FYS3150 - PROJECT 2 - AUTUMN 2015

Mari Dahl Eggen

October 12, 2015

Abstract

In this project we will use Jacobi's method to retrieve the eigenvalues and eigenvectors from a matrix eigenvalue problem. We will also analyze how effective the algorithm for Jacobi's method is, compared to the Armadillo eigenvalue solver. Unit tests is explained and added to the Jacobi algorithm, such that the code can be changed with less risk.

1 Introduction

In quantum mechanics you can use the matrix of the Hamiltonian to retrieve the energy of a system, that can possibly be measured. The observable energies are equivalent to the matrix eigenvalues, and therefore, we can find these by use of Jacobi's method. First of all one has to write the Shrödinger equation on a matrix form, then we implement an algorithm that uses Jacobi's method to retrieve the eigenvalues and eigenvectors of the matrix, and then we can find a energy probability distribution for the system in some given state. Armadillo has a built-in-function that computes the eigenvalues and eigenvectors of a matrix by one simple command. We will test the effectiveness of this built-in-function, and compare it with our Jacobi algorithm. The algorithm is a bit big, and therefore we also implement some unit tests, so that we can develop the algorithm without the risk of destruction.

2 Theory

2.1 Similarity transformations

In the following we assume that the matrix $\mathbf{A}$ is a real and symmetric matrix, $\mathbf{A} \in \mathbb{R}^{n\times n}$. Then, there exists a set of real orthogonal matrices $\mathbf{S_i}$, such that

$$\mathbf{S_n^T}...\mathbf{S_1^T}\mathbf{A}\mathbf{S_1}...\mathbf{S_n} = \mathbf{D}, \tag{1}$$

where $\mathbf{S_i^T}\mathbf{S_i} = \mathbf{S_i}\mathbf{S_i^T} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix, and $\mathbf{D}$ is given by

$$\mathbf{D} = \begin{bmatrix} \lambda_1 & 0 & 0 & \dots & 0 \\ 0 & \lambda_2 & 0 & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & \dots & \dots & \dots & \lambda_n \end{bmatrix},$$

where $\lambda_1, \lambda_2, ..., \lambda_n$ is the eigenvalues of $\mathbf{A}$. If we look at a single similarity transformation, where $\mathbf{B} = \mathbf{S^T A S}$ is a similarity transform of $\mathbf{A}$, we can show that $\mathbf{B}$ and $\mathbf{A}$ have the same eigenvalues, but in general not the same eigenvectors. We starts out with the eigenvalue equation of $\mathbf{A}$.

$$\mathbf{Ax} = \lambda\mathbf{x} \quad \Rightarrow \quad \mathbf{S^T A I x} = \lambda\mathbf{S^T x}$$

$$\Rightarrow \quad (\mathbf{S^T A S})(\mathbf{S^T x}) = \lambda(\mathbf{S^T x}) \quad \Rightarrow \quad \mathbf{B}(\mathbf{S^T x}) = \lambda(\mathbf{S^T x})$$

We then see that $\mathbf{B}$ and $\mathbf{A}$ have the same eigenvaules, but when $\mathbf{A}$ has the eigenvector $\mathbf{x}$, $\mathbf{B}$ has the eigenvector $\mathbf{S^T x}$. We then knows for sure that the matrix $\mathbf{D}$ in Equation (1) gives the eigenvalues of $\mathbf{A}$.

2.2 Jacobi's method

This is a method one can use to obtain the eigenvalue matrix $\mathbf{D}$ in Equation (1). It uses an orthogonal transformation matrix $\mathbf{S} \in \mathbb{R}^{\mathbf{n\times n}}$ with elements

$$s_{kk} = s_{ll} = \cos\theta, \quad s_{kl} = -s_{lk} = -\sin\theta, \quad s_{ii} = 1 \quad \text{for} \quad i \neq k, l,$$

and all the other elements are equal to zero. $\mathbf{S}$ is unitary, and performs a plane rotation in the Euclidean n-dimensional space by an angle θ. By performing a similarity transformation $\mathbf{B} = \mathbf{S^T A S}$ the elements in $\mathbf{B}$ is given by

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

In our case the matrix $\mathbf{A}$ is symmetric around the diagonal, that is, the elements $(\mathbf{A})_{i,j} = (\mathbf{A^T})_{i,j}$. For the similarity transform we have that

$$(\mathbf{B^T})_{i,j} = ((\mathbf{S^T A S})^T)_{i,j} = (\mathbf{S^T A^T S})_{i,j},$$

and we see that also $\mathbf{B}$ is symmetric around its diagonal. Then we have that $b_{ki} = b_{ik}$, $b_{li} = b_{il}$ and $b_{lk} = b_{kl}$.

Now we want to choose the angle θ such that the elements $b_{kl} = b_{lk} = 0$, that is

$$(a_{kk} - a_{ll})\cos\theta\sin\theta + a_{kl}(\cos^2\theta - \sin^2\theta) \quad \Rightarrow \quad (a_{kk} - a_{ll})\cos\theta\sin\theta = a_{kl}(\cos^2\theta - \sin^2\theta)$$

$$\Rightarrow \quad \frac{a_{ll} - a_{kk}}{2}\sin(2\theta) = a_{kl}\cos(2\theta) \quad \Rightarrow \quad \frac{a_{ll} - a_{kk}}{2a_{kl}} \cdot \frac{\sin 2\theta}{\cos(2\theta)} = 1$$

$$\Rightarrow \quad \frac{a_{ll} - a_{kk}}{2a_{kl}} \cdot \tan(2\theta) = \tau\left(\frac{2\tan\theta}{1 - \tan^2\theta}\right) = 1$$

$$\Rightarrow \quad 2\tau\tan\theta = 1 - tan^2\theta \quad \Rightarrow \quad tan^2\theta + 2\tau\tan\theta - 1 = 0$$

$$t^2 + 2\tau t - 1 = 0, \tag{2}$$

where $t \equiv tan\theta$. In the following we will also use the definitions $c \equiv \cos\theta$ and $s \equiv \sin\theta$.

With the second degree formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a},$$

for a general second degree equation $ax^2 + bx + c = 0$, one can show that Equation (2) has the solutions

$$t = -\tau \pm \sqrt{1 + \tau^2}. \tag{3}$$

The values of c and s can be obtained from the expressions

$$c = \frac{1}{\sqrt{1 + t^2}} \quad \text{and} \quad s = tc. \tag{4}$$

With the formulas we now have obtained, it is easy to make an algorithm that

performs Jacobi's method. We have to perform this method a number of iterations, until we obtain the similarity transformation $\mathbf{B} = \mathbf{D}$, that is, when all the elements $b_{i,j}$, for $i \neq j$ is equal to zero.

2.3 Radial part of Schrödinger's equation for one electron in a harmonic oscillator potential

The equation for one electron in a harmonic oscillator potential in three dimensions reads

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

where the harmonic oscillator potential is $V(r) = \frac{1}{2}kr^2$, where $k = m\omega^2$, with ω as the oscillator frequency. The energies of the harmonic oscillator is given by

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

with $n = 0, 1, 2, \ldots$ and $l = 0, 1, 2, \ldots$, where n and l are respectively the energy quantum number and the orbital momentum quantum number.

The radial limits are $r \in [0, \infty)$. We substitute $R(r) = \frac{1}{r}u(r)$, and obtain

$$-\frac{\hbar^2}{2m}\frac{d^2}{dr^2}u(r) + \left(V(r) + \frac{l(l+1)}{r^2}\frac{\hbar^2}{2m}\right)u(r) = Eu(r),$$

where the boundary conditions are $u(0) = 0$ and $u(\infty) = 0$. We the introduce a dimensionless variable $\rho = \frac{r}{\alpha}$, where α is a constant with dimension length, and get

$$-\frac{\hbar^2}{2m\alpha^2}\frac{d^2}{d\rho^2}u(\rho) + \left(V(\rho) + \frac{l(l+1)}{\rho^2}\frac{\hbar^2}{2m\alpha^2}\right)u(\rho) = Eu(\rho).$$

In this project we will just consider the case where $l = 0$, so we use that, and the fact that $V(\rho) = \frac{1}{2}k\alpha^2\rho^2$, to obtain

$$-\frac{\hbar^2}{2m\alpha^2}\frac{d^2}{d\rho^2}u(\rho) + \frac{k}{2}\alpha^2\rho^2u(\rho) = Eu(\rho). \tag{5}$$

We now multiply the equation with $\frac{2m\alpha^2}{\hbar^2}$ and gets

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

We fix the dimensionless constant α such that

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

and defines

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

so that the Schrödinger equation reads

$$-\frac{d^2}{d\rho^2}u(\rho) + \rho^2u(\rho) = \lambda u(\rho). \tag{6}$$

We can then use the standard expression for the second derivative of a function u to solve Equation (6), that is

$$u'' = \frac{u(\rho + h) - 2u(\rho) + u(\rho - h)}{h^2} + O(h^2), \tag{7}$$

where h is the step length, and is defines as

$$h = \frac{\rho_{\max} - \rho_{\min}}{n_{\text{step}}}.$$

n_{step} is a given number of steps, and in this case we have that $\rho_{min} = 0$ and $\rho_{max} = \infty$. In the algorithm we can not use $\rho_{max} = \infty$, an optimal value has to be decided during the numerical calculations. An arbitrary value of ρ is then defined as

$$\rho_i = \rho_{\min} + ih \quad \text{for} \quad i = 0, 1, 2, \dots, n_{\text{step}}$$

We can then rewrite the Schrödinger equation as

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

where $V_i = \rho_i^2$ is the harmonic oscillator potential, $u_{i+1} = u(\rho_i + h)$, $u_i = u(\rho_i)$ and $u_{i-1} = u(\rho_i - h)$. If we define

$$d_i = \frac{2}{h^2} + V_i \quad \text{and} \quad e_i = -\frac{1}{h^2},$$

the Schrödinger equation takes the form

$$d_i u_i + e_{i-1} u_{i-1} + e_{i+1} u_{i+1} = \lambda u_i, \tag{8}$$

where u_i is unknown.

Equation (8) can be rewritten as a matrix eigenvalue problem on the form

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

$$\begin{pmatrix} d_1 & e_1 & 0 & 0 & \dots & 0 & 0 \\ e_1 & d_2 & e_2 & 0 & \dots & 0 & 0 \\ 0 & e_2 & d_3 & e_3 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & \dots & d_{n_{\text{step}}-2} & e_{n_{\text{step}}-1} \\ 0 & \dots & \dots & \dots & \dots & e_{n_{\text{step}}-1} & d_{n_{\text{step}}-1} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \dots \\ \dots \\ \dots \\ u_{n_{\text{step}}-1} \end{pmatrix} = \lambda \begin{pmatrix} u_1 \\ u_2 \\ \dots \\ \dots \\ \dots \\ u_{n_{\text{step}}-1} \end{pmatrix}. \quad (9)$$

A is now the matrix of the Hamiltonian, and by use of Jacobi's method explained in the previous section, we can find the energy eigenvalues with its belonging eigenvectors. It is important to add the boundary conditions described above after the matrix equation is solved.

2.4 Radial part of Schrödinger's equation for two electrons in a harmonic oscillator potential, which interacts via a repulsive Coulomb interaction

If we use Equation (5), the Schrödinger equation for two electrons with no repulsive Coulomb interactions, is given by

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

where $E^{(2)}$ means that we deals with a two-electron energy. This equation can be written out as the product of two single-electron wave functions, that means that we have a solution on closed form. We define new coordinates, the relative coordinate $\mathbf{r} = \mathbf{r}_1 - \mathbf{r}_2$, and the center-of-mass coordinate $\mathbf{R} = 1/2(\mathbf{r}_1 + \mathbf{r}_2)$. Then the radial Schrödinger equation reads

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

Bethe ansatz is an ansatz method for finding the exact solutions of certain one-dimensional quantum many-body models, and the equations for r and R can be separated via the ansatz for the wave function, $u(r, R) = \psi(r)\phi(R)$. The energy is given by the sum of the relative energy E_r, and the center-of-mass energy E_R, that is $E^{(2)} = E_r + E_R$. The repulsive Coulomb interaction between two electrons is given by

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

where $\beta e^2 = 1.44$ eVnm. If we add the electron interaction, and use the separation via the ansatz, the r-dependent Schrödinger equation becomes

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

We again introduce the dimensionless variable $\rho = \frac{r}{\alpha}$. If we substitute $\rho\alpha$ for r and multiply with $\frac{m\alpha^2}{\hbar^2}$ on both sides, we get

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

We will now manipulate this equation to make it as similar to Equation (6) as possible. We therefore define

$$\omega_r^2 = \frac{1}{4}\frac{mk}{\hbar^2}\alpha^4 \quad \text{and} \quad \lambda = \frac{m\alpha^2}{\hbar^2}E,$$

and fix the constant α such that

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

We can then rewrite Schrödinger's equation as

$$-\frac{d^2}{d\rho^2}\psi(\rho) + \omega_r^2\rho^2\psi(\rho) + \frac{1}{\rho} = \lambda\psi(\rho). \tag{10}$$

ω_r is in this case a parameter which reflects the strength of the oscillator potential.

Equation (10) can be rewritten as a matrix eigenvalue problem in the same way we did for Equation (6) in the previous section, which results in Equation (9), with

$$d_i = \frac{2}{h^2} + \omega_r^2 \rho_i^2 + \frac{1}{\rho_i} \quad \text{and} \quad e_i = -\frac{1}{h^2}.$$

The matrix **A** is now the matrix of the Hamiltonian that describes the energy states of the two-electron system. By use of Jacobi's method, we can find the energy eigenvalues with its belonging eigenvectors. It is important to add boundary conditions after the matrix equation is solved.

3 Method

3.1 Algorithm for solving the matrix eigenvalue problem for a single electron in a harmonic potential well

The algorithm needed to solve the matrix eigenvalue problem obtained from Equation (6), is straight forward from the formulas described in section 2.2. A few changes is done to make the program run safer and faster, and these changes will be described in this section.

3.1.1 Round-off

In the algorithm t, c and s is implemented such that the elements $b_{kl} = b_{lk}$ in the similarity transformation is equal to zero. In Equation (3) t is expressed as a function of $\tau = \frac{a_{ll}-a_{kk}}{2a_{kl}}$. This expression for t can give round off errors, so we want to write it in another form:

$$t_{1,2} = -\tau \pm \sqrt{1+\tau^2} \cdot \frac{-\tau \mp \sqrt{1+\tau^2}}{-\tau \mp \sqrt{1+\tau^2}} = \frac{\tau^2 \pm \tau\sqrt{1+\tau^2} \mp \tau\sqrt{1+\tau^2} - 1 - \tau^2}{-\tau \mp \sqrt{1+\tau^2}}$$

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

We then have

$$t_1 = \frac{1}{\tau + \sqrt{1+\tau^2}} \quad \text{and} \quad t_2 = -\frac{1}{-\tau + \sqrt{1+\tau^2}},$$

that is a more safe expression when it comes to round-off errors in the calculations.

3.1.2 Efficiency

We want to minimize the the difference between **B** and **A**, because the less the difference is, the smaller the elements in **A** is, and then we need less iterations before we get the matrix **D** from Jacobi's method. Since

$$||\mathbf{B} - \mathbf{A}||_F^2 = 4(1-c) \sum_{i=1, i\neq k,l}^{n} (a_{ik}^2 + a_{il}^2) + \frac{2a_{kl}^2}{c^2},$$

we see that the difference is minimized when c has the biggest value possible. We look at some limits of the expressions $t_{1,2}$.

$$\lim_{\tau \to \infty} t(\tau)_{1,2} \quad \Rightarrow \quad t_1 \to 0 \quad \text{and} \quad t_2 \to -\infty$$

$$\lim_{\tau \to -\infty} t(\tau)_{1,2} \quad \Rightarrow \quad t_1 \to \infty \quad \text{and} \quad t_2 \to 0$$

$$\lim_{\tau \to 0} t(\tau)_{1,2} \quad \Rightarrow \quad t_1 = 1 \quad \text{and} \quad t_2 = -1$$

We introduce restrictions implemented in the algorithm as

```
double tau = (A(l,l) - A(k,k))/(2*A(k,l));

        if ( tau > 0 ) {
            t = 1.0/(tau + sqrt(1.0 + tau*tau));
        }
        else {
            t = -1.0/( -tau + sqrt(1.0 + tau*tau));

        }
```

With this we avoid that t goes to infinity, and from Equation (4) we see that this gives the highest values possible for c. With the restrictions made we have that $t \in [-1, 1)$, which means that we always have $|\theta| \leq \frac{\pi}{4}$

3.2 Algorithm for solving the matrix eigenvalue problem for a two-electron system in a harmonic potential well

In section 2.4 we can see that also this problem can be solved as a matrix eigenvalue problem, and that the only difference from the single-electron case, is that the potential energy in the equations is changed. After the execution of Jacobi's algorithm, the eigenvalues and eigenvectors of the Hamiltonian matrix is retrieved, sorted, and used to graph the probability distribution of the two-particle system energy.

3.3 Unit testing

In this case we chose to code our own unit tests for the Jacobi algorithm. We made four tests, each explained in the following:

1. We made a 2×2-matrix with known eigenvalues, that is run through the Jacobi algorithm. If the numerically calculated eigenvalues is wrong, the program is interrupted.

2. We made two 3×3-matrices that is run through the loop that finds the element in a matrix with the biggest absolute value. In the matrices made these elements is known. One of the matrices has a positive element that has the biggest absolute value, and the other has a negative element that has the biggest absolute value. If the loop do not manage to pick out the known elements of the test matrices, the program is interrupted.

3. In section 2.1 we discussed the real orthogonal matrix $\mathbf{S}$, that is used in Jacobi's method to do a similarity transformation. For every iteration $\mathbf{S}$ is updated, and the updated version of $\mathbf{S}$ is also supposed to be orthogonal, such that the last matrix $\mathbf{S}$ is a matrix consisting of the eigenvectors of the Hamiltonian matrix. Because of this we have made a unit test, that every ten thousand iteration choose two random column vectors in $\mathbf{S}$, and checks if the dot product of them is equal to zero. If this test is fails, the program reads out a warning.

4. The last test in this project is to compare the numerically calculated eigenvalues with exact eigenvalues found from an analytical formula.

4 Results and discussion

4.1 Algorithm for solving the matrix eigenvalue problem for a single electron in a harmonic potential well

The algorithm of Jacobi's method was run several times for different values of the number of steps in the method n_{ntep}, and for different values of the parameter ρ_{max}, to find the combination that gives the most satisfactory results.

$\boldsymbol{n_{step}}$	$\boldsymbol{\rho_{max}}$	$\boldsymbol{\lambda_1}$	$\boldsymbol{\lambda_2}$	$\boldsymbol{\lambda_3}$	**Number of leading digits**
200	2.0	3.54497	11.2538	23.7340	$\lambda_1 : 0$, $\lambda_2 : 0$, $\lambda_3 : 0$
200	4.0	2.99991	7.00319	11.0845	$\lambda_1 : 4$, $\lambda_2 : 3$, $\lambda_3 : 2$
200	5.0	2.99981	6.99905	10.9979	$\lambda_1 : 4$, $\lambda_2 : 3$, $\lambda_3 : 4$
200	8.0	2.99951	6.99755	10.9940	$\lambda_1 : 4$, $\lambda_2 : 3$, $\lambda_3 : 3$
200	10.0	2.99923	6.99617	10.9906	$\lambda_1 : 3$, $\lambda_2 : 3$, $\lambda_3 : 3$

Table 1: The three lowest eigenvalues calculated numerically by Jacobi's method. In the algorithm the number of steps used was 200, and the parameter ρ_{max} was varied, to find out which value of ρ_{max} is the best to use.

$\boldsymbol{n_{step}}$	$\boldsymbol{\rho_{max}}$	$\boldsymbol{\lambda_1}$	$\boldsymbol{\lambda_2}$	$\boldsymbol{\lambda_3}$	**Number of leading digits**
100	5.0	2.99925	6.99625	10.9911	$\lambda_1 : 3$, $\lambda_2 : 3$, $\lambda_3 : 3$
200	5.0	2.99981	6.99905	10.9979	$\lambda_1 : 4$, $\lambda_2 : 3$, $\lambda_3 : 4$
300	5.0	2.99991	6.99957	10.9992	$\lambda_1 : 4$, $\lambda_2 : 4$, $\lambda_3 : 4$
400	5.0	2.99995	6.99976	10.9996	$\lambda_1 : 5$, $\lambda_2 : 4$, $\lambda_3 : 5$

Table 2: The three lowest eigenvalues calculated numerically by Jacobi's method. The number of steps in the algorithm is varied, to find out how the number of leading digits in the eigenvalues varies with it.

In Table 1 we can see that the accuracy of the eigenvalues is getting better until $\rho_{max} = 5.0$. For bigger values of ρ_{max} the accuracy is actually getting slightly worse, so then we know that $\rho_{max} = 5.0$ is a good choice in our algorithm. We therefore fix $\rho_{max} = 5.0$, and vary the number of steps in different executions, to find out what the optimal number of steps in the algorithm is. In Table 2 we can see that, as expected, the bigger n_{step} is, the more accurate the eigenvalues is calculated. But if we look at Table 3, we can also see that the execution time is growing fast with the number of steps in the algorithm. From that, if you want to use Jacobi's algorithm, you have to decide if you want less accuracy in the eigenvalues, or if you want to wait for some minutes to get the answer with good accuracy.

	Jacobi's method		**Armadillo eigenvalue solver**
n_{step}	**Execution time [sec]**	**Rotations [#]**	**Execution time [sec]**
100	2.401	39104	0.0078
150	11.58	88820	0.0057
200	35.22	158341	0.010
250	86.34	247778	0.017
300	169.3	357720	0.022
350	331.3	487615	0.034
400	527.2	637088	0.025

Table 3: The execution time and number of rotations for different values of n_{step} in the Jacobi algorithm. The execution time of the Armadillo eigenvalue solver is also listed, such that the execution time for the two methods can be compared.

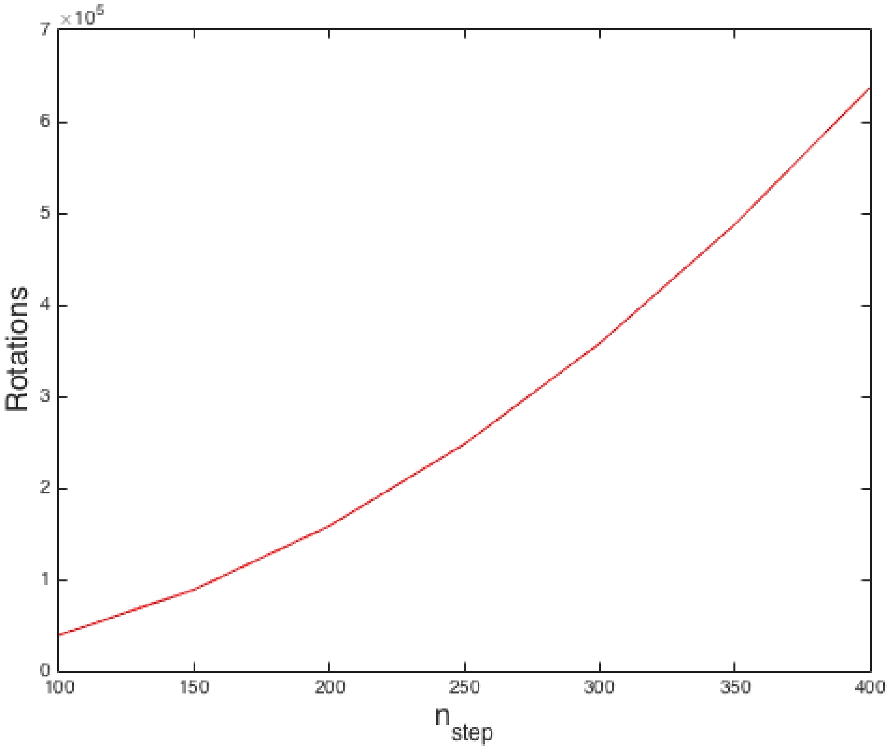

Figure 1: The number of rotations after the execution of the Jacobi algorithm, graphed as a function of the number of steps used in the method.

We can see from Figure 1 that the number of rotations after execution grows fast, as the number of steps used in Jacobi's method is growing. Most likely the graph can be approximated as a second degree equation, and with this graph it is easy to realize why the Jacobi method is using much more time for big n_{step} than for small n_{step}. More rotations means longer execution time.

4.2 Algorithm for solving the matrix eigenvalue problem for a two-electron system in a harmonic potential well

In Figure 2 we can see the probability distribution of a two electron system in its ground state, when it is in a harmonic oscillator potential well, and the repulsive Coulomb interactions is neglected. This is the exact same probability distribution as that of a single-electron system in a harmonic oscillator potential well, and the parameter ω_r does not affect the calculations.

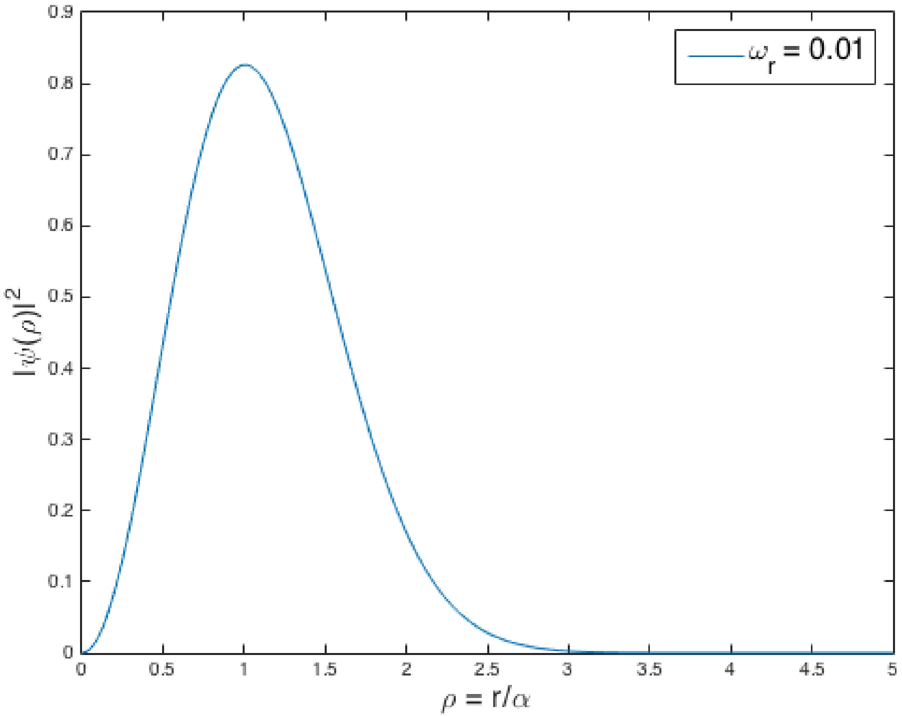

Figure 2: The probability distribution for the ground state of a single-electron system in a harmonic oscillator potential well.

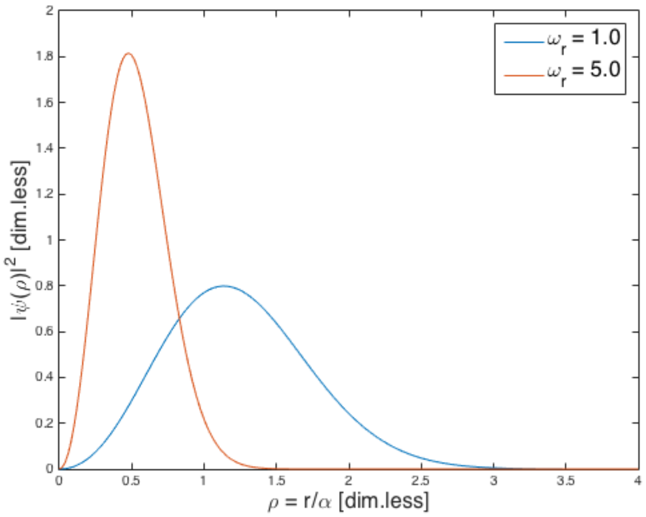

Figure 3: The probability distribution for the ground state of a two-electron system in a harmonic oscillator potential well. The repulsive Coulomb interactions is taken into account, and the strength parameters is $\omega_r = 5.0$ and $\omega_r = 1.0$.

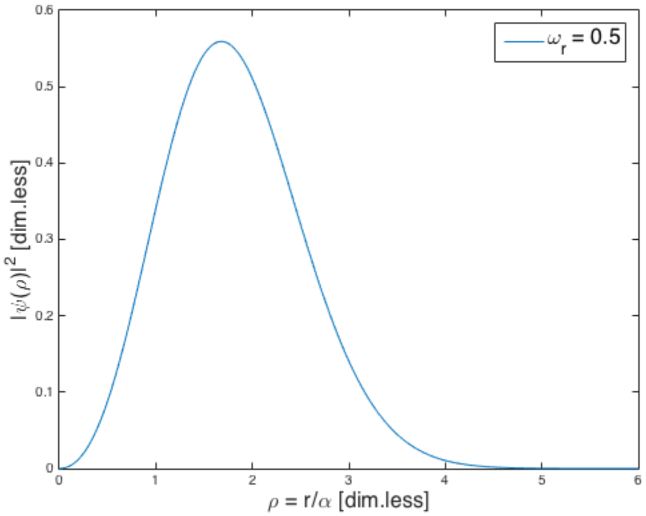

Figure 4: The probability distribution for the ground state of a two-electron system in a harmonic oscillator potential well. The repulsive Coulomb interactions is taken into account, and the strength parameters is $\omega_r = 0.5$.

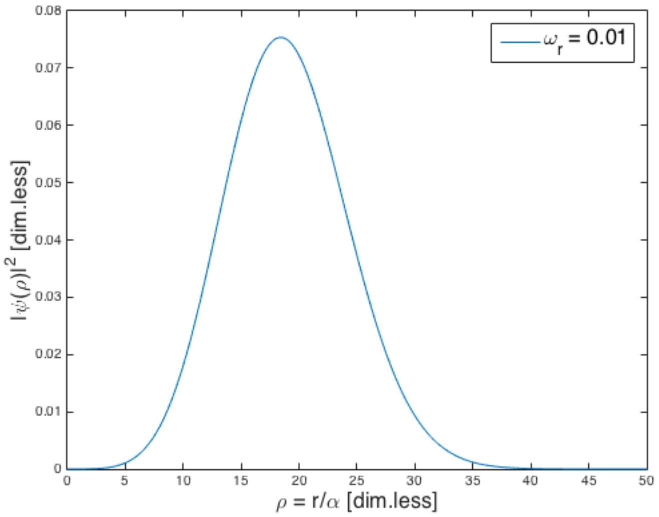

Figure 5: The probability distribution for the ground state of a two-electron system in a harmonic oscillator potential well. The repulsive Coulomb interactions is taken into account, and the strength parameter is $\omega_r = 0.01$.

In Figure 3, 4 and 5 we can see the probability distribution for a ground state two-electron system, in a harmonic oscillator potential well, for different strengths of the repulsive Coulomb interactions. The results is as expected. The higher the the strength of the repulsive Coulomb interactions is, the bigger the area the system can move in is. This is because the two electrons repel each other increasingly for bigger values of ω_r, and so the two electrons are more likely to be farther apart, than for small values of ω_r.

4.3 Unit testing

The unit tests are tested, and are working for their purpose. Now we can safely edit the algorithm, and get a warning if we do something with it that gives the wrong results.

4.3.1 Comparison of eigenvalues

The numerically calculated eigenvalues has been compared with some specific strengths of the potential, found analytically. The analytically calculated eigenvalues is taken from a article by M. Taut, Phys. Rev. A 48, 3561 - 3566 (1993). The article can be retrieved from the following web address
`<http://prola.aps.org/abstract/PRA/v48/i5/p3561_1>`.

We tested the eigenvalues for two different potential strength, and chose to do it with $n_{step} = 300$, so that we get 4 leading digits in the numerically calculated eigenvalues (see Table 2). The result off the comparison is listed in Table 5. The numerically calculated eigenvalues are close to the correct answer, but they do not have four leading digits, which we would expect.

$\boldsymbol{\omega_r}$	**Numerically**	**Analytically**	**Relative error**
0.25	0.62483	0.62500	$2.7132e-04$
0.05	0.17499	0.17500	$3.2655e-05$

Table 4: Jacobi's method is used in this case. Numerically calculated eigenvalues compared with analytically calculated eigenvalues, for two different potential strengths.

$\boldsymbol{\omega_r}$	**Numerically**	**Analytically**	**Relative error**
0.25	0.62474	0.62500	$4.2408e-04$
0.05	0.17499	0.17500	$5.1041e-05$

Table 5: The built-in-function of Armadillo is used in this case. Numerically calculated eigenvalues compared with analytically calculated eigenvalues, for two different potential strengths.

We can see that the eigenvalues calculated by use of the built-in-function of Armadillo, gives eigenvalues that is almost as accurate as the ones calculated with Jacobi's method.

5 Conclusion

Jacobi's method is very slow when it comes to big matrices, and the accuracy of the result is not very good compared to the time spent. The eigenvalues computed with Armadillo is almost as accurate, and i would prefer to use that method in stead of Jacobi's method. Armadillo does the calculations very fast, even for huge matrices.

The work of this project can be checked out in more detail by following the link: `<https://github.com/marieggen/FYS3150/tree/master/projects/project_2>`. The program jacobi_tridiag.cpp is the main program.