FYS3150 - Project 2

Vegard Falmår and Sigurd Sørlie Rustad
Universitetet i Oslo[*]
(Dated: September 30, 2020)

I. INTRODUCTION

The harmonic oscillator (HO) potential appears in many areas within quantum physics. This is because many potentials can be approximated, at least in some intervals, with a HO potential. The most famous example of this might be the model for a hydrogen atom, where the electron is bound to the proton by a HO potential. That is why, in this report, we are going to study both one and multiple electrons in a HO potential. We are also going to account for electrical forces between the particles.

When we try to solve Schrodinger's equation for both one and two particles, we end up with a eigenvalue problem (see theory section). This can be represented with a linear set of equations written on matrix-form (a derivation can be found in our project 1[1]). Using Jacobi's rotation algorithm, also described in the theory section, we can diagonalize the matrix to find these eigenvalues and eigenvectors. Then plot the results, comparing them to experimental values.

When implementing algorithms it is important to test your code and make sure it performs as expected. Therefore we perform unit-tests to make sure sections of our program works as expected. We also test the code with a test scenario (buckling beam), which have analytical eigenvalues and eigenvectors. The testing is described in the appendix (VII).

For our studies we have used c++ for heavy computation, python for visualization and bash for automation. All the code along with instructions on how to run it, can be cloned from our GitHub repository here[2].

II. THEORY

A. Unitary transformation

The transposed of a unitary matrix (U) is its inverse.

$$U^T = U^{-1}$$

From this we can prove that a unitary transformation preserves the orthonormality of vectors. Consider the set of orthonormal vectors $\{\mathbf{v}_i\}_i$ and the unitary transformation $\{U\mathbf{v}_i\}_i = \{\mathbf{w}_i\}_i$.

$$\begin{aligned}\mathbf{w}_i^T\mathbf{w}_j &= (U\mathbf{v}_i)^T U\mathbf{v}_j \\ &= \mathbf{v}_i^T U^T U\mathbf{v}_j = \mathbf{v}_i^T\mathbf{v}_j \\ &= \delta_{i,j}\end{aligned}$$

We notice that orthonormality is perserved.

B. Jacobi's rotation algorithm

Jacobi's rotation algorithm uses unitary transformations to diagonalize a matrix. A detailed description of the algorithm can be found here [1], however we will describe it briefly. In order to diagonalize a given matrix A, as mentioned over, we perform a series of unitary transformations.

$$B = U_n^T U_{n-1}^T ... U_0^T A U_0 ... U_{n-1} U_n$$

Here U_i are the unitary matrices and B the resulting diagonal matrix. An example of how U_i can look is given under.

$$\begin{bmatrix} 1 & 0 & ... & 0 & 0 & ... & 0 & 0 \\ 0 & 1 & ... & 0 & 0 & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & 0 & ... \\ 0 & 0 & ... & \cos\theta & 0 & ... & 0 & \sin\theta \\ 0 & 0 & ... & 0 & 1 & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & 1 & ... \\ 0 & 0 & ... & -\sin\theta & 0 & ... & 0 & \cos\theta \end{bmatrix}$$

The geometric interpretation is that U_i performs a rotation on T in order to zero out non-diagonal elements. It turns out that the fastest way to do this, is to zero out the largest non-diagonal matrix-element. First We define:

$$\cot(\theta) = \tau = \frac{a_{ll} - a_{kk}}{2a_{kl}},$$

Where a_{kl} is the largest non-diagonal element in A. Now to shorten notation we use $\tan(\theta) = t = s/c$, where $s = \sin(\theta) \land c = \cos(\theta)$. By defining θ such that a_{kl} becomes zero we get the quadratic equation

$$t^2 + 2\tau t - 1 = 0 \implies t = -\tau \pm \sqrt{1+\tau^2},$$

and can also obtain c and s

$$c = \frac{1}{1+t^2} \land s = tc.$$

[*] vegardfa@uio.no; sigurdsr@gmail.com
[1] github.com/sigurdru/FYS3150/blob/master/Project1/tex/Project1.pdf
[2] github.com/sigurdru/FYS3150/tree/master/Project2

The actual transformation is defined by the set of equations

$$\begin{aligned} b_{ik} &= a_{ik}c - a_{il}s, i \neq k, i \neq l \\ b_{il} &= a_{il}c + a_{ik}s, i \neq k, i \neq l \\ b_{kk} &= a_{kk}c^2 - 2a_{kl}cs + a_{ll}s^2 \\ b_{ll} &= a_{ll}c^2 + 2a_{kl}cs + a_{kk}s^2 \\ b_{kl} &= (a_{kk} - a_{ll})cs + a_{kl}(c^2 - s^2) \end{aligned} \tag{1}$$

Where b_{ij} are elements in B, again see [1] for a more detailed description of the algorithm.

An interesting question is; what happens to the eigenvectors and eigenvalues? Because B is diagonal the eigenvectors are just vectors with a 1 in one element and 0 everywhere else. We consider an arbitrary eigenvector $\mathbf{v}$ and eigenvalue λ that satisfy

$$B\mathbf{v} = \lambda\mathbf{v}.$$

Applying the unitary transforms we get

$$\begin{aligned} B\mathbf{v} &= U_n^T U_{n-1}^T ... U_0^T A U_0 ... U_{n-1} U_n \mathbf{v} \\ &= \lambda\mathbf{v} \end{aligned}$$

We notice that this is true if

$$\mathbf{v}' = U_0 ... U_{n-1} U_n \mathbf{v} \tag{2}$$

is an eigenvector with eigenvalue λ that satisfy

$$A\mathbf{v}' = \lambda\mathbf{v}'$$

Therefore, if you have the eigenvector $\mathbf{v}$ and matrices U_i you can find $\mathbf{v}'$ with equation (2).

FLOPS

C. Electrons in a harmonic oscillator potential

It turns out that if you try to solve Schrodinger's equation you end up with a eigenvalue problem. We will derive the math here, but everything is taken from here [2]. First we consider the Schrodinger's equation for one electron. Because of spherical symmetry we only need to look at the radial part.

$$-\frac{\hbar^2}{2m}\left(\frac{1}{r^2}\frac{d}{dr}r^2\frac{d}{dr} - \frac{l(l+1)}{r^2}\right)R(r) + V(r)R(r) = ER(r) \tag{3}$$

Where $R(r)$ is the radial wave equation, $V(r)$ the potential, l orbital momentum and r distance from the origin. Now because we only have one electron we end up with the HO potential $V(r) = (1/2)kr^2$. Substituting $R(r) = (1/r)u(r)$ and $\rho = (1/\alpha)r$ (where α is a constant with dimension length) we obtain

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

Setting the orbital momentum $l = 0$, inserting $V(\rho) = (1/2)k\alpha^2\rho^2$ and defining

$$\alpha \equiv \left(\frac{\hbar^2}{mk}\right)^{1/4} \wedge \lambda \equiv \frac{2m\alpha^2}{\hbar}E$$

Schrodinger's equation can then be described by the simple equation (4).

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

Now we can expand our model to two particles with Coulomb potential between them. First introducing the relative position $\mathbf{r}$ and center-of-mass coordinate $\mathbf{R}$

$$\mathbf{r} = \mathbf{r_1} - \mathbf{r_2} \wedge \mathbf{R} \ = \frac{1}{2(\mathbf{r_1} + \mathbf{r_2})}.$$

Where $\mathbf{r_1}$ and $\mathbf{r_2}$ are the particles positions.Then the radial Schrodinger's becomes

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

$E^{(2)}$ denotes the fact that we are looking at the total energy of two particles. From an ansatz for the wave equation, we can perform separation of variable on r and R.

$$u(r,R) = \psi(r)\phi(R) \wedge E^{(2)} = E_r + E_R$$

Adding the Coulomb potential

$$\frac{\beta e^2}{r}$$

between the two electrons we get

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

Again introducing $\rho = r/\alpha$ and defining a few variables

$$\omega_r^2 \equiv \frac{1}{4}\frac{mk}{\hbar^2}\alpha^4 \wedge \alpha \equiv \frac{\hbar^2}{m\beta e^2} \wedge \lambda \equiv \frac{m\alpha^2}{\hbar^2}E$$

We arrive at the Schrodinger's equation (5)

$$-\frac{d^2}{d\rho^2}\psi(\rho) + \omega_r^2\rho^2\psi(\rho) + \frac{1}{\rho}\psi(\rho) = \lambda\psi(\rho) \tag{5}$$

III. METHOD

As mentioned in the theory section, when you study one or more particles in a HO potential, you get a eigenvalue problem. For one particle this is given by equation (4). From Project 1 (see the theory section in [?]) we know this type of differential equation creates a set of equations, that can be represented as follows.

$$\begin{bmatrix} d_1 & e_1 & 0 & 0 & \dots & 0 & 0 \\ e_1 & d_2 & e_2 & 0 & \dots & 0 & 0 \\ 0 & e_2 & d_3 & e_3 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & \dots e_{N-3} & d_{N-2} & e_{N-2} \\ 0 & \dots & \dots & \dots & \dots & e_{N-2} & d_{N-1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \dots \\ \dots \\ \dots \\ u_{N-1} \end{bmatrix} = \lambda \begin{bmatrix} u_1 \\ u_2 \\ \dots \\ \dots \\ \dots \\ u_{N-1} \end{bmatrix} \tag{6}$$

Where $e_i = -1/h^2$, $d_i = 2/h^2 + \rho_i^2$ and $h = (\rho_N - \rho_0)/N$. ρ_0 and ρ_N are the endpoints that in reality goes from $\rho_0 = 0$ to $\rho_N = \infty$. Of course we cannot do this for $\rho_N = \infty$, so we have to decide on a maximum we want to evaluate. We do this with trial end error. Our other problem is to decide the number of integration points N. As with the endpoint ρ_N we try for different values and interpret the solutions.

We can solve this with Jacobi's rotation algorithm described in the theory section. In order to do this we use the sets of equations (1). To make sure algorithm was implemented properly we performed several tests. First unit-tests to make sure small chunks run as expected, then with a buckling beam problem that has analytical eigenvalues and eigenvectors. Further information on the tests can be found in the appendix (VII).

When we introduce another particle we end up with equation (5). This will also have the same shape as equation (6), the only difference being the value of diagonal elements

$$d_i = \frac{2}{h} + \omega_r^2 \rho_i^2 + \frac{1}{\rho_i}.$$

We can then implement Jacobi's rotation algorithm the same way we did for one particle.

IV. RESULTS

To test our algorithm we used two unit tests, one to test that we find expected indexes and one to test right eigenvalues and eigenvectors. All the tests passed and the results can be found in the appendix VII.

Confident in our algorithm we went fourth with the quantum mechanical systems. For one particle the resulting eigenvector can be seen in figure IV. We notice that the eigenstate has a maximum at around $\rho = 1$ and goes towards zero at the ends. In order to produce this plot we tested for different integration points n and endpoints ρ_N. We found $\rho_N = 5$ and $n = 200$ to be reasonable, however all the plots, except those with very low ρ_N and n, are almost indistinguishable.

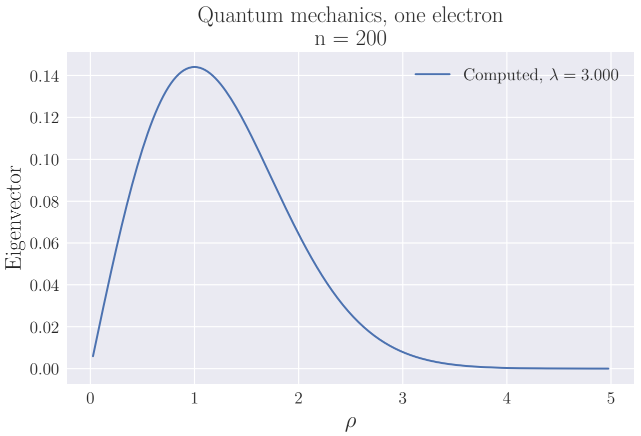

Figure 1. In this figure you see the eigenvector/eigenstate for one particle in a HO potential. Along the x-axis we have dimensionless length ρ and along the y-axis we have the elements of the eigenvector scaled to unit length. In the label we have the corresponding eigenvalue $\lambda = 3.000$.

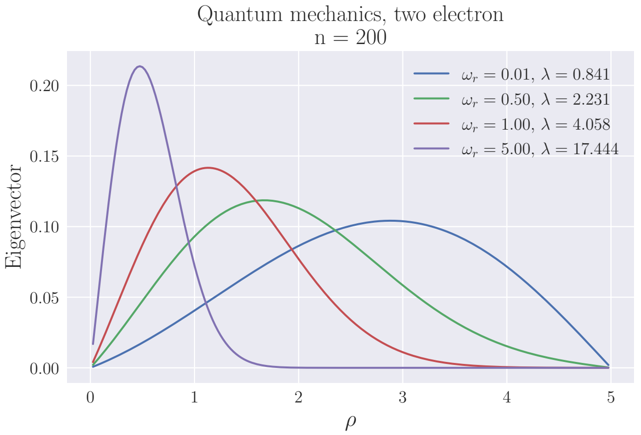

Figure 2.

N	BB	QM1	QM2
10	173	161	178
50	6241	6059	6137
100	25789	25285	25344
200	104882	102937	103182

Table I.

V. DISCUSSION

VI. CONCLUSION

VII. APPENDIX

A. Buckling beam

The simplest use case for our algorithm is a second order differential equation like 4 and 5, but without the potential. One such example is equation 7 for the vertical displacement $u(\rho)$ of a horisontal beam of length L when a force is applied in the x-direction towards the origin. $\rho = x/L \in [0,1]$, where L is the length of the beam, is a dimensionless length. The constant λ can be expressed as $\lambda = FL^2/\gamma$, where F is the applied force and γ is a constant defined by properties like the rigidity of the beam.

$$-\frac{d^2}{d\rho^2}u(\rho) = \lambda u(\rho) \tag{7}$$

Discretizing this equation gives a matrix equation on the form of equation 6 with $e_i = e = -1/h^2$ and $d_i = d = 2/h^2$, where h again is the step size. This is a tridiagonal Toeplitz matrix which has analytical eigenvalues described by 8 and analytical eigenvectors described by 9, both for $j = 1, 2, ..., N-1$.

$$\lambda_j = d + 2e\cos\left(\frac{j\pi}{N}\right) \tag{8}$$

$$\mathbf{u}_j = \left[\sin\left(\frac{j\pi}{N}\right), \sin\left(\frac{2j\pi}{N}\right), ..., \sin\left(\frac{(N-1)j\pi}{N}\right)\right]^T \tag{9}$$

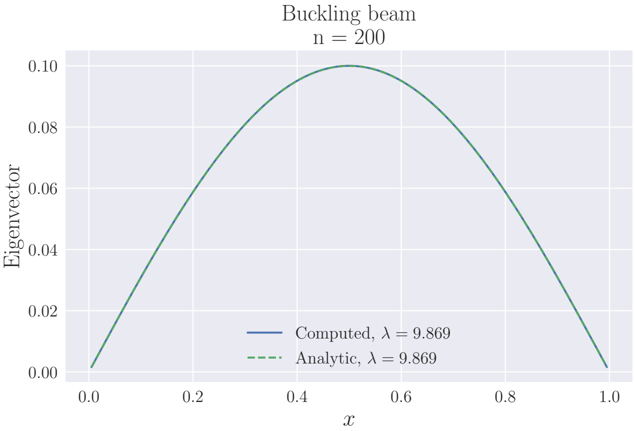

Figure 3.

B. Unit-tests

There are several sections in this code that can be tested independantly. We have developed tests for a general tridiagonal Toeplitz matrix that checks that the program finds the correct

[1] http://compphysics.github.io/ ComputationalPhysics/doc/pub/eigvalues/html/._eigvalues-bs011.html
[2] OPPGAVETEKST REF