathbf{S} = \mathbf{D} = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$$

Transforming our matrix $\mathbf{A}$ to a diagonal matrix is very useful since the diagonal matrix $\mathbf{D}$ must have its eigenvalues on the diagonal. Finding the matrix $\mathbf{S}$ than transforms $\mathbf{A}$ to $\mathbf{D}$ is not easy, so we often have to use multiple transformations $\mathbf{S}_1, \mathbf{S}_2, ..., \mathbf{S}_n$ to get our desired result. An important aspect of these transformations $\mathbf{S}$ is that they are similarity transformations. Mathematically the matrix $\mathbf{B}$ is a similarity transform of $\mathbf{A}$ if

$$\mathbf{B} = \mathbf{S}^T\mathbf{A}\mathbf{S} \qquad \textit{and} \qquad \mathbf{S}^T\mathbf{S} = \mathbf{S}^{-1}\mathbf{S} = \mathbf{I} \tag{1}$$

Where $\mathbf{I}$ represents the identity matrix. We call the matrix $\mathbf{S}$ a unitary matrix since $\mathbf{S}^T = \mathbf{S}^{-1}$. A property of these similarity transformations is that the eigenvalues don't change during the transformation.

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \lambda\mathbf{x} \\ \mathbf{S}^T\mathbf{A}\mathbf{S}(\mathbf{S}^T\mathbf{x}) &= \lambda(\mathbf{S}^T\mathbf{x}) \\ \mathbf{B}(\mathbf{S}^T\mathbf{x}) &= \lambda(\mathbf{S}^T\mathbf{x}) \end{aligned} \tag{2}$$

It's important to note that even though the eigenvalues don't change, the eigenvectors do. From equation (2) we note that the new eigenvectors $\mathbf{y}$ can be found with $\mathbf{y} = \mathbf{S}^T\mathbf{x}$. The orthogonality of the eigenvectors are also conserved during the transformations. Consider two eigenvectors $\mathbf{v}_i$ and $\mathbf{v}_j$ to the matrix $\mathbf{A}$ with different eigenvalues. The orthogonality is expressed as

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

The transformed eigenvectors $\mathbf{w}_i = \mathbf{S}^T\mathbf{v}_i$ and $\mathbf{w}_j = \mathbf{S}^T\mathbf{v}_j$ is then also orthogonal since

$$\mathbf{w}_i^T\mathbf{w}_j = (\mathbf{S}^T\mathbf{v}_j)^T(\mathbf{S}^T\mathbf{v}_i) = \mathbf{v}_i^T\mathbf{S}\mathbf{S}^T\mathbf{v}_j = \mathbf{v}_i^T\mathbf{v}_j = \delta_{ij}$$

We can preform these similarity transforms on our initial symmetric matrix $\mathbf{A}$ where the goal is to transform it to the diagonal matrix $\mathbf{D}$ containing all the eigenvalues. Starting with an orthonormal matrix (say the identity matrix) we can preform the similarity transforms as in equation 3 to obtain the orthonormal eigenvectors. The final eigenvectors will then be stored as columns in $\mathbf{U}$.

$$\mathbf{D} = \mathbf{S}_n^T ... \mathbf{S}_1^T \mathbf{A}\mathbf{S}_1 ... \mathbf{S}_n \quad \textit{and} \quad \mathbf{U} = \mathbf{S}_n^T ... \mathbf{S}_1^T \mathbf{I} \tag{3}$$

[2] Sept. 29, 2020. URL: https://en.wikipedia.org/wiki/Quantum_dot.

[3] Sept. 19, 2020. URL: http://pi.math.cornell.edu/~jerison/math2940/real-eigenvalues.pdf.

3 Implementation

3.1 Jacobi's method

All code used for the algorithm, benchmarking and plotting as well as how to use them can be found at https://github.com/hkve/FYS3150/tree/master/Project2/code

As discussed we will implement the Jacobi eigenvalue algorithm.[4] To preform a similarity transformation as in equation (1) we consider the general $n \times n$ Givens[5] rotational matrix around one axis.

$$\mathbf{S} = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \cdots & \iddots & 0 \\ \vdots & \vdots & \mathbf{R} & \cdots & 0 \\ \vdots & \iddots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Where $\mathbf{R}$ is a $l \times k$ submatrix holding the rotational elements. Defining $c = \cos\theta$ and $s = \sin\theta$, where θ is the degree with which to rotate the matrix around the axis chosen for the Givens matrix. We can now write $\mathbf{R}$ as:

$$\mathbf{R} = \begin{pmatrix} c & 0 & \cdots & 0 & s \\ 0 & 1 & \ddots & \vdots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \vdots & \ddots & 1 & 0 \\ -s & 0 & \cdots & 0 & c \end{pmatrix}$$

Preforming the matrix multiplication from equation (1) we can write the matrix elements of $\mathbf{B}$ in terms of the matrix elements of $\mathbf{A}$ and the rotational elements c and s as

$$\begin{aligned} b_{ii} &= a_{ii},\ i \neq k,\ i \neq l \\ b_{ik} &= a_{ik}c - a_{il}s,\ i \neq k,\ i \neq l \\ b_{il} &= a_{il}c + a_{ik}s,\ i \neq k,\ i \neq l \\ b_{kk} &= a_{kk}c^2 - 2a_{kl}cs + a_{ll}s^2 \\ b_{ll} &= a_{ll}c^2 + 2a_{kl}cs + a_{kk}s^2 \\ b_{kl} &= (a_{kk} - a_{ll})cs + a_{kl}(c^2 - s^2) \end{aligned} \tag{4}$$

To transform $\mathbf{A}$ into the diagonal matrix $\mathbf{D}$ we have to choose the angle θ that sets the off-diagonal elements to 0. This implies that minimizing the off-diagonal elements would move us closer to $\mathbf{D}$. Since we don't want to discriminate between positive and negative elements we minimize the norm of the off-diagonals. Reaching 0 for all these elements will take a lot of time and computational power. We settle for preforming transformations while the norm of the off-diagonal is greater than some tolerance ϵ. That is, preform transformations while

$$\text{off}(\mathbf{A}) = \sqrt{\sum_{i=1}^{n}\sum_{j=1,j\neq i}^{n} a_{ij}^2} > \epsilon$$

However, calculating this norm for every rotation will also be very computationally inefficient, resulting a lot of unnecessary FLOPs. Thus, we will perform the rotations just so long as the the absolute value of the largest non-diagonal element is greater than some ϵ. We pick out the element with the largest absolute value and choose θ such that this element is set to 0. When finding the largest absolute value we can increase performance by only checking the upper diagonal matrix elements, since our matrix is symmetric. We select the matrix element in $\mathbf{A}$ with the largest absolute values $\max|a_{ij}| > \epsilon$ and choose this as a_{kl} (the element we want to set to 0). Looking back at equation (4) and setting the last equation to 0 we obtain.

[4] Sept. 26, 2020. URL: https://en.wikipedia.org/wiki/Jacobi_eigenvalue_algorithm.

[5] Sept. 20, 2020. URL: https://en.wikipedia.org/wiki/Givens_rotation.

$$
\begin{aligned}
(a_{kk} - a_{ll})cs + a_{kl}(c^2 - s^2) &= 0 \\
(a_{kk} - a_{ll})t + a_{kl}(1 - t^2) &= 0 \\
\frac{a_{ll} - a_{kk}}{a_{kl}}t - (1 - t^2) &= 0 \\
t^2 + 2\tau - 1 &= 0 \\
\Rightarrow t &= -\tau \pm \sqrt{1 + \tau^2}
\end{aligned}
$$

Where we have defined $t = \tan(\theta) = s/c$, divided by $-c^2/a_{kl}$ and defined $\tau = (a_{ll} - a_{kk})/2a_{kl}$. We defined τ with $2a_{kl}$ in the numerator instead of a_{kl} to get a prettier solution for t. But there is a problem hidden in this equation. After many rotations a_{kl} will go to zero (since we want the non diagonal elements to be less than ϵ). The term under the square root will then be very close to τ and t would be close to zero that in turn can lead to loss of numerical precision. A simple rewrite gives us the equation

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

Now we choose the smallest angle for the fastest convergence[6] which gives if τ is positive the $+$ solution in the numerator or if τ is negative the $-$ solution. Now that we have found a solution for t that sets $b_{kl} = 0$ we can easily obtain our desired values for s and c using some trigonometric identities

$$
c = \frac{1}{\sqrt{1 + t^2}} \quad \text{and} \quad s = tc
$$

Now when calculating the new elements in matrix $\mathbf{B}$ we can manually set $b_{kl} = b_{lk} = 0$ (due to the symmetry of the matrix) and calculate the other elements using the rest of the equations (4). With the symmetric matrix $\mathbf{A}$ with elements a_{ij} and a eigenvector matrix $\mathbf{U}$ with elements u_{ij} the algorithm is written as

Set A
U = Identity matrix
while $\max_{i \neq j}|a_{ij}| > \epsilon$ **do**
  get k, l for $\max_{i \neq j}(a_{ij})^2$ ▷ max off-diag
  $\tau = (a_{ll} - a_{kk})/2a_{kl}$
  if $\tau > 0$ **then** ▷ Smallest angle
    $t = 1/(\tau + \sqrt{1 + \tau^2})$
  else if $\tau < 0$ **then**
    $t = 1/(\tau - \sqrt{1 - \tau^2})$
  end if
  $c = 1/(1 + t^2)$
  $s = tc$
  Akk = a_{kk}, All = a_{ll}
  $a_{kk} = c^2$Akk - $2cs \cdot a_{kl} + s^2$All ▷ diag elms
  $a_{ll} = s^2$Akk + $2cs \cdot a_{kl} + c^2$All
  $a_{kl} = a_{lk} = 0$
  for i=1 to N **do** ▷ compute $\mathbf{S}^T\mathbf{AS}$
    if $k \neq i$ and $l \neq i$ **then**
      Aik = a_{ik}, Ail = a_{il}
      $a_{ik} = c \cdot a_{ik} - s \cdot a_{il}$
      $a_{ki} = a_{ik}$
      $a_{il} = c \cdot$ Ail $+ s \cdot$ Aik
      $a_{li} = a_{il}$
    end if
    Uik = u_{ik}, Uil = u_{il}
    $u_{ik} = c \cdot$ Uik $- s \cdot$ Uil
    $u_{il} = c \cdot$ Uil $+ s \cdot$ Uik
  end for
end while

For a more lengthy discussion of the convergence of the this method, see appendix section C.

3.2 Buckling beam

For the problem of the buckling beam, we need to solve the differential equation

$$
-\frac{\mathrm{d}^2u(\xi)}{\mathrm{d}\xi^2} = \lambda u(\xi)
$$

with Dirichlet boundary conditions $u(0) = u(1) = 0$. This is derived in appendix A. Here, ξ is a dimensionless length going from 0 to 1, where 0 is the startpoint of the beam and 1 is the endpoint. λ is a constant, and is dependent on the characteristics of the beam, and the force applied to it.
This differential equation can be approximated using the most common approximation of the second

[6] M. Hjort-Jensen. *Computational physics, lecture notes.* 7.4 Jacobi's method p. 215-218: Department of Physics, University of Oslo, 2015.

derivative

$$\frac{\mathrm{d}^2u(\xi)}{\mathrm{d}\xi^2} \approx \frac{u(\xi+h)+u(\xi-h)-2u(\xi)}{h^2}$$

which holds for small stepsizes h. Discretizing the variable $\xi \in [0,1]$ as $\xi_i = ih$, where $i = 0,1,2,...,N+1$ and $h = 1/(N+1)$, and defining $u(\xi_i) = u_i$, we get

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

which can be written as the eigenvalue equation

$$\mathbf{Au} = \lambda\mathbf{u}$$

where

$$\mathbf{A} = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & 0 & \dots & 0 \\ -1 & 2 & -1 & \dots & 0 \\ 0 & -1 & 2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 2 \end{pmatrix}$$

Thus we have reduce the problem to something we can solve with Jacobi's method.
We will solve the equation for various number of integration points $N \in [1,200]$. We will plot the solution for the lowest eigenvalues, and benchmark the time it takes for diagonalizing the matrix $\mathbf{S}$ against the time the armadillo library takes. We will also plot the number of iterations required to diagonalize the matrix, as a function of its size.

3.3 Quantum harmonic oscillator for one electron

We will study a harmonic oscillator potential with one and two electrons trapped. At first, we will look at the simplest case; a single trapped electron, with a minimal angular momentum, given by the quantum number $l = 0$. The Schrödinger equation for this system leaves us with the following differential equation for the substitution $u(\rho) = \rho R(\rho)$ for the radial part of the wave function of the electron

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

with Dirichlet boundary conditions $u(0) = u(\infty) = 0$. Here ρ is a dimensionless length given in units of $\alpha = \sqrt{\frac{\hbar}{m_e\omega}}$, where m_e is the electron mass and ω indicates the strength of the oscillator potential. λ is a dimensionless quantity giving the energy levels of the system in units of $\frac{1}{2}\hbar\omega$.
Discretizing the equation like in the case of the buckling beam, we can get the following matrix eigenvalue problem

$$\mathbf{Au} = \lambda\mathbf{u}$$

where

$$\mathbf{A} = \frac{1}{h^2}\begin{pmatrix} 2+h^2\rho_0^2 & -1 & 0 & \dots & 0 \\ -1 & 2+h^2\rho_1^2 & -1 & \dots & 0 \\ 0 & -1 & 2+h^2\rho_2^2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 2+h^2\rho_{\max}^2 \end{pmatrix}$$

This does however pose a challenge when discretizing ρ, as it goes to infinity. We will therefore choose a $\rho_{\max}$ that is sufficient to create accurate solutions, yet not so large that we get unstable solutions. We will solve the system using Jacobi's method for and study the behaviour with of the eigenvalues for $\rho_{\max} \in [1,15]$. We will also increase the stepsize from $N = 50$ to see what precision we can expect compared to the analytical values for the eigenvalues. The analytical eigenvalues for this system, given our dimensionless variables are $\lambda = 3,7,11,...$[7]

3.4 Quantum harmonic oscillator for two electrons

We will do a similar study of the same system, but with two electrons trapped within the harmonic oscillator potential. Again we will only look at the simplest case where $l = 0$, and we also simplify the problem further by looking only at radial part of the relative wavefunction between the electrons. Substituting $u_r(\rho) = \rho R_r(\rho)$ for the relative radial part of the complete wavefunction of the system, the Schrödinger

[7] Sept. 27, 2020. URL: https://en.wikipedia.org/wiki/Quantum_harmonic_oscillator#Example:_3D_isotropic_harmonic_oscillator.

equation reads

$$-\frac{\mathrm{d}^2 u_r(\rho)}{\mathrm{d}\rho^2} + \omega_r^2 \rho_2 u_r(\rho) + \frac{u_r(\rho)}{\rho} = \lambda u_r(\rho)$$

with Dirichlet boundary conditions $u_r(0) = U_r(\infty) = 0$. This is derived in appendix B. Here λ represents the relative energy between the electrons in units of $\frac{\hbar^2}{m_e \beta^2 e^4}$, where m_e is the mass of an electron and $\beta e^2 = 1.44\,\mathrm{eV\,nm}$, and ω_r is a parameter for the harmonic oscillator potential strength. ρ is a dimensionless length given in units of $\frac{\hbar^2}{m_e \beta e^2}$.
This can again be discretized the same way as before, with $\rho_{\max}$ serving as the cutoff point before infinity, and using the same approximation for the second derivative as earlier. Defining discretized quantities

$$\begin{aligned} \rho_i &= ih \\ u_i &= u(\rho_i) \\ V_i &= \omega_r^2 \rho_i^2 + \frac{1}{\rho_i} \end{aligned}$$

leaving us with the following eigenvalue matrix equation

$$\mathbf{A}\mathbf{u} = \lambda \mathbf{u}$$

where

$$\mathbf{A} = \frac{1}{h^2} \begin{pmatrix} 2 + h^2 V_0 & -1 & 0 & \dots & 0 \\ -1 & 2 + h^2 V_1 & -1 & \dots & 0 \\ 0 & -1 & 2 + h^2 V_2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 2 + h^2 V_{\max} \end{pmatrix}$$

We will explore solutions to this equation, using our Jacobi rotation algorithm, for various values for $\omega_r \in [0.01, 0.5, 1, 5]$ and find suitable values for $\rho_{\max}$ and N with which the solutions are stable. More specifically we will see how the eigenvector for the lowest eigenvalues (the ground-state) changes with the harmonic oscillator potential strength ω_r.[8]

[8] When comparing the eigenvalues for different sets of parameters, we must make sure to normalize the eigenvectors, as they represent wavefunctions that are normalized slightly differently. This is described in detail in appendix E.

4 Results & discussion

4.1 Buckling beam

In figure 1 we can see the two first eigenvectors of the buckling beam, sorted after the magnitude of the eigenvalue with lowest being first. The numerical solutions corresponds nicely with the normalized analytical solutions. We can see that the eigenvector solutions are standing waves with the endpoints fixed at $\xi = 0$ and $\xi = 1$. The first solution has zero nodes and the second one node. This pattern continues for higher eigenvectors where the n'th eigenvector contains n-1 nodes.

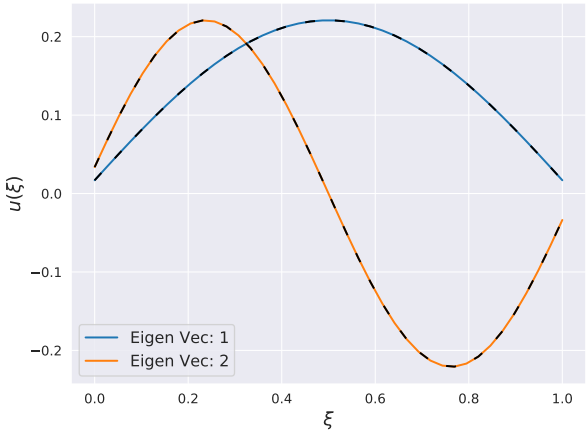

Figure 1: The two first eigenvectors of the buckling beam. $u(\xi)$ represents the vertical deviation from the stable beam and ξ is the scaled length of the beam. Colored lines is the numerical solution, while the dashed lines are the normalised analytical solution.

From figure 2 we can see that the the number of iterations (rotations) before reaching a desired tolerance seems to be growing with n^2 where n is the matrix size, as a $\log_{10}$ based linear fit gave us a slope of 2.06 ± 0.01.
In figure 3 we see that for very small matrices ($N = 2$ to $N = 12$) our algorithm preforms better than the armadillo function. After $N = 12$ we are quickly outmatched for matrices over $N = 50$ the armadillo

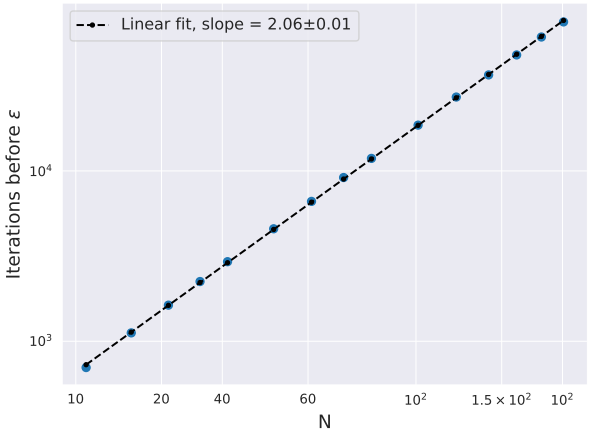

Figure 2: Plot of number of iterations to reach a set tolerance ϵ against matrix sizes. Here we have chosen $\epsilon = 10^{-10}$ and testing for matrix sizes $N = 20, 25, 30, 35, 40, 50, 60, 70, 80, 100, 120, 140, 160, 180$ and 200. The test was preformed using the code from the BucklingBeam program.

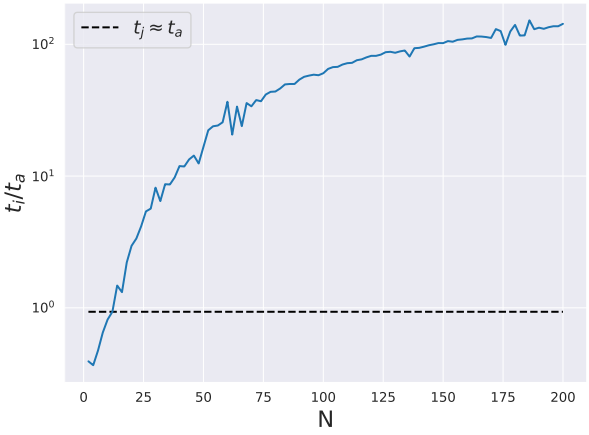

Figure 3: The relative time t_j/t_a, where t_j is the time used for our implementation of the Jacobi method and t_a is the time used by armadillo's **eig_sym** function. Plotted against matrix sizes $N \in [2, 100]$ with increments of 2. Black line represents approximately where armadillo's function surpasses our Jacobi method, at $N = 12$. The timing was preformed using the code from the BucklingBeam program.

function performs faster by an order of 10. This is probably due to armadillo recognizing that the matrix is a tridiagonal Toeplitz matrix and using faster specialized methods such as Householders method.[9]

4.2 Quantum harmonic oscillator with one electron

Solving our differential equation for the harmonic oscillator system with a single electron for various values of $\rho_{\max}$ and $N = 50$ integration points we get the results tabulated in table 1.

Table 1: Table of the first four energy eigenvalues of the harmonic oscillator system with a single electron. All the values use 50 integration points, but changes the parameter $\rho_{\max}$ from 1 to 15.

$[N, \rho_{\max}]$	λ_1	λ_2	λ_3	λ_4
50, 1	10.15	39.75	88.90	157.4
50, 3	3.011	7.321	12.92	20.52
50, 5	2.997	6.985	10.96	14.94
50, 7	2.994	6.970	10.93	14.87
50, 9	2.990	6.951	10.88	14.78
50, 11	2.985	6.927	10.82	14.66
50, 13	2.980	6.897	10.75	14.53
50, 15	2.973	6.862	10.66	14.36

The relationship between the numerical error and the two variables N and $\rho_{\max}$ is presented in figure 4. As we can see there seems to be some "sweet-spot" around $\rho_{\max} = 5$ which is represented by the "valley" in the plot. The error also seems to decrease with the matrix size N. On the other hand this valley is not

[9] Sept. 28, 2020. URL: https://en.wikipedia.org/wiki/Householder_transformation.

perfectly straight so there does not seem to be some perfect $\rho_{\max}$ for every N.

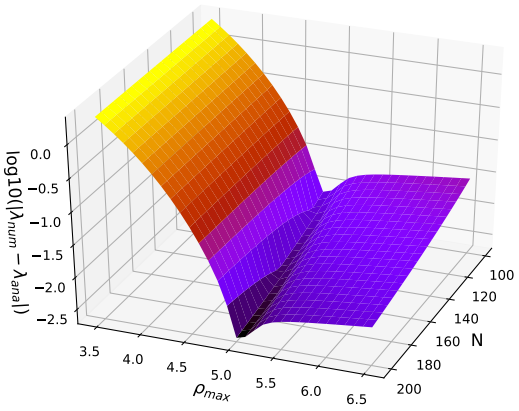

Figure 4: Showing the $\log_{10}$ of the difference between the four lowest eigenvalues as a function of matrix size N and ρ_{max}. The error chosen was the maximum absolute difference between the analytical and numerical eigenvalues for the four lowest eigenvalues.

Tabulating some of the results for $\rho_{\max}$ values around 5 in table 2.

Table 2: A further investigation of the first four eigenvalues for the harmonic oscillator system with a single electron, using $\rho_{\max}$ around 5 and $N = 50$ integration points.

$[N, \rho_{\max}]$	λ_1	λ_2	λ_3	λ_4
50, 3.5	2.999	7.039	11.49	17.09
50, 4	2.998	6.994	11.05	15.53
50, 4.5	2.998	6.988	10.98	15.03
50, 5	2.997	6.985	10.96	14.94
50, 5.5	2.996	6.982	10.96	14.92
50, 6	2.996	6.978	10.95	14.90
50, 6.5	2.995	6.975	10.94	14.88

$\rho_{\max} = 4.5$ seems to be the best compromise for accuracy for the 4 lowest eigenvalues. We can see that the lower values for $\rho_{\max}$ give better results for the smaller eigenvalues, while the larger values $\rho_{\max}$ are more accurate for the higher ones. This makes sense, as the higher energy eigenvalues should create states where the electron is farther from the center, and as such, the point at which it is reasonable to cut off the wavefunction is pushed farther out.
Exploring the accuracy as we increase the number of integration points N, we get the values tabulated in table 3.

Table 3: An exploration of the accuracy of the numerical values for the first four eigenvalues of the harmonic oscillator potential with a single electron as the number of integration points N is increased.

$[N, \rho_{\max}]$	λ_1	λ_2	λ_3	λ_4
50, 4.5	2.9976	6.9879	10.976	15.029
100, 4.5	2.9994	6.9970	10.998	15.073
150, 4.5	2.9997	6.9987	11.002	15.081
200, 4.5	2.9998	6.9993	11.004	15.084
250, 4.5	2.9999	6.9996	11.005	15.085
300, 4.5	2.9999	6.9998	11.005	15.086
350, 4.5	2.9999	6.9999	11.005	15.086
400, 4.5	3.0000	6.9999	11.005	15.087

We can see here that we needed 400 integration points to have four correct leading digits for the first eigenvalue. However, with our current $\rho_{\max} = 4.5$ we can see that the higher eigenvalues actually diverge from the analytical values as N is increased. This is probably because an insignificant portion of the wavefunction is cut off, and as we increase the number of integration points, our numerical "solution" converges on a wrong eigenfunction for the higher eigenvalues.

4.3 Quantum harmonic oscillator with two electrons

For the harmonic oscillator system with two electrons, the numerically solved groundstates are plotted in figure 5 using $N = 150$ and varying $\omega_r = 0.01, 0.5, 1, 5$.

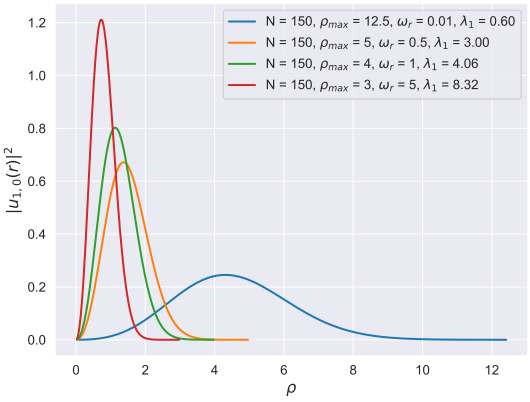

Figure 5: The probability densities of the relative radial part of the two electrons in a harmonic oscillator system. It is run for dimensionless harmonic oscillator frequencies $\omega_r = 0.01, 0.5, 1, 5$. The potential is described in detail in appendix B.

When choosing appropriate $\rho_{\max}$ for these values, we took wisdom from the discoveries for the single electron system, making sure that the wavefunction for the ground state of our solution flattened out completely for the given $\rho_{\max}$. The resulting ground states seem reasonable; the smaller the ω_r value, the farther away from each other we can expect to find the electrons of the ground state. As the Coulomb force between the electrons is repulsive, it would only be natural for the electrons to free themselves of each other as the harmonic potential dissipates. As the harmonic potential grows stronger, we can see that it manages to force the electrons closer together. This results in a greater potential energy from the Coulomb potential, and results in higher energy levels for the ground state, as can be seen from the eigenvalues.

5 Concluding remarks

We found that even though the Jacobi method is intuitive and relies on relatively simple linear algebra, it does not scale well with larger matrices; as the number of rotations before reaching a set tolerance ϵ grows approximately as N^2. When comparing with armadillo's eigenvalue solver, even for relatively small matrices, we can not come close with the more modern methods. As our applications only dealt with tridiagonal Toeplitz matrices, a specialized method such as the Householder method would increase performance drastically.

For one electron in the harmonic oscillator we explored how different choices of $\rho_{\max}$ and N affected the numerical error for the lowest eigenvalues. With $\rho_{\max} = 4.5$ and $N = 400$ we managed to replicate the lowest eigenvalue with a precision of four leading decimals, the eigenvalues up to 4 were only reproduced with three. In the two electron case we found that increasing the potential frequency brings the two electrons closer together as the harmonic oscillator potential competes with the Coulomb force, and yields greater energy levels for the ground state. It is clear that it is entirely feasible to attack very relevant differential equations of interesting physical system with simpler, more intuitive methods like Jacobi's. However, to diver deeper into the equation and reproduce result with great accuracy and efficiency requires some perhaps more convoluted, modern approaches.

A Buckling beam

A force F is applied to parallel to a beam fastened in two endpoints, the vertical displacement of the beam, $u(x)$, is given by the following differential equation:

$$\gamma \frac{\mathrm{d}^2 u(x)}{\mathrm{d}x^2} = -Fu(x)$$

where γ is a material constant describing how well the beam holds up to deformation. We let the beam have a length L such that $x \in [0, L]$. As the endpoints are fixed, we get Dirichlet boundary conditions $x(0) = x(L) = 0$. Defining a new dimensionless length $\xi = x/L$, we can rewrite the differential equation

$$\frac{\mathrm{d}^2 u(\xi)}{\mathrm{d}\xi^2} = -\frac{FL^2}{\gamma} u(\xi) \equiv -\lambda u(\xi)$$

Where we define $\lambda = FL^2/\gamma$.

B Quantum harmonic oscillator

B.1 Single electron

For the single electron in a spherical harmonic oscillator potential $V(r) = \frac{1}{2}m\omega^2 r^2$ the Hamiltonian for the radial part of the state, $R(r)$, reads

$$\hat{H} = -\frac{\hbar^2}{2m}\left(\frac{1}{r^2}\frac{\mathrm{d}}{\mathrm{d}r}\Big(r^2\frac{\mathrm{d}}{\mathrm{d}r}\Big) - \frac{1}{r^2}l(l+1)\right) + \frac{1}{2}m\omega^2 r^2$$

The operator can be simplified by substituting $u(r) = rR(r)$ such that

$$\begin{aligned}\hat{H}R(r) &= -\frac{\hbar^2}{2m}\left(\frac{1}{r^2}\frac{\mathrm{d}}{\mathrm{d}r}\Big(r^2\frac{\mathrm{d}}{\mathrm{d}r}\Big) - \frac{l(l+1)}{r^2}\right)\frac{u}{r} + V\frac{u}{r} \\ &= -\frac{\hbar^2}{2m}\left(\frac{1}{r}\frac{\mathrm{d}^2u}{\mathrm{d}r^2} - \frac{l(l+1)}{r^3}u\right) + \frac{m\omega^2}{2}r \\ &\Downarrow \\ \hat{H}u(r) &= -\frac{\hbar^2}{2m}\left(\frac{\mathrm{d}^2}{\mathrm{d}r^2} - \frac{l(l+1)}{r^2}\right)u(r) + \frac{m\omega^2}{2}r^2u(r)\end{aligned}$$

Schrõdinger's eigenvalue equation for this operator then reads

$$-\frac{\hbar^2}{2m}\left(\frac{\mathrm{d}^2}{\mathrm{d}r^2} - \frac{l(l+1)}{r^2}\right)u(r) + \frac{m\omega^2}{2}r^2u(r) = E_{nl}u(r)$$

We can simplify this further by introducing the dimensionless length $\rho = r/\alpha$ and get

$$\begin{aligned}\Big(\frac{1}{\alpha^2}\frac{\mathrm{d}^2}{\mathrm{d}\rho^2} - \frac{l(l+1)}{\alpha^2\rho^2} - \frac{m^2\omega^2\alpha^2}{\hbar^2}\rho^2\Big)u(\rho) &= -\frac{2m}{\hbar^2}E_{nl}u(\rho) \\ \Big(\frac{\mathrm{d}^2}{\mathrm{d}\rho^2} - \frac{l(l+1)}{\rho^2} - \frac{m^2\omega^2\alpha^4}{\hbar^2}\Big)\rho^2u(\rho) &= -\frac{2m\alpha^2}{\hbar^2}E_{nl}u(\rho)\end{aligned}$$

Setting $\alpha = \sqrt{\frac{\hbar}{m\omega}}$ and introducing $\lambda = \frac{2}{\hbar\omega}E_{nl}$ we get

$$-\frac{\mathrm{d}^2u(\rho)}{\mathrm{d}\rho^2} + \Big(\frac{l(l+1)}{\rho^2} + \rho^2\Big)u(\rho) = \lambda u(\rho)$$

We will only be looking at the case where $l = 0$ and we get our final simplification

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

with the parameter $\rho \in [0, \infty)$. For the solution to be normalizable, we must also impose boundary conditions on the solutions such that $u(0) = u(\infty) = 0$.

B.2 Two electrons

For two electrons in the spherical harmonic oscillator, the solution becomes a bit more complex. Looking at the combined radial state for the two electrons $u(r_1, r_2)$, where u is defined analogously to the one-electron case, we get a combined Schrödinger equation simply by combining the individual Hamiltonians

$$\left(\sum_{i=1}^{2} -\frac{\hbar^2}{2m}\frac{\mathrm{d}^2}{\mathrm{d}r_i^2} + \frac{1}{2}m\omega^2 r_i^2\right)u(r_1, r_2) = Eu(r_1, r_2)$$

where we are assuming $l_1 = l_2 = 0$ and no Coulomb-repulsion for simplicity. We will then rewrite this equation using more natural quantities like the relative coordinate $\mathbf{r} = \mathbf{r}_1 - \mathbf{r}_2$ and the coordinate for the center of mass $\mathbf{R} = \frac{1}{2}(\mathbf{r}_1 + \mathbf{r}_2)$. Using this we the Schödinger equation reads

$$\left(-\frac{\hbar^2}{m}\Big(\frac{\mathrm{d}^2}{\mathrm{d}r^2} + \frac{\mathrm{d}^2}{\mathrm{d}R^2}\Big) + m\omega^2\Big(\frac{1}{4}r^2 + R^2\Big)\right)u(r, R) = Eu(r, R)$$

We will use the separation of variables on this equation, such the $u(r) \equiv u_r(r)u_R(R)$, and equivalently breaking up the energy into a relative part and a mass center-part respectively as $E = E_r + E_R$.
Now we can add the Coulomb repulsion between the particles as

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

where $\beta e^2 = 1.44\,\mathrm{eV\,nm}$.
Now we will explore the relative part of our Schrödinger equation in greater detail, adding the Coulomb term to the Hamiltonian such that we get

$$\left(-\frac{\hbar^2}{m}\frac{\mathrm{d}^2}{\mathrm{d}r^2} + \frac{1}{4}m\omega^2 r^2 + \frac{\beta e^2}{r}\right)u_r(r) = E_r u_r(r)$$

Again, we will introduce the dimensionless length variable $\rho = r/\alpha$, getting

$$\left(-\frac{\mathrm{d}^2}{\mathrm{d}\rho^2} + \frac{m^2\omega^2}{4\hbar^2}\alpha^4\rho^2 + \frac{m\beta e^2\alpha}{\rho\hbar^2}\right)u_r(\rho) = \frac{m\alpha^2}{\hbar^2}E_r u_r(\rho)$$

This time, we will choose $\alpha = \frac{\hbar^2}{m\beta e^2}$ such that the coefficient of the $1/\rho$ term becomes 1. Furthermore, we will define a dimensionless frequency

$$\omega_r = \frac{m\omega}{2\hbar}\alpha^2$$

Finally, we define

$$\lambda = \frac{\hbar^2}{m\beta^2 e^4}E_r$$

Our final two-electron Schrödinger equation thus reads

$$\left(-\frac{\mathrm{d}^2}{\mathrm{d}\rho^2} + \omega_r^2\rho^2 + \frac{1}{\rho}\right)u_r(\rho) = \lambda u_r(\rho)$$

Similarly to the single electron case, for the wavefunction to be normalizable, we require boundary conditions $u(0) = u(\infty) = 0$.

C Convergence

Convergence of the Jacobi method is reliant on the Frobenius norm.[10] The Frobenius norm is defined as

$$||\mathbf{A}||_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}|a_{ij}|^2}$$

When the matrix $\mathbf{A}$ undergoes an orthogonal transformation the Frobenius norm is always conserved. When transforming $\mathbf{A}$ to $\mathbf{B}$ as in equation 1 the Frobenius norm of the two matrices will be equal, that is $||\mathbf{A}||_F = ||\mathbf{B}||_F$. The off-diagonal elements can then be written as

$$\mathrm{off}(\mathbf{A})^2 = ||\mathbf{A}||_F - \sum_{i=1}^{n} a_{ii}^2$$

$$\mathrm{off}(\mathbf{B})^2 = ||\mathbf{B}||_F - \sum_{i=1}^{n} b_{ii}^2$$

[10] Hjort-Jensen, *Computational physics, lecture notes*.

Writing out the matrix multiplication for the 2×2 case one can show that the off-diagonal of $\mathbf{B}$ can be expressed as

$$\mathrm{off}(\mathbf{B})^2 = \mathrm{off}(\mathbf{A})^2 - 2a_{kl}^2$$

Thus systematically choosing the element a_{kl} with the largest absolute value and setting this to zero ensures that after enough iterations the transformed matrix will converge to a diagonal matrix.

D Analytical solutions to a tridiagonal Toeplitz matrix

For a tridiagonal Toeplitz matrix where all the diagonal elements are similar, and the equivalently for the upper and lower diagonals, there are analytical expressions for the eigenvalues and eigenvectors of the matrix.[11] Letting the diagonal elements be d and the upper and lower diagonal elements be a, the eigenvalues λ_i are given as

$$\lambda_i = d + 2a\cos(i\pi h)$$

where $h = 1/(N+1)$ for a $N \times N$ matrix. The eigenvectors $\mathbf{v}_i$ given as

$$\mathbf{v}_i = \begin{pmatrix} \sin(i\pi h) \\ \sin(2\pi h) \\ \vdots \\ \sin(N\pi h) \end{pmatrix}$$

It should be noted that these eigenvectors are orthogonal, but they are not normalized to 1.

E On the normalization of quantum wavefunctions

When we discretize wavefunctions to solve them as a matrix eigenvalue problem, we create normalized

[11] T. Lyche. *Lecture Notes for Mat-inf 4130*. 2017, p. 49. URL: https://www.uio.no/studier/emner/matnat/math/MAT-INF4130/h17/book2017.pdf.

(hopefully) eigenvector solutions, $\tilde{\mathbf{u}}^n$, that fulfill the orthonormality requirement

$$\sum_{i=1}^{N} \tilde{u}_i^n \cdot \tilde{u}_i^{n'} = \delta_{n,n'}$$

where $\delta_{n,n'}$ is the Kronecker delta. However, the normalization criteria for eigenfunctions in a Hilbert space[12] is

$$\int_{\infty} (\psi_{n'}(r))^* \psi_n(r) \, \mathrm{d}r = \delta_{n,n'}$$

In our case we can rewrite this critera using our approximation for this integral product of the substitution $u(\rho)$ for the radial part of our 3D wavefunction in spherical coordinates

$$\int_0^{\infty} u_n(\rho) u_{n'}(\rho) \, \mathrm{d}\rho \approx \sum_{i=1}^{N} u_i^n \cdot u_i^{n'} \cdot h$$

Now letting the discretized eigenvectors have a normalization factor α such that

$$\mathbf{u}_n = \alpha \tilde{\mathbf{u}}_n$$

we get

$$\sum_{i=1}^{N} h u_i^n \cdot u_i^{n'} = \alpha^2 \frac{\rho_{\max}}{N+1} \delta_{n,n'} \equiv \delta_{n,n'}$$

And as such, we must normalize the eigenvectors we compute using the Jacobi rotation method by the factor

$$\alpha = \sqrt{\frac{N+1}{\rho_{\max}}}$$

[12] Such as those occupied by wavefunctions in quantum physics.

References

Jacobi, C. G. J. "Über ein leichtes Verfahren die in der Theorie der Säcularstöorungen vorkommenden Gleichungen numerisch aufzulösen." In: *de Gruyter* 30.4 (1864), p. 51.

Hjort-Jensen, M. *Computational physics, lecture notes*. 7.4 Jacobi's method p. 215-218: Department of Physics, University of Oslo, 2015.

Lyche, T. *Lecture Notes for Mat-inf 4130*. 2017, p. 49. URL: https://www.uio.no/studier/emner/matnat/math/MAT-INF4130/h17/book2017.pdf.

Sept. 29, 2020. URL: https://en.wikipedia.org/wiki/Quantum_dot.

Sept. 19, 2020. URL: http://pi.math.cornell.edu/~jerison/math2940/real-eigenvalues.pdf.

Sept. 26, 2020. URL: https://en.wikipedia.org/wiki/Jacobi_eigenvalue_algorithm.

Sept. 20, 2020. URL: https://en.wikipedia.org/wiki/Givens_rotation.

Sept. 27, 2020. URL: https://en.wikipedia.org/wiki/Quantum_harmonic_oscillator#Example:_3D_isotropic_harmonic_oscillator.

Sept. 28, 2020. URL: https://en.wikipedia.org/wiki/Householder_transformation.