

User Guide: Expectation Values and Probability Distributions for Quantum Eigenfunctions

Overview

This document provides an explanation of how to work with the Python code that calculates expectation values of position and momentum for quantum eigenfunctions. The code supports three quantum systems:

1. Particle in a Box
2. Quantum Harmonic Oscillator
3. Symmetric Double-Well Potential

The script numerically evaluates and visualizes probability densities, position expectation values, and momentum expectation values.

Required Libraries

To run the code, the following Python libraries are required:

- numpy for numerical computations
- matplotlib for plotting graphs
- scipy.special for Hermite polynomials (used in harmonic oscillator and double-well eigenfunctions)

The required libraries can be installed using:

pip install numpy matplotlib scipy

Equations Used

1. Eigenfunctions for Different Quantum Systems

(a) Particle in a Box: Consider an infinite potential well through which quantum particle cannot penetrate through.

The normalized eigenfunction for an infinite potential well of width L is:

$$\psi_n(x) = \sqrt{\frac{2}{L}} \times \sin\left(\frac{n\pi x}{L}\right)$$

Where n is a positive integer that describe the quantum state of particle (i.e. $n = 1$ for ground state, and $n > 1$ for excited states), x is the position co-ordinate/operator.

For this implementation, L is assumed to be π , leading to:

$$\psi_n(x) = \sqrt{\frac{2}{L}} \times \sin(n x)$$

(b) Harmonic Oscillator

The quantum harmonic oscillator describes a particle subjected to a quadratic finite potential of the form:

$$V(x) = \left(\frac{1}{2}\right) m \omega^2 x^2$$

where m is the mass of the particle, and ω is the angular frequency. This potential leads to discrete energy levels and corresponding eigenfunctions expressed in terms of Hermite polynomials ($H_n(x)$):

$$\psi_n(x) = \left(\frac{1}{\sqrt{2^n n!} \sqrt{\pi}}\right) \times H_n(x) \times \exp\left(-\frac{x^2}{2}\right)$$

The eigenfunctions exhibit a Gaussian envelope multiplied by Hermite polynomials, resulting in oscillatory behavior that increases in complexity with higher quantum numbers.

(c) Symmetric Double-Well Potential (Approximation)

A double-well potential is characterized by two minima separated by a potential barrier. A common form of this potential is:

$$V(x) = x^4 - \alpha x^2$$

where α determines the depth of the wells. This potential is often used to model tunneling effects and quantum superposition states. For the double-well potential, an approximate eigenfunction is:

$$\psi_n(x) \approx \exp(-x^4) \times H_n(x)$$

This formulation ensures wavefunction localization near the potential wells, capturing the essential quantum behavior of a particle confined in a double-well system.

2. Expectation Values

(a) Position Expectation Value

The expectation value of position is calculated as:

$$\langle x \rangle = \int x |\psi_n(x)|^2 dx$$

(b) Momentum Expectation Value

Momentum expectation is calculated using Fourier transforms:

$$\langle p \rangle = \int k |\psi(k)|^2 dk$$

How to Use the Code

1. Run the script in a Python environment.
2. Enter the desired quantum number indices when prompted.
3. Choose the potential type ('particle_in_box', 'harmonic_oscillator', or 'double_well').
4. The script computes and displays:
 - Expectation values of position and momentum.
 - Probability density graph for $|\psi(x)|^2$.
 - Momentum probability distribution graph for $|\psi(k)|^2$.

Output and Visualization

The script generates:

- Probability Density Plot: Shows the likelihood of finding a particle at different positions.
- Momentum Distribution Plot: Represents the spread of momentum values for the quantum state.

These graphs help visualize quantum behavior in different potential systems.

Conclusion

This code provides a fundamental numerical approach to understanding quantum mechanical expectation values. By modifying parameters, different quantum states can be explored under varying conditions.