PH3205-Computational Physics

Spring 2022

Bipradeep Saha (19MS135)

Assignment - 5

Problem Write a program to calculate Probability-Density as a function of time for a particle trapped in a tipple-well potential.

Solution: We solve the problem in Matlab. The required Matlab code is in the file *Assignment5.m*

We first define the potential

```
L = 5;                   % Interval Length.
N = 1000;                % No of points.
x = linspace(-L, L, N).';% Coordinate vector.
dx = x(2) - x(1);        % Coordinate step.
a = L/30;                % Width
b = L/50;                % Separation
D = 200;                 % Depth
U = -D*(heaviside(x+1.5*a+b)-heaviside(x+0.5*a+b)+heaviside(x+0.5*a) ...
    -heaviside(x-0.5*a)+heaviside(x-0.5*a-b)-heaviside(x-1.5*a-b));
```

Then we set up the hamiltonian

```
% Finite-difference representation of Laplacian and Hamiltonian,
% where hbar = m = 1.
hbar = 1;
m = 1;
e    = ones(N,1);
Lap  = spdiags([e -2*e e],[-1 0 1],N,N) / dx^2;
H    = -(1/2)*(hbar^2/m)*Lap + spdiags(U,0,N,N);
```

And we then obtain the first 2 eigenvalues and eigenstates.

```
% Find and sort lowest nmodes eigenvectors and eigenvalues of H.
nmodes  = 2;
[V,E]   = eigs(H,nmodes,'smallestreal');
[E,ind] = sort(diag(E)); % Convert E to vector and sort low to high.
V       = V(:,ind);      % Rearrange coresponding eigenvectors
% Rescale eigenvectors so that they are always
% positive at the center of the right well.
for c = 1:nmodes
    V(:,c) = V(:,c) / sign(V((3*N/4),c));
end
```

Then we obtain the probability density function

```
%----------------------------------------------------------------
% Compute and display normalized prob. density rho(x,t).
%----------------------------------------------------------------
% Parameters for solving the problem in the interval 0 < t < TF.
TF = 4*pi*hbar/(E(2)-E(1)); % Length of time interval.
NT = 100;                   % No. of time points.
t  = linspace(0,TF,NT);     % Time vector.
% Compute probability normalization constant (at T=0).
psi_o = 0.5*V(:,1) + 0.5*V(:,2);        % Wavefunction at T=0.
sq_norm = psi_o' * psi_o * dx;          % Square norm = |<ff|ff>|^2.
Usc = 4*U*max(abs(V(:))) / max(abs(U)); % Rescale U for plotting.
% Compute and display rho(x,t) for each time t.
figure1 = figure;
% Plot lowest 2 eigenfunctions.
plot(x,V(:,1), x,V(:,2),'--g',x,U/500);
legend('\psi_{E_0}(x)','\psi_{E_1}(x)','Double well potential (rescaled)','Location','EastOutside');
xlabel('x (m)');
ylabel('unnormalized wavefunction (1/m)');
ax = gca; % Get the Current Axes object
ax.XLim = [-1 1];
```

And we then animate the probability density for different times. The final animation is included in the .zip file with the name "tripleWell.avi". Here we just show the final density.