FYS3150/4150 - Project 2
Nummerical methods for solving Schröedinger equation for two different potentials

Aram Salihi[1], Adam Niewelgowski[1]

[1]Department of Physics, University of Oslo, N-0316 Oslo, Norway

September 23, 2018

Abstract

Contents

1 Introduction:

2 Theoretical Model:

2.1 Jacobi rotation:

Jacobi rotation or the so called Jacobis algorithm, is a alogirthm which turns any given matrix $\mathbf{A}$ into a diagonal matrix $\mathbf{A_D}$. This is done by multiplying a set of orthogonal transformation $\mathbf{J}_i$, N times until a diagonal matrix is formed. Thus each time $\mathbf{J}_i$ is multiplied an entry in $\mathbf{A}$ is zeroed out.

$$\mathbf{A_D} = \mathbf{J}_N^T....\mathbf{J}_1^T\mathbf{A}\mathbf{J}_1....\mathbf{J}_N \tag{1}$$

A more detailed expalination is provided (ref her). Since $\mathbf{J}_i$ is a orthogonal transformation matrix it preserves the properties and the eigenvalues of $\mathbf{A}$. Thus, this method is a outstanding method to calculate the eigenvalues for a any given matrix due to its simplicity, but slow compared to algorithms (this will be discussed in detailed later). Since $\mathbf{A}_D$ is a diagonal matrix the eigenvlaues is along the diagonal:

$$\mathbf{A}_D = \begin{pmatrix} \lambda_1 & ... & .... & 0 & ... & 0 \\ 0 & \lambda_2 & ... & ... & ... & . \\ . & . & \lambda_3 & ... & ... & . \\ . & .. & . & \lambda_4 & ... & . \\ . & .. & ... & . & ... & 0 \\ 0 & .. & ... & ... & 0 & \lambda_N \end{pmatrix} \tag{2}$$

Where λ_i is the eigenvalues of matrix $\mathbf{A}$ and $\mathbf{A}_D$.

2.1.1 Properties of orthogonal transformation:

We will now show that the orthogonal transformation preserves the dot product and orthogonality. Consider a orthonormal vector $\mathbf{v}$ and transformation matrix $\mathbf{J}$. This orthogonal transformation matrix has the property:

$$\mathbf{J}^T = \mathbf{J}^{-1} \tag{3}$$

Notice that our orthonormal vectors form a basis $\mathcal{B} = \{\mathbf{v}\}$, thus $v_i^T v_j = \delta_{ij}$, where δ_{ij} is the kronecker delta . Further we will consider a vector $\mathbf{w} = \mathbf{J}\mathbf{v}$. The question is: "What will happen if we take the dot product of $\mathbf{w}$ with itself?". Consider the following:

$$\mathbf{w} \cdot \mathbf{w} = w_i^T w_j = (\mathbf{J}v_i)^T \mathbf{J}v_j = v_i^T \mathbf{J}^T \mathbf{J} v_j = v_i^T \mathbf{J}^{-1} \mathbf{J} v_j = v_i^T v_j = \delta_{ij} \tag{4}$$

Thus:

$$\mathbf{w} \cdot \mathbf{w} = w_i^T w_j = \delta_{ij} \tag{5}$$

2.2 Schröedinger Equation:

As previously mentioned we will be solving the three dimensional Schröedinger equation for harmonic oscillator and the coloumb potential:

$$V(r) = \frac{1}{2} m\omega^2 \mathbf{r}^2 \tag{6}$$

Where $\mathbf{r}$ is the position vector for center of mass for the electrons and ω is the frequency. We are going to define the potential as following.

$$V(r) = \frac{1}{2} k\mathbf{r}^2 \tag{7}$$

The constant k defines the steepness of the well. The coloumb potential:

$$V(r) = \frac{q}{4\pi\epsilon_0 r} \tag{8}$$

The time independent Schröedinger equation is mathematically defined as:

$$\frac{\hbar^2}{2m} \nabla^2 \Psi + V(r)\Psi = E\Psi \tag{9}$$

Since the potential is radially symmetric, the angular part can be dropped. Thus:

$$-\frac{\hbar^2}{2m} \left(\frac{1}{r^2} \frac{\mathrm{d}}{\mathrm{d}r} r^2 \frac{\mathrm{d}}{\mathrm{d}r} - \frac{l(l+1)}{r^2} \right) R(r) + V(r)R(r) = ER(r) \tag{10}$$

Where l is the oribtal angular momentum and E is the energy in the system. Doing the following substituion $R(r) = u(r)/r$ we can rewrite our equation to:

$$-\frac{\hbar^2}{2m} \frac{d^2}{dr^2} u(r) + \left(V(r) + \frac{l(l+1)}{r^2} \frac{\hbar^2}{2m} \right) u(r) = Eu(r) \tag{11}$$

We will now introduce a dimensionless distance variable $\rho = r/\alpha$ where α has dimension length. After some algebra, the Schröedinger equation can then be formulated as (in this project we have choosen to set $l = 0$):

$$-\frac{d^2}{d\rho^2} u(\rho) + V(\rho)u(\rho) = \lambda u(\rho) \tag{12}$$

where λ is the dimensionless energy eigenvalues defined as:

$$\lambda = \frac{2m\alpha^2}{\hbar^2}E \tag{13}$$

Note that:

$$\alpha^4 = \frac{\hbar^2}{mk} \tag{14}$$

The potential will look like: Non-interacting:

$$V_{HarmonicOscillator}(\rho) = \omega_r \rho^2 \tag{15}$$

where ω_r is the characteristic frequency defined in (ref) interacting:

$$V = V_{HarmonicOscillator}(\rho) + V_{Coloumb}(\rho) = \omega_r \rho^2 + 1/\rho \tag{16}$$

2.3 Schröedinger's equation and the eigenvalue problem:

Recall from previous the reformulation of Schröedingers equation:

$$-\frac{d^2}{d\rho^2}u(\rho) + V(\rho)u(\rho) = \lambda u(\rho) \tag{17}$$

This can be turned into a eigenvalue problem. We will now define a vector $\mathbf{u}$, a square diagonal matrix $\mathbf{V}$ (with the potentials on its diagonal element). The derivative operator can be written as toeplitz matrix $\mathbf{T}$ with $2/h^2$ on its main diagonal and $-1/h^2$, where h is the stepsize.

$$\underbrace{(\mathbf{T} + \mathbf{V})}_{\mathbf{A}} \mathbf{u}(\rho) = \lambda \mathbf{u}(\rho) \tag{18}$$

where we will define the main, upper and lower diagonal as:

$$d_i = \frac{2}{h^2} + V_i \qquad e_i = -\frac{1}{h^2} \tag{19}$$

Thus this can be discretized, and will have following look:

$$d_i u_i + e_{i-1}u_{i-1} + e_{i+1}u_{i+1} = \lambda u_i \tag{20}$$

2.4 boundary conditions:

In this project we will be solving the Schröedinger equation for Dirichlet boundaries. For $u_0 = u_N = 0$ we have:

$$(d_1 - \lambda)u_1 + e_2 u_1 = 0 \tag{21}$$

$$(e_{N-1} + e_{N+1})u_1 = 0 \tag{22}$$

This means we can remove these values the rows and coloumns and simply add zero on the top and bottom on the eigenvector.

3 Method

3.1 Algorithm

To study different situations of electrons in a harmonic oscillator potential, we had to develope Jacobi method algorithm. The pseudo code of the transformations is (see github repository for all the source codes implemented with c++)

```
while(biggest_of_diagonal < eps AND iterations < max_iterations)
 JacobiRotation(A, k, l)
 FindMaxOffDiagonal(A, k, l)
```

The idea behind this algorithm is to perform the rotation on the biggest off diagonal element in the matrix A, which has index k and l. After every rotation, we are updating the indexes with *FindMaxOffDiagonal*. Diagonalization is done when the biggest off diagonal element is smaller than given *eps*, which is an approximation to zero.

3.2 Unit tests

We applied both of the functions shown in the pseudo code on a small test matrices, to ensure that our algorithm is working the way we want it to before we moved on the bigger matrices to solve quantum problems. First, we checked the *FindMaxOffDiagonal* by simply checking if it returned the correct value

```
        find_max_offdiag(A,k,l,n);

    if((abs((A(l,k))-(abs(biggest_element))) < eps) && (k == 2) && (l == 1))
        {
        //PASSED
    }
    else{
        cout << "Finding max offdiag element test FAILED - need to check the
             find_max_offdiag function" << endl;
        exit(1);
    }
```

where *biggest_ element* is the biggest value placed on the off-diagonal in the test matrix.

To test the *JacobiRotation* we had to implement the whole algorithm (together with *FindMaxOffDiagonal* as shown in pseudo code above) to be able to check

1. eigenvalues produced by code against the analytical eigenvalues

2. orthogonality of the eigenvectors produced

3. symmetry conversation of a test matrix

3.3 Parameters

After passing all of the tests, we were ready to solve quantum problems. We were interested in two scenarios, one and two electrons in a potential well. With only one electron we had harmonic oscillator potential. With two electrons we had to add the Coulomb repulsive force between the particles. We were interested to study how the eigenfunction will behave with different ω_r in range between 0.01 to 5.0.

In addition we needed to find a good fit for the $\rho_{max} = \infty$ to match a given range of ω_r. We chose $\rho_{max} = 40$ for $\omega_r = 0.01$ and $\rho_{max} = 5$ for $\omega_r = \{0.5, 1.0, 5.0\}$.

4 Results

BLABLA