Physics-Informed Neural Networks to Solve the Many-Electron Time-Dependent Schrödinger Equation

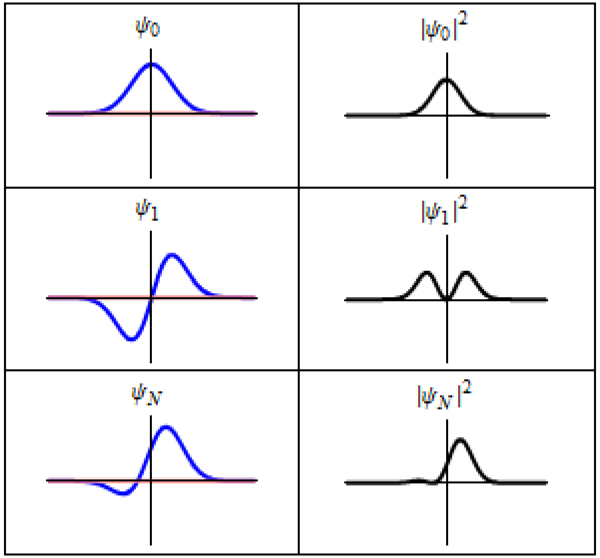
Afthash Sahal Ubaid Puzhakkal
a1913863

The Problem: Solving many electron TDSE

- The **Schrödinger Equation** describes how quantum systems (like electrons) evolve over time
- It predicts how the state of particles changes based on forces acting on them (like electric fields, potentials)
- The wavefunction, Ψ, contains all the measurable information about the particle at a given space and time, like energy, momentum, position, speed etc.
- For a single electron, solving the equation is manageable. But for many electrons interacting together, solving it becomes complex.

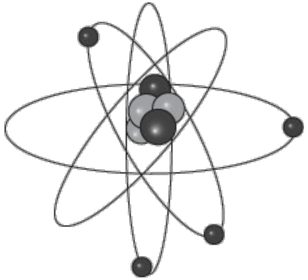
Research Question

- Can Physics-Informed Neural Networks (PINNs) accurately and efficiently solve the many-electron time-dependent Schrödinger equation?
- Can PINNs maintain accuracy over longer time?
- How to modify PINNs to handle electron electron interactions?



Research Approach

1. Single-electron time-independent one-dimensional model
2. Single-electron time-dependent one-dimensional models
    1. Moving quantum dot
    2. Double quantum dots and silicon charge qubits
3. Many electron models
    1. Quantum dot array



What Are Physics-Informed Neural Networks (PINNs)?

- PINNs are special neural networks designed to solve partial differential equations (PDEs)
- Instead of training on labeled examples, PINNs learn by making sure their predictions satisfy known physics laws
- The training goal: Minimize how much the neural network “breaks” the physical law measured through the loss function
- No labeled data is needed. Only the governing equation and an initial and boundary conditions are used.

PINNs vs Numerical Methods

Feature	PINNs	Analytical Methods
Grid-Free	Yes	No
Handles higher dimensions	Better	Poor (exponential scaling)
Accuracy	High (If well trained)	Limited to grid resolution
Computational Efficiency	Training takes time	Efficient for lower dimension
Generalisation	Yes	No

Loss Functions in PINN

- Physics Loss (L_{PDE}): Residual of the PDE we are trying to solve

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

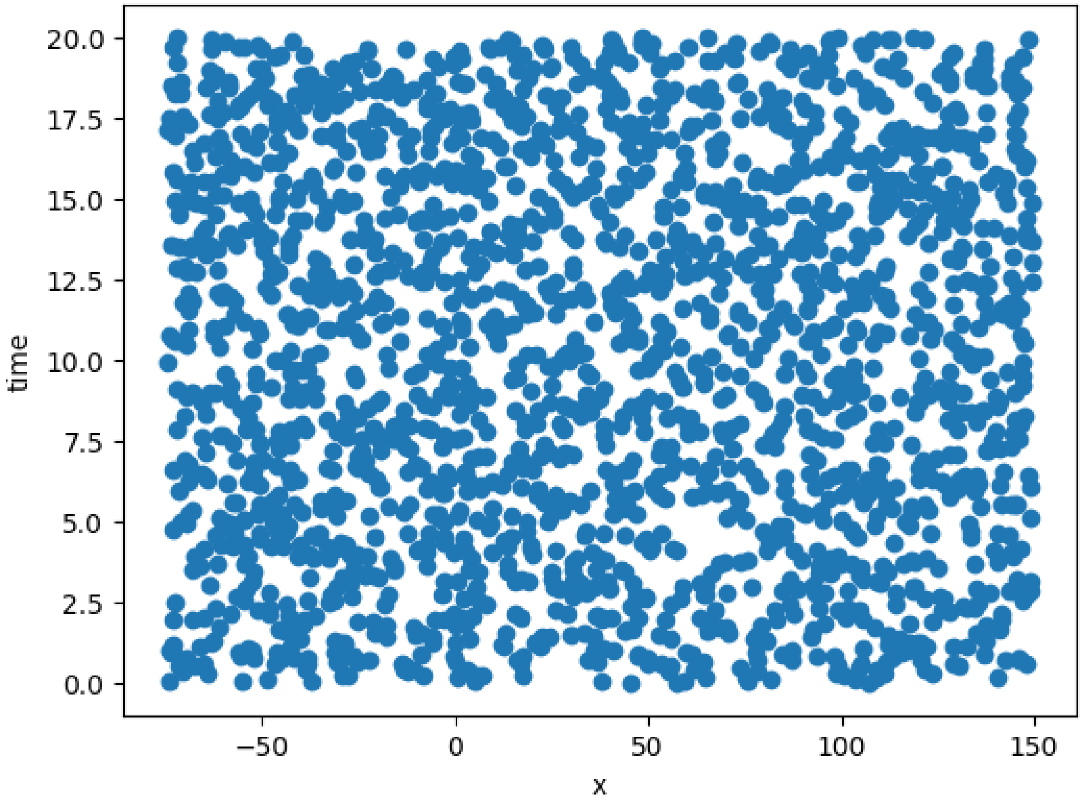
$$\mathcal{R} = i\hbar\frac{\partial\psi}{\partial t}+\frac{\hbar^2}{2m}\left(\frac{\partial^2\psi}{\partial x^2}+\frac{\partial^2\psi}{\partial z^2}\right)-V(x,z,t)\psi$$

- Initial condition loss ($L_{initial}$): Model will be forced to match the predicted Ψ at t=0 with a predefined analytical initial condition
- Boundary condition loss ($L_{boundary}$): Model will be forced to match the predicted wavefunction at spatial boundaries with a boundary condition value, zero in our case
- Total Loss = $L_{PDE} + L_{initial} + L_{boundary}$

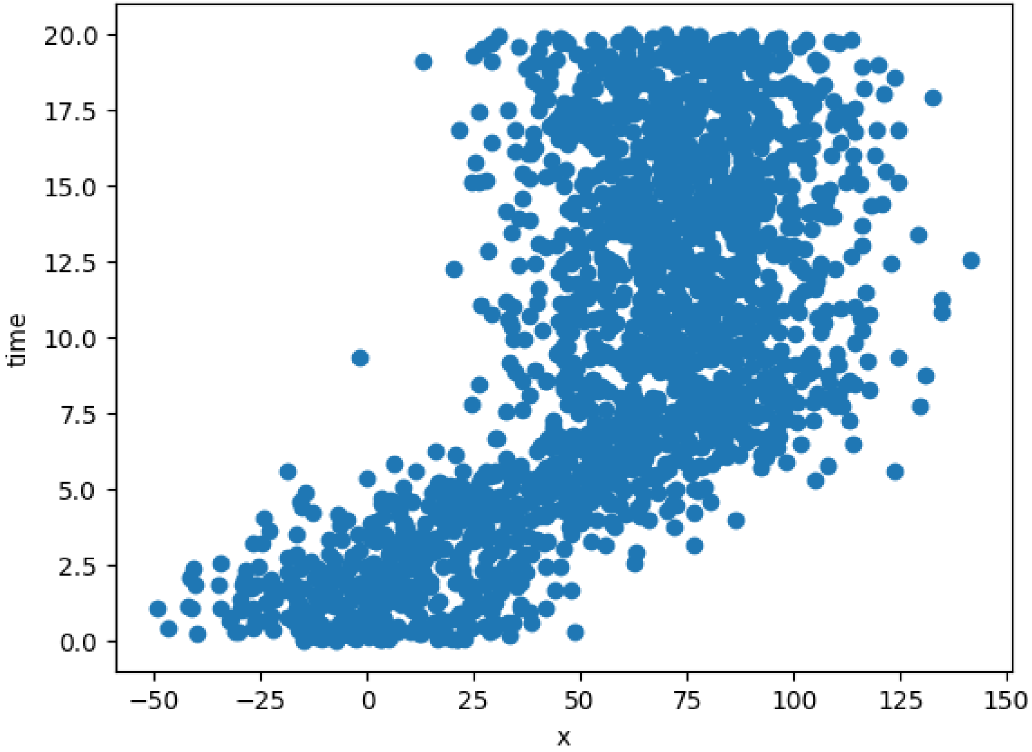
Training Data: Random Points

- Random points are sampled across the time and space domains
- Model will be trained to minimise all the loss functions based on these points
- Sampling method will be different for all three loss functions
- Random points generation could follow Uniform, Normal or other distribution depending upon the problem complexity

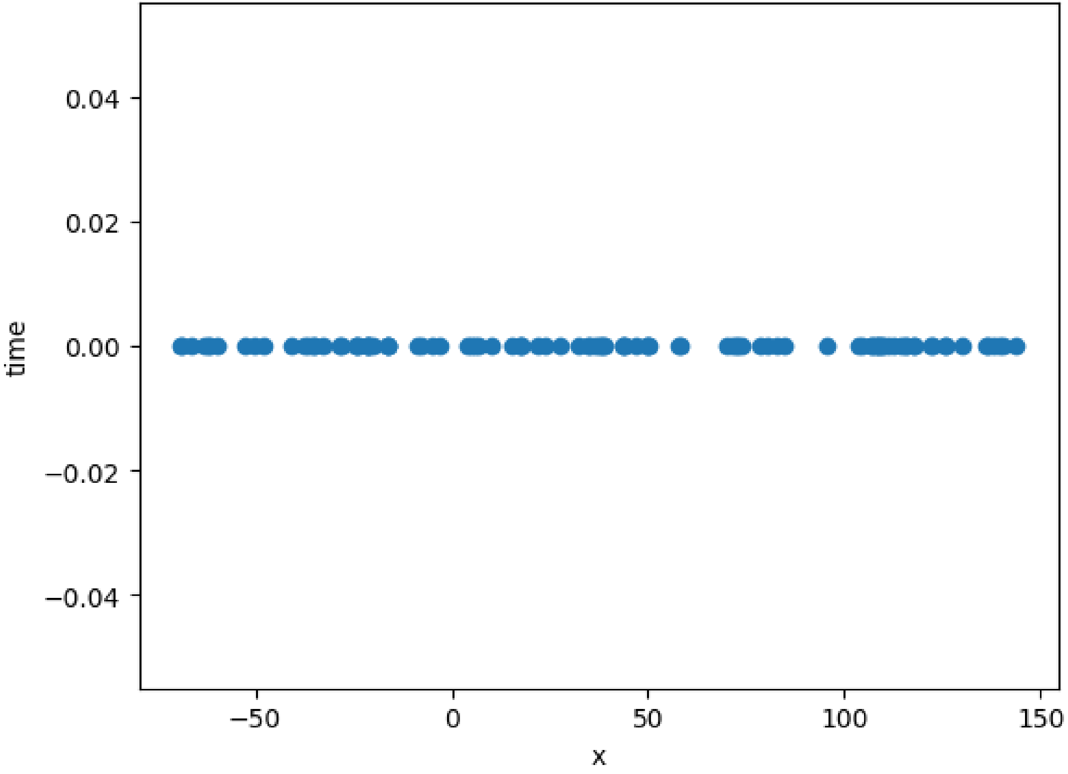
Physics Loss: Random Uniform Points



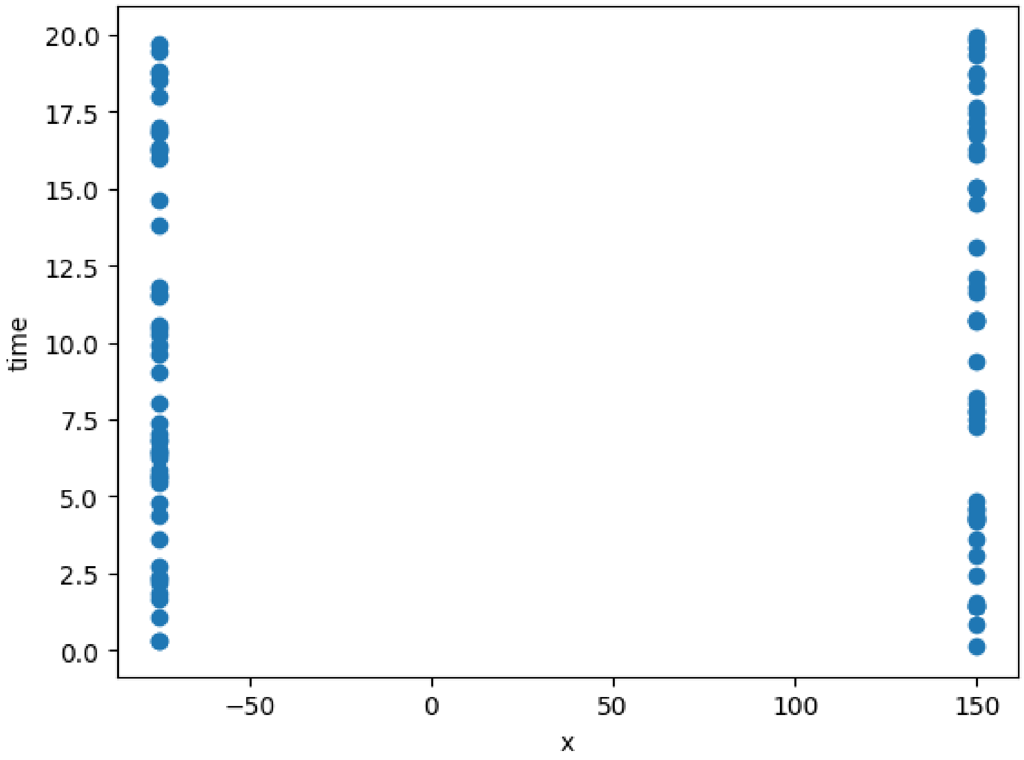
Physics Loss: Random Normal Points



Initial Condition Loss: Random Points



Boundary Condition Loss: Random Points



Testing Data: Analytical Solutions

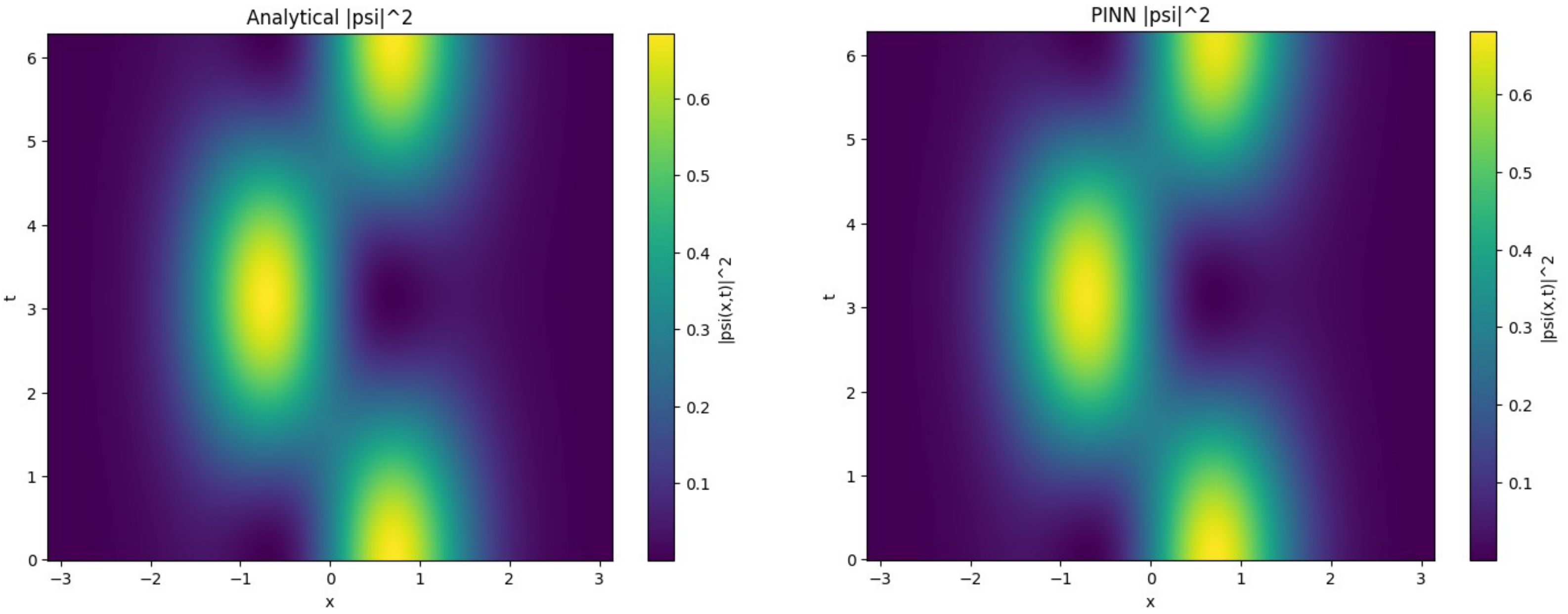
- Pre-existing analytical solution of the problem will be used to test the model performance
- Analytical solution will be only used for testing
- For more complex problems, analytical solution is either hard to solve or it does not exist
- Other method to test the model is to check if the wavefunction created maintain normalisation through out the time domain

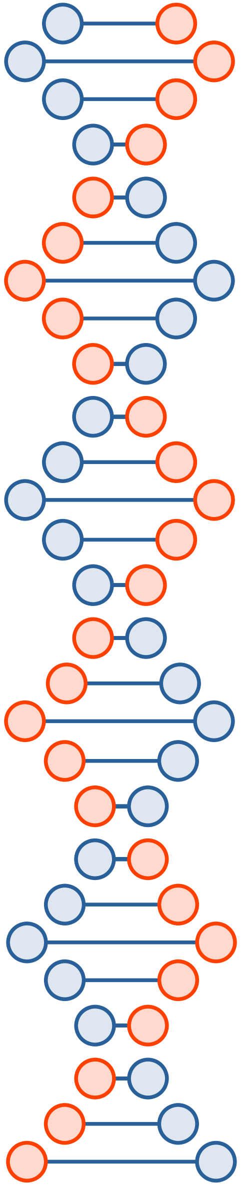
Progress on Model Development

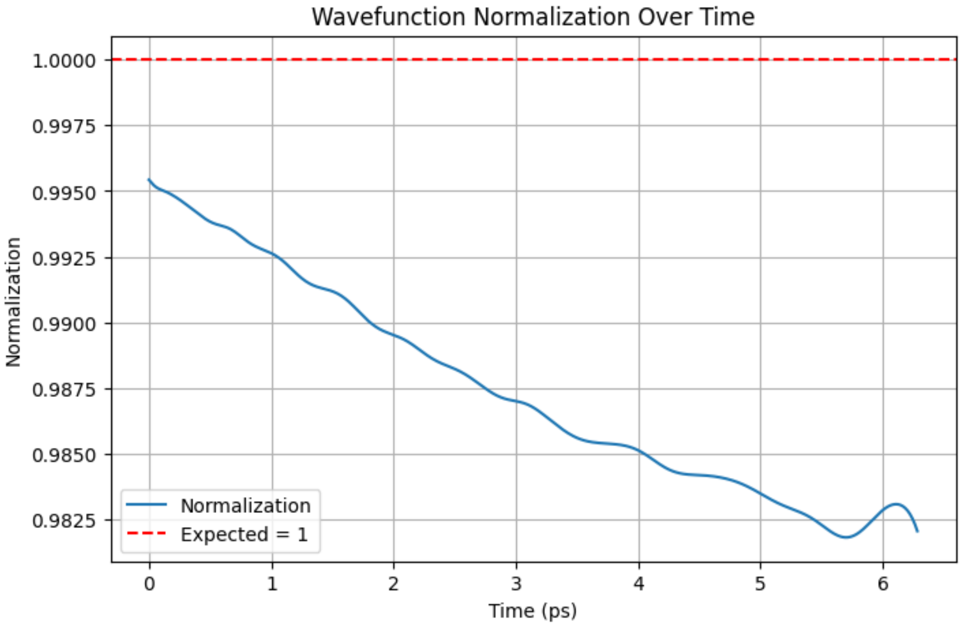
- Started by recreating a model from the paper Shah et al. 2022, single electron in 1D harmonic potential, using atomic units
- Extended previous model with different initial conditions
- Used actual units instead of atomic units
- Single electron in a 1-D moving quantum dot

Recreating 1D Model (Atomic Units)

- Started by recreating a model from the paper Shah et al. 2022, single electron in 1D harmonic potential, using atomic units
- Atomic Units: m = 1, hbar = 1, $\omega = 1$
- Initial condition: Superposition of ground and first excited states
- Mean square error (MSE) $\approx 10^{-5}$


Analytical |psi|^2
PINN |psi|^2
t
x
|psi(x,t)|^2
0 1 2 3 4 5 6
−3 −2 −1 0 1 2 3
0.1 0.2 0.3 0.4 0.5 0.6

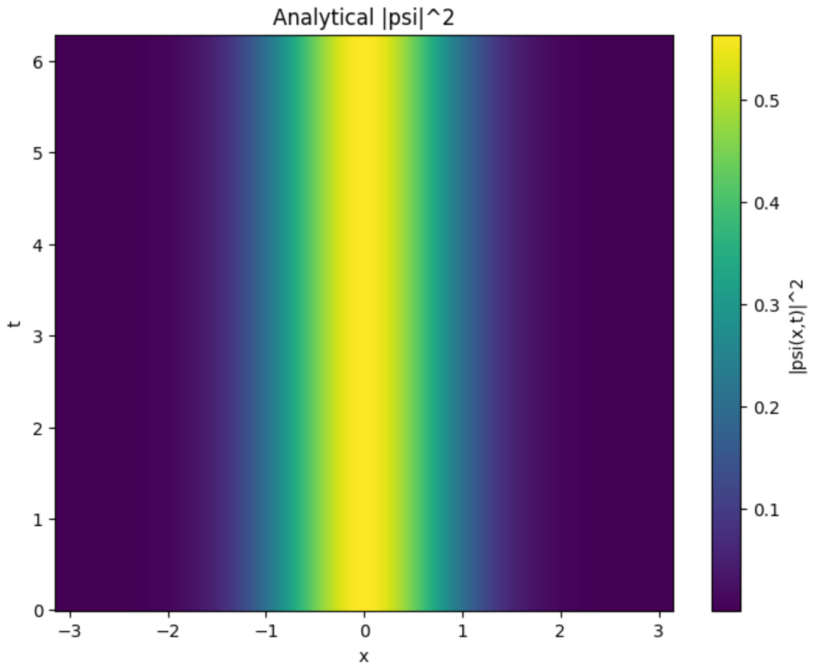


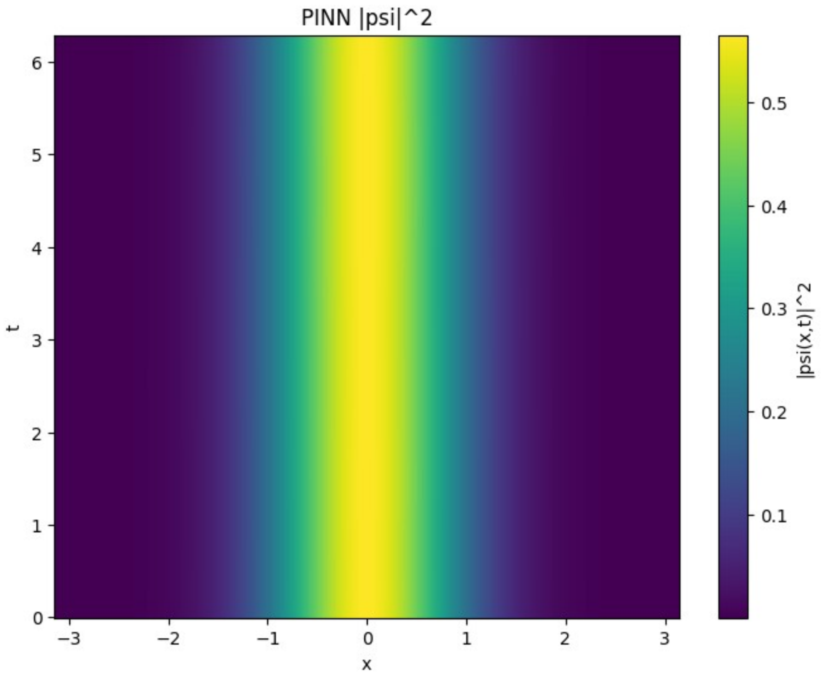

Wavefunction Normalization Over Time
1.0000
0.9975
0.9950
0.9925
0.9900
0.9875
0.9850
0.9825
Normalization
0
1
2
3
4
5
6
Time (ps)
Normalization
Expected = 1

Testing different Initial conditions

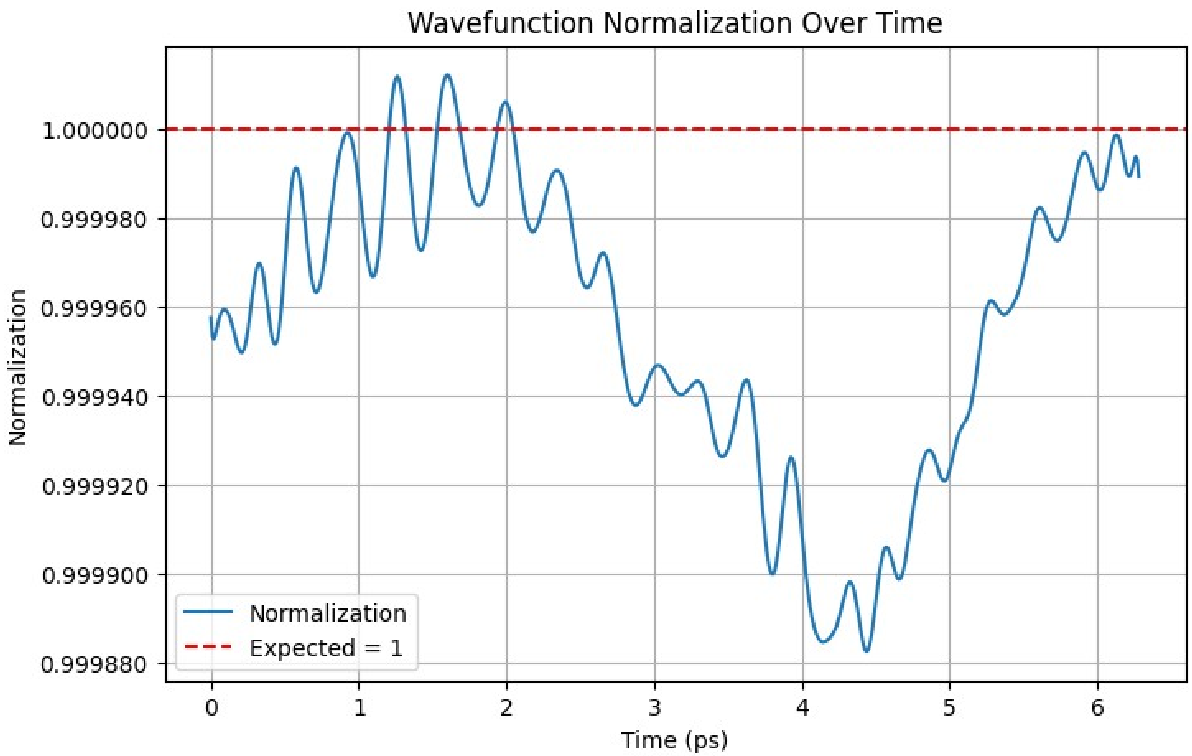
- Model trained with initial condition:
    - Ground state alone
    - First excited state alone
- Observed better stability for ground state
    - MSE ≈ 10^{-8}
- First excited state had slightly higher training error
    - MSE ≈ 10^{-5}

Initial Condition: Ground State

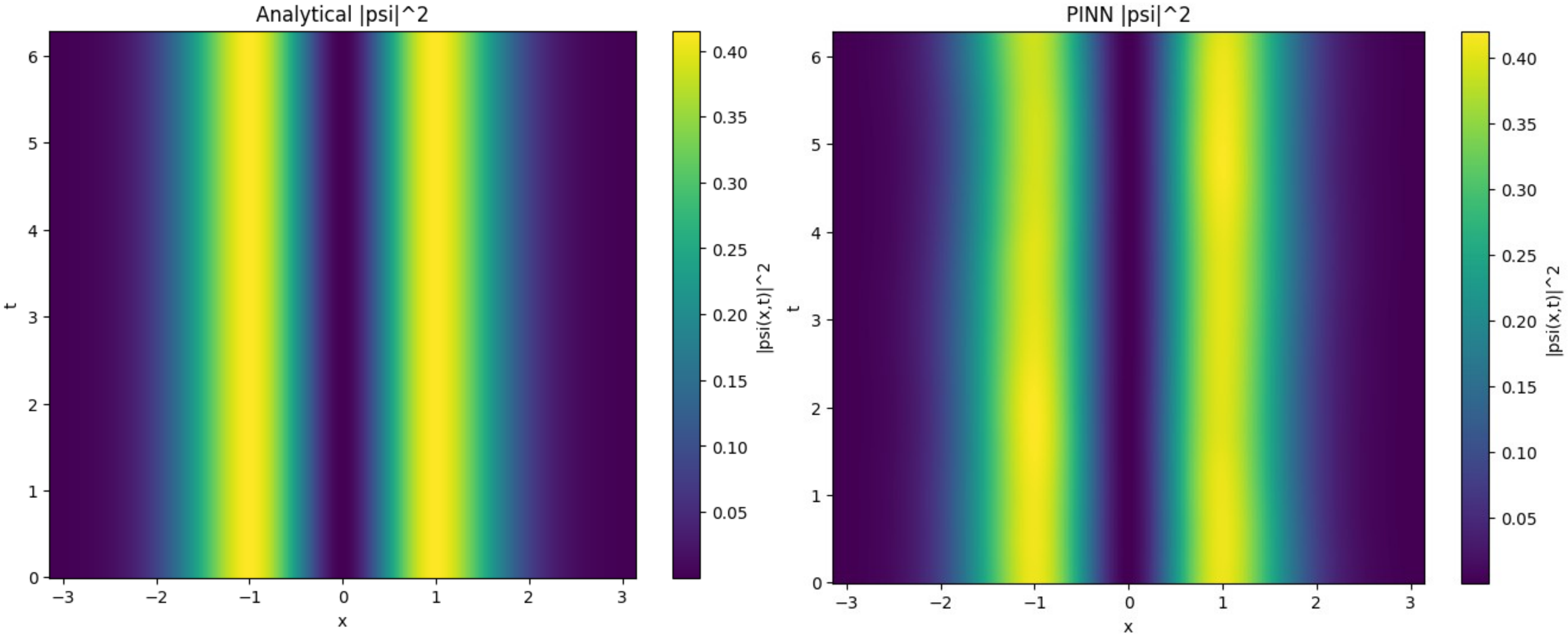




Initial Condition: Ground State

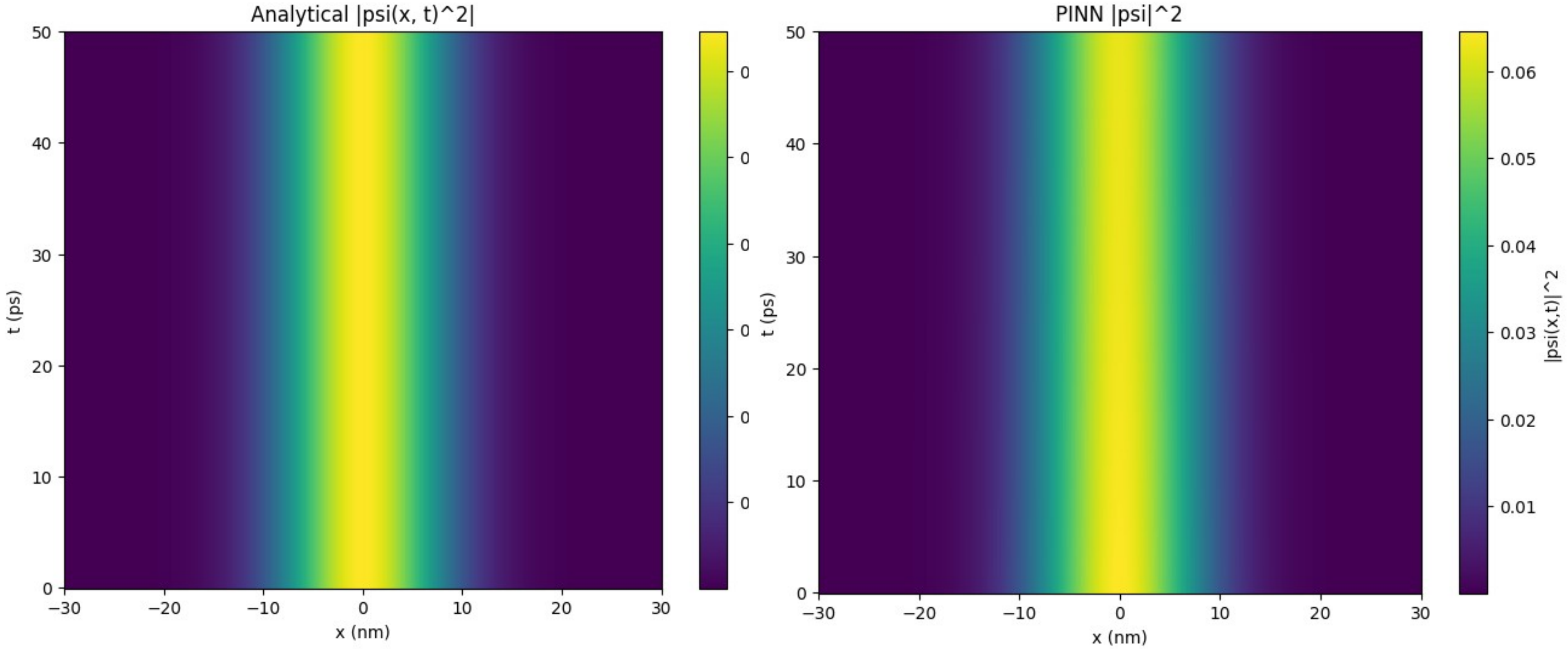


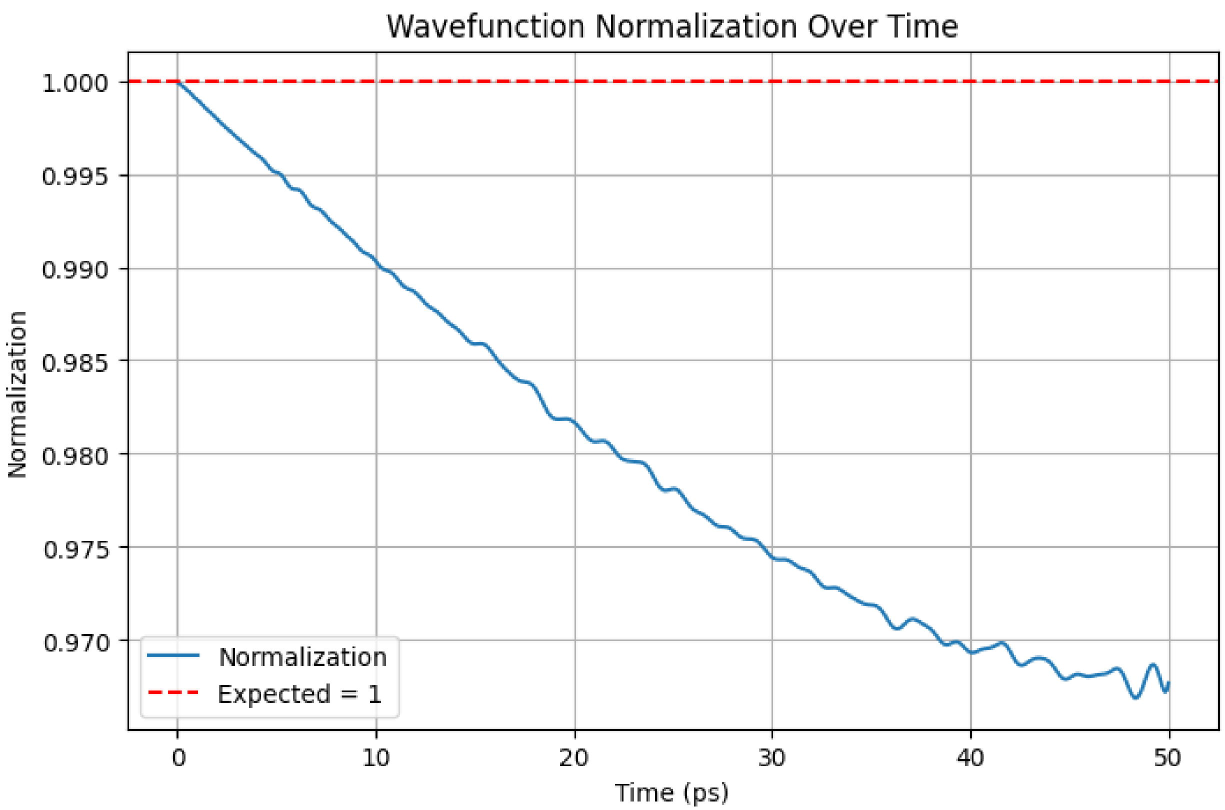
Initial Condition: First Excited State



Model with Physical Units

- Trained the model using nm-ps-meV units
  - hbar = 0.658 meV.ps
  - me = 5.68×10^{-3} meV.ps^2/nm^2
  - ω = 1 / hbar
  - x in nm
  - t in ps
- Adjusted model architecture accordingly

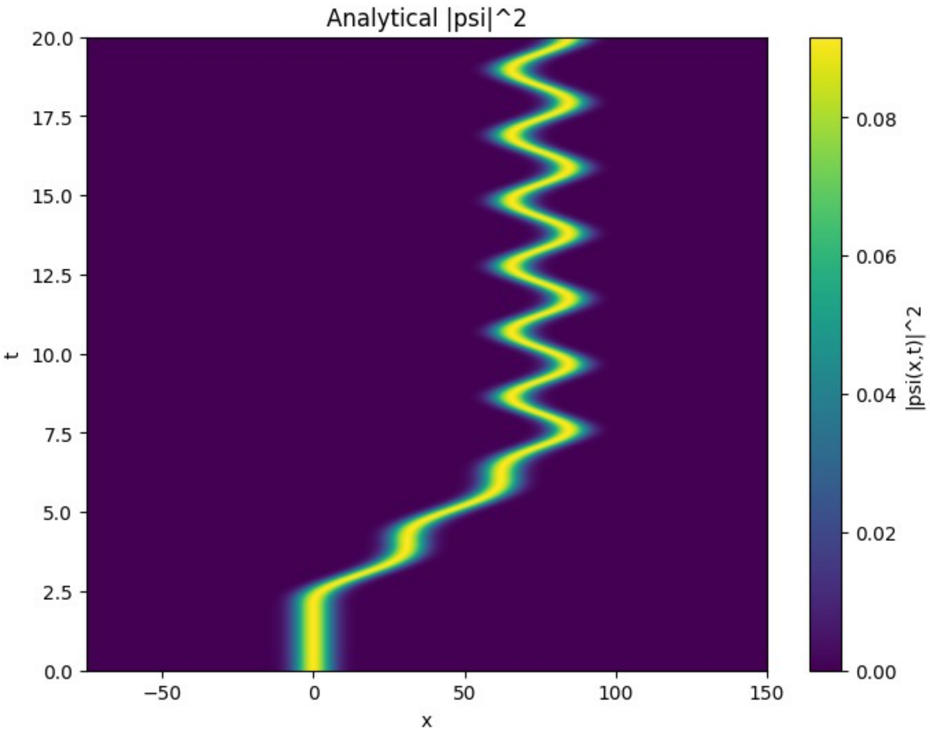

Analytical |psi(x, t)^2|
t (ps)
x (nm)
0
0
0
0
0
0
PINN |psi|^2
t (ps)
x (nm)
|psi(x,t)|^2
0.06
0.05
0.04
0.03
0.02
0.01
50
40
30
20
10
0
−30
−20
−10
0
10
20
30

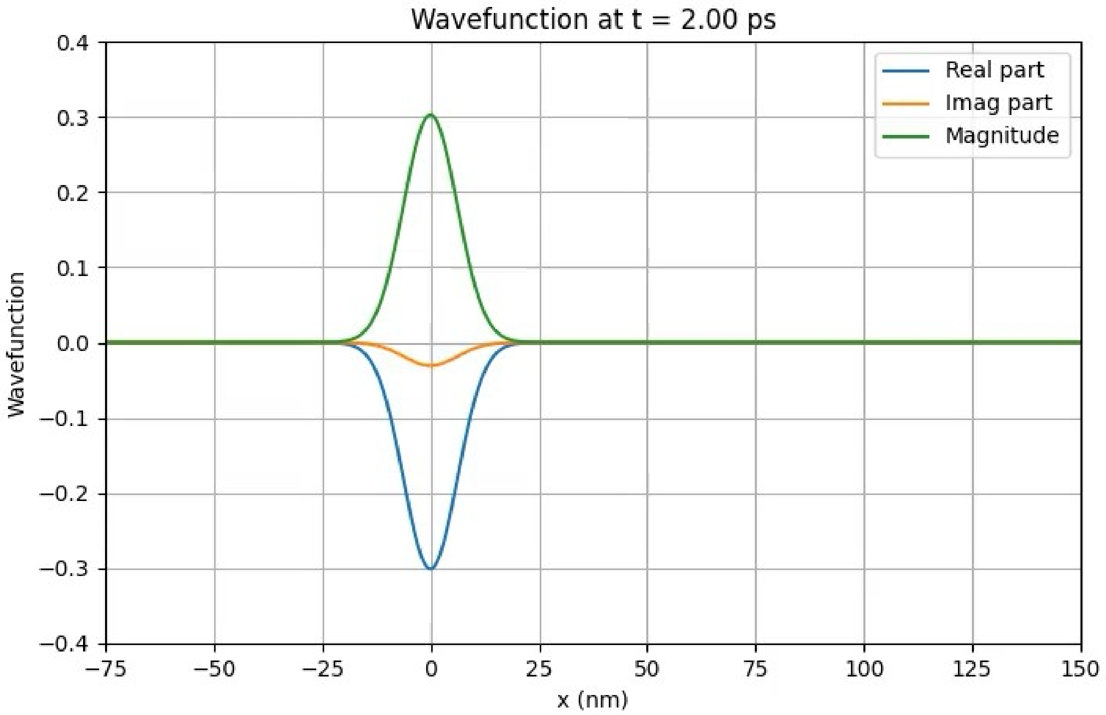


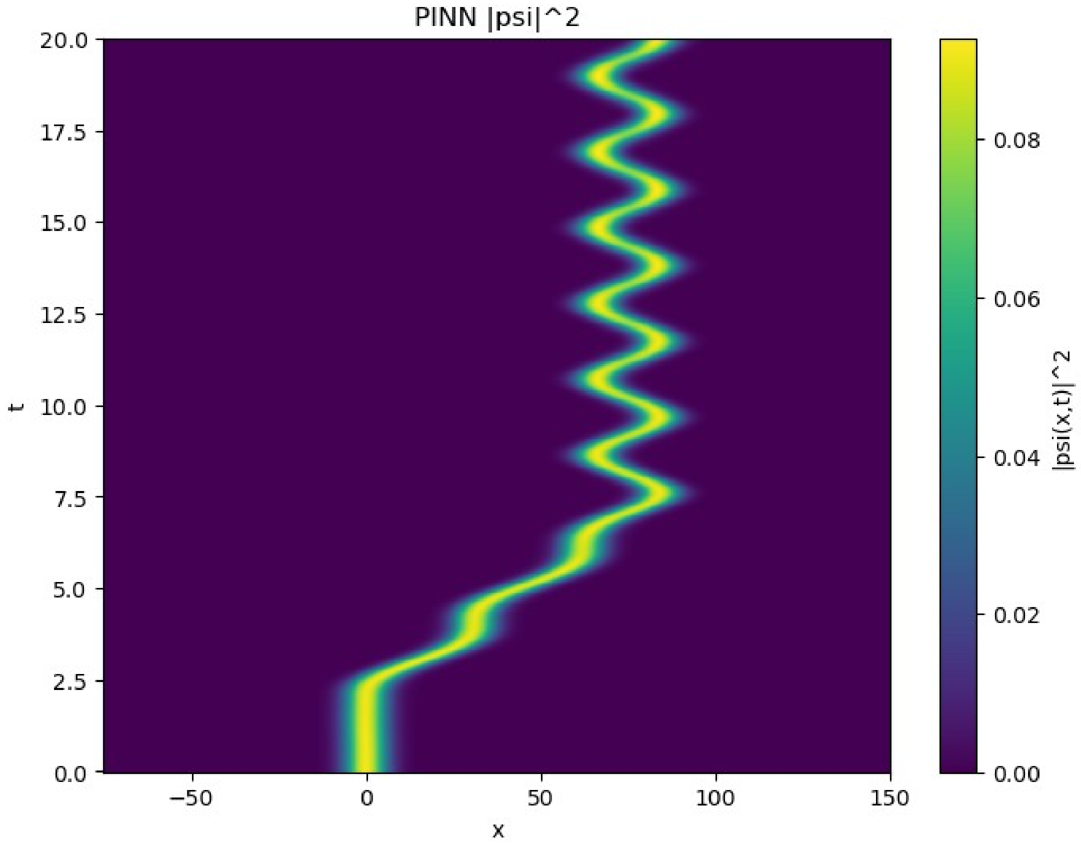
Wavefunction Normalization Over Time
1.000
0.995
0.990
0.985
0.980
0.975
0.970
Normalization
0
10
20
30
40
50
Time (ps)
Normalization
Expected = 1

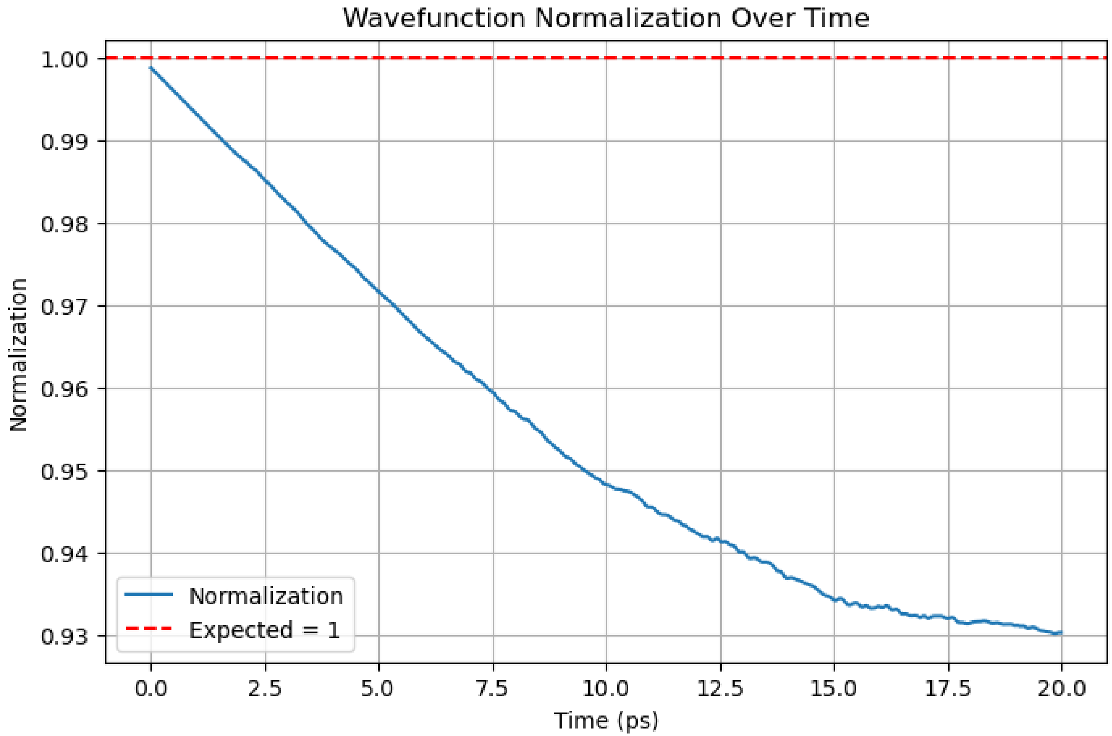
Moving Quantum Dot Model

- Introduced time dependent quantum dot that move along the x axis
- Quantum dot shift over time → electron follows the trap
- Trained the PINN to follow the dynamic behaviour


Analytical |psi|^2
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
t
−50
0
50
100
150
x
0.08
0.06
0.04
0.02
0.00
|psi(x,t)|^2


Wavefunction at t = 2.00 ps
Real part
Imag part
Magnitude
Wavefunction
x (nm)
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
−0.3
−0.4
−75
−50
−25
0
25
50
75
100
125
150


PINN |psi|^2
t
x
|psi(x,t)|^2
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
−50
0
50
100
150
0.08
0.06
0.04
0.02
0.00


Wavefunction Normalization Over Time
Normalization
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Time (ps)
Normalization
Expected = 1

Conclusion

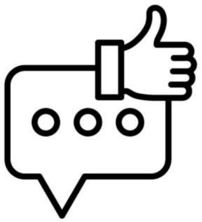
- Successfully validated PINN using multiple 1D models
- Correctly handled:
  - Superposition of states
  - Real physical units
  - Moving quantum dot systems

Next Steps

- Improve the stability of the moving quantum dot model
- Improve the long time evolution stability
- Create a more generalised quantum dot model where v_{QD} can be given as an input
- Explore 1D two electron model with and without electron electron interaction

Feedback Requested

- Suggestions to improve long time training stability
- Best practices to handle electron-electron interaction
- Ideas to extend PINN to higher dimensional systems



.end