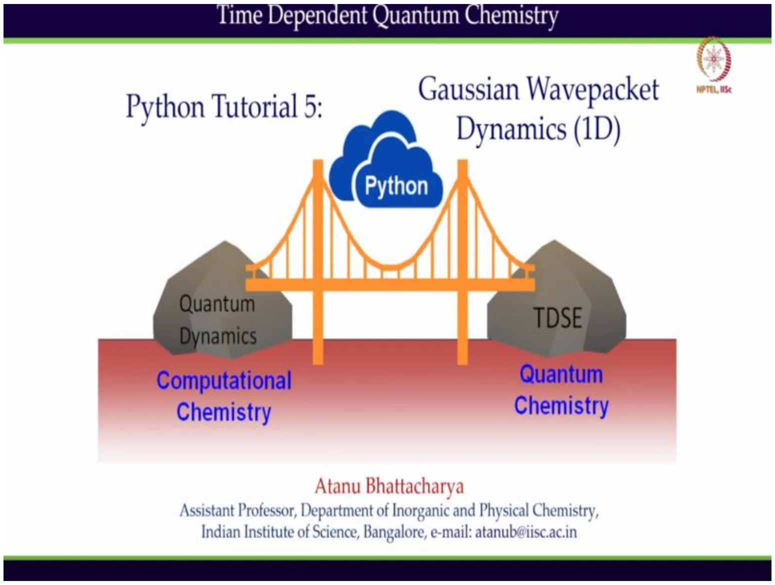
Time Dependent Quantum Chemistry
Professor. Atanu Bhattacharya
Department of Inorganic and Physical Chemistry
Indian Institute of Science, Bengaluru
Mod 05 Lecture 35
Wavepacket Dynamics under zero interaction potential

Welcome to Python tutorial 5 of the course Time Dependent Quantum Chemistry. In this tutorial, we will go over the one dimensional wavepacket dynamics, which will be solved numerically. So, this is the ultimate motivation we had in this course. Systematically, we have learned how to -- what does it mean by Python programming, we have presented that.

Then we have presented how to get expectation value, how to normalize discretized wavefunction. Then we have checked how to get the eigen state and eigen values associated with a particular state, system or Hamiltonian. And this is the ultimate motivation which, where we would like to take a look at how the wavepacket can propagate in time, particularly one dimensional wavepacket propagate in time numerically, so that any potential can be used to check the propagation.

(Refer Slide Time: 01:37)



Time Dependent Quantum Chemistry

Python Tutorial 5:

Gaussian Wavepacket Dynamics (1D)

Python

Quantum Dynamics

TDSE

Computational Chemistry

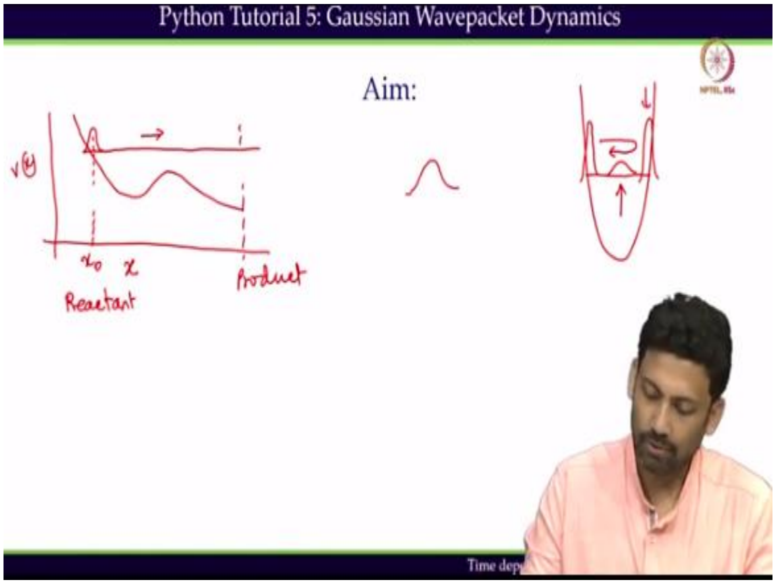
Quantum Chemistry

Atanu Bhattacharya

Assistant Professor, Department of Inorganic and Physical Chemistry,
Indian Institute of Science, Bangalore, e-mail: atanub@iisc.ac.in

We have already outlined the split operator approach to solve the TDSE and we will see the numerical implementation of in Python programming for this split operator approach to, to propagate that one.

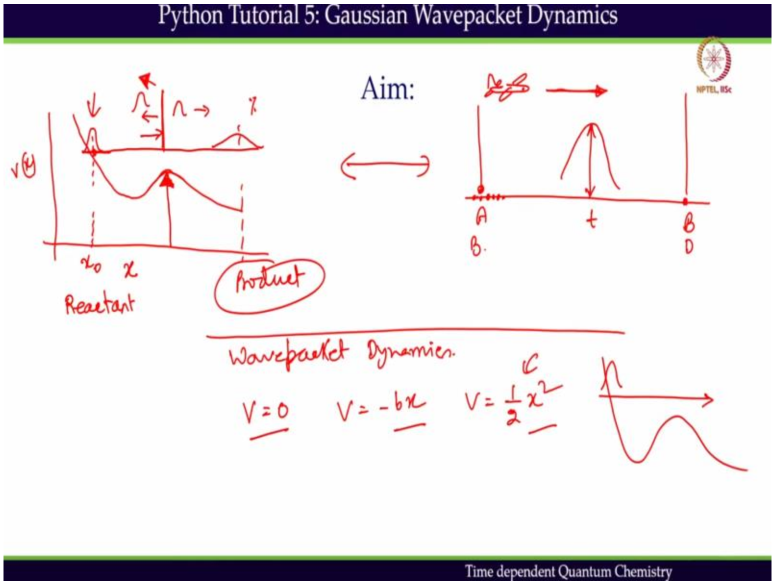
(Refer Slide Time: 01:55)



Python Tutorial 5: Gaussian Wavepacket Dynamics

Aim:

Time dep



Python Tutorial 5: Gaussian Wavepacket Dynamics

Aim:

Time dependent Quantum Chemistry

So, the aim is following. We may have a potential like this. This is let us say potential and this is the x-axis and this is V x. And I may create a wavepacket here. When I am creating wavepacket and drawing wavepacket like this way through this envelope function, I am not just showing the entire wavepacket, the actual wavepacket looks like this, there is an oscillatory part in addition to this envelope function. But to reduce the complexity, we just simply use this envelope function to show the wavepacket.

So, the wavepacket has been created, let us say, at this initial position, x_0 and it is should propagate. So, it may so happen that if this is the potential I have, so in chemistry, we talk about the reactant, and on this side, you have the product, that is all about the simplest form of the chemical reaction. Reactant is converting to product. And when reactant is converting to product, there is not a wavepacket dynamics going on the potential energy surface. And if we know the dynamics, if we know the timescale of this propagation, we know the dynamics of this reaction, this reactant converting to product.

So, wavepacket dynamics constitute the very basic quantum dynamics in chemistry. And it is very easy to understand also. We know that wavepacket, analytically, we have solved it. When the wavepacket propagates, it will spread out and it will decrease the amplitude. In general, that is the basic idea, but it is not true always. If the wavepacket has been created in the harmonic potential, so what happens in the harmony potential, it is, it is a narrow wavepacket we create, then it forms, it spreads out in the middle. And then again, it becomes narrow. So, the, the exact nature of wavepacket and its dynamics depends on what kind of potential it is experiencing.

But mostly, wavepacket when it would be propagating, it will be spreading out. And when, let us say, when wavepacket is propagating in time, this is time axis wavepacket is propagating in time is spreading out, it does not mean that the particle size was previously very narrow and now it is spreading out. It is not representing the particle size. Particle itself is a quantum particle. Quantum particle is a very tiny little particle, it is a point particle.

But the problem is that its exact position, I cannot determine, it is just what will happen that I will have a distribution. So, single experiment can give me one experiment, one exact, I mean, one tiny, I mean, point position. But if you, if you repeat the experiment, we will see that that will have a pollution distribution.

So, this is the distribution function which is evolving as a function of time. That is the, that is the meaning. We should not, we should never say that, size of the particle looks like a camel, let us say this, this, this, this, this size, it looks like a camel. And this camel is converted to let us say our dinosaur. It is not like. The size of the particle does not change; size of the particles remains to be the tiny little particle. That is why quantum effect is showing up. It is a point particle, but its distribution function, its distribution function is going to spread out. That is the, that is the meaning of it. And so this is something which you should remember.

Another thing is that when wavepacket, let us say a flight, let us say a flight, if I have a mountain here and a flight is flying over the mountain, when a flight is flying over the mountain, it can reach from point A to point B. That is the classical picture. It can reach from point A to point B from let us say, Bangalore to Kochi or Delhi. It can reach there with 100 percent probability as long as it is above the, this barrier of the mountain.

But whenever we think about the quantum dynamics, you look at this, this particle is actually having energy above this barrier. And when it is moving, classically we would say that, okay, 100 percent, this entire probability will come down here, but that does not happen. Because of this barrier, existence of this barrier here, always, there will be a scattering at this point, even if it is above the barrier, it will have a scattering. There is a probability of the wavepacket going back and there is a probability of wavepacket moving forward, always. And that limits the percent yield or the kinetics or dynamics of the entire chemical process.

So, we should always remember this one. This is something which we will be, if time permits, we will study it in the quantum scattering part. This is, so the quantum scattering is an important subject in chemical reactivity. And we study that one, and it suggests that, it is a very unusual outcome of quantum scattering. Even if the particle is above the barrier, it will be scattered because of the presence of this barrier. The barrier is present, and that is why it will be scattered. And to some, what percentage will be scattered, that depends on where you are with respect to the barrier, but there will be a scattering.

And if there is a scattering, there is no chance of 100 percent probability transmission here. So, product formation is limited by the barrier and where you are. So, this is something which you should keep it in mind, and these are the effects, like quantum scattering effects and all these things, we can understand better with the help of this wavepacket dynamics.

So this contrary, this contradictory picture should be kept in your mind always, that classical particle can transmit the barrier, I mean, can go over the barrier with 100 percent probability, but part, quantum particle, when it is going even over the barrier, it cannot transfer the probability 100 percent. There is, always there will be a scattering and a percentage will be reflected back.

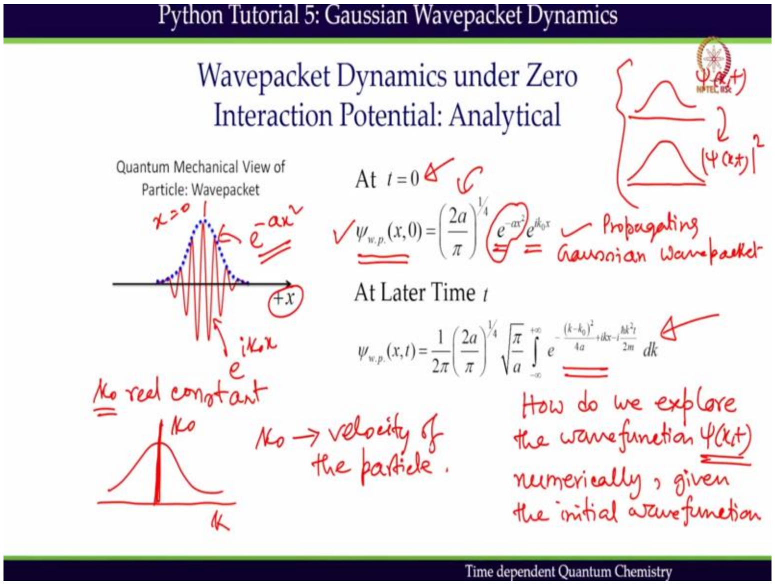
So, with this, and question is why we need numerical methods? We have used analytical approach to follow wavepacket dynamics. Wavepacket dynamics. We have seen analytical approach, and we have seen that it works for 0 interaction potential when the particle is not experiencing the potential, it works for linear potential when the particular is experiencing linear potential and it works for, analytical approach works for when you have this, the quadratic potential.

$$\boldsymbol{V = 0, V = -bx, V = \frac{1}{2}x^2}$$

So, these are the three situations where analytical approach works and we can solve it analytically. But if we go beyond this quadratic potential or linear potential, or this 0 potential, you need to solve it numerically. And these are the cases very, in practical situation, we do not get this linear potential, we do not get this quadratic potential even.

In practical scenario, my potential may have this kind of potential, and there is neither linear potential nor this quadratic potential. We have to create our own potential, and based on that potential, we have to propagate it. And in that case, we need the numerical solution to this wavepacket dynamics. So, numerical solution to this wavepacket dynamics is very important for practical application and we will go over it.

(Refer Slide Time: 11:26)



Python Tutorial 5: Gaussian Wavepacket Dynamics

Wavepacket Dynamics under Zero Interaction Potential: Analytical

Quantum Mechanical View of Particle: Wavepacket

At $t = 0$

$$\psi_{w.p.}(x,0) = \left(\frac{2a}{\pi}\right)^{1/4} e^{-ax^2} e^{ik_0 x}$$

Propagating Gaussian Wavepacket

At Later Time t

$$\psi_{w.p.}(x,t) = \frac{1}{2\pi}\left(\frac{2a}{\pi}\right)^{1/4} \sqrt{\frac{\pi}{a}} \int_{-\infty}^{+\infty} e^{-\frac{(k-k_0)^2}{4a} + ikx - i\frac{\hbar k^2 t}{2m}} dk$$

k_0 real constant

$k_0 \rightarrow$ velocity of the particle.

How do we explore the wavefunction $\psi(x,t)$ numerically, given the initial wavefunction

Time dependent Quantum Chemistry

Parts of the techniques we are going to use for this entire wavepacket dynamics that the part of those techniques we have already gone over Fourier Transform the normalization. We will just integrate them and then we will come up with the solution. So, we will make use of, as I have told earlier, that we will, we are going to make use of split operator approach in this numerical problem. And the, conceptually, the probability of finding the particle at any given position and time is determined by the square of absolute value of the wavefunction.

So, if I have an, a function like this, then probability of finding the particle, this is, this gives you quantum mechanics, always gives the probabilistic answer to the question, where is the particle? And it gives the probability. So, if **Ψ**(x,t) is the wavefunction, then the probability function is become $|\boldsymbol{\Psi}.(x,t)|^2$ absolute, square of the absolute value of the wavefunction.

So the, this is something which we have understood in this course. As we did before during the translational motion, discussion of translational motion, let us assume that, that initial wavefunction. So, the entire wavepacket dynamics is based on the, or any quantum dynamics is based on few steps.

The steps is that we have to construct the initial wavefunction that is the initial guess we have to make. And what kind of initial guess, how to make up good initial guess, that depends on our understanding of the experimental condition, because the particle, how it is starting its evolution, that has to be decided based on the experimental condition.

$$\boldsymbol{\Psi_{w.p}(x,0) = (\frac{2a}{\pi})^{1/4} e^{-ax^2} e^{ik_0x}}$$

And if the, and that is why we need to understand the experiment or experimental condition very well so that we can come up with a proper initial wavefunction representing the system or representing the particle. But here, we will assume that At equals 0 initial time, I have the wavefunction like this.

$$\boldsymbol{\Psi_{w.p}(x,0) = (\frac{2a}{\pi})^{1/4} e^{-ax^2} e^{ik_0x}}$$

This is called propagating wavefunction, propagating Gaussian wavefunction. So, we have seen that. This is called propagating Gaussian wavepacket.

So, this part is responsible for the propagation and this part is responsible for the envelope. So, this fast varying component is represented by $\boldsymbol{e^{ik_0x}}$ and this part envelop function, blue part is given by you $\boldsymbol{e^{-ax^2}}$. These are the two components we have, and this is normalized as we see. It is normalized and that is the another requirement we have to always begin with normalized wavefunction and then go for the solution.

Good thing about TDQC is that it preserves norm of the wavefunction. So, normalization is not time dependent anymore, normalization will be constant, and that is a good thing, because every time we have to confirm that at any given time, the total probability of finding the particle from minus infinity to plus infinity has to be 1, because I have single particle which is moving. So, this is the only interpretation of quantum mechanics through this probabilistic idea.

Now so this is, this is our initial wavefunction, that is the assumption we have made, and we know that this k_o, this k_o is going to, this is a real constant, real constant representing the angular wave number of the carrier wave of the wavepacket, these, the meaning of this angular wave number is now clear to us in two different context, we have presented in Fourier Transform we have presented it and also in the context of translational motion we have presented it.

And we have also understood that this k_o, the moment, this is x domain representation of the wavepacket. If we do the Fourier Transform, we get another, because it is a Gaussian shape in the x domain, so in the K domain, Fourier domain, we also get a Gaussian and the center of the Gaussian is actually represented by k_o. This, all these consequences and features of our wavepacket is known to us already based on the Fourier Transform discussion we had previously

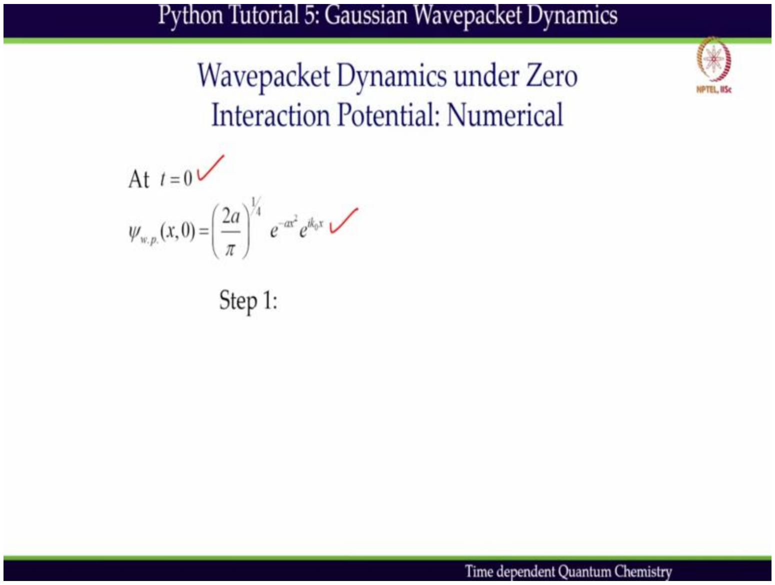
And also we know that the, this k_o is actually defining the velocity of the particle. This is defining the velocity of the particle. So, k_o has multiple perspective. It is defining that velocity of the particle, it is defining momentum of the particle, it is defining the center of the spectrum, this is a momentum domain spectrum. It is, so k_o has many perspective and those perspectives should be clear to us before we go for our or before we make use of this wavefunction for our wavepacket dynamics.

And another thing we remind here that the way we have taken this function, this envelope function e^{-ax^2}, we know that, that this function is centered at x equals 0 that we know. So, $e^{-a(x-x_0)^2}$, this is centered at x equals x_o. But if I do not have x_o, then it is centered at x equals 0. So, this is something we should remember that we are starting with a wavepacket, which is centered at x equals 0 and we will see how it is evolving.

Now question in this entire problem is that, how do we explore, how do we explore the wavefunction, the initial wavefunction numerically given the initial wavefunction? So, the question is how do we explore **Ψ**(x). And analytically, we have seen that this is an analytical approach we previously used and we have seen that this is the wavepacket we get. And finally, we have got this integration term and one final wavepacket expression we have been able to produce in the earlier one of the modules.

What we are going to do is that we are going to do the same thing numerically, and we can check what we previously got analytically and how does it compare with the numerical solution?

(Refer Slide Time: 20:08)



Python Tutorial 5: Gaussian Wavepacket Dynamics

Wavepacket Dynamics under Zero Interaction Potential: Numerical

NPTEL, IISc

At $t = 0$ ✓

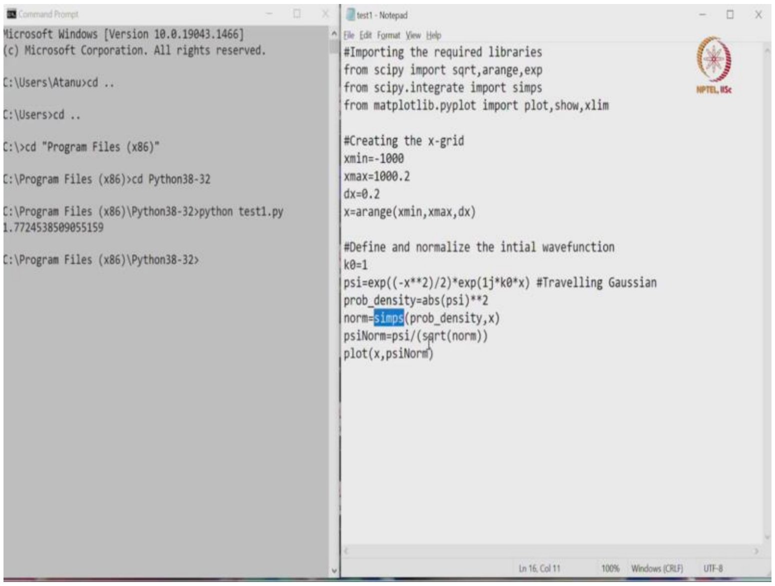
$$\psi_{w.p.}(x,0) = \left(\frac{2a}{\pi}\right)^{1/4} e^{-ax^2} e^{ik_0x} \checkmark$$

Step 1:

Time dependent Quantum Chemistry

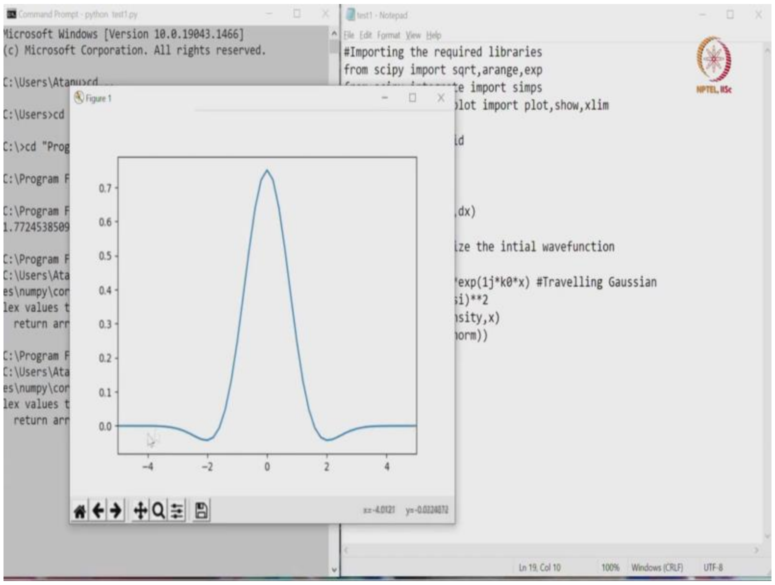
So, in order to do this, if we now jump into the numerical problem, it is clear that we are going to use this t equals 0 time, At t equals 0 time, we are going to use this wavefunction. And the first step of this entire procedure is to normalize the wavefunction. So, first, we will check how to normalize the wavefunction, this is quite clear to us now.

(Refer Slide Time: 20:31)



```
Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Atanu>cd ..

C:\Users>cd ..

C:\>cd "Program Files (x86)"

C:\Program Files (x86)>cd Python38-32

C:\Program Files (x86)\Python38-32>python test1.py
1.7724538509055159

C:\Program Files (x86)\Python38-32>
```

```python
#Importing the required libraries
from scipy import sqrt,arange,exp
from scipy.integrate import simps
from matplotlib.pyplot import plot,show,xlim

#Creating the x-grid
xmin=-1000
xmax=1000.2
dx=0.2
x=arange(xmin,xmax,dx)

#Define and normalize the intial wavefunction
k0=1
psi=exp((-x**2)/2)*exp(1j*k0*x) #Travelling Gaussian
prob_density=abs(psi)**2
norm=simps(prob_density,x)
psiNorm=psi/(sqrt(norm))
plot(x,psiNorm)
```



```
Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Atanu>cd ..

C:\Users>cd

C:\>cd "Prog

C:\Program F

C:\Program F
1.7724538509

C:\Program F
C:\Users\Ata
es\numpy\cor
lex values t
  return arr

C:\Program F
C:\Users\Ata
es\numpy\cor
lex values t
  return arr
```

```python
#Importing the required libraries
from scipy import sqrt,arange,exp
                  te import simps
                 plot import plot,show,xlim

                 id



                 ,dx)

                 ize the intial wavefunction

                 *exp(1j*k0*x) #Travelling Gaussian
                 si)**2
                 nsity,x)
                 norm))
```

I will move to laptop and we will write down the initial wavefunction as follows. First, we will be importing certain modules and sub-modules for our work. Importing the required library from scipy import, we will import square root functionality, arange functionality, and exponential functionality. From scipy dot integrate, this procedure is quite similar to what we did to find out the normalized wavefunction in Python tutorial 2, and that is the procedure, so we are, because we are familiar with it and that is the reason why I can directly write down what I need to import.

So, from scipy integrate, I have to import Simpson's method, because I have to do this integration, numerical integration through Simpson's method from matplot library, pyplot, I will import plot x limit and show. All these features are known to you now, now I'll be first creating the x grid. When I first create the x grid, I will write out x minimum equals minus 1000 x maximum equals plus 1000, plus 1000.2 I will write down because the, this point will be excluded from the list prepared by the arange functionality. And then dx I will place 0.2.

So, we have -- this will create the x domain. And then I will define x equals arange functionality, x minimum, x maximum, and dx. So, this will create the array. Now I will be able to define the, define and normalize the initial wavefunction. Good comment line will help you remember what you are doing right now, because it is very easy to forget this programming, unless you do the practice regularly exponential of minus x square.

I will make minus x square by 2. It is just, that is the way I am selecting. It does not need to be like that, anything you can select as long as it is a Gaussian envelope, it is perfectly fine. And then this propagating part, which is 1J, we know that the complex number is written in Python as j, then k naught, then multiplied by x. Now x is actually T, this is x domain, yes this is x domain. And so once we have defined this one, this definition we can write down, this is a traveling Gaussian form.

Then we have to find out the probability density, probability density would be given by absolute value of psi. This is the probability density, then I can find out norm, norm equals Simpson's method, y, this y and x I have to give and y is going to be the probability density comma x, so it will be sampled on each point of x and it will be done, it will be, the numerical integration will be done. Numerical integration in the end, giving you the area under the curve.

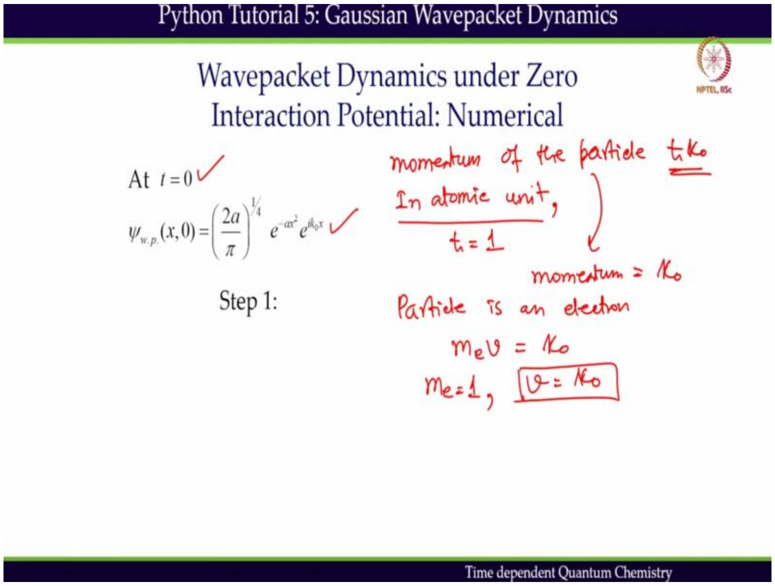
And finally, I will be plotting this psi, sorry psiNorm I am getting as the wavefunction divided by the norm, divided by sqrt off the norm, that is the definition of, that is the way you can, you normalize the wavefunction. So, this is the way we have done it. And as we can see that this plotting part we can avoid right now, we can just check whether what we get from here and this is the way it can normalize it. So, I can write down print norm.

So, if we run the program, we see that it is printing the -- and the value, the norm, it has given 1.77 value. And it is, it can be assumed that and I can also plot if we want to check the wavefunction which we are dealing with. So, in that case, I need this matplot library dot py plot, we have to use plot, show, x limit, these are the things we need. So, we can plot on even plot, psiNorm comma.

This is x coma, this part is x comma psiNorm. So, I will remind, this also I have reminded before, but I can remind again, see when you use Simpson's method, you write y value first and then x value. But when you use plot, you drive x value first and then y value. So, y array. These are all array. So, that is the way it does, the syntaxes are. So, I can again run the program and I can see that it can be plotted as this and we have, we can make the, we can do one thing, we can change the x limit, we can control the x limit as let us say minus 5 to plus 5. Let us see what happens.

If we use minus 5 to plus 5, yeah, this is, this is the wavepacket we have right now, which is having an oscillation, and if we forget that oscillation, it is going to give you the envelope function. So, this is all we have, the initial wavefunction we have defined already, and we can see that it is centered at a 0, which is x equals 0 position.

(Refer Slide Time: 29:21)



Python Tutorial 5: Gaussian Wavepacket Dynamics

Wavepacket Dynamics under Zero Interaction Potential: Numerical

At $t=0$ ✓

$$\psi_{w.p.}(x,0)=\left(\frac{2a}{\pi}\right)^{1/4} e^{-ax^2} e^{ik_0x}$$ ✓

Step 1:

momentum of the particle $\underline{\hbar k_0}$

$\underline{\text{In atomic unit,}}$

$\hbar = 1$

momentum = k_0

Particle is an electron

$m_e v = k_0$

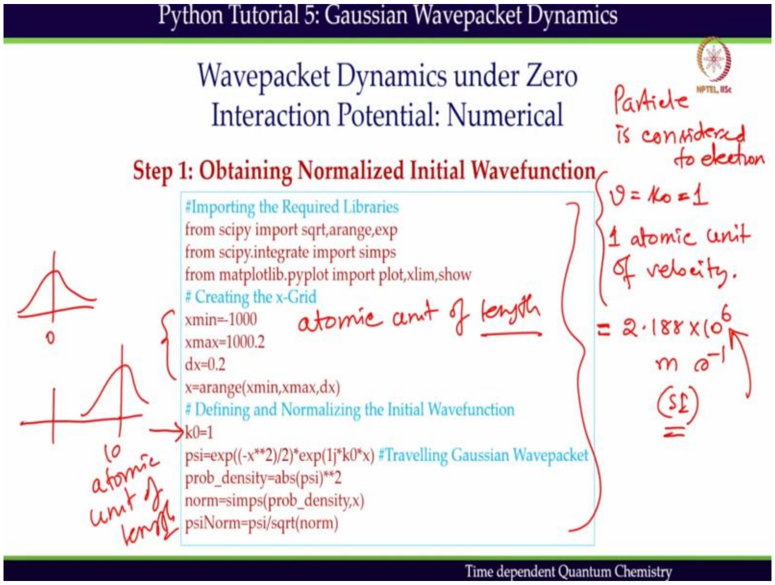
$m_e = 1$, $\boxed{v = k_0}$

Time dependent Quantum Chemistry

So, these are the, these are the points we have found from the from this. So, now we will go back to the discussion here. What we have done in this is that the momentum of the free particle, if we look at the momentum of the free particle, momentum of the particle, that is going to be $\frac{hk_0}{2\pi}$, that is the momentum we get. All these calculations in general we do in atomic unit. It is an unit system which is very frequently used in calculation of atoms and molecules, electronic structure, calculations of atoms and molecules, and then wavepacket dynamics, all these calculations are used based on atomic unit.

Under the atomic unit, there are certain things we assume or we consider. Under atomic unit, $h/2\pi$ is considered to be 1 atomic unit. And in that case, momentum, we get as k_0 in atomic unit. So, when I say momentum in close k_0 in atomic unit, it is k_0. Now if the particle is considered to be an electron, if we consider the particle, is an electron, in that case, this momentum will be represented by this k_0 and Me is considered to be again an in atomic unit, mass of the electorn is considered to be 1 atomic unit. In that case, velocity is equal to k_0. That is the -- and this is in atomic unit only.

(Refer Slide Time: 31:37)



Python Tutorial 5: Gaussian Wavepacket Dynamics

Wavepacket Dynamics under Zero Interaction Potential: Numerical

Step 1: Obtaining Normalized Initial Wavefunction

```
#Importing the Required Libraries
from scipy import sqrt,arange,exp
from scipy.integrate import simps
from matplotlib.pyplot import plot,xlim,show
# Creating the x-Grid
xmin=-1000
xmax=1000.2
dx=0.2
x=arange(xmin,xmax,dx)
# Defining and Normalizing the Initial Wavefunction
k0=1
psi=exp((-x**2)/2)*exp(1j*k0*x) #Travelling Gaussian Wavepacket
prob_density=abs(psi)**2
norm=simps(prob_density,x)
psiNorm=psi/sqrt(norm)
```

Particle is considered to electron

$\vartheta = k_0 = 1$

1 atomic unit of velocity.

$= 2.188 \times 10^6$ $m\,s^{-1}$ (SI)

atomic unit of length

Time dependent Quantum Chemistry

So, when I have considered in the, in the, in this numerical procedure, I have considered you can see that this k_0 we have considered to be 1. It means that I have considered velocity of the particle, the particle is considered to be electron, to be electron and velocity equals k_0 equals 1 in atomic unit and it means that it is 1 atomic unit of velocity.

So, what is the velocity with which the particle electron is traveling in my problem? In this problem, we are assuming that the particle is traveling, initial velocity was 1 atomic unit. Which is equivalent one atomic unit may not make sense to you but which is equivalent to in the SI unit if we write down, it is going to be 2.188 into 10 to the power 6 meter per second. This is the SI unit equivalent, SI unit component, SI unit value.

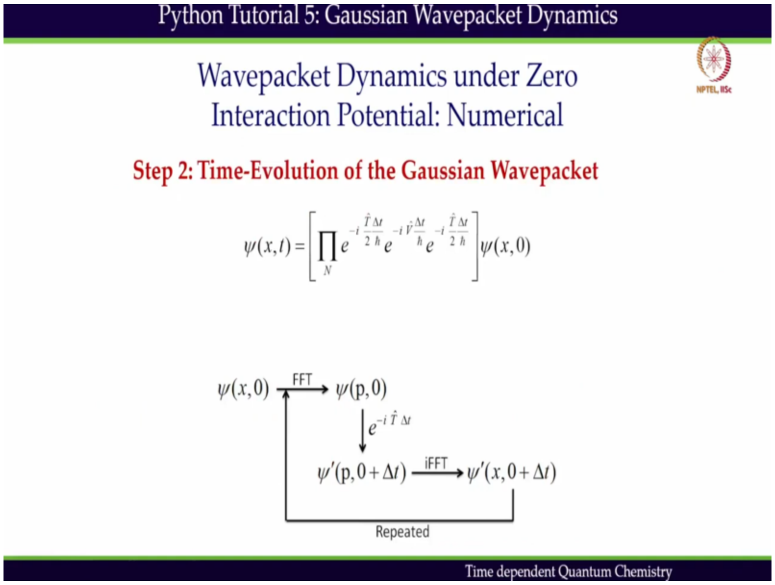
So, we are assuming that particle is traveling with this velocity. So, what is the idea of atomic unit? Atomic unit was proposed by Hartree in the beginning of the, I mean, when the quantum mechanics was being proposed. Hartree actually came up with this idea that we should use atomic unit in these calculations, because we are going to deal with very large number or very small number. Sometimes we may deal with minus 10, this kind of number or sometimes we can deal with this kind of large number.

And to use this kind of large number, it is going to be very difficult to use the SI unit for this calculation, it will be much convenient to do with this atomic unit. And slowly entire quantum mechanics community has accepted that, yes, it is the convenient way to go ahead. So, when I have written here x maximum, x minimum, that 1000 and 1000.2, it is assuming the atomic unit, atomic unit of length. So, when I say that, the wavepacket is centered at 0, that is 0 atomic unit of length, if the wavepacket has moved from 0 to some other value, let us say 10, then this is the 10 atomic unit of length.

Everything has been represented in the atomic unit. And based on this representation only, if I think about the kinetic energy of the particle, kinetic energy of the particle is going to be half m_e, we are considering electron as a particle, half me and then v^2 , which is nothing but half me is one and v equals k which is k_0, so half naught square is the kinetic energy of the particle in atomic unit.

So, as you can already realize that if we use an atomic unit, many, factors is being simplified. If we use the atomic unit. So, that is the basic idea. Remaining part, whatever we have written in the program is equivalent to what the way we have learned the normalization thing, how to normalize a discretized wavefunction. So, this is a discretized wavefunction representation, and we are normalizing it. So, first thing we have to do is the, we have to normalize the wavefunction, and initially wavefunction.

(Refer Slide Time: 35:54)



Python Tutorial 5: Gaussian Wavepacket Dynamics

Wavepacket Dynamics under Zero Interaction Potential: Numerical

Step 2: Time-Evolution of the Gaussian Wavepacket

$$\psi(x,t) = \left[\prod_N e^{-i\frac{\hat{T}}{2}\frac{\Delta t}{\hbar}} e^{-i\hat{V}\frac{\Delta t}{\hbar}} e^{-i\frac{\hat{T}}{2}\frac{\Delta t}{\hbar}}\right]\psi(x,0)$$

$$\begin{tikzcd}
\psi(x,0) \arrow[r, "\text{FFT}"] & \psi(\mathrm{p},0) \arrow[d, "e^{-i\hat{T}\Delta t}"] & \\
& \psi'(\mathrm{p},0+\Delta t) \arrow[r, "\text{iFFT}"] & \psi'(x,0+\Delta t) \arrow[ull, bend left, "\text{Repeated}"]
\end{tikzcd}$$

Time dependent Quantum Chemistry

And then we have to propagate the initial wavefunction. The propagation part we will look at in the in the next session.