Explicit Derivation of the Transmission Coefficient in Quantum Tunneling

1 Understanding Quantum Motion

The Schrödinger equation describes the evolution of the wave function Ψ:

$$\hat{H}\Psi(r,t) = i\hbar\frac{\partial}{\partial t}\Psi(r,t) \tag{1}$$

where $\hat{H}$ is the Hamiltonian operator, representing the total energy.

In classical mechanics, the Hamiltonian is given by:

$$H = \frac{p^2}{2m} + V(r) \tag{2}$$

In quantum mechanics, we replace momentum p with its operator form:

$$\hat{p} = -i\hbar\nabla \tag{3}$$

which leads to the quantum Hamiltonian:

$$\hat{H} = -\frac{\hbar^2}{2m}\nabla^2 + V(r) \tag{4}$$

For time-independent states, we assume:

$$\Psi(r,t) = \psi(r)e^{-iEt/\hbar} \tag{5}$$

Substituting this into the Schrödinger equation, we obtain:

$$\left[-\frac{\hbar^2}{2m}\nabla^2 + V(r)\right]\psi(r) = E\psi(r) \tag{6}$$

2 Quantum Tunneling and the Potential Barrier

Consider a potential barrier:

$$V(x) = \begin{cases} 0, & x < 0 \text{ or } x > a \\ V_0, & 0 \le x \le a \end{cases} \tag{7}$$

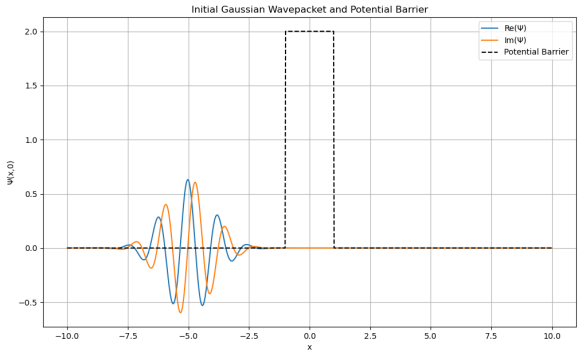

Figure 1: Example of our situation, with a Gaussian Wavepacket

A particle with energy E encounters this barrier. The wave function solutions in the three regions are:

Region I ($x < 0$):

$$\psi_I(x) = A_r e^{ikx} + A_l e^{-ikx}, \quad k = \sqrt{\frac{2mE}{\hbar^2}} \tag{8}$$

Region II ($0 \leq x \leq a$):

$$\psi_{II}(x) = B_r e^{iqx} + B_l e^{-iqx}, \quad q = \sqrt{\frac{2m(V_0 - E)}{\hbar^2}} \tag{9}$$

If $E < V_0$, q is imaginary, leading to an exponentially decaying function.

Region III ($x > a$):

$$\psi_{III}(x) = C_r e^{ikx} \tag{10}$$

3 Solving for the Transmission Coefficient t

The boundary conditions at $x = 0$ are:

$$1 + r = B_r + B_l \tag{11}$$

$$ik(1 - r) = iq(B_r - B_l) \tag{12}$$

At $x = a$:

$$B_r e^{iqa} + B_l e^{-iqa} = te^{ika} \tag{13}$$

$$iq(B_r e^{iqa} - B_l e^{-iqa}) = ikte^{ika} \tag{14}$$

Solving for B_r and B_l:

$$B_r = (1 + r) - B_l \tag{15}$$

$$ik(1 - r) = iq\left[(1 + r) - 2B_l\right] \tag{16}$$

$$B_l = -\frac{k}{2q}(1 - r) + \frac{1}{2}(1 + r) \tag{17}$$

Similarly,

$$B_r = \frac{k}{2q}(1 - r) + \frac{1}{2}(1 + r) \tag{18}$$

Substituting into the equation at $x = a$:

$$(k(1 - r) + q(1 + r))e^{iqa} = (q + k)te^{ika} \tag{19}$$

Solving for t:

$$t = \frac{4kqe^{i(q-k)a}}{(q + k)^2 - (q - k)^2e^{2iqa}} \tag{20}$$

4 Finding the Transmission Probability T

The probability is given by $T = |t|^2$:

$$T = \frac{16k^2q^2e^{-2qa}}{(q + k)^4 + (q - k)^4e^{-4qa} - 2(q^2 - k^2)^2e^{-2qa}} \tag{21}$$

Using the identity $1 - e^{-2qa} = \sinh^2(qa)$, we express it in a simpler form:

$$T = \frac{1}{1 + \left(\frac{V_0^2 \sinh^2(qa)}{4E(V_0 - E)}\right)} \tag{22}$$

5 Crank-Nicholsen Derivation

The objective is to model the behavior of a localized Gaussian wave packet as it approaches and interacts with the barrier.

Detailed Derivation and Algorithm:

Import Necessary Libraries:

The simulation utilizes Python libraries for numerical computations and visualization:

- `matplotlib` for plotting and animations.
- `scipy` for numerical operations, including sparse matrix representations and linear algebra.

Define the Wave_Packet Class:

This class encapsulates the properties and methods required to simulate the wave packet's evolution.

Initialize Parameters:

The class constructor (`__init__` method) initializes several parameters:

- `n_points`: Number of spatial grid points.
- `dt`: Time step for the simulation.
- `sigma0`: Initial width of the Gaussian wave packet.
- `k0`: Initial wave number, related to the particle's momentum.
- `x0`: Initial position of the wave packet center.
- `x_begin` and `x_end`: Boundaries of the spatial domain.
- `barrier_height` and `barrier_width`: Parameters defining the potential barrier.

Discretize the Spatial Domain:

The spatial domain is discretized into a uniform grid:

- `self.x`: Array of spatial positions.
- `self.dx`: Spatial step size, calculated as the difference between consecutive points.

Initialize the Wave Function:

The initial wave function is modeled as a Gaussian wave packet, which is a common choice due to its well-defined position and momentum characteristics.

Normalization Factor:

To ensure the total probability density integrates to one, the normalization constant is:

$$\text{norm} = \left(2\pi\sigma_0^2\right)^{-1/4}$$

Gaussian Envelope:

This term defines the spatial localization of the wave packet, centered at x_0 with a width σ_0:

$$\psi(x) = \exp\left(-\frac{(x-x_0)^2}{4\sigma_0^2}\right)$$

Plane Wave Component:

To incorporate momentum into the wave packet, we multiply by a complex exponential with wave number k_0:

$$\psi(x) = \psi(x) \times \exp(ik_0 x)$$

Combined Initial Wave Function:

Combining the above components and applying the normalization factor, we obtain:

$$\psi(x) = \left(2\pi\sigma_0^2\right)^{-1/4} \exp\left(-\frac{(x-x_0)^2}{4\sigma_0^2}\right) \exp(ik_0 x)$$

Set Up the Potential Barrier:

The potential $V(x)$ represents a barrier of height V_0 and width a, positioned between x_1 and x_2:

$$V(x) = \begin{cases} V_0 & \text{if } x_1 \leq x \leq x_2 \\ 0 & \text{otherwise} \end{cases}$$

This step function defines a region where the potential energy is elevated, simulating a barrier that the wave packet will encounter during its evolution.

Construct the Hamiltonian Matrix:

The Hamiltonian H governs the system's energy and is composed of kinetic and potential energy terms:

$$H = -\frac{1}{2}\frac{d^2}{dx^2} + V(x)$$

To numerically approximate H, we discretize the spatial domain into a grid with spacing Δx and apply the finite difference method to approximate the second derivative.

Finite Difference Approximation:

The second derivative of ψ at point x_i is approximated as:

$$\left.\frac{d^2\psi}{dx^2}\right|_{x_i} \approx \frac{\psi_{i+1} - 2\psi_i + \psi_{i-1}}{\Delta x^2}$$

Kinetic Energy Matrix (T):

This leads to a tridiagonal matrix representation for the kinetic energy term:

$$T = -\frac{1}{2}\frac{1}{\Delta x^2}\begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & -2 \end{bmatrix}$$

Potential Energy Matrix (V):

The potential energy is represented as a diagonal matrix with entries corresponding to $V(x_i)$:

$$V = \begin{bmatrix} V(x_1) & 0 & 0 & \cdots & 0 \\ 0 & V(x_2) & 0 & \cdots & 0 \\ 0 & 0 & V(x_3) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & V(x_n) \end{bmatrix}$$

Hamiltonian Matrix (H):

Combining the kinetic and potential energy matrices, we obtain:

$$H = T + V$$

Implement the Crank-Nicolson Method for Time Evolution:

The Crank-Nicolson method is an implicit finite difference scheme used to solve the time-dependent Schrödinger equation. It is unconditionally stable and conserves probability.

Crank-Nicolson Scheme:

The time evolution of the wave function ψ is given by:

$$\left(I + i\frac{\Delta t}{2}H\right)\psi(t + \Delta t) = \left(I - i\frac{\Delta t}{2}H\right)\psi(t)$$

where I is the identity matrix, Δt is the time step, and H is the Hamiltonian matrix.

Matrix Definitions:

- `A Matrix`: Combines the identity matrix with the Hamiltonian for the next time step:

$$A = I + i\frac{\Delta t}{2}H$$

Crank-Nicholson Method for Time Evolution (continued):

The Crank-Nicolson method is an implicit numerical scheme used to solve the time-dependent Schrödinger equation:

$$i\hbar\frac{\partial\Psi(x,t)}{\partial t} = H\Psi(x,t)$$

Using the Crank-Nicolson discretization, the time evolution equation becomes:

$$\left(I + i\frac{\Delta t}{2\hbar}H\right)\Psi(x,t+\Delta t) = \left(I - i\frac{\Delta t}{2\hbar}H\right)\Psi(x,t)$$

This equation can be rewritten in matrix form as:

$$A\Psi(t+\Delta t) = B\Psi(t)$$

where:

$$A = I + i\frac{\Delta t}{2\hbar}H$$

$$B = I - i\frac{\Delta t}{2\hbar}H$$

To advance the wave function in time, we solve:

$$\Psi(t+\Delta t) = A^{-1}B\Psi(t)$$

Since A is a tridiagonal matrix due to the finite-difference discretization of the Hamiltonian, we can efficiently solve for $\Psi(t+\Delta t)$ using numerical linear algebra techniques, such as LU decomposition.

Constructing the Tridiagonal Matrix Representation:

From the finite-difference approximation of the kinetic energy term:

$$\frac{d^2\Psi}{dx^2} \approx \frac{\Psi_{i+1} - 2\Psi_i + \Psi_{i-1}}{\Delta x^2}$$

we obtain the kinetic energy operator in matrix form:

$$T = -\frac{\hbar^2}{2m\Delta x^2}\begin{bmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & -2 \end{bmatrix}$$

This matrix, when combined with the potential matrix, provides the Hamiltonian for the system.

Implementing the Time Evolution in Code:

The algorithm progresses as follows:

1. Initialize the wave function $\Psi(x, 0)$ based on the initial conditions.
2. Define the potential barrier and the Hamiltonian matrix.
3. Apply the Crank-Nicolson method iteratively to compute the wave function at each time step.