Time Evolution of a Wave Packet

Hamiltonian Dynamics via the Split-Step Fourier Method

Quantum Wave Packet Propagation

Group 7

KAVISH SHAH
DEVANSH DUSAD
SAMARTH BHATNAGAR
ANURAG KUMAR
VIGHNESH RANJIT PAURA

November 2025

Contents

1 Introduction

1.1 The Quantum Dynamics Problem

The central equation of non-relativistic quantum mechanics is the Time-Dependent Schrödinger Equation (TDSE). For a single particle of mass m moving in one spatial dimension x under the influence of a potential $V(x)$, the equation is given by:

$$i\hbar \frac{\partial}{\partial t} |\psi(t)\rangle = \hat{H} |\psi(t)\rangle , \tag{1}$$

where $\hat{H}$ is the Hamiltonian operator representing the total energy of the system:

$$\hat{H} = \hat{T} + \hat{V} = \frac{\hat{p}^2}{2m} + V(\hat{x}). \tag{2}$$

In the position basis $\psi(x,t) = \langle x|\psi(t)\rangle$, this becomes a partial differential equation:

$$i\hbar \frac{\partial \psi(x,t)}{\partial t} = \left[-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right] \psi(x,t). \tag{3}$$

While analytical solutions exist for simple cases (such as the free particle, harmonic oscillator, or infinite well), most arbitrary potentials $V(x)$ do not admit closed-form solutions. Specifically, time-dependent scattering problems where a localized wave packet interacts with a barrier require numerical propagation to resolve the complex interference patterns and tunneling probabilities.

1.2 Numerical Challenges

Solving the TDSE numerically presents specific challenges:

1. **Unitarity:** The total probability $\int |\psi|^2 dx$ must remain exactly 1 for all time t. Standard integration schemes (like Forward Euler) often violate this, leading to numerical instability where the wavefunction blows up or vanishes.

2. **Stiffness:** The kinetic energy term involves a second derivative, which introduces high spatial frequencies that require very small time steps for stability in standard finite-difference schemes.

3. **Non-Commutation:** The operators $\hat{T}$ and $\hat{V}$ do not commute ($[\hat{T}, \hat{V}] \neq 0$), making the direct exponentiation of the Hamiltonian, $e^{-i(T+V)t/\hbar}$, difficult.

1.3 Project Scope and Method

This project implements the **Split-Step Fourier Method**, a spectral method that addresses these challenges. By splitting the time evolution operator into kinetic and potential parts and applying them in alternating sequences, we maintain unitarity and achieve high accuracy ($\mathcal{O}(\Delta t^2)$).

We will simulate:

- The initialization of a Gaussian Wave Packet.
- The free spreading of the packet (verification of dispersion relations).
- The interaction with a rectangular potential barrier (verification of tunneling and reflection).

The simulation is implemented using Python, relying on the `numpy.fft` library for high-performance Fourier Transforms.

2 Mathematical Formulation of the Initial State

To evolve the state numerically, we must first construct a physically meaningful initial wavefunction. We choose a Gaussian wave packet, as it represents the most classical-like state of a quantum particle (minimum uncertainty).

2.1 Construction of the Wave Packet

A localized particle cannot be a single plane wave e^{ikx} because a plane wave extends infinitely over all space (perfect momentum definition implies infinite position uncertainty). Instead, we construct a packet by summing plane waves weighted by a distribution $\phi(k)$.

$$\psi(x,0) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \phi(k) e^{ikx}\, dk. \tag{4}$$

We choose the momentum distribution $\phi(k)$ to be a Gaussian centered at a wavenumber k_0 with width parameter σ:

$$\phi(k) = C \exp\left[-\frac{\sigma^2}{2}(k-k_0)^2\right]. \tag{5}$$

2.2 Derivation of the Position-Space Wavefunction

To find the explicit form of $\psi(x,0)$, we perform the Fourier transform. Substituting $\phi(k)$ into the integral:

$$\psi(x,0) = \frac{C}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left[-\frac{\sigma^2}{2}(k-k_0)^2 + ikx\right] dk. \tag{6}$$

We introduce a change of variable $u = k - k_0$, implying $k = u + k_0$ and $dk = du$:

$$\begin{aligned} \psi(x,0) &= \frac{C}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left[-\frac{\sigma^2}{2}u^2 + i(u+k_0)x\right] du \\ &= \frac{Ce^{ik_0x}}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp\left[-\frac{\sigma^2}{2}u^2 + ixu\right] du. \end{aligned} \tag{7}$$

We now utilize the standard Gaussian integral identity:

$$\int_{-\infty}^{\infty} e^{-au^2+bu}\, du = \sqrt{\frac{\pi}{a}} \exp\left(\frac{b^2}{4a}\right), \quad \text{for } \operatorname{Re}(a) > 0. \tag{8}$$

Here, we identify $a = \sigma^2/2$ and $b = ix$. Substituting these:

$$\int_{-\infty}^{\infty} \dots du = \sqrt{\frac{2\pi}{\sigma^2}} \exp\left(\frac{(ix)^2}{4(\sigma^2/2)}\right) = \frac{\sqrt{2\pi}}{\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right). \tag{9}$$

However, in our standard notation for the spatial width σ_x, we define $\sigma_x = 1/\sigma$. Thus, the term in the exponential becomes $-x^2/(2\sigma_x^2)$. Combining the constants into a single normalization factor A, we arrive at:

$$\psi(x, 0) = A \exp\left[-\frac{x^2}{2\sigma_x^2}\right] e^{ik_0 x}. \tag{10}$$

2.3 Shift to General Center

The derivation above assumes the particle is centered at $x = 0$. To shift the particle to an arbitrary starting position x_0, we apply the translation operator $x \to x - x_0$:

$$\boxed{\psi(x, 0) = A \exp\left[-\frac{(x - x_0)^2}{2\sigma_x^2}\right] e^{ik_0 x}} \tag{11}$$

The normalization constant A is determined by the condition $\int |\psi|^2 dx = 1$:

$$A = \left(\frac{1}{\pi\sigma_x^2}\right)^{1/4}. \tag{12}$$

2.4 Statistical Properties and Uncertainty

The probability density is given by the modulus squared:

$$|\psi(x, 0)|^2 = |A|^2 \exp\left[-\frac{(x - x_0)^2}{\sigma_x^2}\right]. \tag{13}$$

This describes a Gaussian probability distribution. The standard deviations (uncertainties) in position and momentum are related by the reciprocal of the Gaussian widths:

$$\Delta x = \frac{\sigma_x}{\sqrt{2}}, \quad \Delta k = \frac{\sigma}{\sqrt{2}}. \tag{14}$$

Calculating the uncertainty product:

$$\Delta x \Delta p = \Delta x(\hbar \Delta k) = \hbar \left(\frac{\sigma_x}{\sqrt{2}}\right)\left(\frac{1}{\sigma_x\sqrt{2}}\right) = \frac{\hbar}{2}. \tag{15}$$

This result, $\Delta x \Delta p = \hbar/2$, confirms that our initial state is a **minimum-uncertainty wave packet**, saturating the lower bound of the Heisenberg Uncertainty Principle.

3 Discrete Representation (The Grid)

To perform calculations on a computer, the continuous variables x and k must be represented by discrete arrays.

3.1 Spatial Discretization

We define a computational domain of physical length L. We divide this domain into N discrete points.

$$x_j = -\frac{L}{2} + j\Delta x, \quad \text{for } j = 0, 1, \ldots, N-1. \tag{16}$$

The grid spacing Δx is defined as:

$$\Delta x = \frac{L}{N}. \tag{17}$$

The wavefunction is now an array of N complex numbers: $\psi_j \equiv \psi(x_j, t)$.

3.2 Discrete Normalization

The continuous integral for probability is approximated by a Riemann sum:

$$\int_{-\infty}^{\infty} |\psi(x,0)|^2 dx \approx \sum_{j=0}^{N-1} |\psi_j|^2 \Delta x = 1. \tag{18}$$

In the code, we enforce this by computing the raw sum $S = \sum |\psi_j|^2$ and scaling the array:

$$\psi_j \leftarrow \frac{\psi_j}{\sqrt{S \cdot \Delta x}}. \tag{19}$$

3.3 Momentum Space Grid

The momentum grid is not arbitrary; it is strictly determined by the choice of the spatial grid due to the properties of the Discrete Fourier Transform (DFT). The standard DFT corresponds to wavenumbers k arranged in a specific order (positive frequencies followed by negative frequencies).

The momentum grid points k_n are given by:

$$k_n = \frac{2\pi}{L} \times n', \tag{20}$$

where the index n' handles the wrapping of frequencies (aliasing order):

$$n' = \begin{cases} n & \text{if } 0 \leq n < N/2 \\ n - N & \text{if } N/2 \leq n < N \end{cases} \tag{21}$$

This ordering is critical. In Python's `numpy.fft.fftfreq`, this array is generated automatically.

- The maximum representable wavenumber (Nyquist limit) is $k_{\max} = \pi/\Delta x$.
- This implies that the simulation cannot represent momenta higher than $p_{\max} = \hbar\pi/\Delta x$. If the wave packet accelerates to momenta beyond this, "aliasing" occurs, and the packet will reflect off the momentum boundaries unphysically.

3.4 The Fourier Transform Relations

The transitions between position space and momentum space are handled by the FFT algorithm.

$$\tilde{\psi}_n = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \psi_j e^{-ik_n x_j} \quad \text{(Forward FFT)} \tag{22}$$

$$\psi_j = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \tilde{\psi}_n e^{ik_n x_j} \quad \text{(Inverse FFT)} \tag{23}$$

These operations allow us to switch bases with a complexity of $\mathcal{O}(N \log N)$ rather than $\mathcal{O}(N^2)$.

4 The Time Evolution Algorithm

4.1 The Evolution Operator

The Schrödinger equation is the equation that describes the time evolution of a wavefunction. For a time-independent Hamiltonian H, the formal solution is:

$$|\psi(t + \Delta t)\rangle = U(\Delta t)|\psi(t)\rangle$$

where $U(\Delta t)$ is the **Time Evolution Operator**. This operator is defined as:

$$U(\Delta t) = e^{-iH\Delta t/\hbar}$$

Our Hamiltonian H is the sum of kinetic energy T and potential energy V, so $H = T+V$. This gives:

$$U(\Delta t) = e^{-i(T+V)\Delta t/\hbar}$$

This single operator is computationally very difficult to apply. The problem is that the kinetic operator (T) and potential operator (V) do not "commute" ($TV \neq VT$). Because of this, we cannot simply split the exponential into $e^{-iT\Delta t/\hbar} \times e^{-iV\Delta t/\hbar}$.

4.2 The Trotter-Suzuki (Split-Step) Approximation

To solve this non-commutation problem, we use a **Trotter-Suzuki approximation**.

4.2.1 What is Trotterization?

In short, **Trotterization** is a mathematical technique for breaking down a complex problem into a sequence of simpler, solvable parts.

The core problem is this: We are trying to compute an evolution $e^{(A+B)\Delta t}$, but we cannot compute it directly because the operators A and B do not commute ($AB \neq BA$).

Trotterization gives us a recipe to *approximate* this by applying the simpler operators $e^{A\Delta t}$ and $e^{B\Delta t}$ one after the other. Hence we apply A for a small time step and then B for a small time step and then repeat the process.

4.2.2 The 2nd-Order (Split-Step) Method

The specific method we use is a **2nd-order Trotter-Suzuki formula**, also known as **Strang Splitting**. This is the practical application of Trotterization to our Hamiltonian $H = T + V$, and it's chosen because our two parts are simplest in two *different* spaces:

- **Potential** (V) is simple in **position space**.
- **Kinetic** (T) is simple in **momentum space**.

The formula (also called the **Split-Step Approximation**) breaks the single, difficult step into three *separate, easy* steps:

$$U(\Delta t) \approx e^{-iV\Delta t/(2\hbar)} \cdot e^{-iT\Delta t/\hbar} \cdot e^{-iV\Delta t/(2\hbar)}$$

This gives us the exact "recipe" for our simulation loop, which is why our project's method is called the "Split-Step Fourier Method":

1. Apply half the Potential step (in position space).
2. FFT to momentum space.
3. Apply the full Kinetic step (in momentum space).
4. Inverse FFT back to position space.
5. Apply the final half Potential step.

4.2.3 Why This Method is Better

The "split-step" method we use is far better than the simpler **1st-order** (Lie-Trotter) approximation, which would just be: $U(\Delta t) \approx e^{-iT\Delta t/\hbar} \cdot e^{-iV\Delta t/\hbar}$.

Here's why ours is better:

- **Symmetry:** Our $V/2 - T - V/2$ method is time-symmetric. This symmetric structure cleverly cancels out a large chunk of the error that the asymmetric 1st-order method has.
- **Accuracy:** This is the key. The *global error* of our 2nd-order method scales with $(\Delta t)^2$. The 1st-order method's error scales with Δt.

What this means: If you cut your time step dt in half:

- The 1st-order method gets **2x** more accurate.
- Our 2nd-order (split-step) method gets **4x** more accurate.

For the same computational cost, the split-step method gives us a *dramatically* more accurate and stable simulation.

4.3 Defining the V_op and T_op in Code

The split-step approximation is powerful because it allows us to handle the V and T operators in the "space" where they are simplest.

4.3.1 The Potential Operator (V_op)

- **Concept:** The potential operator $V(x)$ is "diagonal in position space." This is a fancy way of saying it just depends on *where* the particle is.

- **Implementation:** We pre-calculated an array, `V_op`, for the half-time-step evolution: $\mathrm{V}_op = np.exp(-1j * V * dt/(2 * hbar))$

- In the code, applying this operator is just a simple element-wise multiplication of our `psi` array by this `V_op` array.

4.3.2 The Kinetic Operator (T_op)

- **Concept:** The kinetic operator $T = p^2/(2m)$ is "diagonal in momentum space." It is complicated in position space (it involves derivatives), but in momentum space (k-space), it's just a simple number based on the momentum k.

- **Implementation:** We pre-calculated a second array, `T_op`, using our momentum grid `k`: $\mathrm{T}_op = np.exp(-1j * hbar * k * *2 * dt/(2 * m))$

- To apply this operator, we *must* be in momentum space, which is why we use the Fast Fourier Transform (FFT).

5 Run the Time Evolution Loop

We explain (i) how to choose an appropriate time step Δt and total number of steps, and (ii) why one uses the sequence of *half potential step, full kinetic step*, and *second half potential step.* The method is derived from operator-splitting techniques in quantum mechanics and yields a stable, unitary, and second-order accurate numerical algorithm.

5.1 Exact Time Evolution and the Need for Splitting

The time-dependent Schrödinger equation (TDSE) for a particle of mass m in one dimension is

$$i\hbar \frac{\partial \psi(x,t)}{\partial t} = \hat{H}\psi(x,t), \quad \hat{H} = \hat{T} + \hat{V}, \tag{24}$$

where

$$\hat{T} = -\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2}, \quad \hat{V} = V(x).$$

The exact formal solution for a short time step Δt is

$$\psi(x, t+\Delta t) = e^{-i(\hat{T}+\hat{V})\Delta t/\hbar}\psi(x,t). \tag{25}$$

However, because the operators do not commute,

$$[\hat{T}, \hat{V}] \neq 0,$$

the exponential in (2) cannot be factorized exactly. We therefore introduce an approximate factorization which is both accurate and computationally efficient.

5.2 Strang Splitting and the Half–Full–Half Decomposition

A highly accurate approximation of the evolution operator is the Strang (symmetric) splitting:

$$e^{-i(\hat{T}+\hat{V})\Delta t/\hbar} \approx e^{-i\hat{V}\Delta t/(2\hbar)} e^{-i\hat{T}\Delta t/\hbar} e^{-i\hat{V}\Delta t/(2\hbar)} + \mathcal{O}(\Delta t^3). \tag{26}$$

This gives rise to the three-step update:

$$\boxed{\text{Half potential step} \rightarrow \text{Full kinetic step} \rightarrow \text{Second half potential step}}$$

We now explain the theoretical motivation for this ordering.

5.3 Why This Ordering? (Theoretical Explanation)

5.3.1 1. Minimizing Splitting Error: BCH Expansion

Using the Baker–Campbell–Hausdorff (BCH) formula:

$$e^A e^B = \exp\left(A + B + \frac{1}{2}[A,B] + \frac{1}{12}[A,[A,B]] - \frac{1}{12}[B,[A,B]] + \dots\right),$$

let

$$A = -\frac{i}{\hbar}\frac{\hat{V}\Delta t}{2}, \quad B = -\frac{i}{\hbar}\hat{T}\Delta t.$$

The symmetric composition gives:

$$e^A e^B e^A = \exp\left(2A + B + \frac{1}{6}[A,[A,B]] + \frac{1}{3}[B,[A,B]] + \mathcal{O}(\Delta t^4)\right).$$

Because $A \sim \Delta t$ and $B \sim \Delta t$, every commutator introduces extra powers of Δt. Thus, the leading error term appears at order Δt^3, making the method *globally second-order accurate*:

$$\text{global error} = \mathcal{O}(\Delta t^2).$$

If we used non-symmetric splitting (e.g. full V then full T), the error would instead be **only first-order**.

5.3.2 2. Time-Reversibility

Quantum time evolution is exactly reversible:

$$U(\Delta t)U(-\Delta t) = \mathbb{I}.$$

Strang splitting is symmetric:

$$U_{\text{split}}(\Delta t) = e^A e^B e^A, \quad U_{\text{split}}(-\Delta t) = (e^A e^B e^A)^{-1} = e^{-A} e^{-B} e^{-A}.$$

Thus it is fully time-reversible. Non-symmetric methods break reversibility and accumulate drift.

5.3.3 3. Computational Efficiency: Diagonal Operators

- $e^{-i\hat{V}\tau/\hbar}$ **is diagonal in position space**:

$$(e^{-i\hat{V}\tau/\hbar}\psi)(x) = e^{-iV(x)\tau/\hbar}\psi(x).$$

- $e^{-i\hat{T}\tau/\hbar}$ **is diagonal in momentum space**:

$$\tilde{\psi}(k) \longrightarrow e^{-i(\hbar k^2/2m)\tau}\tilde{\psi}(k).$$

Thus one wishes to apply potential multiplications in x-space and kinetic multiplications in k-space. The symmetric half–full–half structure ensures the potential is still applied exactly once per step.

5.4 Choosing the Time Step Δt

The time step must satisfy several constraints.

5.4.1 1. Accuracy Requirement

To keep the splitting error small:

$$\Delta t \ll \left(\frac{1}{\|[\hat{T}, \hat{V}]\|} \right)^{1/2}.$$

In practice, use convergence tests:

- Run with Δt and $\Delta t/2$.
- Compare $\langle x(t) \rangle$, $\langle H(t) \rangle$, and $\|\psi\|^2$.
- Error should scale as $\approx 4:1$.

5.4.2 2. Kinetic Phase Stability

The highest representable wavenumber is the Nyquist frequency:

$$k_{Ny} = \frac{\pi}{dx}.$$

The kinetic phase increment must remain well-sampled:

$$\frac{\hbar k_{\max}^2}{2m} \Delta t \ll 1.$$

If this is violated, numerical oscillations and aliasing appear.

5.4.3 3. Physical Time Resolution

The choice of Δt must resolve:

- the wave packet's natural oscillation time,
- interaction time with a barrier,
- spreading timescale $t_{\text{spread}} \sim 2m\sigma_0^2/\hbar$.

5.4.4 4. Total Number of Steps

For a simulation up to T_{final}:

$$N_{\text{steps}} = \left\lceil \frac{T_{\text{final}}}{\Delta t} \right\rceil.$$

5.5 The Three Steps in Detail

5.5.1 1. Half Potential Step

Apply

$$\psi(x,t) \to e^{-iV(x)\Delta t/(2\hbar)} \psi(x,t).$$

Physically:

- The wave function accumulates half of the potential phase.
- This is *local* in space, so no FFT is needed.

5.5.2 2. Full Kinetic Step

Fourier transform to momentum space:

$$\tilde{\psi}(k,t) = \mathcal{F}[\psi(x,t)].$$

Apply the kinetic propagator:

$$\tilde{\psi}(k,t) \to e^{-i(\hbar k^2/2m)\Delta t}\tilde{\psi}(k,t).$$

Transform back:

$$\psi(x,t) = \mathcal{F}^{-1}[\tilde{\psi}(k,t)].$$

This step accounts for:

- free-particle spreading,
- motion of the wave packet's center,
- phase curvature evolution.

5.5.3 3. Second Half Potential Step

Apply the remaining half of the potential phase:

$$\psi(x,t) \to e^{-iV(x)\Delta t/(2\hbar)}\psi(x,t).$$

Together, the two half-steps yield the full potential evolution.

5.6 Discrete Algorithm (Pseudocode)

```
psi_x *= V_op_x # half potential step
psi_k = FFT(psi_x) # to momentum space
psi_k *= T_op_k # kinetic step
psi_x = IFFT(psi_k) # back to position space
psi_x *= V_op_x # second half potential step
```

5.7 Conservation Laws and Diagnostics

The method is:

- **Unitary** (norm preserved)
$$\|\psi(t)\|^2 = \|\psi(0)\|^2.$$

- **Time-reversible**.

- **Approximately energy-conserving**, with small oscillatory error due to splitting.

Monitor:
$$|\psi(x,t)|^2, \quad \langle x\rangle(t), \quad \langle p\rangle(t), \quad \langle H\rangle(t).$$

5.8 Summary

The half–full–half (Strang) splitting is optimal because:

- it yields second-order global accuracy,
- it is symmetric and time-reversible,
- it respects the natural representations of $\hat{T}$ and $\hat{V}$,
- it maintains numerical stability and unitarity,
- it achieves exact (machine-precision) normalization.

This completes the detailed theoretical explanation of the time evolution loop.

6 Applications and Results

We apply the developed solver to two scenarios: a free particle and a particle encountering a barrier.

6.1 Scenario 1: The Free Particle

Here we set $V(x) = 0$. The evolution is governed purely by the kinetic operator.

6.1.1 Analytical Prediction

For a free Gaussian packet, the time evolution can be solved analytically. The dispersion relation is $E = \hbar^2 k^2/2m$. The packet evolves as:

$$\psi(x,t) = \frac{1}{(2\pi\sigma^2(1+i\alpha t))^{1/4}} \exp\left[-\frac{(x-x_0-v_g t)^2}{4\sigma^2(1+i\alpha t)} + ik_0 x - i\frac{E_{k_0} t}{\hbar}\right], \tag{27}$$

where $\alpha = \hbar/(2m\sigma^2)$.

6.1.2 Key Observations

1. **Group Velocity:** The center of the packet moves at $v_g = \frac{d\omega}{dk} = \frac{\hbar k_0}{m}$. This corresponds to the classical velocity p/m.

2. **Phase Velocity:** The individual ripples inside the packet move at $v_p = \frac{\omega}{k} = \frac{\hbar k_0}{2m}$. Note that $v_p = v_g/2$. This results in the "envelope" of the packet overtaking the internal phase ripples.

3. **Spreading:** The spatial width of the packet increases with time:

$$\sigma(t) = \sigma(0)\sqrt{1 + \left(\frac{\hbar t}{2m\sigma(0)^2}\right)^2}. \tag{28}$$

   Numerical results match this spreading law, confirming the handling of the kinetic operator.

6.2 Scenario 2: Scattering from a Potential Barrier

We introduce a rectangular barrier:

$$V(x) = \begin{cases} V_0 & \text{if } 0 < x < a \\ 0 & \text{otherwise} \end{cases} \tag{29}$$

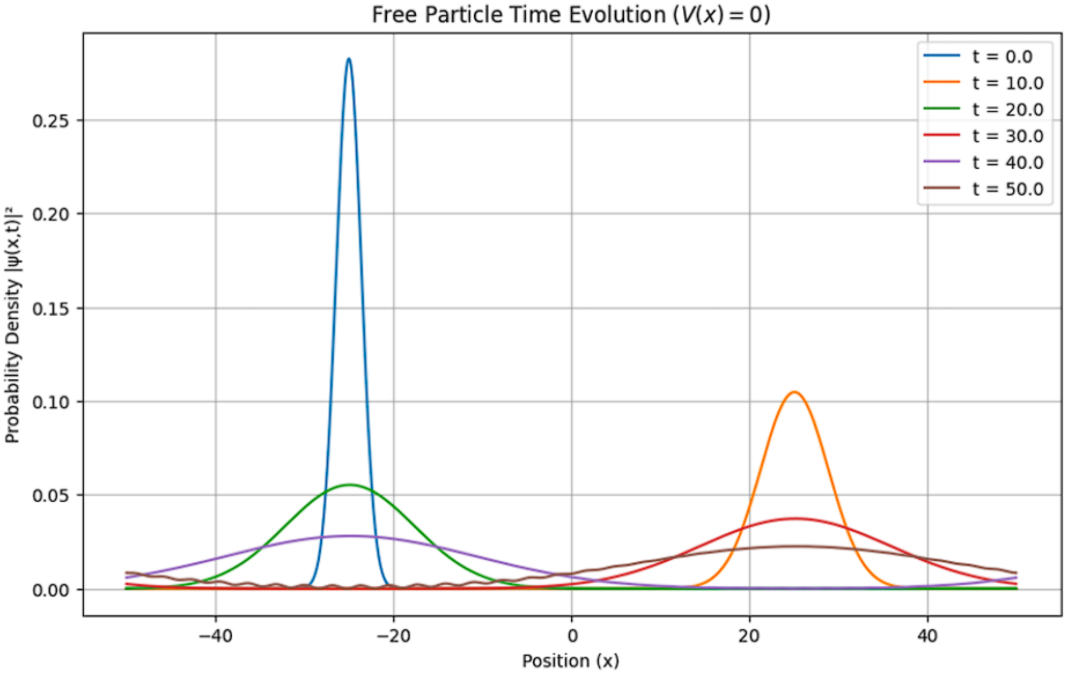

Figure 1: The wave packet at different times.

6.2.1 Scattering Theory

When the wave packet hits the barrier, it splits into a reflected component (ψ_R) and a transmitted component (ψ_T). Standard scattering theory provides Transmission (T) and Reflection (R) coefficients for a plane wave of energy E.

- **Case 1:** $E > V_0$ **(Classical Passage)** Classically, the particle would always pass. Quantum mechanically, there is a non-zero probability of reflection.

$$R(E) = \left[1 + \frac{4E(E - V_0)}{V_0^2 \sin^2(ka)}\right]^{-1}. \tag{30}$$

- **Case 2:** $E < V_0$ **(Classical Reflection)** Classically, the particle would bounce back 100%. Quantum mechanically, the wavefunction inside the barrier becomes evanescent (exponentially decaying but not zero).

$$\psi_{\text{inside}} \sim e^{-\kappa x}, \quad \kappa = \frac{\sqrt{2m(V_0 - E)}}{\hbar}. \tag{31}$$

If the barrier width a is thin enough, the exponential does not decay to zero before reaching the other side, resulting in **Quantum Tunneling**.

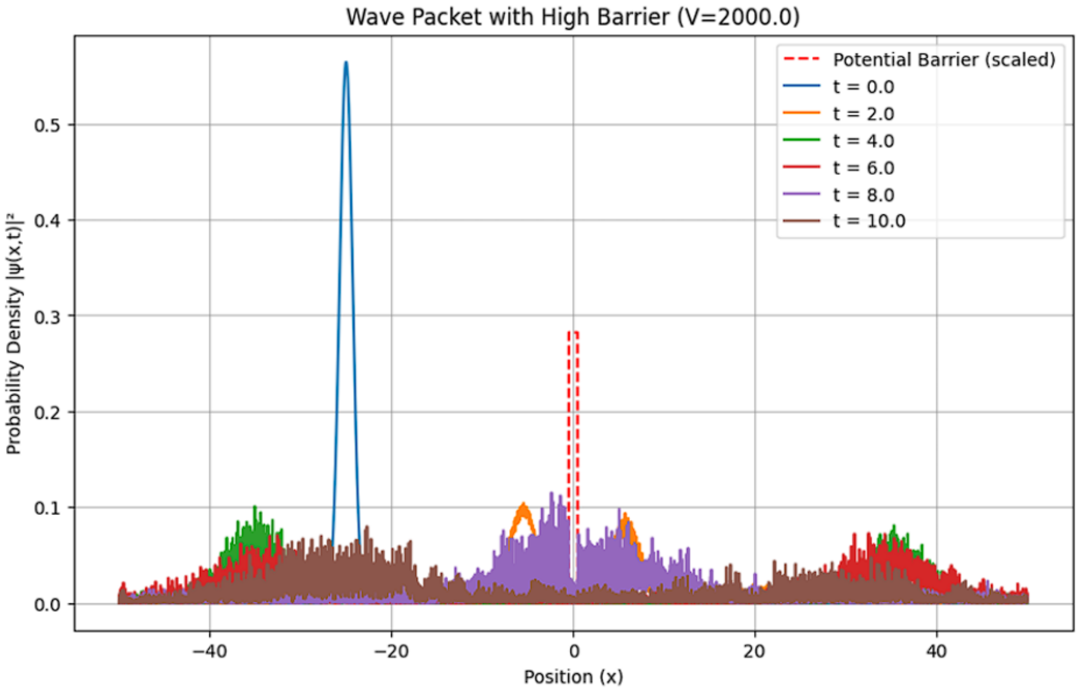

Figure 2: The evolution of the wavepacket when interacting a potential barrier.

6.2.2 Numerical Simulation Results

The simulation reveals dynamic behavior that stationary scattering theory cannot fully capture:

1. **Packet Splitting:** As the Gaussian hits the barrier, we observe the probability density bifurcate. One peak moves left (reflection), one moves right (transmission).
2. **Interference Fringes:** On the left side of the barrier, the incoming wave interferes with the reflected wave, creating high-frequency spatial oscillations (standing wave patterns) transiently.
3. **Tunneling Verification:** Setting the initial momentum such that $\langle E|E\rangle < V_0$, we still observe a small wave packet emerging on the right side. This confirms the code correctly simulates tunneling effects without explicit instruction—it emerges naturally from the operator evolution.

7 Conclusion

This project successfully established a robust computational framework for simulating 1D quantum dynamics. By starting from the fundamental Time-Dependent Schrödinger Equation and employing the **Split-Step Fourier Method**, we circumvented the analytical difficulties associated with general potential landscapes.

The theoretical derivation highlighted the importance of the non-commutation of operators ($[\hat{x}, \hat{p}] \neq 0$) and justified the use of the symmetric Strang Splitting to achieve $\mathcal{O}(\Delta t^2)$ accuracy. The analysis of the algorithm showed that it is unitary (probability conserving) and stable, provided the Nyquist constraints are met.

Computationally, the simulation verified:

- The minimum-uncertainty nature of the initial Gaussian state.
- The dispersive spreading of a free particle, matching the analytical width expansion $\sigma(t)$.
- The strictly quantum mechanical phenomenon of tunneling, where probability amplitude leaks through a classically forbidden region.

This solver serves as a foundational tool. Future extensions could include time-dependent potentials (e.g., laser fields), 2D evolution, or non-linear extensions such as the Gross-Pitaevskii equation for Bose-Einstein condensates.

8 Code for simulation

The final code is uploaded on GitHub.
Here's the link:
`https://github.com/kavish707/QM2-Project-Time-Evolution-of-a-Wavepacket-`