In this module, we will solve the problem of a particle confined to a one-dimensional box with an arbitrary potential within the box in the time domain. To do this, we will solve the *time-dependent* Schroedinger's equation:

$$\mathcal{H}\psi(x,t) = i\hbar\frac{\partial\psi(x,t)}{\partial t} \tag{1}$$

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

This equation may not have been covered yet in the paradigms, but they'll catch up by the end of the term.

0.1 Integerating Schroedinger's equation

Schroedinger's equation isn't suitable for Verlet's method, since it is only a *first-order* differential equation with respect to time. In addition, it is complicated by the fact that it is inherently complex. If the wavefunction starts out real, then it will immediately gain an imaginary part. We will integrate this equation using an approach similar to the *Yee lattice* that is used in the Finite Difference Time Domain (FDTD) method in photonics.

The key is that we always want to perform *centered derivatives*. Provided all our derivatives are centered, things will be okay.

We begin by writing the complex differential equation as two coupled real equations:

$$\psi(x,t) = \psi_R(x,t) + i\psi_I(x,t) \tag{3}$$

$$-\frac{\hbar^2}{2m}\frac{\partial^2\psi_R(x,t)}{\partial x^2} + V(x)\psi_R(x,t) = -\hbar\frac{\partial\psi_I(x,t)}{\partial t} \tag{4}$$

$$-\frac{\hbar^2}{2m}\frac{\partial^2\psi_I(x,t)}{\partial x^2} + V(x)\psi_I(x,t) = \hbar\frac{\partial\psi_R(x,t)}{\partial t} \tag{5}$$

Now we need to rewrite the derivatives as centered finite differences. Let's begin with the first time derivative.

$$-\frac{\hbar^2}{2m}\frac{\partial^2\psi_R(x,t)}{\partial x^2} + V(x)\psi_R(x,t) \approx -\hbar\frac{\psi_I(x,t+\frac{\Delta t}{2}) - \psi_I(x,t-\frac{\Delta t}{2})}{\Delta t} \tag{6}$$

At this point, we notice that we need to know ψ_I at *half-integer* time steps. Of course, we could have divided by $2\Delta t$, but this approach is more conventional. So now we rewrite the imaginary part using a centered finite difference equation, keeping in mind that we need to know its value at half-integer timesteps.

$$-\frac{\hbar^2}{2m}\frac{\partial^2\psi_I(x,t+\frac{\Delta t}{2})}{\partial x^2} + V(x)\psi_I(x,t+\frac{\Delta t}{2}) \approx \hbar\frac{\psi_R(x,t+\Delta t) - \psi_R(x,t)}{\Delta t} \tag{7}$$

So now we see that things are looking good: both equations involve the real part at integral timesteps and the imaginary part at half-integer timesteps. We now just need to convert the spatial derivatives into finite differences, and then we'll be almost done.

$$-\frac{\hbar^2}{2m}\frac{\psi_R(x+\Delta x,t)+\psi_R(x-\Delta x,t)-2\psi_R(x,t)}{\Delta x^2}+V(x)\psi_R(x,t)\approx-\hbar\frac{\psi_I(x,t+\frac{\Delta t}{2})-\psi_I(x,t-\frac{\Delta t}{2})}{\Delta t} \tag{8}$$

$$-\frac{\hbar^2}{2m}\frac{\psi_I(x+\Delta x,t+\frac{\Delta t}{2})+\psi_I(x-\Delta x,t+\frac{\Delta t}{2})-2\psi_I(x,t+\frac{\Delta t}{2})}{\Delta x^2}+V(x)\psi_I(x,t+\tfrac{\Delta t}{2})\approx\hbar\frac{\psi_R(x,t+\Delta t)-\psi_R(x,t)}{\Delta t} \tag{9}$$

At this point, we can see the light at the end of the tunnel. If we know the imaginary part at time $t-\frac{\Delta t}{2}$ and the real part at time t, then we can solve the first equation to find the imaginary part at time $t+\frac{\Delta t}{2}$. Similarly, if we know the real part at time t and the imaginary part at time $t+\frac{\Delta t}{2}$, we can solve for the real part at time $t+\Delta t$.

You just need to solve for the "future" wave function for each of the above equations.

This gives us:

$$\psi_I\left(x,t+\tfrac{\Delta t}{2}\right)=\psi_I\left(x,t-\tfrac{\Delta t}{2}\right)+\frac{\hbar\Delta t}{2m}\left(\frac{\psi_R(x+\Delta x,t)+\psi_R(x-\Delta x,t)-2\psi_R(x,t)}{\Delta x^2}\right)-\frac{\Delta t}{\hbar}V(x)\psi_R(x,t) \tag{10}$$

$$\psi_R\left(x,t+\Delta t\right)=\psi_R(x,t)-\frac{\hbar\Delta t}{2m}\frac{\psi_I(x+\Delta x,t+\frac{\Delta t}{2})+\psi_I(x-\Delta x,t+\frac{\Delta t}{2})-2\psi_I(x,t+\frac{\Delta t}{2})}{\Delta x^2}+\frac{\Delta t}{\hbar}V(x)\psi_I\left(x,t+\tfrac{\Delta t}{2}\right) \tag{11}$$

This means we can represent $\psi_R(x,t)$ and $\psi_I(x,t)$ as

$$\psi_R(x,t) \doteq \begin{pmatrix} \psi_R(0,0) & \psi_r(\Delta x,0) & \psi_R(2\Delta x,0) & \cdots \\ \psi_R(0,\Delta t) & \psi_r(\Delta x,\Delta t) & \psi_R(2\Delta x,\Delta t) & \cdots \\ \psi_R(0,2\Delta t) & \psi_r(\Delta x,2\Delta t) & \psi_R(2\Delta x,2\Delta t) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \tag{12}$$

$$\psi_I(x,t) \doteq \begin{pmatrix} \psi_I(0,-\frac{1}{2}\Delta t) & \psi_I(\Delta x,-\frac{1}{2}\Delta t) & \psi_I(2\Delta x,-\frac{1}{2}\Delta t) & \cdots \\ \psi_I(0,\frac{1}{2}\Delta t) & \psi_I(\Delta x,\frac{1}{2}\Delta t) & \psi_I(2\Delta x,\frac{1}{2}\Delta t) & \cdots \\ \psi_I(0,1.5\Delta t) & \psi_I(\Delta x,1.5\Delta t) & \psi_I(2\Delta x,1.5\Delta t) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \tag{13}$$

0.2 Creating a wave packet

We often like to work with plane waves:

$$\psi(\mathbf{r},t)=e^{i(\mathbf{k}\cdot\mathbf{r}-\omega t)} \tag{14}$$

but for many purposes they aren't really very convenient. A plane wave describes a particle with a precisely known momentum $\hbar\mathbf{k}$, but that is equally likely to be anywhere in space. At the other extreme, we could work with the eigenstate of position:

$$\psi(x,t=0)=\delta(x) \tag{15}$$

which describes a particle that is precisely known to be located at the origin. This has a couple problems associated with it. One is that we can't very effectively represent a delta function on a grid

with finite resolution. Another is that because of the uncertainty principle, a particle whose position is known so precisely could have any momentum, which means it won't stay well-known for very long.

A standard approach for dealing with these issues is to construct a *wave packet*. A wave packet is a compromise, allowing us to describe a particle with a reasonably well-known position and a reasonably well-known momentum. We generally use *Gaussian* wave packets, which provide the best compromise in terms of total certainty, and correspond to a product of a Gaussian with a plane wave:

$$\psi(x, t = 0) \approx e^{-\frac{(x-x_0)^2}{2\sigma^2}} e^{ikx} \tag{16}$$

This describes a particle that is located at position x_0 and is moving with momentum $\hbar k$.

1. Write a program that uses a function $V(x)$ and solves for the wave function as a function of time, given initial values for $\psi_I(x, -\Delta t/2)$ and $\psi_R(x, 0)$. To begin with, use $V(x) = 0$ to study the free particle in a box. **Use separate 2D arrays to store the real and imaginary parts of $\psi(x, t)$.**

2. Construct a wave packet and observe its time dependence. You may choose to either use a static visualization or an animation to do so… and eventually both.

3. Given that the particle described by this wave packet has momentum $\hbar k$, what should the velocity and energy of the particle be? (This is a paper and pencil portion.) With this velocity determine how many times your particle should traverse your box in the time you are simulating.

   Extra paper fun How certain should you be about the velocity and energy?

4. Create a static visualization of your wave function. I suggest visualizing just the real or imaginary part by itself. There are other approaches, but this simple approach is quite effective.

5. Confirm numerically from your static visualization that your particle is traveling with the expected velocity.

6. Try seeing what happens when you modify the wave packet in different ways. You can try changing the packet width, the momentum, and you can try using the complex conjugate or the real or imaginay part of a wave packet. This is a chance to have fun! (And even better, it's *required* rather than optional *extra* fun!) To start with, you may leave the potential $V(x)$ as zero, so you are looking at a simple particle confined to a box.

Extra potential barrier fun Now that you've got a wave packet, try bouncing it off of something. Create a potential barrier, and see what happens when your particle hits it. It will help to find the kinetic energy of your wave packet and compare that with the height of your barrier.

Extra fun Try other potentials. See what you can make your packet do! Also, try to explain what you see it doing.

0.3 Saving and loading your file

You may have noticed that your code takes a while to solve Schrödinger's equation. And it can be frustrating to have to rerun that whole calculation just to tweak your visualization. One solution to this would be to make your code faster, but in python that isn't so simple. Another solution is to store the wave function data that you computed in a file or two, so you can just reread it each time you want to attempt a modified visualization.

1. Create two new python programs. One should compute ψ_R and ψ_I arrays and then save them files (or possibly a single file). The other program will read those files and perform the same visualizations that you've been working on. **Note that you should be able to copy and paste most of the code from your existing simulation.**

0.4 The classical limit

To understand your simulations, we'll need to talk about the *uncertainty principle,* which you should be covering in class this week. The uncertainty principle (or one of them, anyhow) states that

$$\Delta p \Delta x \geq \frac{1}{2}\hbar \tag{17}$$

where Δp and Δx are the uncertainties in momentum and position, respectively, **which is different from the Δx that we use in our simulation**. This arises from the fact that in quantum mechanics a particle is represented as a wave, and you'll learn much more about the *why* of this in Periodic Systems, particularly in terms of Fourier transforms into momentum space. For this class, we just need the qualitative sense. The position of our wave packet is approximately x_0, with an uncertainty of σ. This just comes from the form of the Gaussian. The momentum of our wave packet is $\hbar k$, and to find its uncertainty you will want to make use of the uncertainty principle. The inequality is actually an equality for a Gaussian wave packet, which is one of the reasons why we used this form.

1. Work out the expected speed of your particle, and add to your static visualization a line that tracks the expected trajectory $x(t)$ of your particle. You can just use `plt.plot` as usual for this line. Verify that your wave packet is tracking the line.

2. Work out the uncertainty in the speed, and plot a couple more lines that track the expected trajectory $x(t)$ of your particle if the initial momentum were Δp higher or lower than $\hbar k$.

3. Play with different values of k and σ to see that your predictions seem to work properly.

4. Make the potential $V(x)$ have some functional form for which you know how to solve for the classically expected trajectory $x(t)$. Plot the classically expected trajectory and compare this with your quantum simulation.

Extra fun Try another potential. But save a copy of the code for each potential you create, so we can do presentations later.

0.4.1 Numerical stability

Video of short lecture on numerical instability

An algorithm is numerically unstable if small errors grow exponentially. In the particular case of our finite difference integration of Schroedinger's equation, our numerical stability is determined by the relationship between the resolution in space and time, Δx and Δt. To understand numerical stability physically, it is often helpful to consider the dimensions and behavior in the relevant dimensions.

In this case, it is useful to ask how quickly the wave function is going to change, and then to ensure that our Δt is considerably smaller than that value. The Schroedinger equation governs how quickly the wave function will change:

$$-\frac{\hbar^2}{2m}\frac{\partial^2\psi(x,t)}{\partial x^2} + V(x)\psi(x,t) = -i\hbar\frac{\partial\psi(x,t)}{\partial t} \tag{18}$$

We can now ask ourselves how quickly might the wave function be expected to change. The maximum magnitude of the second derivative $\frac{\partial^2\psi}{\partial x^2}$ will be approximately $\frac{\psi}{\Delta x^2}$, so we can write that

$$\Delta\psi \approx \left(\mp\frac{\hbar^2}{2m}\frac{\psi}{\Delta x^2} + V\psi\right)\Delta t \tag{19}$$

so the maximum possible fractional change in ψ over one time step is given by

$$\left|\frac{\Delta\psi}{\psi}\right|_{max} \approx \left(\frac{\hbar^2}{2m}\frac{1}{\Delta x^2} + |V|_{max}\right)\Delta t \tag{20}$$

We expect that there will be problems if in one time step the wave function changes by anything close to 100%, so we have an upper limit on Δt.

$$\Delta t < \frac{1}{\frac{\hbar^2}{2m}\frac{1}{\Delta x^2} + V_{max}} \tag{21}$$

This gives us an approximate value for the largest possible Δt. If we use too large a value for Δt, the most likely result is that our wave function will develop huge oscillations that grow exponentially. But we do want to pick a value for Δt that is as large as possible, so as to enable our simulation to run quickly.