NB: The graded, first version of the report must be returned if you hand in a second time!

H3b: Time dependent quantum mechanics

Victor Nilsson and Simon Nilsson (930128-1854)

December 22, 2016

Task №	Points	Avail. points
$\sum$		

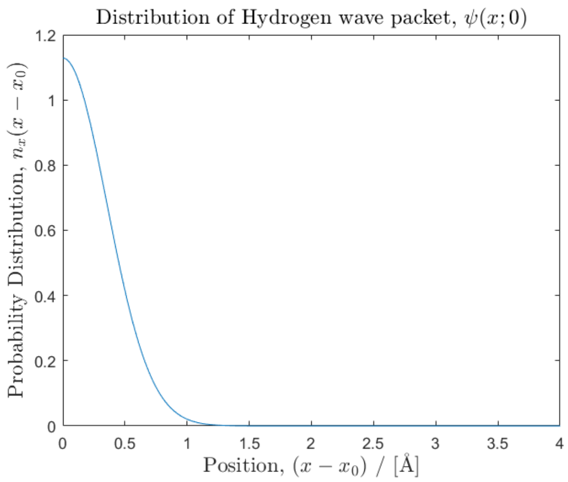
Introduction

Problem 1

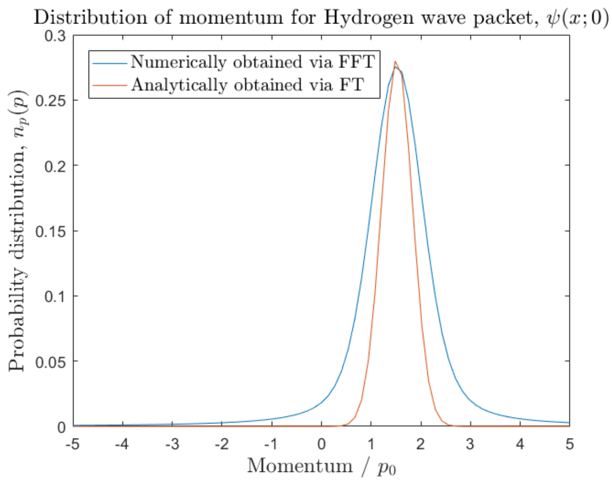
The initial distribution of the Gaussian wave packet,

$$\psi(x;0) = \frac{1}{(\pi d^2)^{1/4}} \exp(-\frac{(x-x_0)^2}{2d^2}) \exp(ip_0(x-x_0)/\hbar), \tag{1}$$

where d is the witdth of the packet, x_0 and p_0 are the corresponding initial position and momenta.



(a)



(b)

Figure 1: *(a): * (b): **

Fourier of the gaussian,

$$\mathcal{F}\left[\psi(x;0)\right](p) = \int_{-inf}^{inf} e^{-ipx}\psi(x;0)dx \tag{2}$$

$$= \frac{1}{(\pi d^2)^{1/4}} \int_{-inf}^{inf} e^{-ipx} e^{-\frac{(x-x_0)^2}{2d^2}} e^{ip_0(x-x_0)/\hbar} dx \tag{3}$$

$$\left\{ a = \frac{e^{-ipx_0}}{(\pi d^2)^{1/4}} \right\} = a \int_{-inf}^{inf} e^{-ip(x-x_0)} e^{-\frac{(x-x_0)^2}{2d^2}} e^{ip_0(x-x_0)/\hbar} dx \tag{4}$$

$$\left\{ \begin{array}{ll} x' & = x - x_0 \\ dx' & = dx \\ x' & \rightarrow x \end{array} \right\} = a \int_{-inf}^{inf} e^{-i(p-p_0/\hbar)x} e^{-\frac{x^2}{2d^2}} dx \tag{5}$$

$$\left\{ \begin{array}{ll} p' & = p - p_0/\hbar \\ b & = \frac{1}{2d^2} \\ x' & = x - \frac{ip}{2b} \\ dx' & = dx \\ x' & \rightarrow x \end{array} \right\} = a \int_{-inf}^{inf} e^{-bx^2} e^{-p'^2/4b} dx \tag{6}$$

$$= a e^{-p'^2/4b} \int_{-inf}^{inf} e^{-bx^2} dx \tag{7}$$

$$\left\{ \int_{-int}^{int} e^{-bx^2} dx = \sqrt{\frac{\pi}{2b}} \right\} = a e^{-p'^2/4b} \sqrt{\frac{\pi}{2b}} \tag{8}$$

$$= (\pi d^2)^{1/4} e^{-ipx_0 - (p\hbar - p_0)^2 d^2/2\hbar^2} \tag{9}$$

References

A Source code

A.1 main1.m

```
% ==================================
% HOMEWORK 3B IN COMP.PHYS. - TASK 1
% ==================================
% By Victor Nilsson, Simon Nilsson
% 2016
%
% Length scale: 1
% Time scale:   1 fs = 1e-15 s
% Energy scale: 1 eV

clear all, clc, close all

% ------ SIMULATION PARAMETERS ---------
hbar        = 1.054/1.602; % JS -> f eV s
d           = 0.5;
m           = 1.66/1.6*1e2;
p_0         = sqrt(0.1*2*m);
x_0         = 0;
dx          = 0.01;
n_points    = 1024;
dp          = 2*pi/(n_points*dx);


% ----------- VARIABLES ------------
% space samples
x = x_0 + dx*(0:n_points-1);
% and the corresponding samples in momentum space
p = dp*((0:n_points-1)-n_points/2);
% ---- Functions handles ----
Gaussian_Wave_Packet = @(x)1/(pi*d^2)^(1/4)*exp(-(x-x_0).^2/(2*d^2)).*exp(1i*p_0*(x-x_0)/hbar);
% Fourier transform obtained via Mathematica as the 'Inverse Fourier
% Transform', due to differences in FT defintion
Gaussian_Packet_Fourier = @(p)exp(-1i*p*x_0 - (d^2*(p*hbar - p_0).^2)./(2*hbar^2))*(pi*d^2)^(1/4);
% --------------------------------

% Sample-discretize the wave packet function
wave_packet = Gaussian_Wave_Packet(x)*dx;
theoretic_prob = abs(Gaussian_Packet_Fourier(p)).^2;

```

```matlab
prob = abs(wave_packet/dx).^2;
fft_prob_momentum = abs(fftshift(fft(wave_packet))).^2*dp;

% Plot prob.distr. in momentum space
figure(1); clf;
plot(p/p_0, fft_prob_momentum)
hold on
plot(p/p_0, theoretic_prob)
hold off
xlim([-5 5])
xlabel('Momentum / $p_0$', 'interpreter', 'latex', 'fontsize', 14)
ylabel('Probability distribution, $n_p(p)$','interpreter','latex', 'fontsize', 14)
title('Distribution of momentum for Hydrogen wave packet, $\psi(x;0)$','interpreter','latex','fontsize',14)
L = legend({'Numerically obtained via FFT', 'Analytically obtained via FT'},'interpreter','latex','fontsize',12);
set(L, 'location', 'northwest')

% Plot prob.distr. in normal space
figure(2); clf;
plot(x, prob)
xlim([0 4])
ylabel('Probability Distribution, $n_x(x-x_0)$','interpreter','latex', 'fontsize', 14)
xlabel('Position, $(x-x_0)$ / [\AA]', 'interpreter', 'latex', 'fontsize', 14)
title('Distribution of Hydrogen wave packet, $\psi(x;0)$','interpreter','latex','fontsize',14)
```

A.2 main2.m

```matlab
% ==================================
% HOMEWORK 3B IN COMP.PHYS. - TASK 2
% ==================================
% By Victor Nilsson, Simon Nilsson
% 2015
%
% Length scale: 1
% Time scale:   1 fs = 1e-15 s
% Energy scale: 1 eV

clear all, clc, close all

% ------ SIMULATION PARAMETERS ---------
hbar        = 1.054/1.602; % JS -> f eV s
d           = 0.5;
m           = 1.66/1.6*1e2;
p_0         = sqrt(0.1*2*m);
x_0         = 0;
dx          = 0.1;
n_points    = 2^10;
dp          = 2*pi/(n_points*dx);
dt          = 1;


% ----------- VARIABLES ------------
% space samples
x = x_0+dx*(0:n_points-1);
% and the corresponding samples in momentum space
p = ((0:n_points-1)-n_points/2)*dp;
% Functions handles
Gaussian_Wave_Packet = @(x)1/(pi*d^2)^(1/4)*exp(-(x-x_0).^2/(2*d^2)).*exp(1i*p_0*(x-x_0)/hbar);
Potential_Function = @(x) 0;
% ----
step_three=Gaussian_Wave_Packet(x);

potential = Potential_Function(x);
exp_potential = exp(-1i/hbar.*potential*dt);
inv_pot = exp(-1i/hbar*(hbar^2*p.^2./(2*m))*dt);


% Plot initial prob.distr.
figure(1); clf;
plot(x(1:n_points/2), abs(step_three(1:n_points/2).^2))
xlim([0 2])
xlabel('Position / [\AA]', 'interpreter', 'latex', 'fontsize', 14)
ylabel('Probability distribution', 'fontsize', 14)
title('$\left| \psi (0) \right|^2$', 'interpreter', 'latex', 'fontsize', 18)

% Plot the rest in a separate figure
figure(2); clf;
step_one = step_three;
step_two = fftshift(fft(step_one.*exp_potential));
step_three = ifft(ifftshift(inv_pot.*step_two));

```

```
plotHandle = plot(x(1:n_points/2), abs(step_three(1:n_points/2).^2));
xlabel('Position / [\AA]', 'interpreter', 'latex', 'fontsize', 14)
ylabel('Probability distribution', 'fontsize', 14)
title('$\left| \psi (t = 256 \mathrm{fs}) \right|^2$', 'interpreter', 'latex', 'fontsize', 18)

for j=1:n_points/4-1
    step_one = step_three;

    step_two = fftshift(fft(step_one.*exp_potential));
    step_three = ifft(ifftshift(inv_pot.*step_two));
    set(plotHandle, 'YData', abs(step_three(1:n_points/2).^2))
    pause(0.01)
end
```

A.3 main3.m

```
% ==================================
% HOMEWORK 3B IN COMP.PHYS. - TASK 2
% ==================================
% By Victor Nilsson, Simon Nilsson
% 2015
%
% Length scale: 1
% Time scale:   1 fs = 1e-15 s
% Energy scale: 1 eV

clear all, clc, close all

% ------ SIMULATION PARAMETERS ---------
hbar        = 1.054/1.602; % JS -> f eV s
d           = 0.5;
m           = 1.66/1.6*1e2;
p_0         = sqrt(0.12*2*m);
x_0         = 0;
dx          = 0.1;
n_points    = 2^10;
dp          = 2*pi/(n_points*dx);
dt          = 1;
v_0=0.1;
alpha=2.0;


% ---------- VARIABLES ------------
% space samples
x = x_0+dx*(0:n_points-1);
% and the corresponding samples in momentum space
p = ((0:n_points-1)-n_points/2)*dp;
% Functions handles
Gaussian_Wave_Packet = @(x)1/(pi*d^2)^(1/4)*exp(-(x-x_0).^2/(2*d^2)).*exp(1i*p_0*(x-x_0)/hbar);
Potential_Function = @(x) v_0*cosh(x/alpha).^(-2);
% ----
step_three=Gaussian_Wave_Packet(x);

potential = Potential_Function(x);
exp_potential = exp(-1i/hbar.*potential*dt);
inv_pot = exp(-1i/hbar*(hbar^2*p.^2./(2*m))*dt);

for j=1:n_points/4
    step_one = step_three;

    step_two = fftshift(fft(step_one.*exp_potential));
    step_three = ifft(ifftshift(inv_pot.*step_two));

    plot(x-max(x)/2,abs(ifftshift(step_three)).^2)
    pause(0.01)
end
```

A.4 main4.m

```
% ==================================
% HOMEWORK 3B IN COMP.PHYS. - TASK 2
% ==================================
% By Victor Nilsson, Simon Nilsson
% 2015
%
% Length scale: 1
% Time scale:   1 fs = 1e-15 s
% Energy scale: 1 eV

```

```
clear all, clc, close all

% ------ SIMULATION PARAMETERS ---------
hbar        = 1.054/1.602; % JS -> f eV s
d           = 0.5;
m           = 1.66/1.6*1e2;
p_0         = sqrt(0.12*2*m);
x_0         = 0;
dx          = 0.1;
n_points    = 2^10;
dp          = 2*pi/(n_points*dx);
dt          = 1;
a           = 0.3;
b           = 0.4;
d           = 0.7;
c           = 0.05; % 0.1;

tmp = V_d(0,a,b,c,d);
[R,v]=eig(tmp);
```