

1_quantum_dynamics

May 28, 2024

1 Time Dependent Quantum Mechanics

In Lecture 10, we developed some solutions for the stationary Schrödinger equation. We solved the harmonic oscillator or the particle in a box model. All of this did not give any dynamics of particles in potentials but rather a static picture. Today we would like to consider the time dependent Schrödinger equations which yields the dynamics of the system. We will develop a solution based on Fourier transforms, as we still have to practice that a bit.

1.1 Time dependent Schrödinger equation

The time dependent Schrödinger equation is given by

$$-i\hbar \frac{\partial \Psi}{\partial t} = -\frac{\hbar^2}{2m} \nabla^2 \Psi + V(\vec{r}) \Psi \quad (1)$$

in 3 dimensions. We have already shortly mentioned this equation in 1 dimension

$$-i\hbar \frac{\partial \Psi(x, t)}{\partial t} = \left(-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x, t) \right) \Psi(x, t) \quad (2)$$

during the last lecture. We are using the 1-dimensional equation today again to study the dynamics of a particle in a box and the tunneling of a particle. To do so, we first have a look at wavepackets.

1.2 Wavepackets

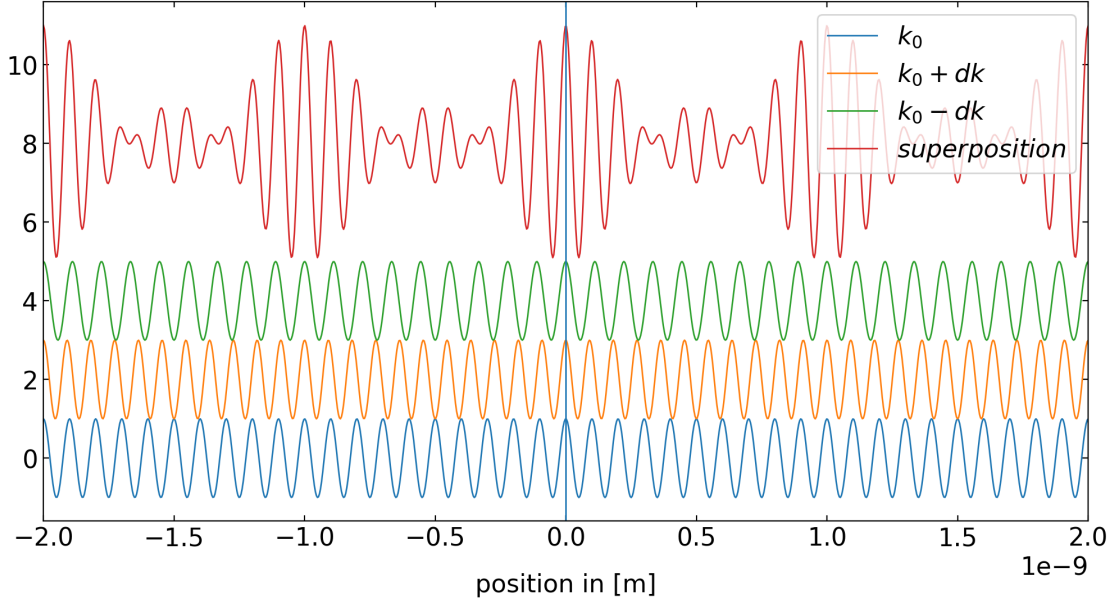
As particles are typically localized objects we need to construct a wavepacket. A wavepackets can be constructed from a superposition of plane waves. A wavepacket can be constructed from a superposition of plane waves with different wavenumber k or different frequency ω . We write down examples only for the wavenumber, but equivalently, they may be developed for the frequency as well.

$$\Psi(x) = A \int_{+\infty}^{-\infty} \phi(k) e^{ikx} e^{-i\omega t} dk \quad (3)$$

The function $\phi(k)$ gives the amplitude of the wavefunction with the value k in the wavepacket. If for example only one wave at k_0 is contributing to the wavepacket, then the function $\phi(k_0) = 1$ but for all other k this function is zero. Therefore the integral results only in a single plane wave. The value of ω is given by $\omega = \frac{\hbar k^2}{2m}$.

1.2.1 Demonstration of superposition of plane waves

The code below just shows how waves of different wave vector can be combined to result in a wave packet, i.e. an object that is localized in space. This is nothing else than a Fourier decomposition of a spatially extended object.



1.2.2 Wavepacket

If the distribution of amplitudes $\phi(k)$ extends over a larger region, i.e. from $k_0 - \Delta k/2$ to $k_0 + \Delta k/2$ with equal amplitudes over a whole range $\phi(k) = \text{const.}$ our first wavepacket arises. For matter waves we have to take into account that ω is a function of the wavevector as well.

$$\omega = \frac{\hbar k^2}{2m} \quad (4)$$

To calculate the shape of the wavepacket we approximate the above dispersion relation by

$$\omega = \omega_0 + \frac{\partial \omega}{\partial k}(k - k_0) \quad (5)$$

which replaces the above dispersion relation by an approximate linear one.

As a final result we obtain the wavepacket

$$\Psi(x, t) = 2A \frac{\sin\left(x - \frac{\partial \omega}{\partial k} t\right) \frac{\Delta k}{2}}{x - \frac{\partial \omega}{\partial k} t} e^{i(k_0 x - \omega_0 t)} \quad (6)$$

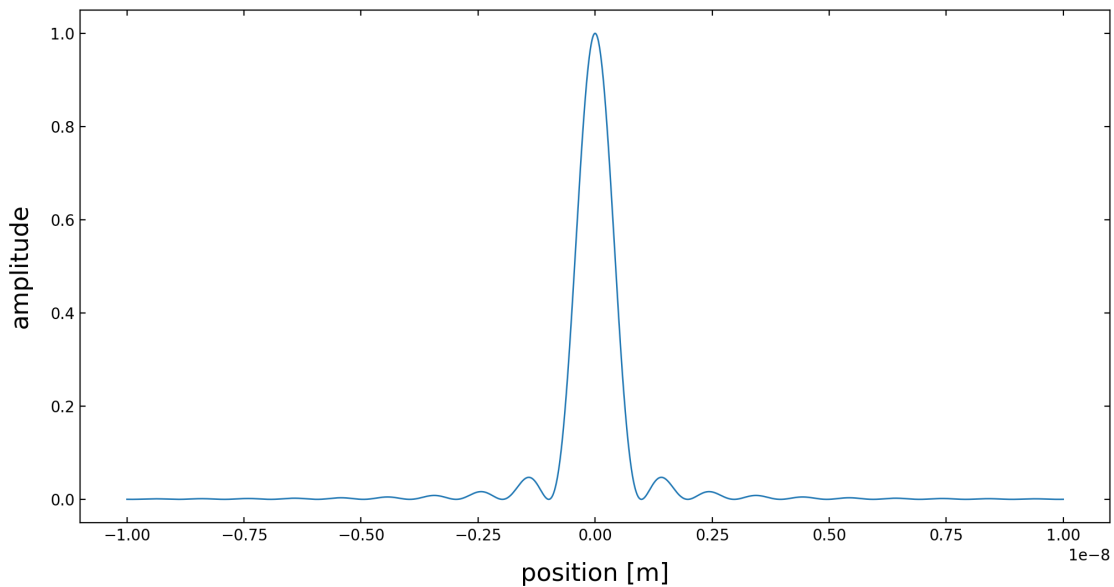
Obviously the wavepacket has an underlying plane wave which is the rightmost exponential factor. The factor in front of the plane wave defines the envelope of the wavepacket. The envelope appears

through the superposition of all plane waves. It propagates with the group velocity $v_g = \partial\omega/\partial k$, while the plane wave propagates with the phase velocity $v_{ph} = \omega_0/k_0$, which can be different than the group velocity.

1.2.3 Wavepacket with rectangular amplitude

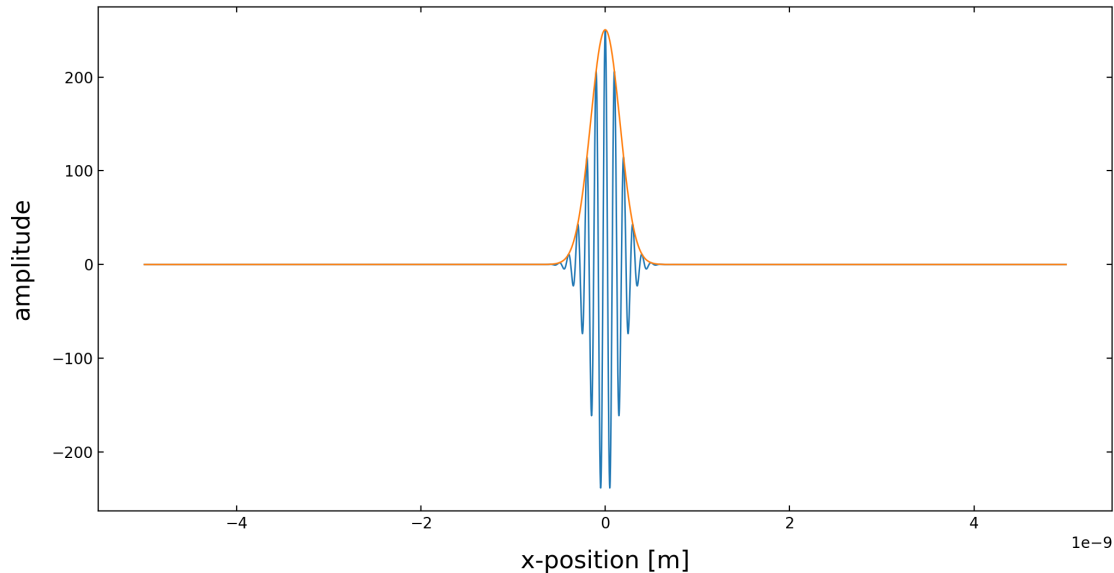
The wavepacket above had just constant amplitudes over a range of wavenumbers. While we have an analytical function for this type of wave, we just program a function to show that the superposition works as well.

The plot below now shows the magnitude square of the wavefunction, i.e. $|\Psi|^2$. You may recognize the side lobes besides the main peak. The whole distribution is reminiscent of the diffraction pattern of a single slit in the far field as tackled also in our lectures on electromagnetic waves. You may want to think about this similarity a bit longer.



1.2.4 Gaussian Wave Packet

We next create a Gaussian wave packet, which means that the amplitudes $\phi(k)$ of the interfering plane waves follow a Gaussian distribution. The result is then also a Gaussian function for the spatial distribution.



1.3 Time evolution of a Gaussian Wavepacket

We would like to plot in the next sections the time evolution of the wavefunction. Of course, we want to see the function developing live over time in a nice diagram. We could use the `matplotlib animate` method, which is commonly used to create a movie file that is later embedded in the notebook. Here we want to take a different route. We use `matplotlib` to generate the plot. We store the image data for the plot in a numpy array, which we then show with the help of our `ipycanvas` module. When calculations are not taking too long, this is a good way to animate in real time.

```
Canvas(height=300, width=800)
```

The next lines do the magic here. You see that there is essentially a normal `matplotlib` plotting in the first lines. The special lines are

```
plt.draw()
background = fig.canvas.copy_from_bbox(ax.bbox)
points=ax.plot(x*1e9,np.abs(gauss_wave_packet(x,t))**2)[0]
plt.close()
```

which draws the diagram without the data into the buffer. We store that buffer as the background with `fig.canvas.copy_from_bbox(ax.bbox)`, because it is everytime the same. The third line obtains the data point list as a `matplotlib` object. This list can be later updated with new data for the animation. Finally, we close the figure with `plt.close()` to prevent extra drawing below the cell.

The animation is then done by a for loop through time calculating the new wavefunction data and supplying it to the `points` object as new data. In detail it looks this way.