# Surrogate models using PINNs (Physics-Informed Neural Networks)

NVIDIA Modulus Report

1 July 2024

Astraea Software Co., Ltd

# Overview

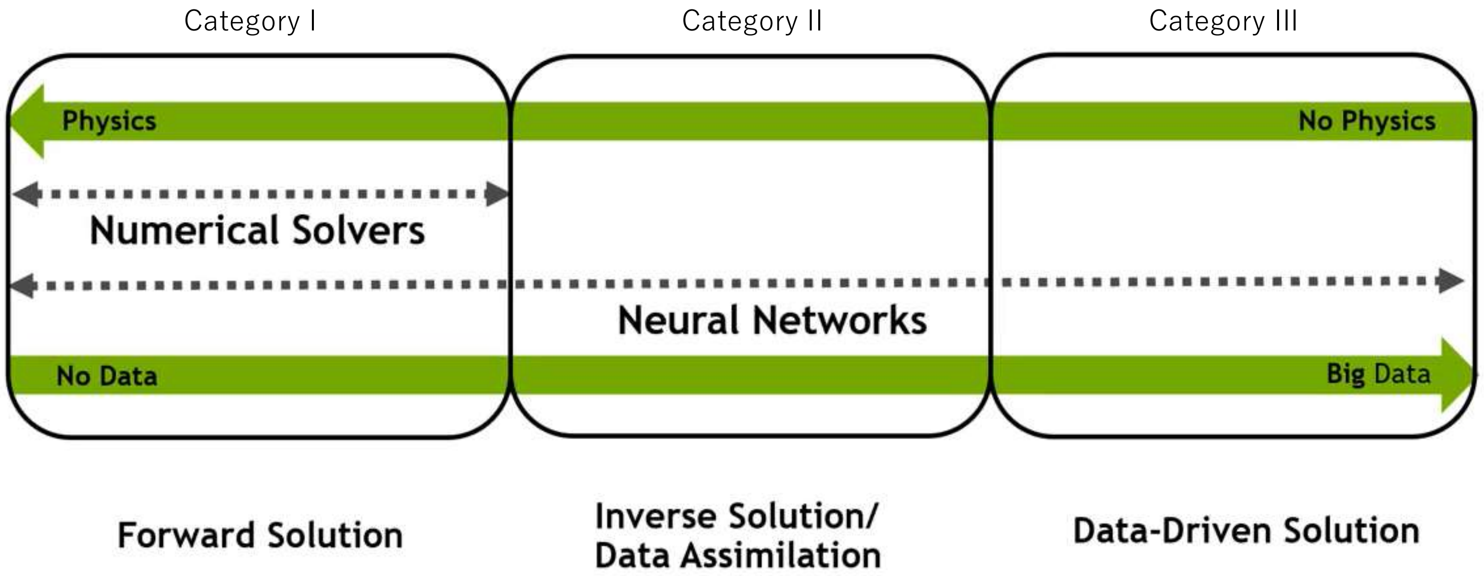
- PINNs (Physics-Informed Neural Networks) is a method that aims at more accurate prediction by incorporating physics-based constraint formulas into neural networks. Recently, the PINNs module, Modulus, is being offered by NVIDIA, and is attracting a lot of attention.
- In this study, NVIDIA Modulus was applied to a simple cantilevered beam static analysis surrogate model to evaluate its performance.
- When training PINNs, there are cases where only physical equations are used, and other cases where data such as FEM analysis results and theoretical solutions are referenced together. In this case, we proceeded with the verification using the procedure on the next page.
- Furthermore, the conclusions are summarized at the end of this paper.

# Verification Procedure

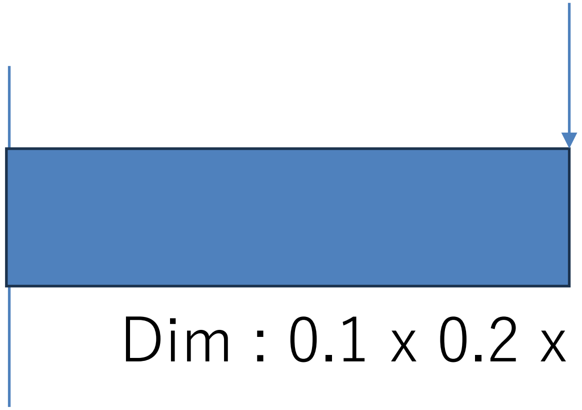
1. Training was performed on a cantilevered beam model with fixed geometry, using only the physical equations as loss functions. (page 5)
2. As good result were obtained, theoretical solution reference data were added to the loss function with the aim of reducing training time. (page 9)
3. As the time savings due to the theoretical solution references were confirmed, the shape change compliance was verified by using the length direction of the shape as a parameter. (page 10)
4. A good result was not obtained, and a theoretical solution reference was added. Accuracy was improved, but there were inaccuracies, e.g. in the middle of the beam. (page 12)
5. The theoretical solution reference position was appropriately increased and further parameterized in all three directions of the geometry to validate a more advanced shape parameter model. (page 15)

# NVIDIA Modulus

- To utilize PINNs, NVIDIA Modulus is used to solve the mechanics problem using Category I (no data references) and Category II (data integration/small data references).



# Verification by simple beam model



F = 1e5N

Dim : 0.1 x 0.2 x 1 m

Mat = E : 210e9 Pa

nu : 0.3

Theoretical value :

$$\delta_{\max} = \frac{FL^3}{3EI}$$

Displacement = 2.38 mm

- Training was attempted for this problem using PINNs and without theoretical solution data references. Therefore, only physical equations are used.

# Equations used for PINN

Astraea

- The equations used are already included in the modulus package."LinearElasticity"

**Equilibrium Equations**

The equilibrium equations in a 3D continuum are derived from the conservation of linear momentum. In the absence of body forces, the equations of equilibrium are given by:

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} = 0$$

$$\frac{\partial \sigma_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z} = 0$$

$$\frac{\partial \sigma_{zx}}{\partial x} + \frac{\partial \sigma_{zy}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} = 0$$

**Traction Boundary Conditions**

On a boundary surface with normal vector $\mathbf{n} = (n_x, n_y, n_z)$, the traction vector $\mathbf{t}$ is given by:

$$\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n} = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{pmatrix} \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} = \begin{pmatrix} \sigma_{xx} n_x + \sigma_{xy} n_y + \sigma_{xz} n_z \\ \sigma_{yx} n_x + \sigma_{yy} n_y + \sigma_{yz} n_z \\ \sigma_{zx} n_x + \sigma_{zy} n_y + \sigma_{zz} n_z \end{pmatrix}$$

**3. Constitutive Relations (Hooke's Law for Linear Elasticity)**

For an isotropic material, the stress-strain relations (Hooke's law) in three dimensions are:

$$\sigma_{xx} = \lambda(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz}) + 2\mu\epsilon_{xx}$$

$$\sigma_{yy} = \lambda(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz}) + 2\mu\epsilon_{yy}$$

$$\sigma_{zz} = \lambda(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz}) + 2\mu\epsilon_{zz}$$

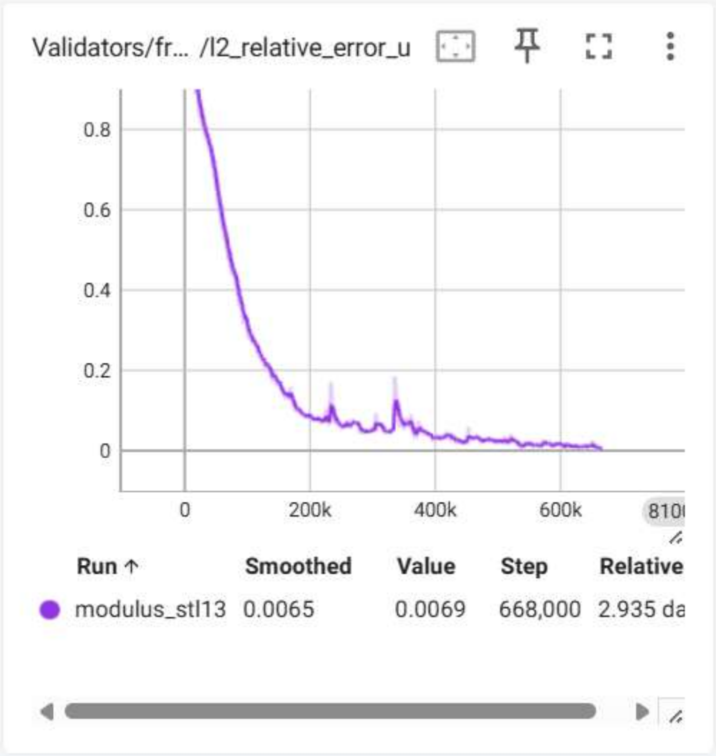
$$\sigma_{xy} = 2\mu\epsilon_{xy}$$

$$\sigma_{xz} = 2\mu\epsilon_{xz}$$

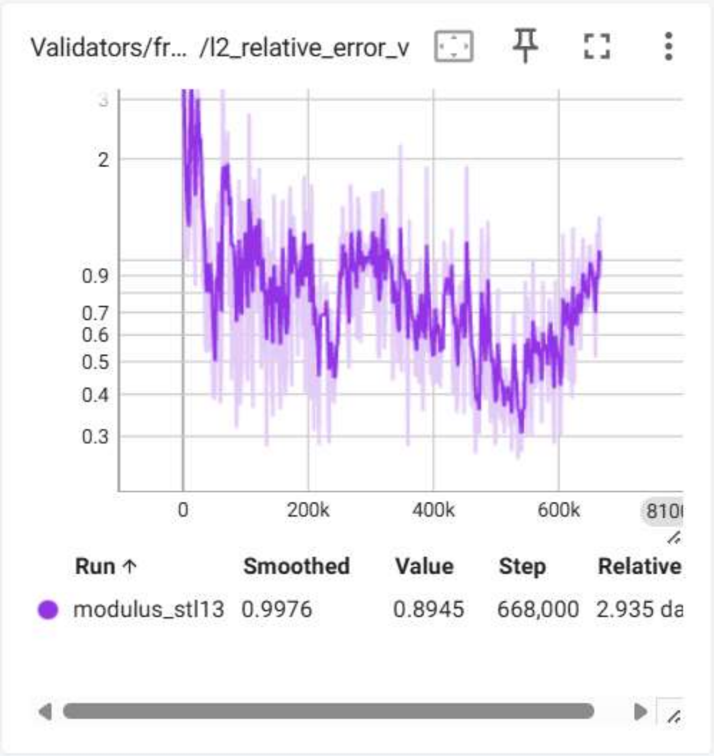
$$\sigma_{yz} = 2\mu\epsilon_{yz}$$

# Training progress graph

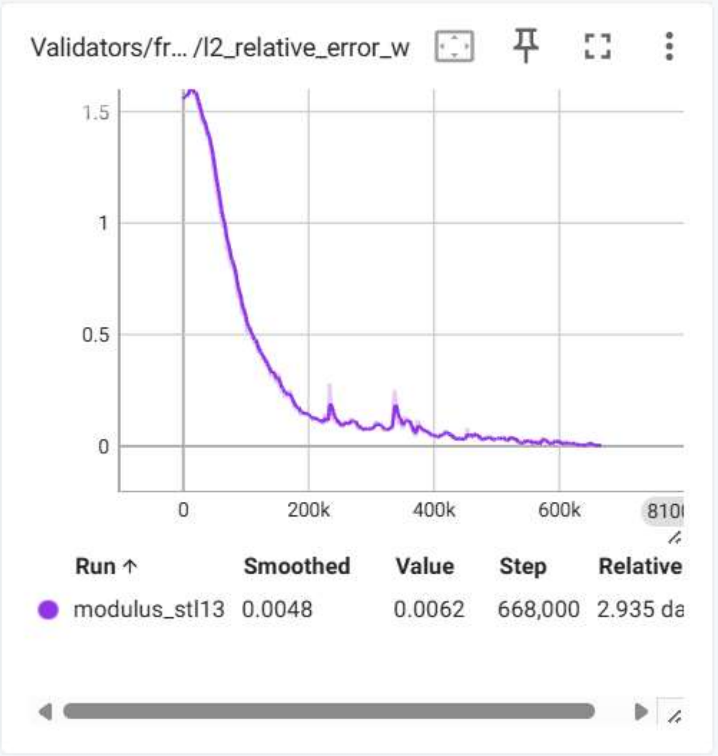
Displacement U (X direction)




Displacement V (Y direction)

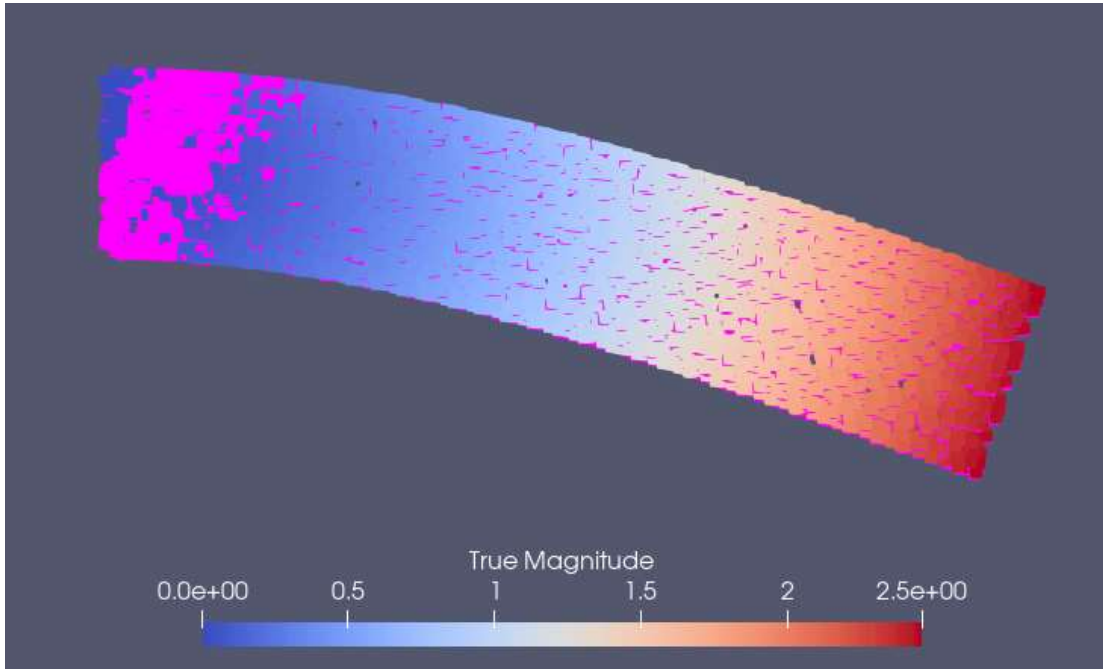



Displacement W (Z direction)

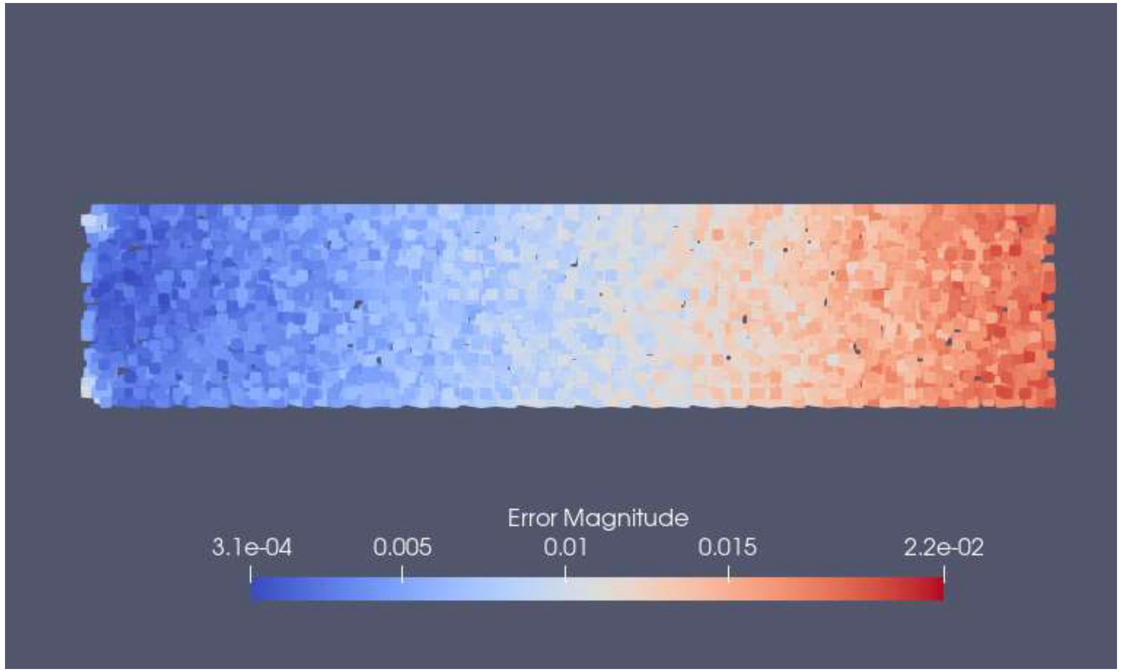



500-600K training steps required to achieve good results

# Comparison of displacement prediction results




displacement magnitude

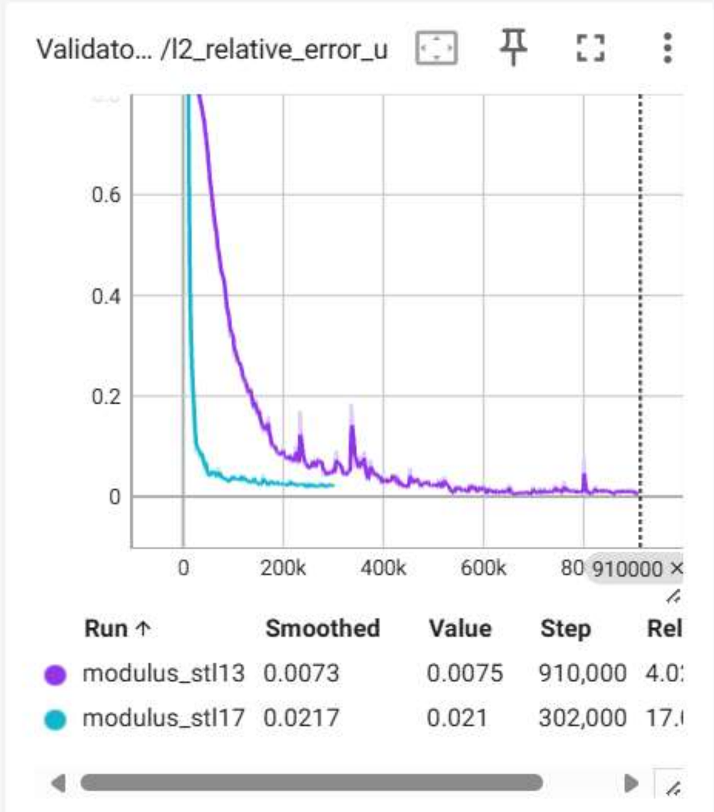



Error Comparison(FEA – PINNs)
Error < 1%

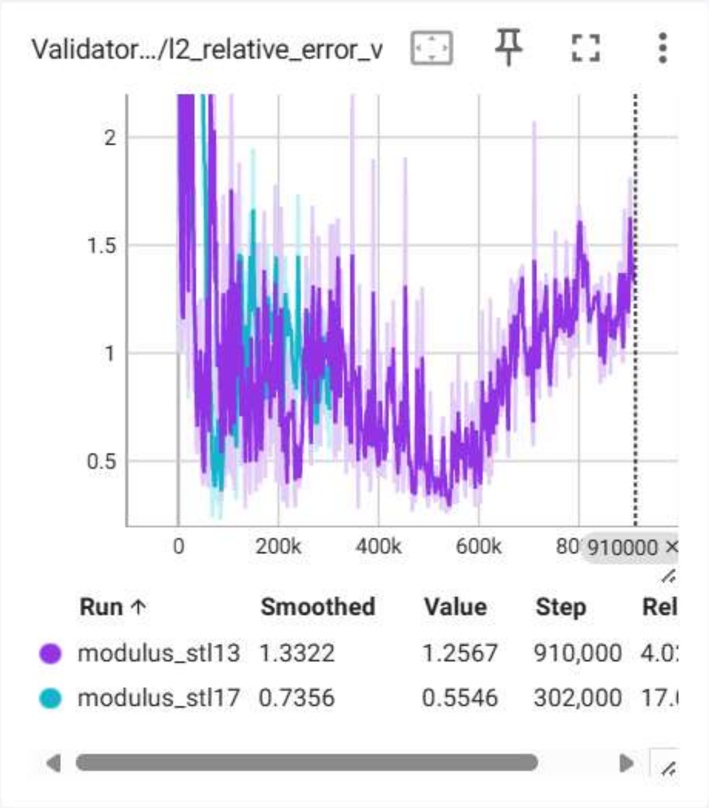
color contour : FEA result
magenta dots : PINNs result

# Verification to reduce training time by referencing theoretical values at the free end

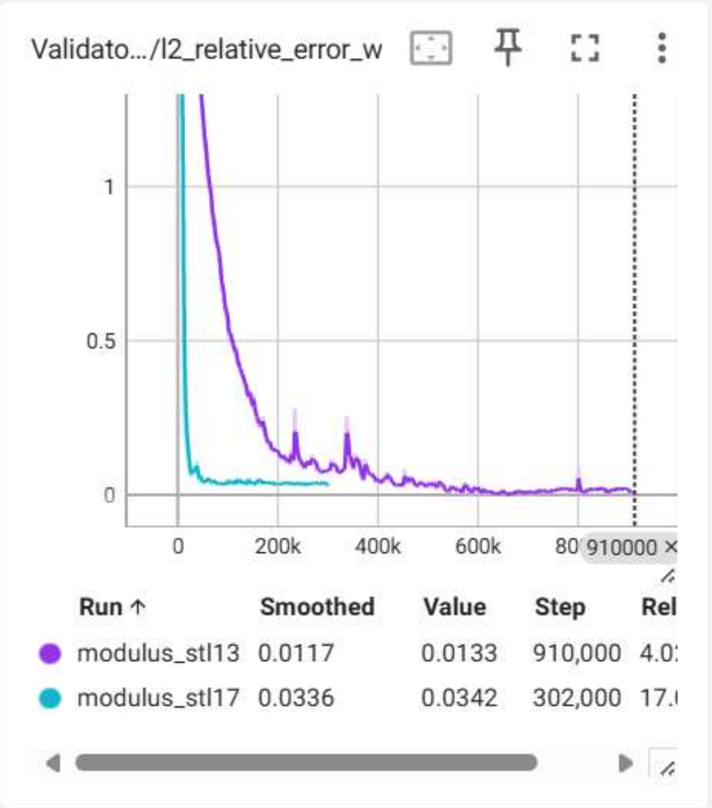
Displacement U (X direction)




Displacement V (Y direction)

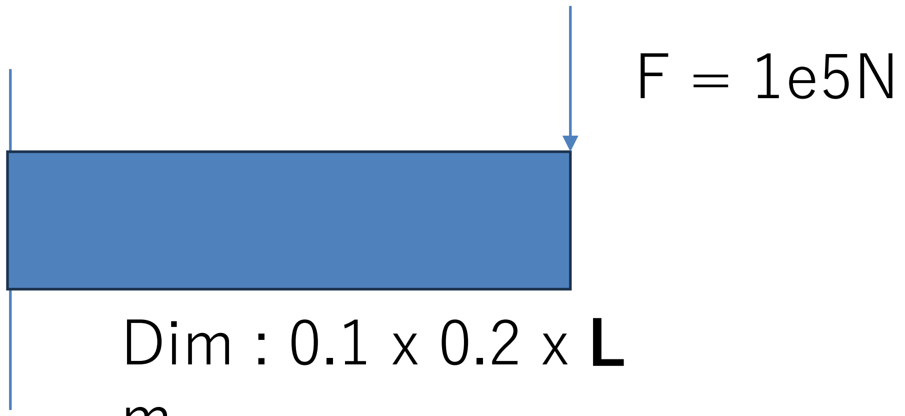



Displacement W (Z direction)




Light blue (see theoretical solution) Purple (physical equation only)
Even if only reference information on disp w is given, disp u also improves the convergence speed.

# Verification by simple parametric geometry




Dim : 0.1 x 0.2 x **L** m

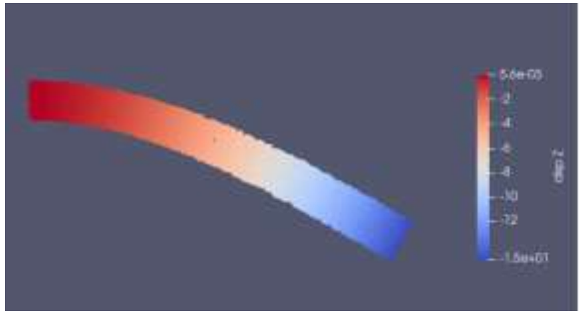
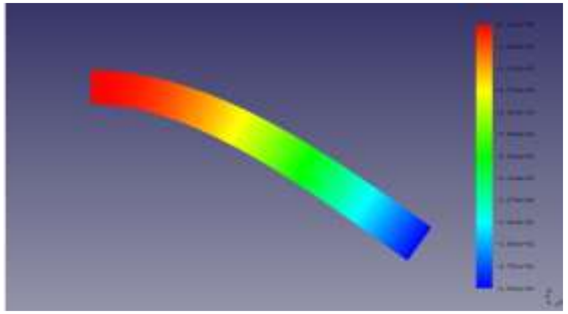
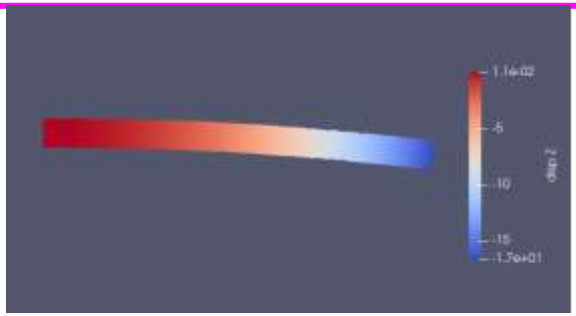
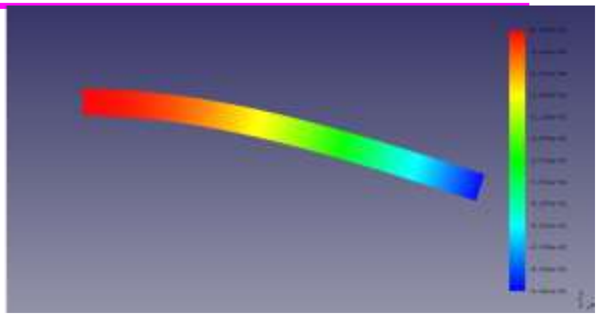
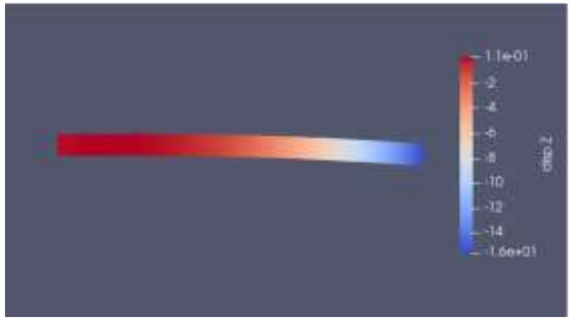
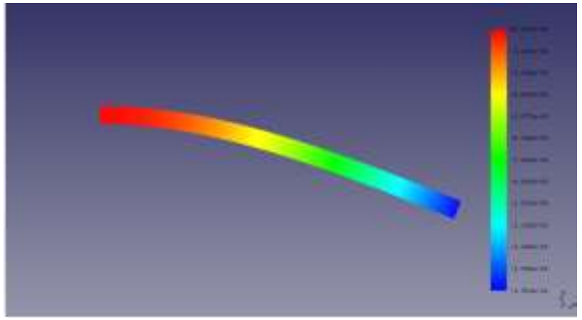
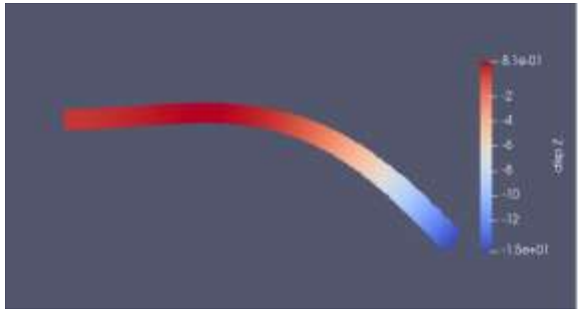
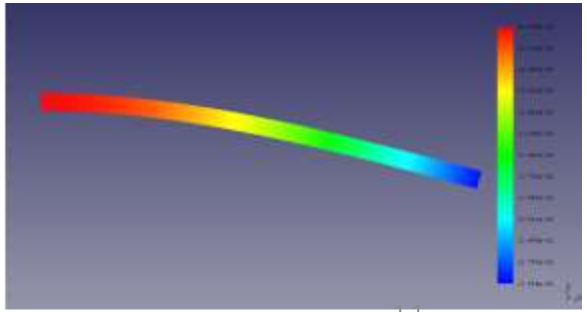
Mat = E : 210e9 Pa

nu : 0.3

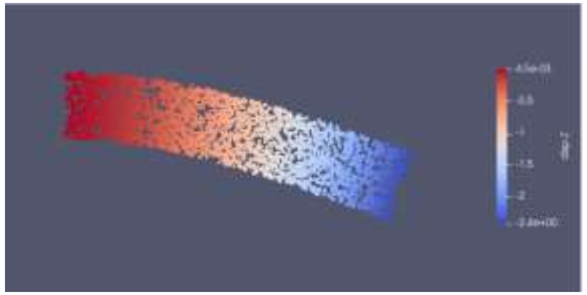
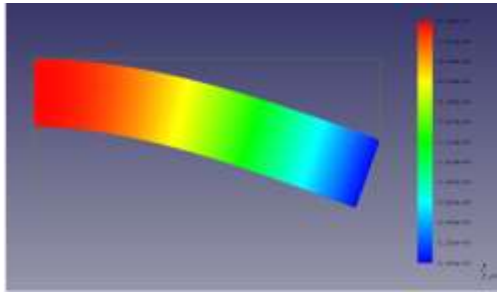
L is the parametric length :
L used for training : 1, 2, 3, 4, 5 m.

Training is performed using only physical losses, so, training data is not referenced.

# Training results
# Left: PINNs Right: FEM

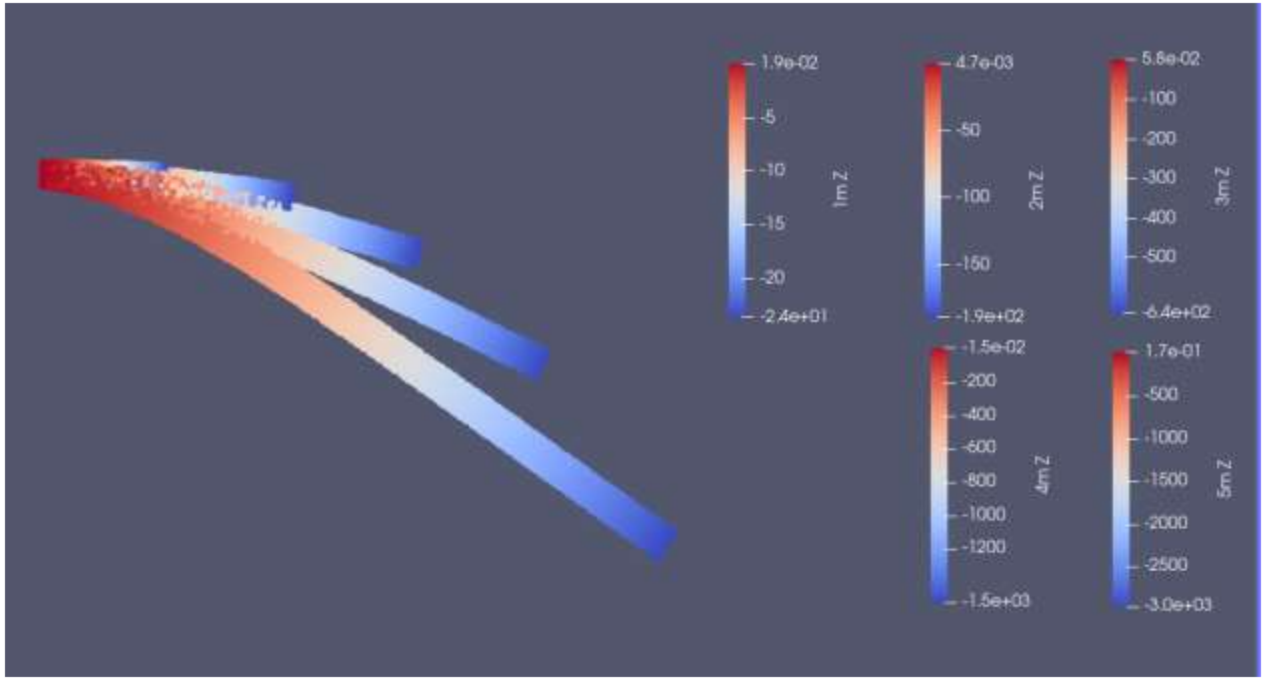









| Length | PINNs | FEM | Error |
|---|---|---|---|
| 1 m | -2.4 mm | -2.4 mm | 0.012 |
| 2 m | -15 mm | -19 mm | 0.2 |
| 3 m | -17 mm | -64 mm | 0.73 |
| 4 m | -16 mm | -152 mm | 0.89 |
| 5 m | -15 mm | -297 mm | 0.94 |

Results for PINNs other than 1m do not agree with FEA results.

# Result of adding theoretical value references to the free end of the parametric model

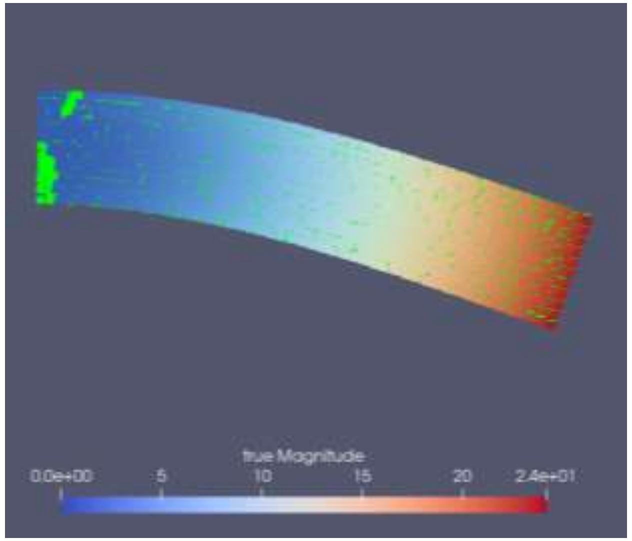



| Length | PINNs | FEM |
|---|---|---|
| 1 m | -2.4 mm | -2.4 mm |
| 2 m | -19 mm | -19 mm |
| 3 m | -64 mm | -64 mm |
| 4 m | -150 mm | -152 mm |
| 5 m | -300 mm | -297 mm |

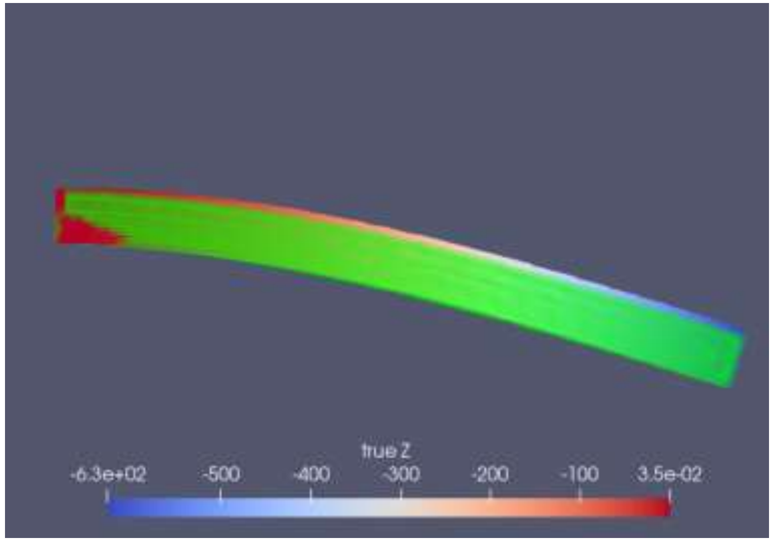
When referring to theoretical values at the free end, PINNs displacement results improve significantly in just less than 200k steps.

# Displacement prediction results

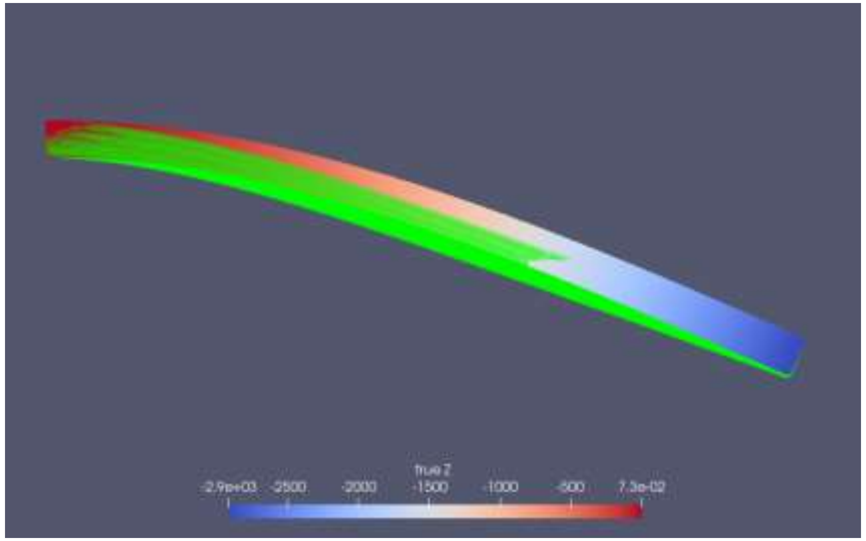
- The maximum displacement at the free end is good, but there are still deviations in the displacement results elsewhere on the model when training a small number of steps.




1 m length



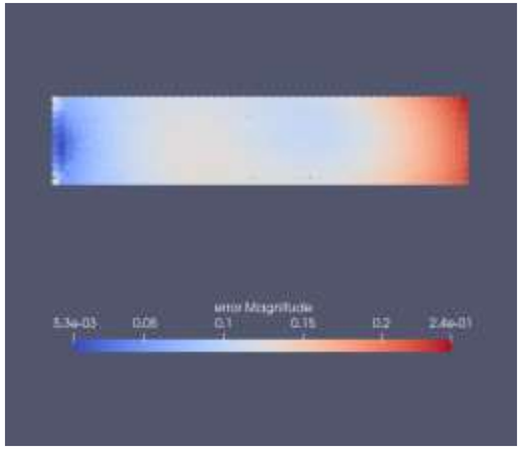

3 m length



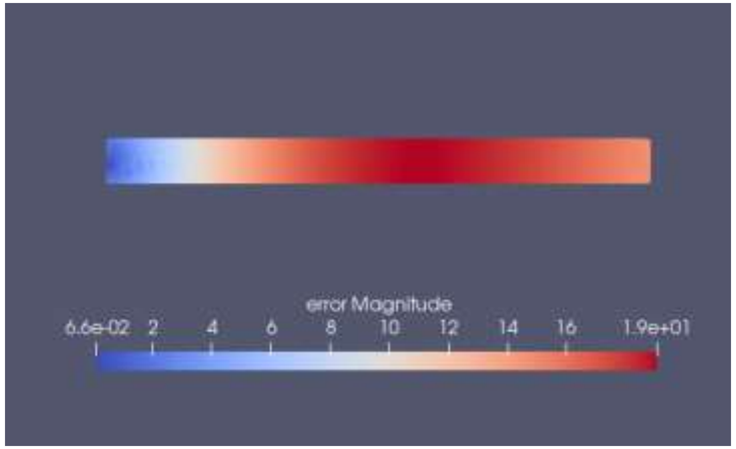

5 m length

color contour : FEA result
Green dots : PINNs result

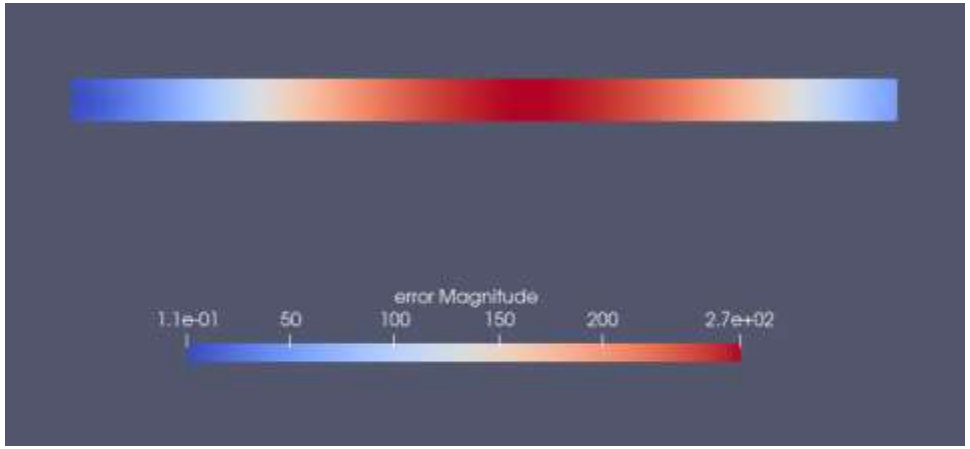
# Relative error distribution




1 m length




3 m length

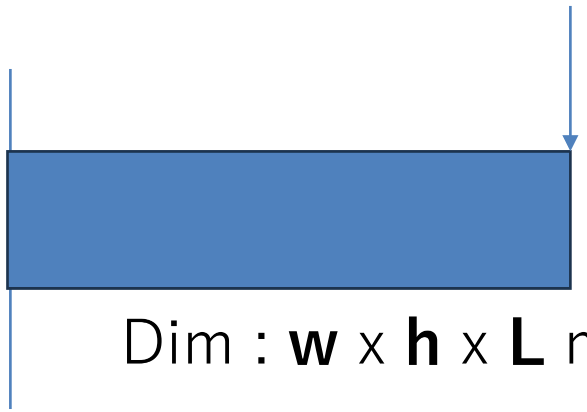



5 m length

Results show more error reduction with longer training times. Above are the results for 200k steps. Accuracy is improved from the edges, as there are more errors at the center position than at both edges.

For the neural network to learn the bending behavior of the beam, it would be effective to also refer to the displacement at the mid-length position. This will be reflected in the next case study.

# Theoretical solution data reference parametric shape verification -Part 2



F = 1e5N

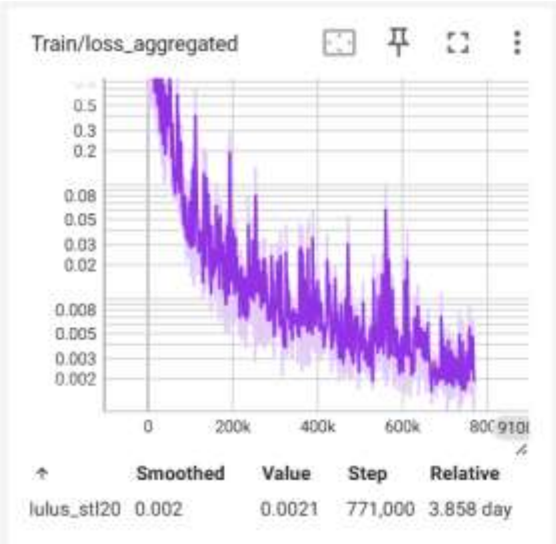
Dim : **w** x **h** x **L** m

Mat = E : 210e9 Pa

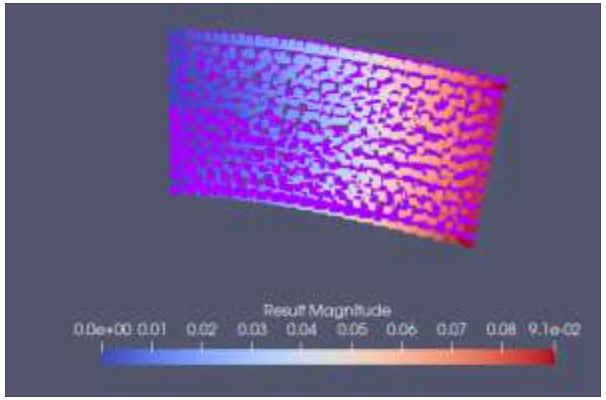
nu : 0.3

L, w, h are parametric lengths:
L used for training: 0.5, 1, 2 m.
w used for training: 0.1, 0.2, 0.3 m.
h used for training: 0.15, 0.25, 0.35 m.

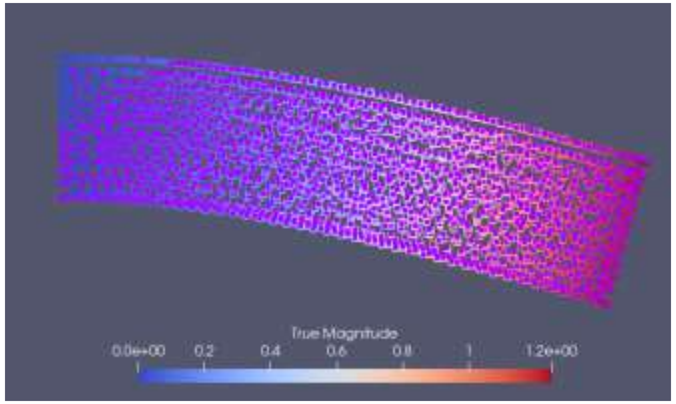
The loss function for this study is optimized from the physical equations, free end displacements, and intermediate length displacements.



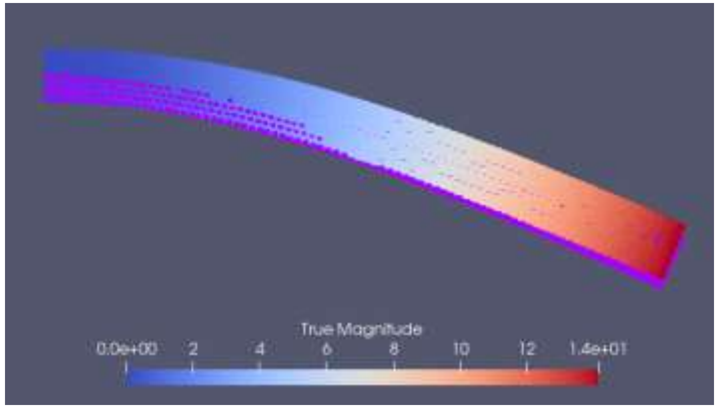
# Predicted result on the same size used for training




0.2, 0.25, 0.5




0.1, 0.25, 1

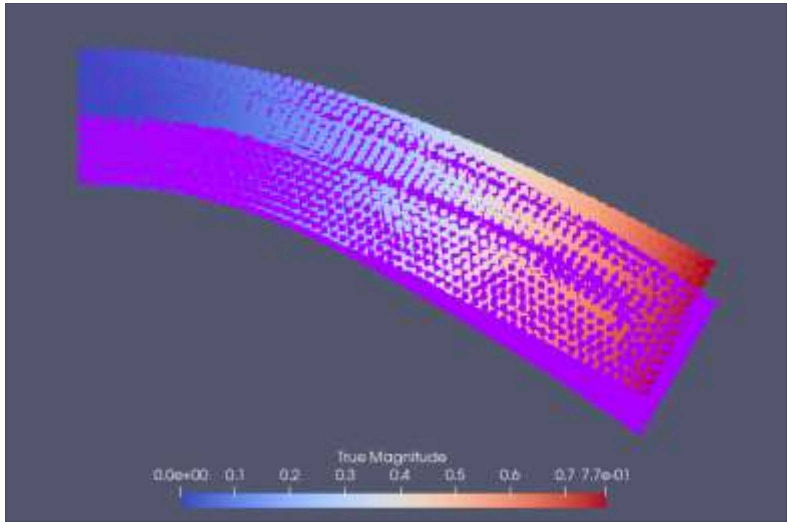



0.3, 0.15, 2

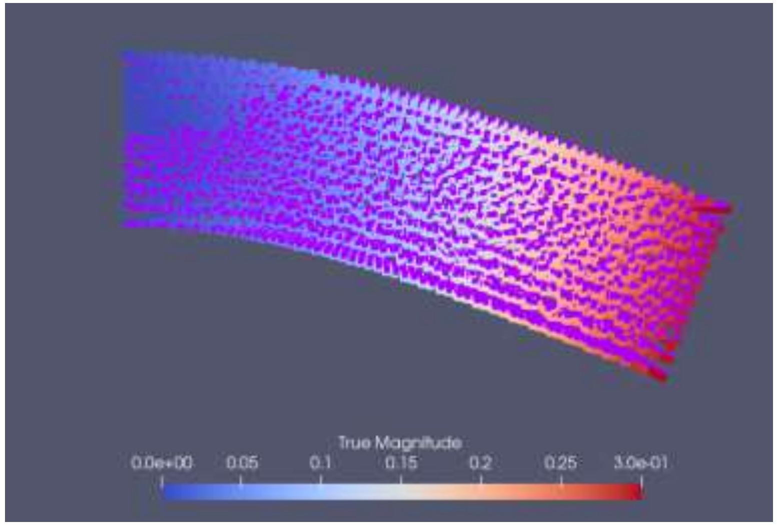
color contour : FEA result
magenta dots : PINNs result

Basically, the PINNs results are close to the FEA, although some differences can be observed in some points.
However, when comparing the FEA results with the PINNs results, the PINNs results are better compared to the theoretical values. (due to the theoretical value reference, not the FEM solution).

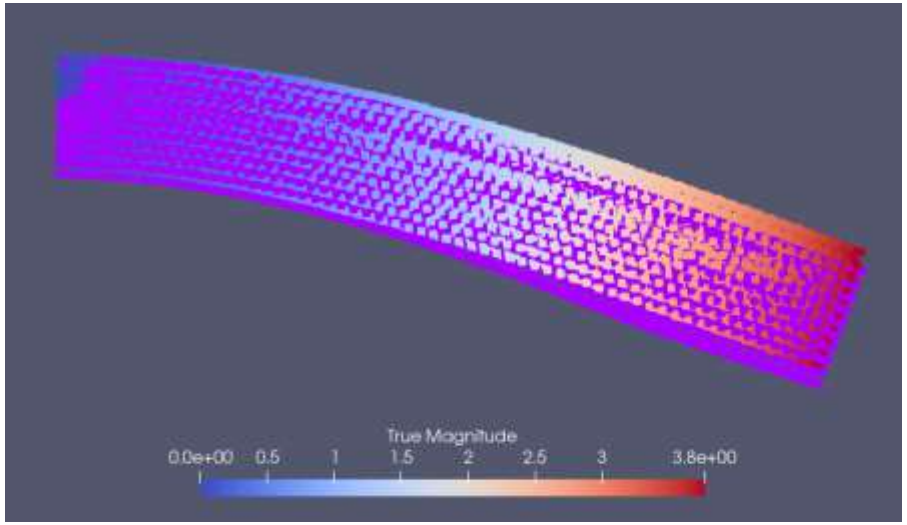
# Prediction results for random shape sizes




(0.255, 0.322, 1.5)




(0.255, 0.222, 0.75)




(0.155, 0.222, 1.5)

color contour : FEA result
magenta dots : PINNs result

Results of randomly predicting unknown shape dimensions using the weights and parameters used in training. Good predictions at 0.75 m length, but more errors occur at 1.5 m length.

# Conclusion.

- NVIDIA Modulus could be used for PINNs for simple mechanical problems.
- Training the network on a geometry containing several shape parameters may allow instant prediction of the solution for an unknown geometry.
- Training takes a long time, but training time can be reduced if some data (FEA/experiments) is referred to for training.

# The end