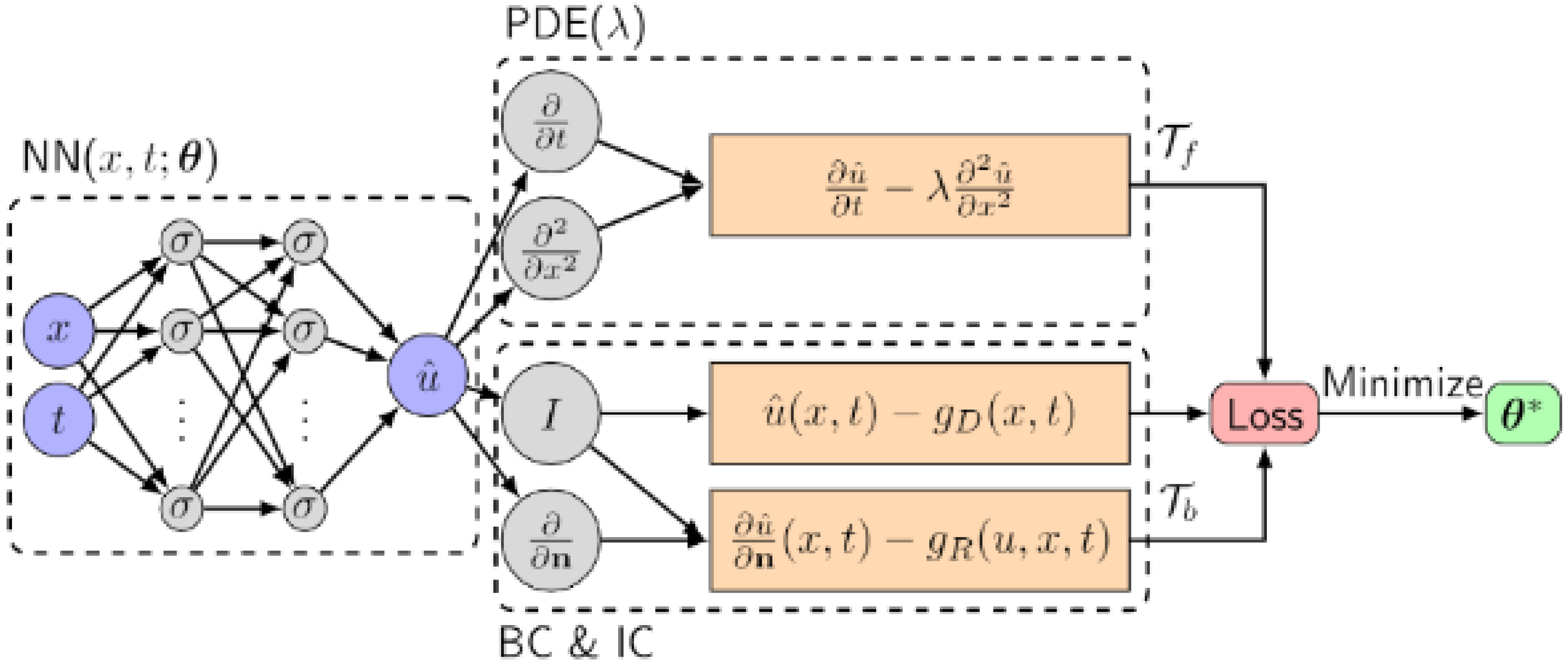
PHYSICS INFORMED NEURAL NETWORK FOR TIMOSHENKO EQUATION

TABLE OF
CONTENT



Introduction

Physics Informed Neural Networks (PINNs) is a machine learning technique that incorporates physical laws and constraints into the neural network training process for solving partial differential equations (PDEs) in various fields of science and engineering, including solid mechanics.

This is particularly useful in cases where it may be difficult or expensive to obtain large amounts of data, such as in many scientific and engineering applications.

Why PIINs ?

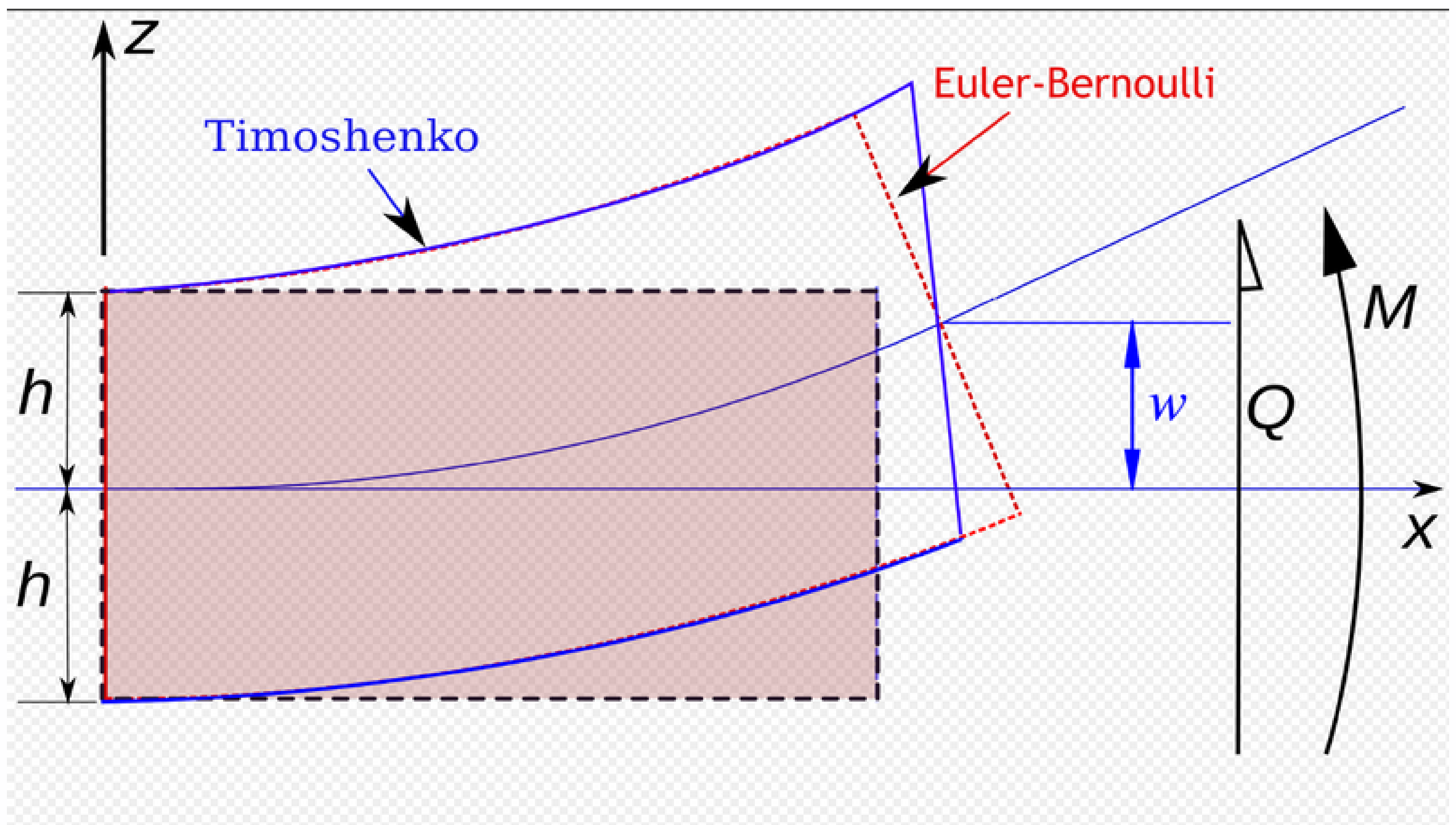
- Normal Neural Networks are used for general-purpose machine learning models that can learn patterns and relationships in any type of data and it must be trained on data while

- The key advantage of PINNs is that they can effectively learn from limited and noisy data by leveraging the known physical principles.

Timoshenko Equation

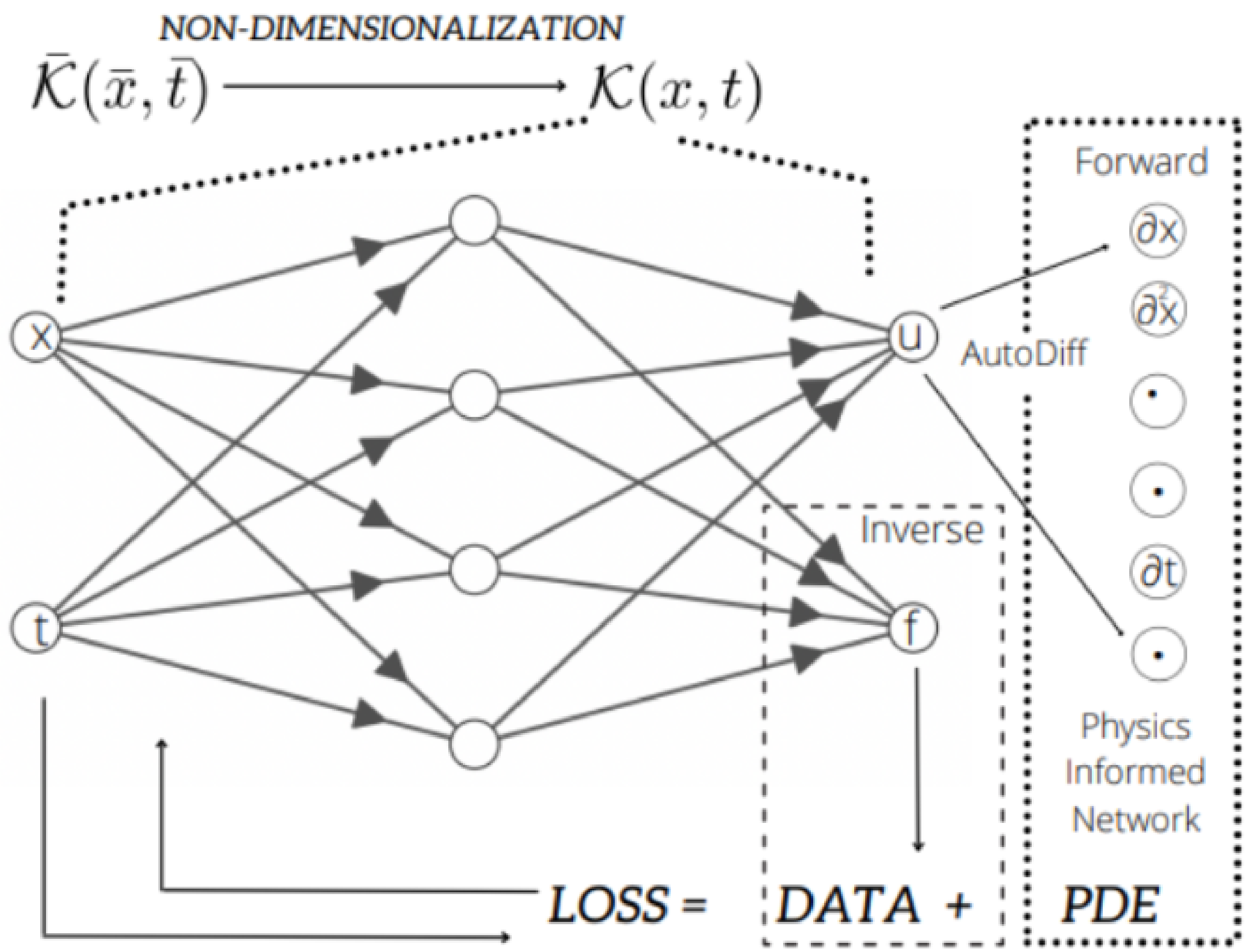
$$\rho A \frac{\partial^2 w}{\partial t^2} - q(x,t) = \frac{\partial}{\partial x}\left[\kappa A G \left(\frac{\partial w}{\partial x} - \varphi\right)\right]$$

$$\rho I \frac{\partial^2 \varphi}{\partial t^2} = \frac{\partial}{\partial x}\left(EI \frac{\partial \varphi}{\partial x}\right) + \kappa A G \left(\frac{\partial w}{\partial x} - \varphi\right)$$



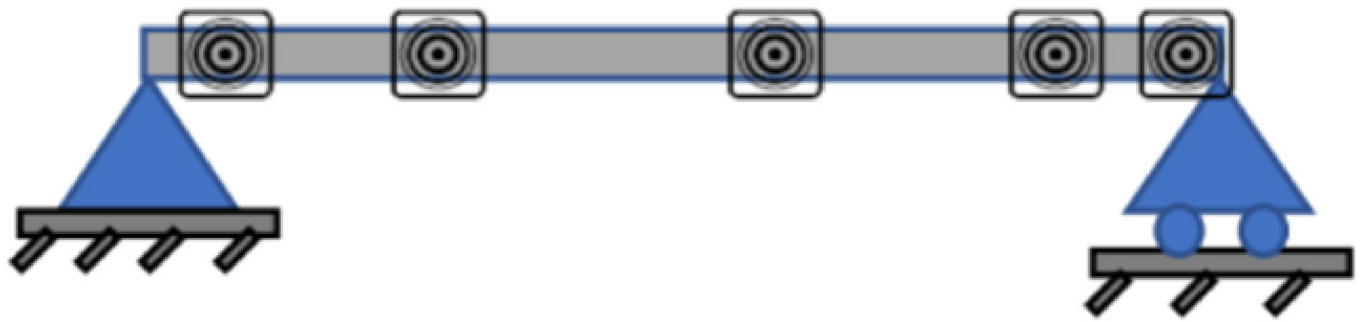
This is the actual
Timoshenko Beam Equation

Non-Dimensionless Equation



By nondimensionalizing the variables and parameters, they are kept within a specific range, resulting in improved performance and generalization of the neural network. Furthermore, dimensionless equations generate more interpretable solutions by eliminating the units of measure.

Non-Dimensionless Equation

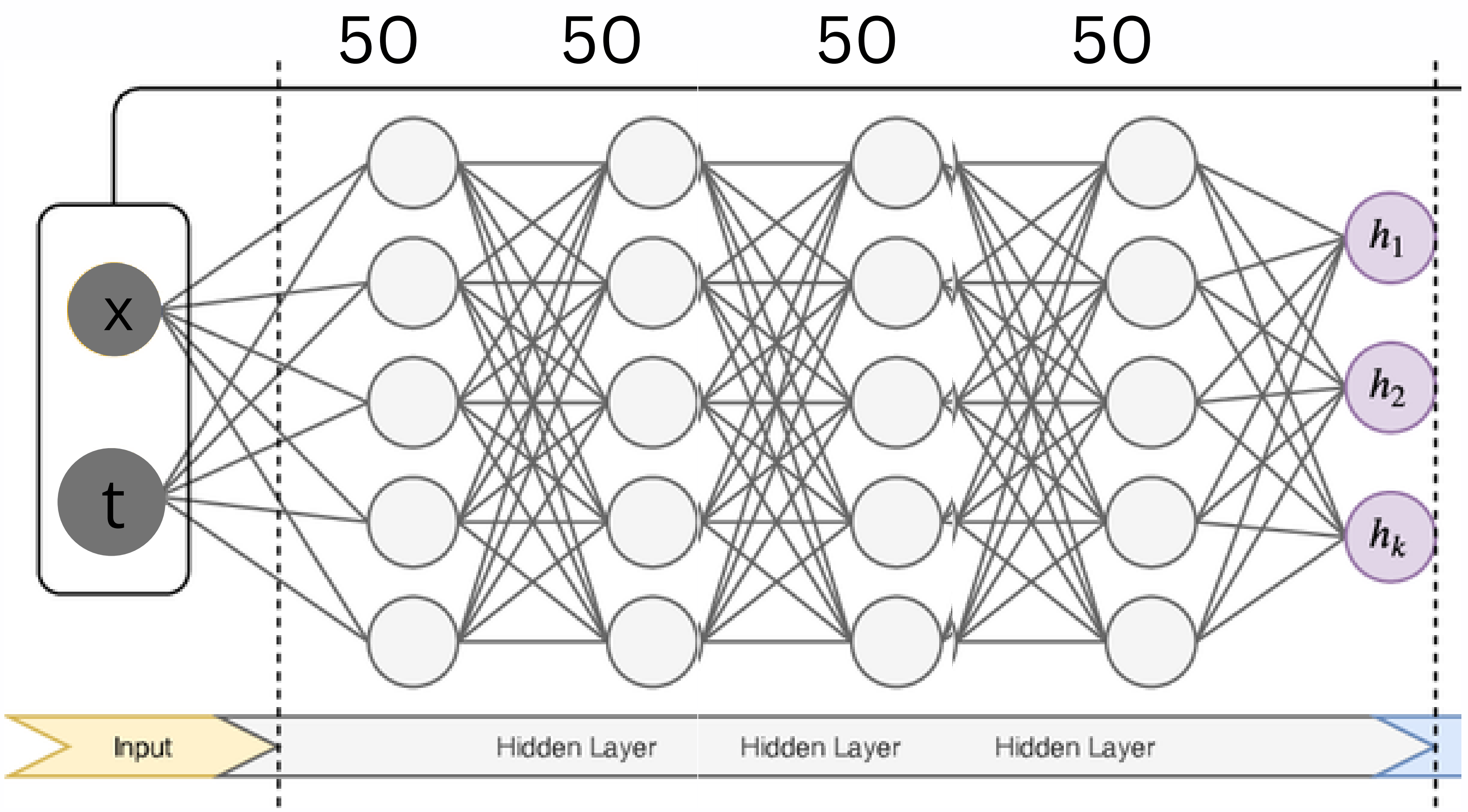


Non-Dimensionless form of Timoshenko Equation

$$\theta_{tt} - \theta_{xx} + (\theta - w_x) = 0$$
$$w_{tt} + (\theta - w_x)_x = g(x, t)$$

g(x,t) is the external force acting on the beam.
w(x,t) transverse is displacement
θ(x,t) is the cross-sectional rotation of the beam
x is position
t is time

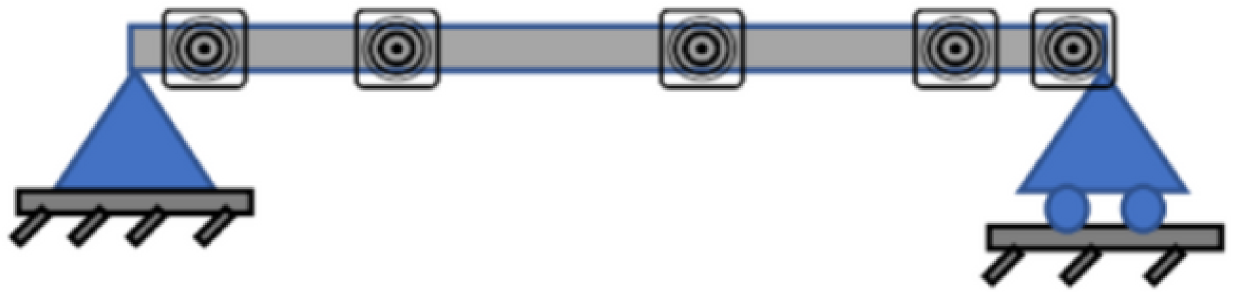
MODEL ARCHITECTURE



I have used FNN Neural Network with
x,t as Input Layer
4 Hidden Layer with 50 neural nets
3 Output Layer

Activation Function = 'tanh'
No of iterations = 4000

Problem :



$$\theta_{\mathrm{tt}} - \theta_{\mathrm{xx}} + (\theta - w_{\mathrm{x}}) = 0$$
$$w_{\mathrm{tt}} + (\theta - w_{\mathrm{x}})_{\mathrm{x}} = g(x, t)$$

g(x) = 0 (zero)

x ∈ [0, π] and t ∈ [0, 1].

BC(Boundary Condtions)

At x = 0 (fixed end):

- **θ**(0, t) = 0
- w(0, t) = 0
- **θ**t(x, 0) = 0

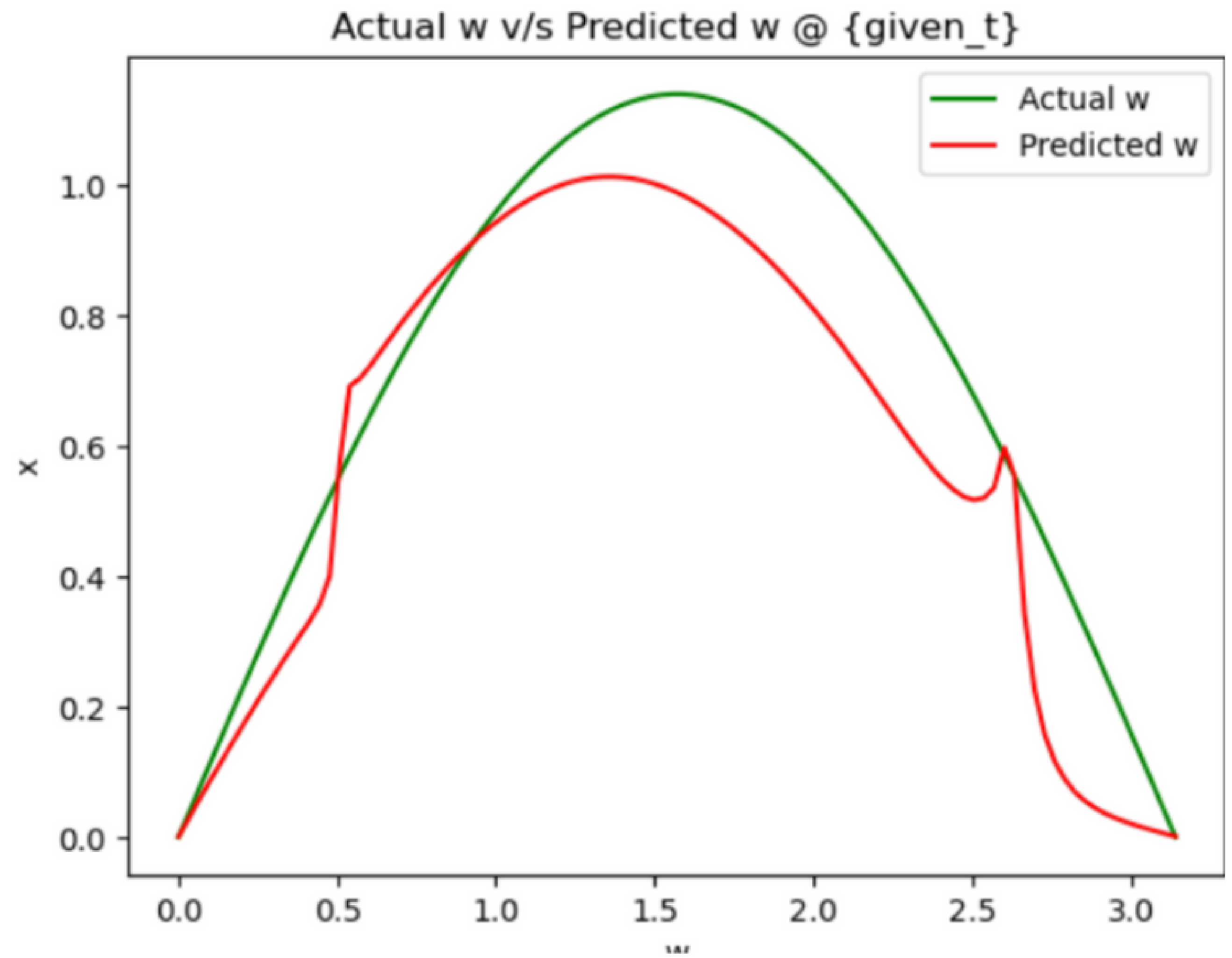
At x = L (free end):

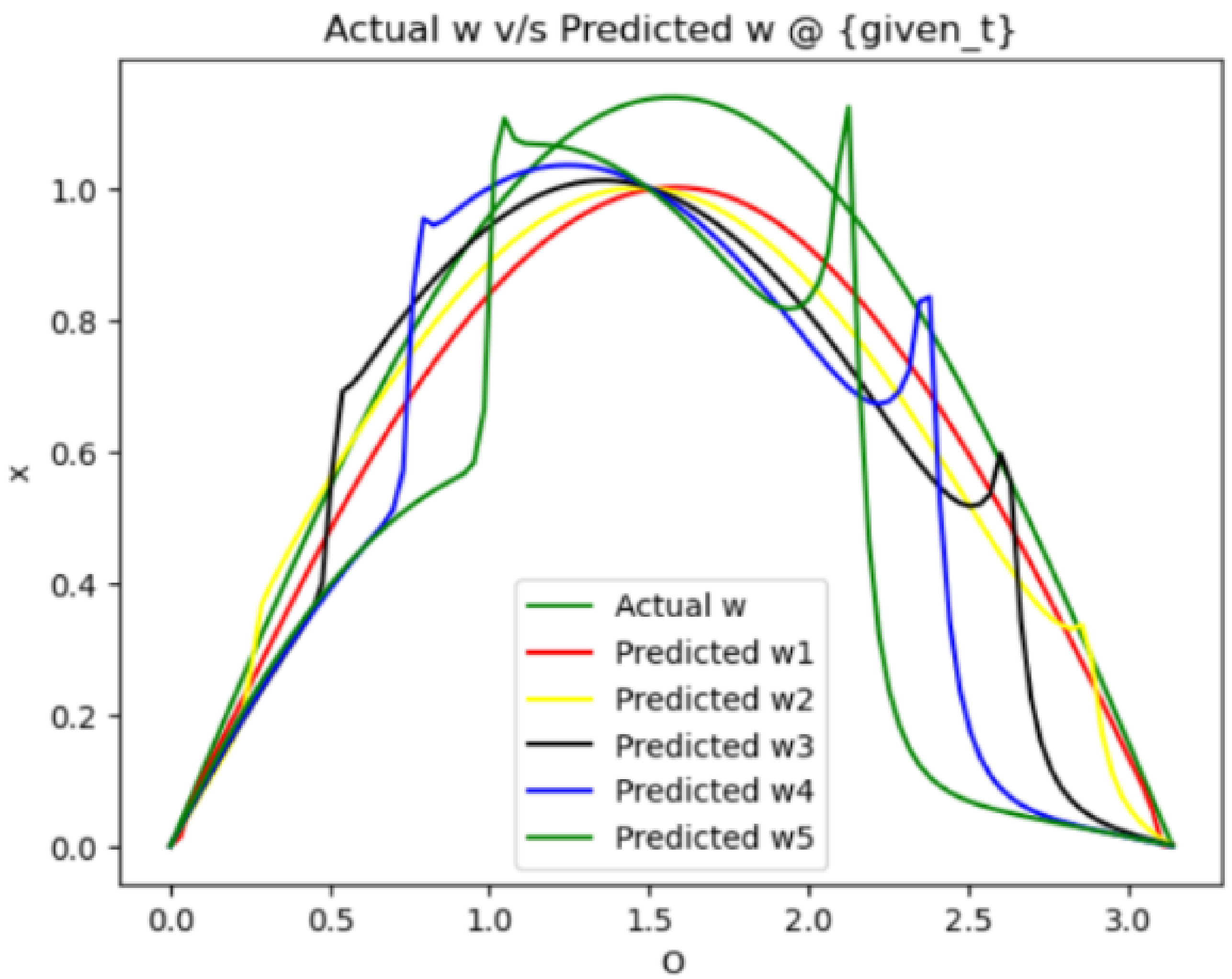
- w(π, t) = 0
- **θ**(π, t) = 0

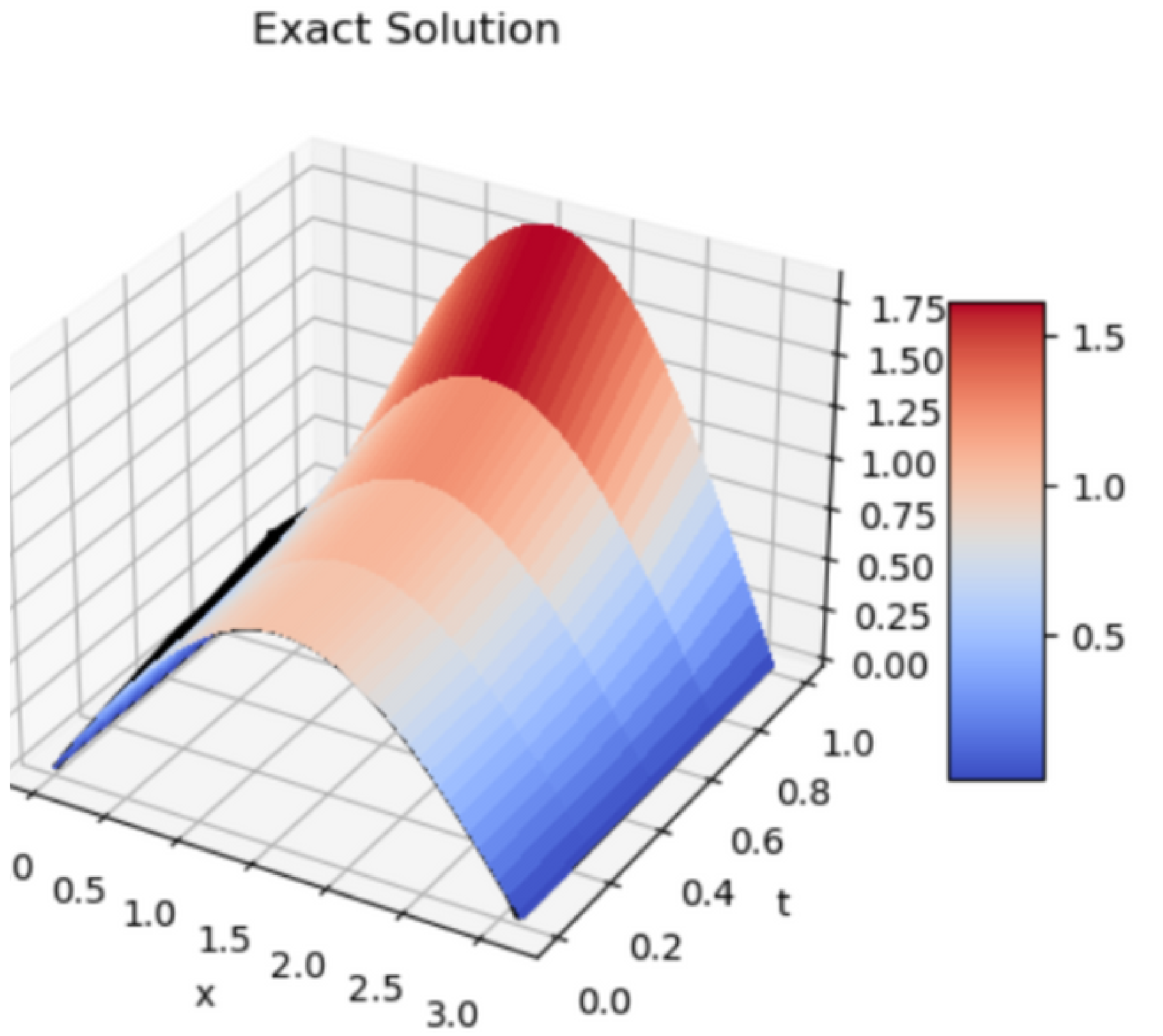
Additionally, we have an Initial Condition at t = 0:

- **θ**(x, 0) = sinx
- w(x,0) = cosx

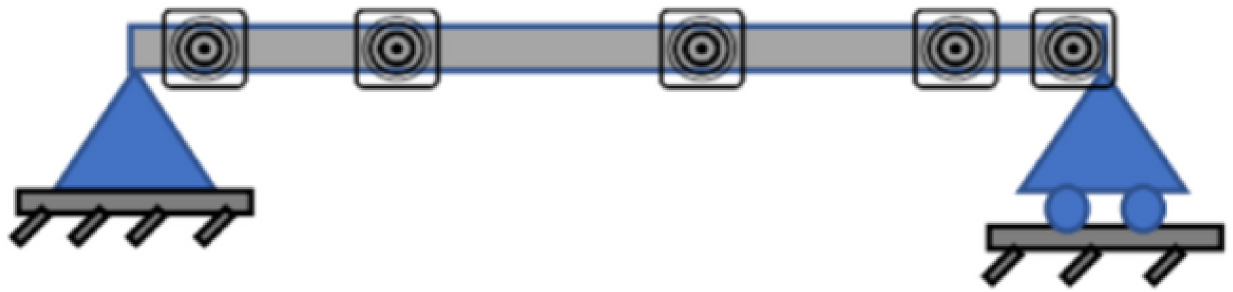
Results


Actual w v/s Predicted w @ {given_t}
Actual w
Predicted w
x
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0


Actual w v/s Predicted w @ {given_t}
Actual w
Predicted w1
Predicted w2
Predicted w3
Predicted w4
Predicted w5
x
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0


Exact Solution
x
t
0
0.5
1.0
1.5
2.0
2.5
3.0
0.0
0.2
0.4
0.6
0.8
1.0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
0.5
1.0
1.5

Problem :



$$\theta_{tt} - \theta_{xx} + (\theta - w_x) = 0$$
$$w_{tt} + (\theta - w_x)_x = g(x, t)$$

$g(x, t) = \cos(t) - \pi/2 \sin(x) \cos(t)$

$x \in [0, \pi]$ and $t \in [0, 1]$.

BC(Boundary Condtions)

At x = 0 (fixed end):

- $\boldsymbol{\theta}(0, t) = 0$
- $w(0, t) = 0$
- $\boldsymbol{\theta}t(x, 0) = 0$

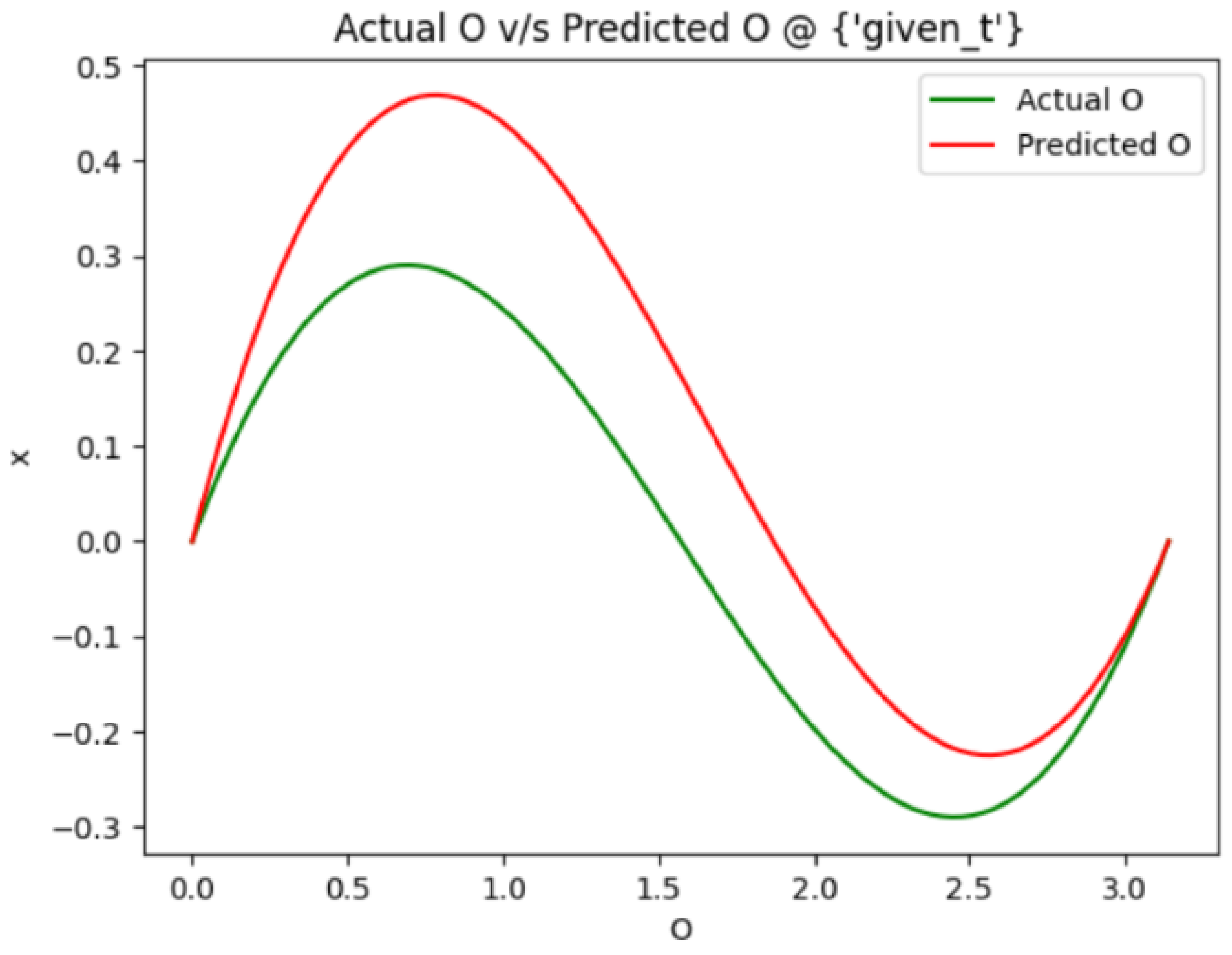
At x = L (free end):

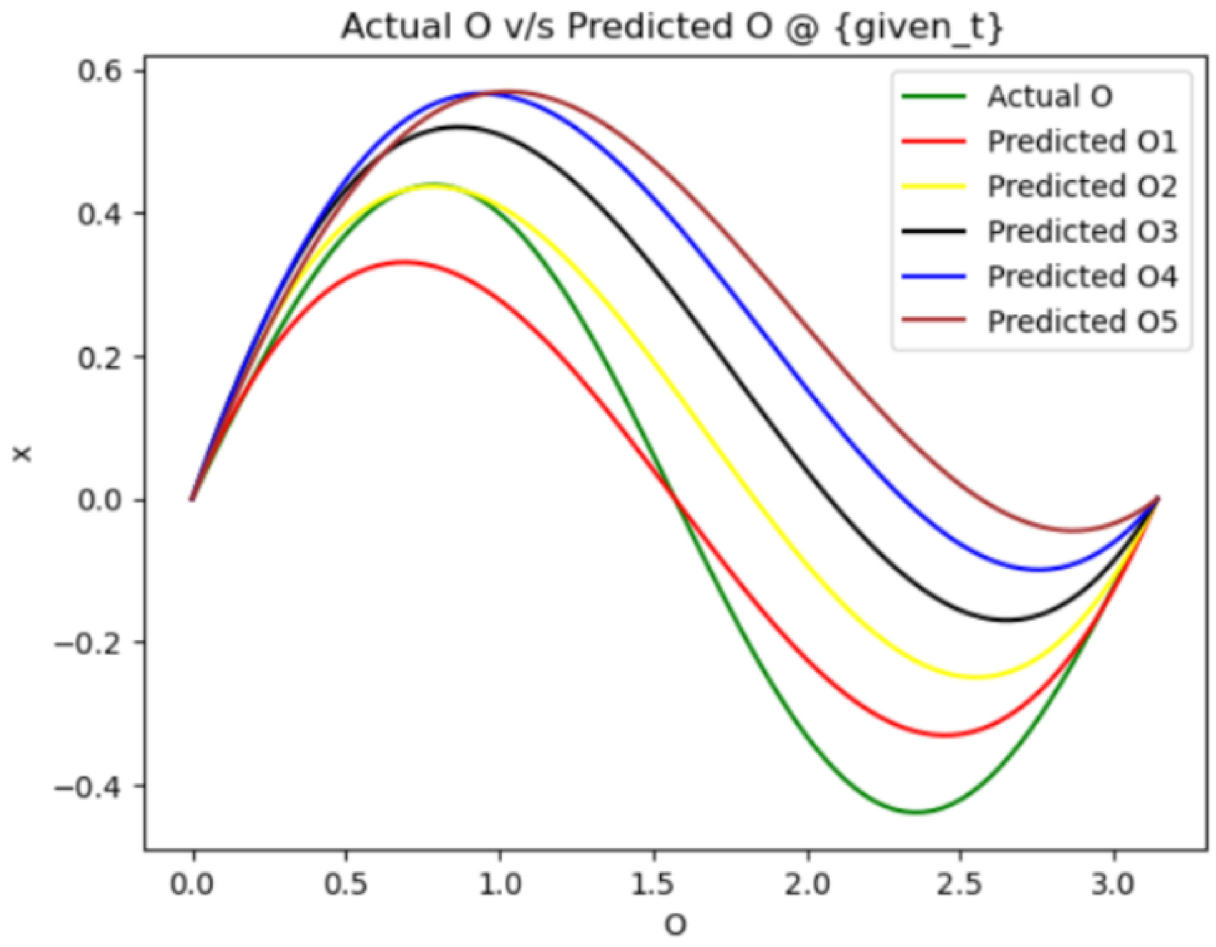
- $\boldsymbol{\theta}(\pi, t) = 0$
- $w(\pi, t) = 0$

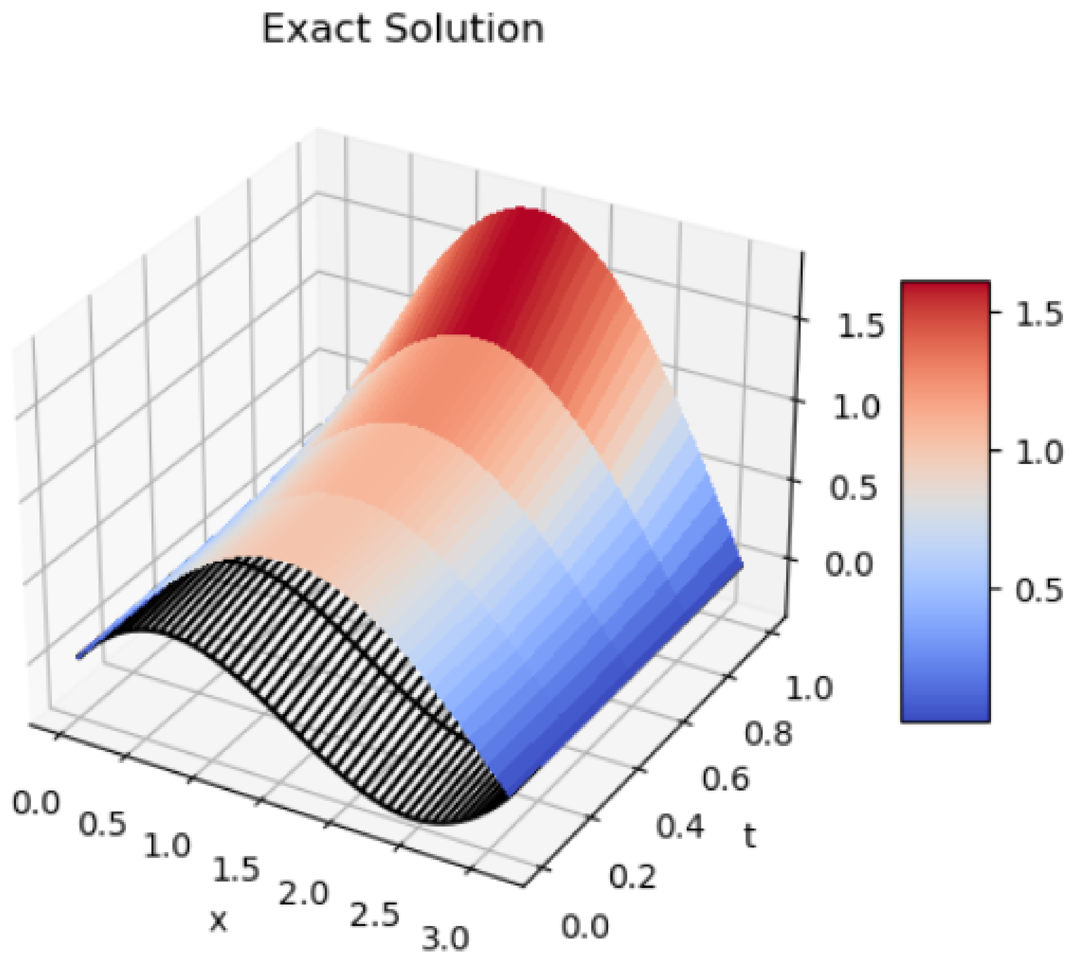
Additionally, we have an Initial Condition at t = 0:

- $\boldsymbol{\theta}(x, 0) = \pi/2 \cos(x) + (x - \pi /2)$
- $\boldsymbol{\theta}t(x, 0) = wt(x, 0) = 0$
- $w(x, 0) = \pi/2 \sin(x)$

Results


Actual O v/s Predicted O @ {'given_t'}
Actual O
Predicted O
x
O
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
−0.3
0.0
0.5
1.0
1.5
2.0
2.5
3.0


Actual O v/s Predicted O @ {given_t}
Actual O
Predicted O1
Predicted O2
Predicted O3
Predicted O4
Predicted O5
x
O
0.6
0.4
0.2
0.0
−0.2
−0.4
0.0
0.5
1.0
1.5
2.0
2.5
3.0


Exact Solution
x
t
1.5
1.0
0.5
0.0
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
1.5
1.0
0.5

CONCLUSION

- In this project, I have developed a PINN model that integrated the Timoshenko equation to predict the deflection of the beam at any distance and time and compared the results with those obtained through the traditional finite element method.

- This study showes that PINNs can be effective in predicting structural mechanics problems, with comparable accuracy to the finite element method. Overall, the combination of machine learning and physics has significant potential for enhancing predictions in a wide range of engineering applications.

Thank You