DELHI TECHNOLOGICAL UNIVERSITY

Department of Applied Physics

EP-205 Quantum Mechanics

Solving the Schrodinger Equation using finite difference method in MATLAB

Prepared by

Sakshi Arora
2K19/EP/084

Parmeet Singh
2K19/EP/066

Under the Guidance of

Dr. R. K. Sinha
Department of Applied Physics

November 27, 2020

1 Schrodinger wave equation

The Schrodinger equation is a linear partial differential equation that describes the wave function or state function for a quantum-mechanical system. The wave function hence obtained give us everything we need to know about the quantum mechanical system.
For one dimension it can be written as

$$i\hbar\frac{\partial\psi}{\partial t} = \frac{-\hbar^2}{2m}\frac{\partial^2\psi}{\partial x^2} + V(x,t)\psi \tag{1}$$

we will be using separation of variables to first separate the equation and we get

$$E\psi = \frac{-\hbar^2}{2m}\frac{d^2\psi}{dx^2} + V(x)\psi \tag{2}$$

and

$$i\hbar\frac{d\phi}{dt} = E\phi \tag{3}$$

where E is the energy of the given state at that time.
Our plan for solving the equation - We will be using the analytic solution of equation(3) and we will use Finite difference method to solve the time independent equation.

2 Motivation for quantum harmonic oscillator

Any oscillatory motion can be expressed as simple harmonic oscillator around minimas.
As we can express V(x) using Taylor series as

$$V(x) = V(x_0) + V'(x_0)(x - x_0) + \frac{1}{2}V''(x_0)(x - x_0)^2 + ... \tag{4}$$

ignoring the higher powers and not considering the $V(x_0)$ (as the force is defined as the potential of the potential we can ignore a constant)we get

$$V(x) = \frac{1}{2}V''(x_0)(x - x_0)^2 \tag{5}$$

which is the potential of the Quantum harmonic oscillator.
This is the reason harmonic oscillator is considered such an important topic.

3 Finite difference method

The finite difference is a numerical method to solve ordinary differential equations.It is mainly used while solving the equations using a advanced calculator or computer based software. These differential equations must have conditions

imposed on the boundary rather than at the initial point.
What we do is we substitute $\frac{dU}{dr}$ with $\frac{\Delta U}{\Delta r}$ and then we solve for the value of U at different x.
Now we can write this as

$$\frac{dU}{dr}_{r=r_i} = \frac{\Delta U}{\Delta r}_{r=r_i} = \frac{U_{i+1} - U_{i-1}}{r_{i+1} - r_{i-1}} \tag{6}$$

takingr_{i+1} as $r_i + d$ and r_{i-1} as $r_i - d$ we get

$$\frac{U_{i+1} - U_{i-1}}{r_{i+1} - r_{i-1}} = \frac{U_{i+1} - U_{i-1}}{2d}$$

Similarly we get

$$\frac{d^2U}{dr^2}_{r=r_i} = \frac{\Delta U^2}{\Delta r^2}_{r=r_i} = \frac{U_{i+1} - 2U_i + U_{i-1}}{d^2} \tag{7}$$

3.1 Using this method in time-independent Schrodinger equation

If we substitute the equations (4) and (5) into the time-independent Schrodinger equation we get

$$\frac{-\hbar^2}{2m}\frac{\psi_{i+1} - 2\psi_i + \psi_{i-1}}{d^2} + V_i\psi_i = E\psi_i \tag{8}$$

writing the same equation in matrix form

$$-\frac{\hbar^2}{2md^2}\begin{bmatrix} -2 & 1 & 0 & 0 & 0 & . & 0 \\ 1 & -2 & 1 & 0 & 0 & . & 0 \\ 0 & 1 & -2 & 1 & 0 & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & -2 & 1 \\ 0 & 0 & . & . & . & 1 & -2 \end{bmatrix}\begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ . \\ . \\ \psi_n \end{bmatrix} + \begin{bmatrix} V_1 & 0 & 0 & 0 & 0 & . & 0 \\ 0 & V_2 & 0 & 0 & 0 & . & 0 \\ 0 & 0 & V_3 & 0 & 0 & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & V_{n-1} & 0 \\ 0 & 0 & . & . & . & & V_n \end{bmatrix}\begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ . \\ . \\ \psi_n \end{bmatrix} = E\begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ . \\ . \\ \psi_n \end{bmatrix}$$

The first part of the LHS of equation written above is the kinetic energy and the second part is given as potential energy and the sum of both the equations is the Hamiltonian.

$$H\psi = E\psi \tag{9}$$

where H is the Hamiltonian operator and the equation (7) is an eigen equation and we will be using in built function in MATLAB, eig(A) which will give us the eigen values and eigen finctions.
The eigen values will be equal to the energy eigen states and the eigen functions will be equivalent to the wave functions.
We will be solving two different cases - one of finite potential well and the other of quantum harmonic oscillator

3.2 Quantum Harmonic Oscillator

The Potential for Quantum harmonic oscillator is defined as

$$V(x) = \frac{1}{2}m\omega^2x^2$$

where the ω is the angular frequency and m is the mass of the quantum particle. We solve the schrodinger wave equation by substituting the above potential in place of V(x).

3.2.1 Code for Simulation using OCTAVE

```
%%% Quantum Harmonic Oscillator
%%% we will be using finite difference method for solving ordinary
%%% differential equations
clc; clear all; close all;



a=-6; b=6; N=1001;
x=linspace(a,b,N);
h=x(2)-x(1);
h_bar=6.626e-34/(2*pi);

%%%defining the kinetic energy for the case
K=eye(N)*(-2);
i=ones(N-1,1);
K=K+diag(i,1)+diag(i,-1);

%%%defining the potential for the harmonic well
V_arr=(x.^2)/800;
V=diag(V_arr);

%%% getting the hamiltonian
H=-K/(2*h^2)+V;

n=input('Enter energy state: ');

%%% we will get the eigan vectros which will be the wave functions
%%% and the eigen value which represents the energy states

%%% this only gives the Time independant wave equation
[wave_func e_val]=eig(H);


%%%
for t=0:5e-37:5e-35
```

```
    % now we multiply the time part in order to get the complete wave
    % equation
    wave_func=wave_func*exp(-j*e_val(n,n)*t/h_bar);
    subplot(1,2,1);
    plot(x,real(wave_func(:,n)));
    hold on;
    plot(x,imag(wave_func(:,n)));
    plot(x,V_arr);
    hold off
    axis([-inf inf -0.05 0.05]);
    grid on
    xlabel("x")
    ylabel("wave function")


    legend("wave function(real part)","wave function(imaginary part)","Potential well")
    subplot(1,2,2);
    plot(x,(abs(wave_func(:,n))).^2);
    hold on
    grid on
    legend('probability density')
    pause(0.001);
    xlabel("x")
    ylabel("probability density")
     hold off
end
```

3.2.2 Output

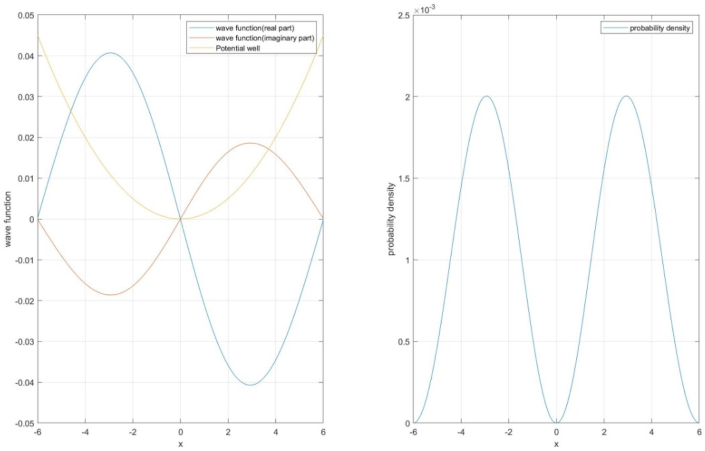

Figure 1: Quantum Harmonic Oscillator

3.3 Finite potential well

The finite potential well is defined as

$$V(x) = \begin{bmatrix} 0 & for & -a < x < a & \\ V_0 & for & x > a & x < -a \end{bmatrix}$$

where 2a is the width of the well and V_0 is the value of potential everywhere except the well.
Firstly we will define the Potential like mentioned above and then use the inbuilt function $diag(V)$ in order to get in proper format for calculation.

3.3.1 Code for Simulation using OCTAVE

```
%%% SIMPLE POTENTIAL WELL
%%% we will be using finite difference method for solving ordinary
%%% differential equations

clear all;
clc;

%%% asking for what energy state do you want
m = input("what energy state do you want?? \n");
%%% writing the constants first
m_e = 9.1e-31;
```

```
h = 6.626e-34/2*pi;
width = 4e-9;
% width of quantum well
V_max = 1; % these units will be in eV
V_min = 0;

%%% Now we are gonna define the space by
d = 0.1e-9;
z = -20e-9:d:20e-9;

%%%defining the potential over space
V = zeros(size(z));
V(z < -width ) = V_max;
V(z > width ) = V_max;
V(z < width & z > -width) = V_min;
V1=V;
%%% we will be making the hamiltonian now
% first calculating the Kinetic ebergy
[x y] = size(z);
K = eye(y,y)*(-2);
a=ones(y-1,1);
K = K + diag(a,1) + diag(a,-1);

%calculating the potential now

V = diag(V);

% Finally adding them both to make the Hamiltonin, we get

H = K*(-(h)^2/2*m_e*(d^2)) + V;

%%% getting the eigen value and eigen vector for these values

[eig_vec , eig_val] =  eig(H); % here the eig value gives the value of energy eigen states
% and  eig_vec in different columns give value of the wave function at different places

%%%%%%%%%%%%%%%%%%%%%%%%
C = 1;
s=10e110;
a=0;
t = a*s;% put the value for time here..
%  this is the time part
T = C*exp(-i*eig_val(m,m)*t/h);
T_real = real(T);
T_imag = imag(T);
```

```
Wave_fucntion = eig_vec(:,m)* T;
Wave_fucntion_r = real(Wave_fucntion);
Wave_fucntion_i = imag(Wave_fucntion);
 hold on

plot(z,Wave_fucntion_r)
plot(z,Wave_fucntion_i)
plot(z,V1)
legend('wave function(real part)','wave function(complex part)')
grid on
xlabel("x")
ylabel("wave function")
title("Finite potential well")
```

3.3.2 Output

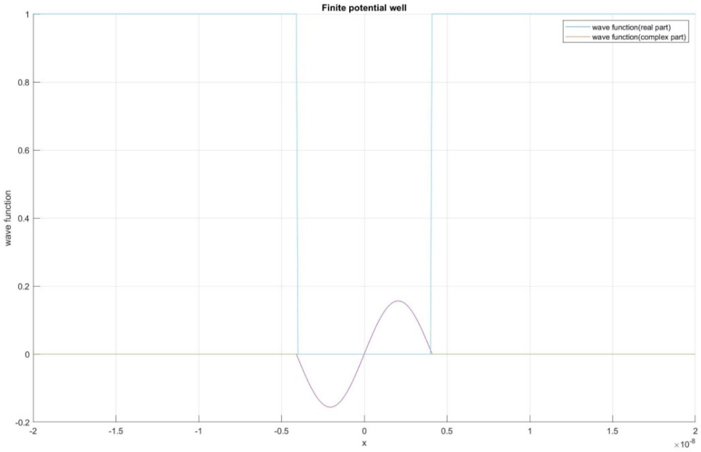

Figure 2: finite potential well

Conclusion

Through this project we have gained a lot of knowledge about both the quantum harmonic potential well and working in the MATLAB/OCTAVE environment. We have observed how the finite difference method, which we have used for this project can be used to solve a number of other functions in quantum mechanics and other fields as well.