

Solution of structural mechanic's problems by machine learning

Himanshu Gaur^{1,2}, Basim Khidhir² & Ram Kishore Manchiryal²

¹Institute of Structural Mechanics, Bauhaus–Universität Weimar, Marienstrasse 15, D-99423 Weimar, Germany

²Middle East College, Knowledge Oasis Muscat, P.B. No. 79, Al Rusayl, Muscat, 124, Sultanate of Oman

Abstract

This article proposes analysis procedure of structural mechanic's problem as integral formulation. The methodology is novel which can be suitable applied for finding the solution of engineering problems with required accuracy either it is linear or nonlinear range (plastic range) of the material behaviour. This methodology, which was proposed as stress-based analysis procedure, exploits the unfolded part of the structural analysis problems which were not so easy to solve such as geometric and material nonlinearity together with simple integration technique [11]. In fracture mechanics, it has already unfolded the misery of physically exploiting plastic behaviour of structures before the start of crack for elastic materials [13]. The formulation is integral formulation rather than differential formulation in which whole stress –strain behaviour is utilised in the analysis procedure by using neural network as regression tool. In this article, one dimensional problem of uniaxial bar, beam bending problem and plane strain axis symmetric problem of cylinder subjected to internal pressure is solved. The results are compared with the existing differential formulation or linear theory.

Key Words: Computational methods, Continuum mechanics, Machine learning, Plastic analysis, Elastic materials.

1. Introduction:

Almost all problems in physics and engineering are derived/dealt based on differential formulation in which each problem, based on the physics, has its differential formulation. In recent years, a considerable development is seen in finding the solution of differential equations by machine learning [17-19]. Applications of machine learning, that are of interest in computational mechanics is based on the solution of differential equations by collocation method and was first developed by [1]. Recently, its applications are further extended to solve many other problems in structural mechanics [2 -6, 20, 28].

Machine learning, the name itself, demonstrates that the computer learns from the available data. Artificial neural network trains the model, for the given data, and are known to make very accurate predictions. It has shown wide applications in different fields such as voice and face recognition, business and commercial applications etc. [14-16]. This methodology utilizes regression, the ability of neural network to predict accurate numerical value based on

the available data. The importance of the methodology is observed when it solves linear as well nonlinear analysis of the structures with relative ease.

In this article, we propose the solution of physical problems for which physical response of the system is non- linear such as problems of structural mechanics. Solution based on these problems cover a wide range of engineering applications in mechanical, civil and infrastructure engineering. In these nonlinear problems, nonlinearity in the physical response may be coming from material nonlinearity or geometric nonlinearity [30-36]. For these kind of problems, we propose integral formulation rather than differential formulation. For the physical nonlinear problems, the method has proved as the most efficient approach of analysis [11].

So far, in formulating the structural problems, all the problems are modelled as displacement based [37-42]. In these problems, the material property is inputted as elastic modulus which is just the ratio of stress over strain. This formulation favours the displacement based or linear analysis procedure. Hence this structural analysis procedure is basically differential formulation. This type of formulation leads difficulty for finding the solution of problems when material behaviour is nonlinear or enters in plastic range. Till date, this remained unfolded part of the research in which no direct solution is applicable. The difficulty can be observed when the complicated, lengthy and time consuming methods are available for finding the solution of fracture mechanics problems of elastic materials [22].

For solving the structural mechanics problems, we utilize the capability/property of neural network of properly mapping the nonlinear data. This is done with the help of more than one layer (hidden layers) and activation functions. The work of neural networks is essentially to optimize weights & biases such that to give accurate output. For different applications different neural networks are being used and they are named differently in the literature [23]. For our application in continuum mechanics, we wish to predict accurate output in our numerical method, so, we have used fully connected neural network as regression tool to make accurate numerical prediction. The architecture and functioning of the neural networks for our application can be understood with the help of figures 5 and 6.

2. From differential formulation towards integral formulation:

Current state of art of structural mechanics is basically differential formulation in which for solving any typical problem, force balance equations i.e. equations of equilibrium are written which are then solved analytically or by numerical methods such as finite element method, mesh-free methods and Isogeometric Analysis (IGA) etc. [24- 27]. It is evident that this formulation has limited capability of solving the problems in nonlinear range of material behaviour.

In this article, we are proposing integral formulation of the analysis procedure for finding the solutions of the problems of structural mechanics. With the integral formulation, strain energy stored in the loaded body is directly found. In the proposed approach, first of all true stress- strain behaviour of the material is mapped to some third parameter which is called as reference coordinate system (r). In the present study, it is taken to be varied between -1 to +1. This variation of stress and strain with reference coordinate system can be viewed in figures 8 and 9 for mild steel. The objective of this mapping is to express stress and strain in terms of this third parameter (r), by which strain energy can be evaluated by simple integration. Once

strain energy of the system is found, other desired results such as strain, deformation etc. can be found suitably. The proposed analysis procedure is novel and is perhaps inspired by the derivation of Piolla Kirchhoffs stress from Euler Lagrange Strain (energy conjugates) of finite Element analysis [29].

3. Methodology- the basic formulation:

As explained previously, the proposed approach is the integral formulation. By integral formulation, we mean to evaluate strain energy of the deformed/stressed body when subjected to loading by the following integration.

$$\text{Strain Energy} = \int_V \boldsymbol{\sigma} \boldsymbol{\varepsilon} dV \quad (1)$$

Here ' $\boldsymbol{\sigma}$ ' and ' $\boldsymbol{\varepsilon}$ ' are the stress and strain tensors generated in a three dimensional structural body in Cartesian coordinates [8]. Figure 1 shows a typical structure loaded with ' F ' at the boundary Γ_F . It has fixed boundaries at ' Γ_o '. Under the application of external loads, total potential (Π) of the structural body can be expressed as follows.

$$\Pi = U - W \quad (2)$$

Here, ' U ' is strain energy stored in the deformed/stressed body which can be evaluated by equation (1) and ' W ' is the work done by the external loads which can be evaluated by the expressions below.

$$W = \int_{\Gamma_F} \mathbf{F} d\mathbf{u} \quad (3)$$

Here, ' \mathbf{F} ' and ' \mathbf{u} ' are force and displacement vectors in three dimensional coordinate system.

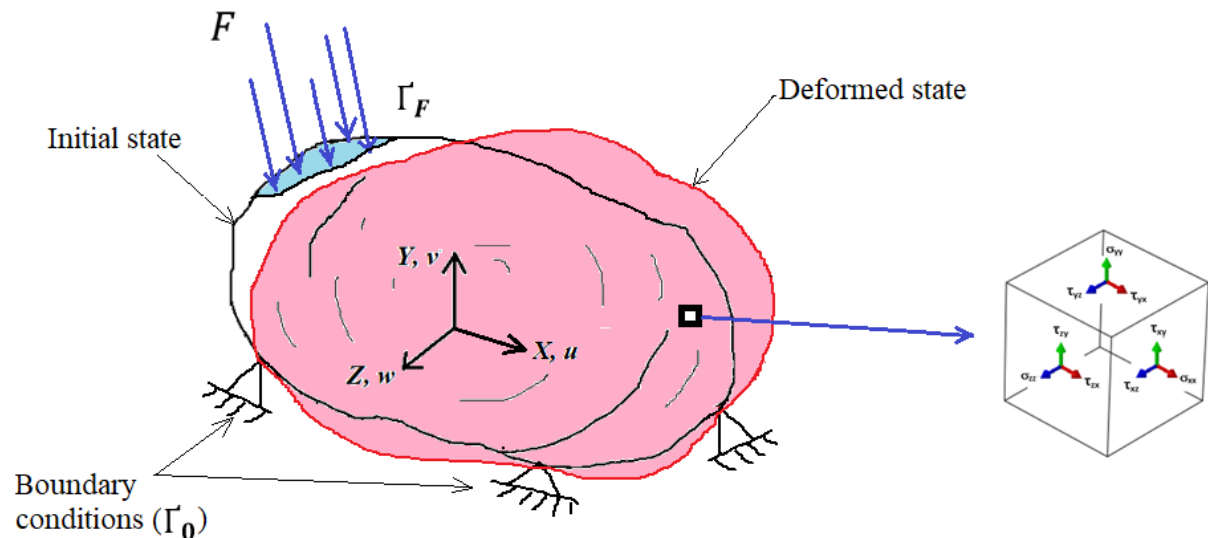


Figure.1: A typical three dimensional body subjected to loading.

It will be interesting to observe how integration of equation (1) is performed for finding the solution of any typical structure. For the structure loaded such that the material deforms in plastic range, such as the bar with the loading condition shown in figure 10. In these cases,

stress and strain components may vary across the volume of the body and hence to evaluate the multiplication, numerical integration is performed in basic Python.

Considering the special case of uniaxial rod of sectional area 'A' and length 'L', loaded in x-direction with the load of 'F_x' as shown in figure 2. As there will be axial stress σ_x and strain ε_x for the given loading condition, total potential of the loaded bar of equation (2) can be expressed as,

$$\Pi = \int_V \sigma_x \varepsilon_x dV - \int_A F_x du_x \quad (4)$$

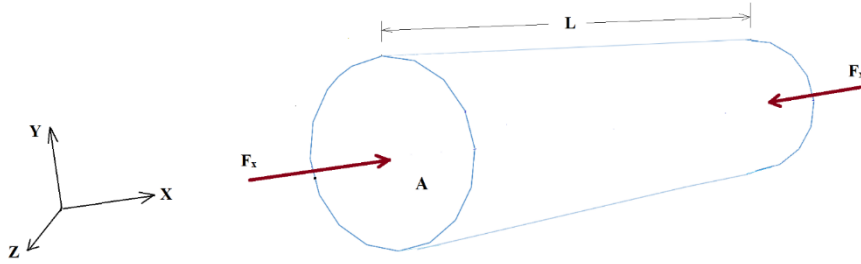


Figure. 2: A cylindrical bar subjected to axial loading.

Here 'u_x' refers to the deformation of the bar in x- direction. Total potential of the bar can further be simplified as follows.

$$\Pi = A \int_L \sigma_x \varepsilon_x dx - \int_A F_x du_x \quad (5)$$

Applying stationarity condition $\delta\Pi = 0$ [7],

$$0 = A (\sigma_x \varepsilon_x) - F_x \frac{\partial u_x}{\partial x} \quad (6)$$

Since $\frac{\partial u_x}{\partial x} = \varepsilon_x$, further simplifying,

$$A \sigma_x = F_x$$

Or,

$$\sigma_x = \frac{F_x}{A} \quad (7)$$

This is force balance equation which is used in the methodology to find reference coordinate system (r), once the stress distribution in the structural body is defined with proper/assumed structural idealization and modelling assumptions.

3.1. Modelling & Structural Idealization:

The proposed methodology is regarded as the stress based analysis procedure as it is required to know the stress distribution within the body of the structure so that strain energy can be calculated [11]. Stress distribution with in the loaded body is determined from the proper structural idealization and hence the accuracy of the solution depends upon the proper modelling assumptions.

Our assumptions for modelling and structural idealizations are same as found in the strength of material theory. For example, Euler-Bernoulli's beam bending theory, torsion theory etc. Hence, the result obtained from the methodology can suitably be applied and used for the engineering applications.

Once the stress distribution with in the loaded structural element is determined, strain energy stored within the body can simply be evaluated by the equations as explained in the methodology part.

3.2. Definition of stress and strain:

As the proposed integral approach is a complete approach i.e. the results are valid for linear as well as nonlinear range of the material behaviour. We are using true stress and true strain in order to accommodate the behaviour of large deformation. Therefore, it should be noticed that the results are valid before necking starts in the tension specimen of elastic material used for measuring stress-strain behaviour.

4. Analysis procedure by Machine Learning as a regression tool:

Neural network is used as the nonlinear regression tool in the analysis procedure. For any structural problem, geometric parameters such as area of cross-section, length do not produce nonlinearity in the solution. Nonlinearity essentially comes from the material behaviour. Therefore, in the neural network, material property is the only parameter that is trained in the neural network.

The basic idea behind the analysis procedure is to find reference coordinate system (r), or to say, locating the point on stress-strain diagram (figure 3). Force balance equation such as equation (7) is used to find reference coordinate system (r) from the known stresses. Once reference coordinate is known, strain can also be located by stress-strain graph. Once stress and strain are known, strain energy stored in the structural body can simply be evaluated by integration.

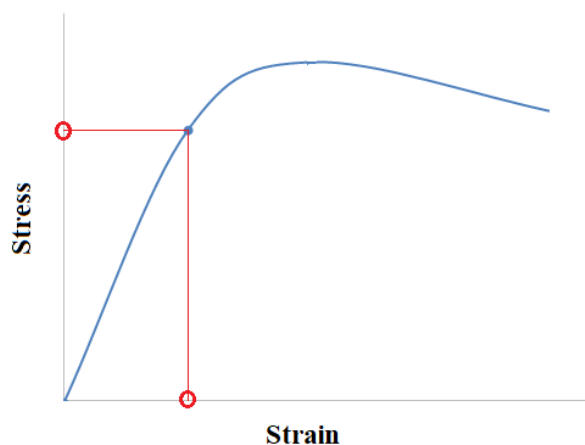


Figure. 3: A typical stress- strain behaviour of material.

Overall analysis procedure of the methodology is summarized with the help of flow chart shown in figure 4.

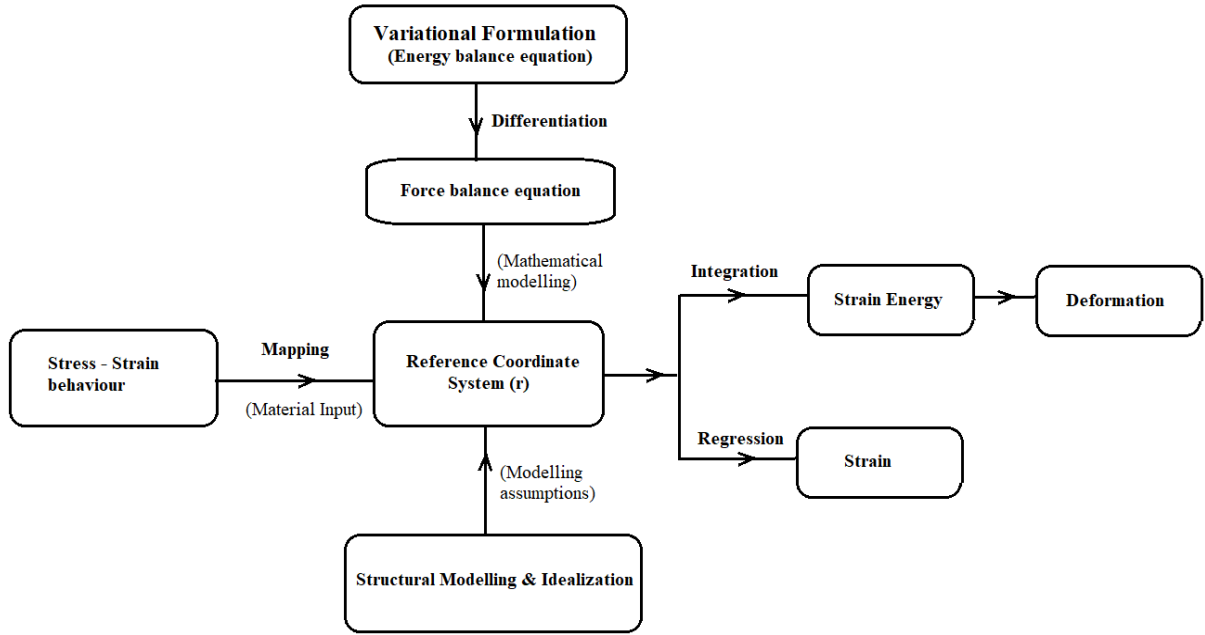


Figure. 4: Flow chart of overall analysis procedure employed.

4.1. Construction of Neural Network:

Figure 5 shows fully connected feed forward neural network that is used in the analysis. Three hidden layers are found to be performing well in comparison to the two, or more than three hidden layers. It is found that changing the number of neurons in the hidden layer did not give any influence in the output. In this study, 64 neurons are used for each hidden layer. Nonlinear mapping is achieved by the ReLU activation function which is used for all hidden layers. For the last hidden layer, which yields the output layer, linear activation function is used (figures 5 & 6).

First of all, for each neuron weights are initialized randomly as per standard normal distribution and all biases are set to zero. Python code used for generating a layer can be glanced underneath.

```

class Dense_layer:
    def __init__(self, n_of_inputs, n_of_neurons):
        self.weights = np.random.randn(n_of_inputs, n_of_neurons)
        self.biases = np.zeros((1, n_of_neurons))

    def forward(self, inputs):
        self.inputs = inputs
        self.output = np.dot(inputs, self.weights) + self.biases
  
```

Data flow following the code can be glanced with the help of neural network drawn in figure 5. In this neural network output from any particular neuron ‘i’ in the 1th layer can be expressed as follows.

$$\sigma_i^l = \sum_{j=1}^{n_{l-1}} a_{l-1}^j (w_{i,j}^l * \sigma_j^{l-1} + b_i^l) \quad (8)$$

Here σ_j^{l-1} is the input from each neuron of the previous layer ($l-1$) to any particular neuron of the current layer (l). b_i^l is the bias used for neuron ‘i’ in layer l and a_{l-1} is the activation function used for the layer $l-1$.

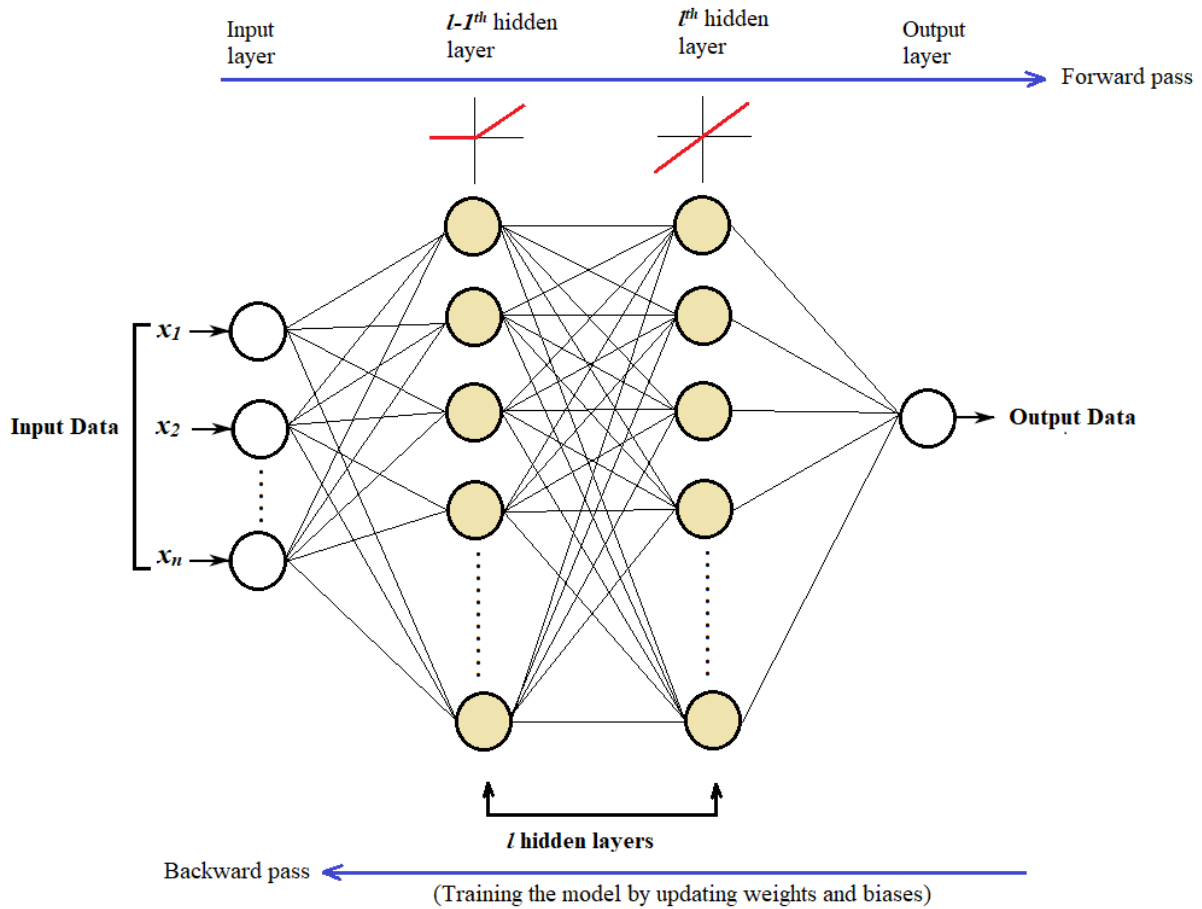


Figure. 5: Architecture and functioning of Neural Network.

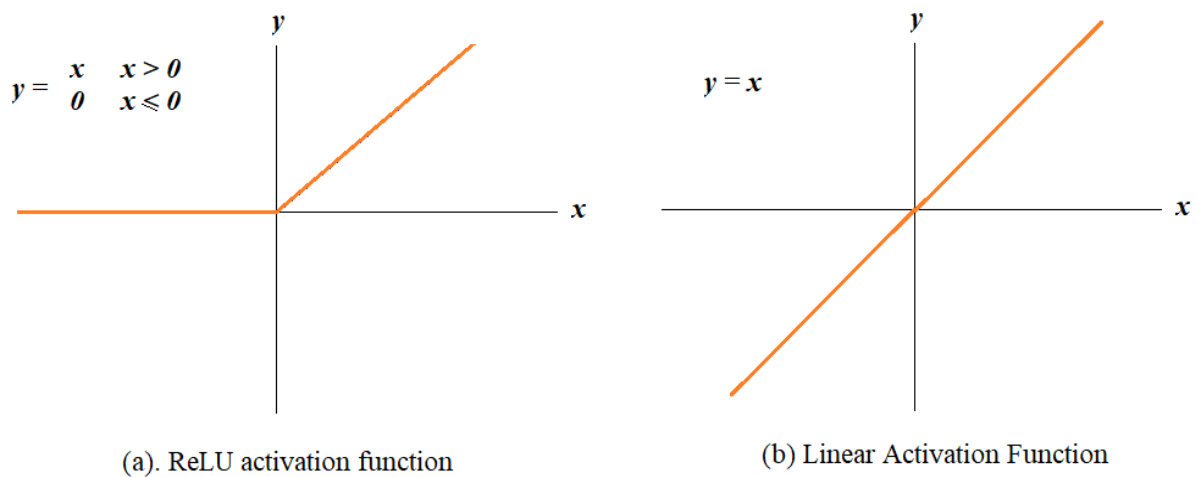


Figure. 6: Schematics of (a) ReLU activation and (b) Linear activation functions

Because of the randomly choosing weights and biases, output of the forward pass yields huge error which has to be minimized by properly tuning weights and biases that is done in the backward pass. Backward pass is performed for training the neural network, or to say adjusting/tuning the weights and biases to produce the accurate output. This is done based on error function (loss or cost function) which is taken as root mean square error (MSE). This root mean squared error is evaluated by the following mathematical formula.

$$\text{Loss function (MSE)} = \mathcal{L}(\hat{y}, w, b) = \frac{1}{n} \sum_{i=1}^n [Y_i - \mathcal{N}(\hat{y}, w, b)]^2 \quad (9)$$

Here, Y_i are the target values which neural network is supposed to predict and $\mathcal{N}(\hat{y}, w, b)$ are the current predicted values by the neural network. Mean square error of equation (9) is the function of inputs (\hat{y}), weights (w) and biases (b). Python code employed for evaluating loss is given below in the snippets.

```
class Loss_calculations:
    def evaluate(self, output, y):
        sample_losses = self.forward(output, y)
        data_loss = np.mean(sample_losses)
        return data_loss

class Loss_Mean_Squared_Error(Loss_calculations):
    def forward(self, y_pred, y_true):
        sample_losses = np.mean((y_true - y_pred)**2, axis=-1)
        return sample_losses
```

Weights and biases are updated in the backward pass. For this purpose, gradient ' $\nabla \mathcal{L}$ ' of this loss function is evaluated.

$$\nabla \mathcal{L}(\hat{y}, w, b) = \begin{bmatrix} \frac{\partial}{\partial \sigma} (\mathcal{L}(\hat{y}, w, b)) \\ \frac{\partial}{\partial w} (\mathcal{L}(\hat{y}, w, b)) \\ \frac{\partial}{\partial b} (\mathcal{L}(\hat{y}, w, b)) \end{bmatrix} \quad (10)$$

Derivatives of this loss function with respect to weights and biases are used to update weights and biases, and derivatives with respect to inputs are used to chain with the previous layer [9]. Adam (Adaptive momentum) optimizer is used to train the model [10].

In order to check the accuracy in regression model, a limiting value is defined which is standard deviation of the target values. This standard deviation is divided by 250 (our choice, this value defines the desired precision in obtaining the accuracy). Following python code is implemented in order to check the accuracy of the output from the neural network.

```
# defining the limiting value as standard deviation of the target values
limiting_value = np.std(target_values) / 250
# predictions are the output of the neural network
predictions = activation3.output
# Accuracy is measured with the mean of absolute values obtained from the following inequality.
# False counts to zero and True counts to one
accuracy = np.mean(np.absolute(predictions - target_values) < limiting_value)
```

It is found that, with this code, even 80 % accuracy gave satisfactory results that were required in the analysis. Full Python code used in the study can be accessed from this Github page: <https://github.com/hgaur0007/Solution-of-Structural-Mechanics-problems-by-ML->

4.2. Implementation of Neural Network in the analysis:

In the analysis procedure basically two neural networks are used. One, which maps true stress to reference coordinate system, and is expressed by equation (11) and the another one, which maps reference coordinate system to true strain, and is expressed by equation (12). Flow chart of the analysis procedure using neural network as regression tool can be viewed in figure 7. Neural networks are trained with 100 % data available from stress-strain behaviour and mapping. The variation of reference coordinate system (test data which is different from the training data) with true stress can be observed in figure (7) and the variation of true strain (test data) with reference coordinate system can be observed in figure (8).

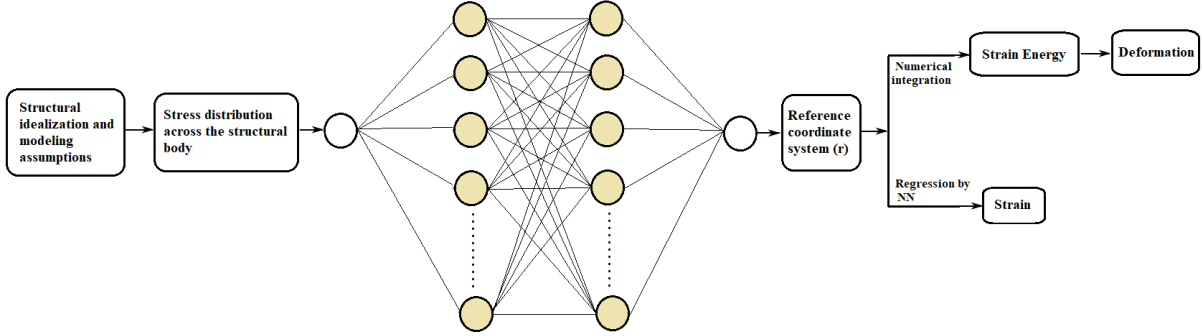


Figure. 7: Flow chart of analysis procedure with neural network as regression tool.

As explained previously, for solving the problem, the methodology requires proper stress distribution across the structural body with proper structural idealization and modelling assumptions. Once stress distribution in the structural body is defined, corresponding reference coordinate (r) is determined from the output of the neural network expressed by the following equation.

$$r = \mathcal{N}^r(\sigma, w^r, b^r) \quad (11)$$

Here, \mathcal{N}^r referees the neural network used for regression analysis from stress to reference coordinate (r), ' σ ' refers to the true stress (input value), ' w^r ' and ' b^r ' refers to weights and biases of the neural network obtained after running the optimization by Adam optimizer. For training this neural network, weights and biases were updated by around 10000000 times (iterations). Even 80 % accuracy gave satisfactory results. Accuracy of the output values from this neural network could be observed in figure 8.

Once, reference coordinate (r) is obtained, true strain can be obtained with the regression analysis of neural network (\mathcal{N}^ε) which maps reference coordinate to strains expressed by the following equation.

$$\varepsilon = \mathcal{N}^\varepsilon(r, w^\varepsilon, b^\varepsilon) \quad (12)$$

Here ' w^ε ' and ' b^ε ' are weights and biases of this neural network after training and ' r ' is the input value of neural network. This mapping is linear which could be observed with the help of figure 9. Around 1000 iterations sufficiently/satisfactorily tuned the weights and biases which produced accurate results.

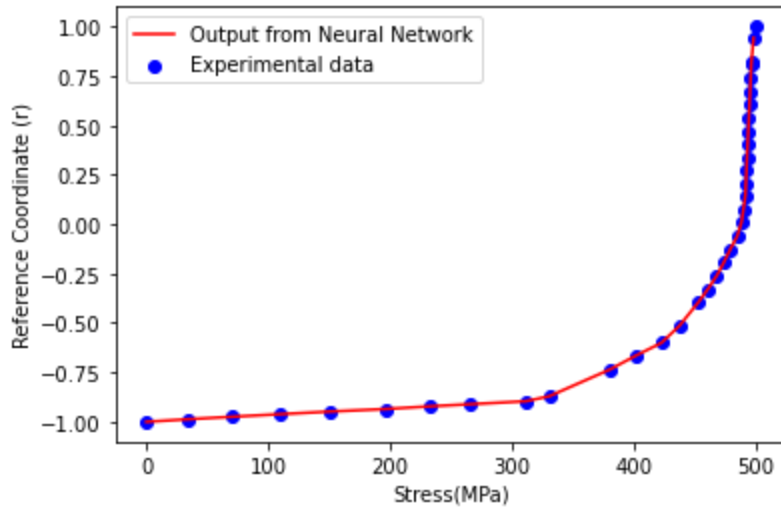


Figure. 8: Variation of true stress with reference coordinate system (r) of mid steel.

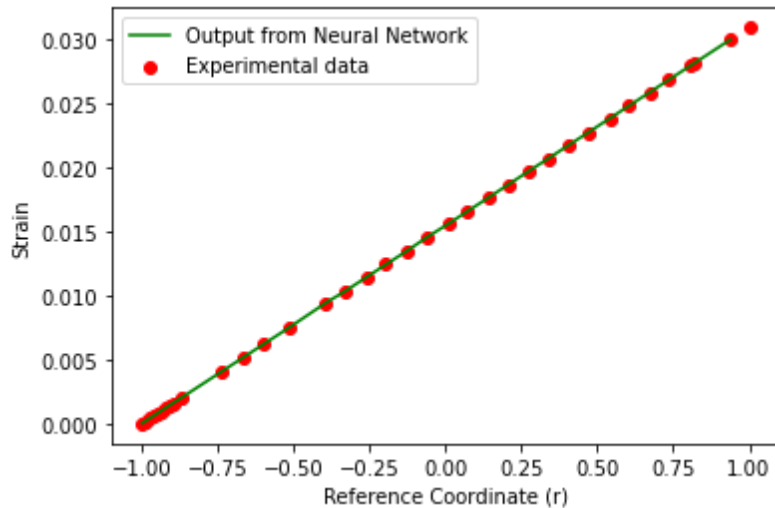


Figure. 9: Variation of true strain with reference coordinate system (r) of mid steel.

5. Numerical problems:

In this section, uniaxial bar loaded with point and uniform load acting axially, beam bending problem and plane strain axis symmetric problem of a long cylinder subjected to internal pressure are solved.

5.1. Uniaxial bar loaded with varying load:

Figure 10 shows uniaxial bar loaded at the free end by the traction force ' F ' acting at the free end and a uniform load ' $f(x)$ ' acting along the length of the bar downward in X-direction. Considering the magnitude of load combination is such that the material of the bar

experiences plastic deformation. One of the possible stress distribution in this condition is shown in figure 10 (b). Stress at the free and fixed end can be evaluated by the following expressions.

$$\sigma_{free\ end} = \frac{F}{A} \quad (13)$$

$$\sigma_{fixed\ end} = \frac{F+f(x)L}{A} \quad (14)$$

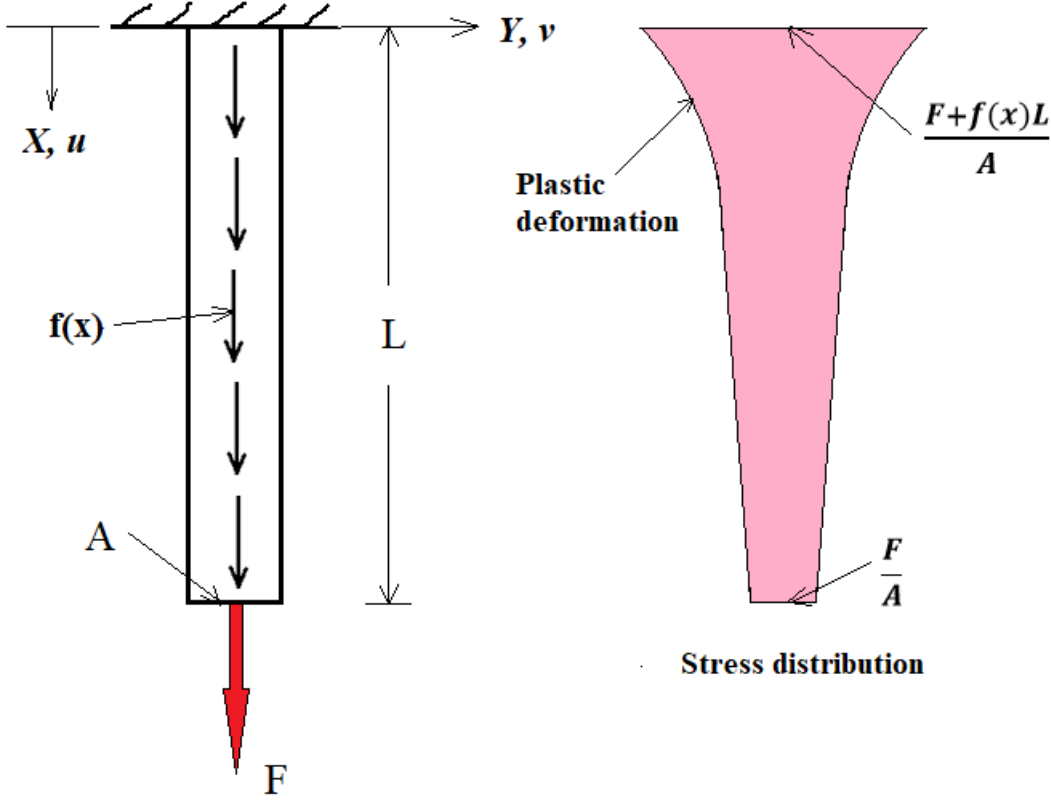


Figure 10: (a). Uniaxial bar with loading condition. (b) stress distribution across the length of the bar is shown.

Once the stress distribution across the body is defined, strain energy stored in the body can be evaluated by volume integral of equation (1). As per the loading, the only stress which is acting in the bar is ' σ_x ' and at any particular section stress remains constant throughout the area, this strain energy expression is modified as follows.

$$Strain\ energy = A \int_L \sigma_x \varepsilon_x dx \quad (15)$$

As the stress varies across the length, this volume integral can further be modified as follows.

$$Strain\ energy = AL \int_{r_1}^{r_2} \sigma_x \varepsilon_x dr \quad (16)$$

Here, the multiplication $\sigma_x \varepsilon_x$ in integrated because it is not constant and varies across the length with the lower and upper bound of r_1 and r_2 which can be obtained from the neural network N^r , expressed with the equation (11) with known stress values (σ_x) across the length of the bar by expressions (13) and (14). True strain (ε_x) can also be obtained by the neural network N^ε expressed with the equation (12) by known values of reference coordinate.

Numerical integration is performed with the Python code. Once, strain energy stored in the bar is evaluated, it's deformation can be evaluated by the Castigliano's Theorem expressed as follows.

$$D_i = \frac{\partial U}{\partial F_i} \quad (17)$$

Analytical solution (deformation at the free end) by differential formulation (linear theory) of this problem is given by the following equation [12].

$$u = \frac{f(x)}{EA} \left(Lx - \frac{x^2}{2} \right) + \frac{F}{EA} x \quad (18)$$

Considering area of cross section of the bar as 20 cm² and length as 50 cm. In this analysis, uniform load $f(x)$ is kept constant and taken as 0.4 MN/m. Point load at free end varies in magnitude. Variation of strain energy as well as deformation of the bar at free end, versus the point load 'F' can be glanced in figures 11 and 12.

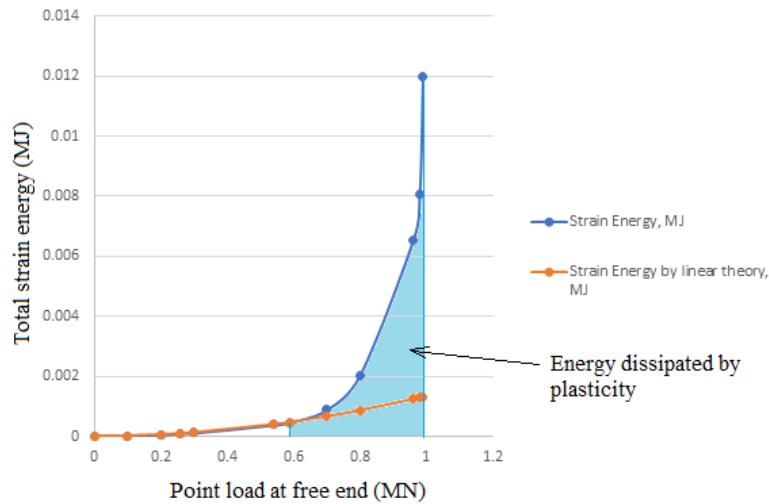


Figure 11: Variation of strain energy versus the point load 'F' at free end. Statics of energy dissipated in plasticity is shown.

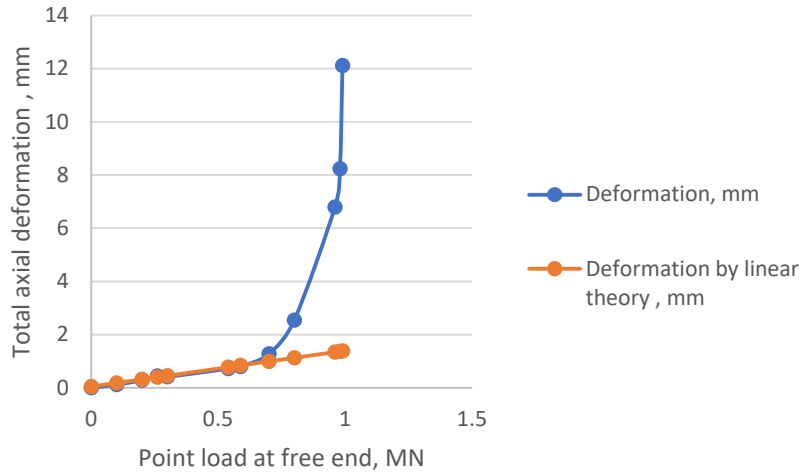


Figure 12: Variation of deflection at free end versus the point load ‘F’ at free end.

Results of this uniaxial bar are compared with the linear theory in which Young’s modulus is taken as 184128.9683 MPa which is evaluated from the stress –strain curve of the material. It is important to notice the results in linear range, which are very accurately matching with linear theory. In nonlinear range, strain energy as well as deformation of the bar increases rapidly, this is because of the plastic deformation of the bar as per the material behaviour. Strain energy dissipated because of plasticity is also shown in figure 11.

5.2. Two Dimensional Problem of Beam Bending:

Figure 13 shows a two dimensional problem of a simply supported beam of length ‘L’, width ‘b’ and depth ‘h’, loaded with a concentrated load ‘F’ at the mid span.

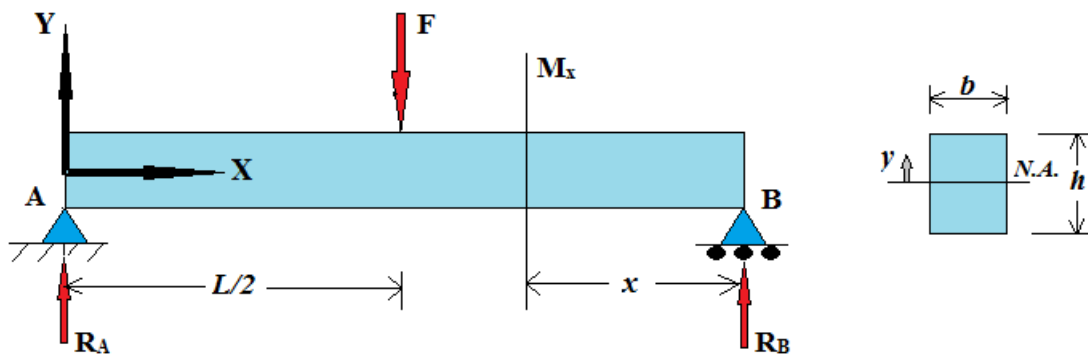


Figure 13: Simply supported beam with concentrated load at mid span.

Current state of art of beam bending theory assumes linear variation of the stresses across the depth of the beam. Interestingly, this methodology is capable of analyzing structures in nonlinear range of the material behavior as well. If the beam is loaded such that it generates stresses corresponding to plastic/nonlinear range of the material behavior, it is reasonable to understand that the stress variation across the depth of the beam at any particular section would resemble the variation shown in figure 14.

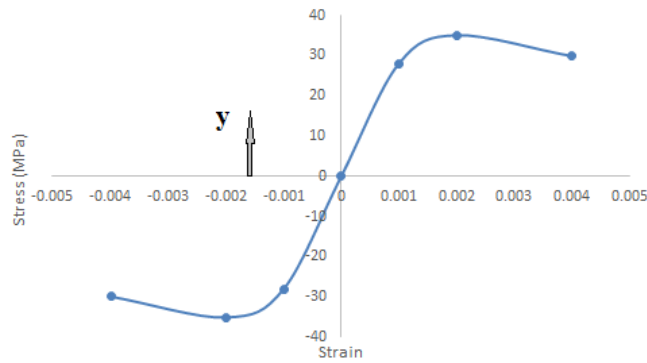


Figure 14: Variation of stress across depth of the beam at a particular section.

Stress variation across the length (for the simply supported beam of figure 13) could be visualized with the help of figure 15. Its variation simply depends upon the variation of bending moment across the length.

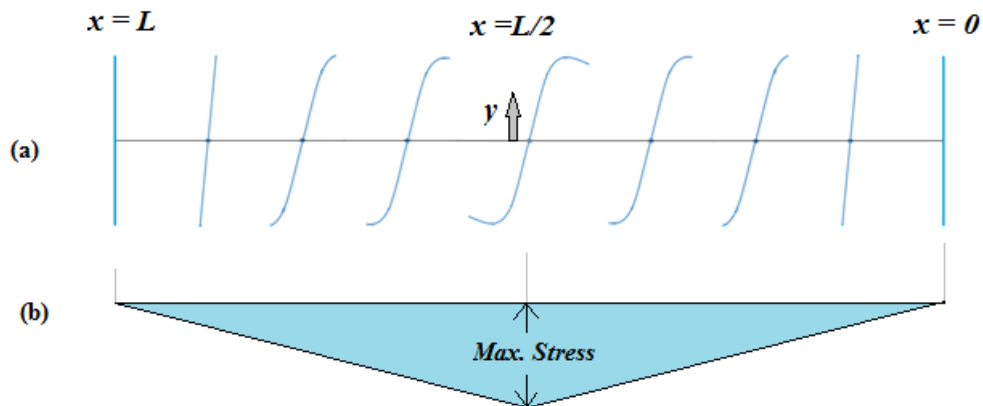


Figure 15: Variation of stress along the depth (a) and length (b) at different sections of the beam.

Since the simply supported beam shown in figure 13 is loaded with the concentrated load at the mid span, its bending moment at any particular section within the range $0 \leq x \leq \frac{L}{2}$ can be calculated as,

$$M_x = R_B x \quad (19)$$

Where, R_B is the reaction at support. Let M_x be the bending moment at any particular section. Considering stress ' σ_x ' in a fiber at any particular depth ' y ' from neutral axis, bending moment at that section can be calculated as,

$$M_x = \int_A \sigma_x y dA \quad (20)$$

or,

$$M_x = \int_A \sigma_x y dz dy$$

Or,

$$M_x = b\sigma_x \int_{-h/2}^{+h/2} y dy = b\sigma_x \left[\frac{y^2}{2} \right]_{-h/2}^{+h/2} = \frac{bh^2}{4} \sigma_x$$

Rearranging,

$$\sigma_x = \frac{4M_x}{bh^2} \quad (21)$$

It is to be noted that $\frac{bh^2}{4}$ resembles with moment of inertia of the section $\left(\frac{bh^3}{12}\right)$ which is found in the derivation of beam bending equations with linear theory of elasticity. Here $\frac{bh^2}{4}$ avoids any linearity assumption in derivation.

Now, bending energy stored in the beam can be calculated as [8],

$$\text{Bending energy} = \int_V \sigma \varepsilon dV \quad (22)$$

As the stress distribution across the width 'b' remains constant at any section, bending energy can be evaluated as,

$$\text{Bending energy} = b \int_A \sigma \varepsilon dx dy$$

Now, considering stress variation across the length of the beam as depicted in figure 15.

$$\text{Bending energy} = b \frac{L}{2} \int_y \sigma \varepsilon dy$$

Now, considering stress variation across the depth of the beam as portrayed with the help of figure 14. Stress varies from zero at neutral axis ($r = -1$) and it becomes maximum at the outer layer. Hence, for this variation of stress, the integral modifies as follows.

$$\text{Bending energy} = 2 * b \frac{L}{2} * \frac{h}{2} \int_{-1}^r \sigma_x \varepsilon_x dr \quad (23)$$

Here, the integral is multiplied by $\frac{h}{2}$ as it is calculating strain energy starting from neutral axis to the extreme fibre, either in compression or tension zone. The expression is multiplied again by two because same energy will be stored, either it is compression or tension zone.

Considering length of the beam as $L = 2 \text{ m}$, $b = 300 \text{ mm}$ and $h = 400 \text{ mm}$. If the beam is loaded with concentrated load of magnitude 2 MN, maximum stress generated (σ_x) at the extreme fiber of the middle span ($x=1\text{m}$) can be evaluated by equations (19) and (21). For the known stress, corresponding value of reference coordinate system (r) could be obtained by neural network \mathcal{N}^r , expressed by the equation (11). Once the value of reference coordinate value (r) is known, corresponding value of strain (ε_x) could be evaluated by the neural network \mathcal{N}^ε expressed by equation (12). Finally, strain energy stored in the beam bending is evaluated by expression (23) for different magnitudes of load. Deformation in the beam at the mid span is evaluated by Castigliano's Theorem expressed with equation (17). The results are compared with the linear theory and can be glanced the figures (16) and (17). Deformation at mid span for this beam with linear theory is evaluated by following expression of equation (24)

$$y = \frac{FL^3}{48EI} \quad (24)$$

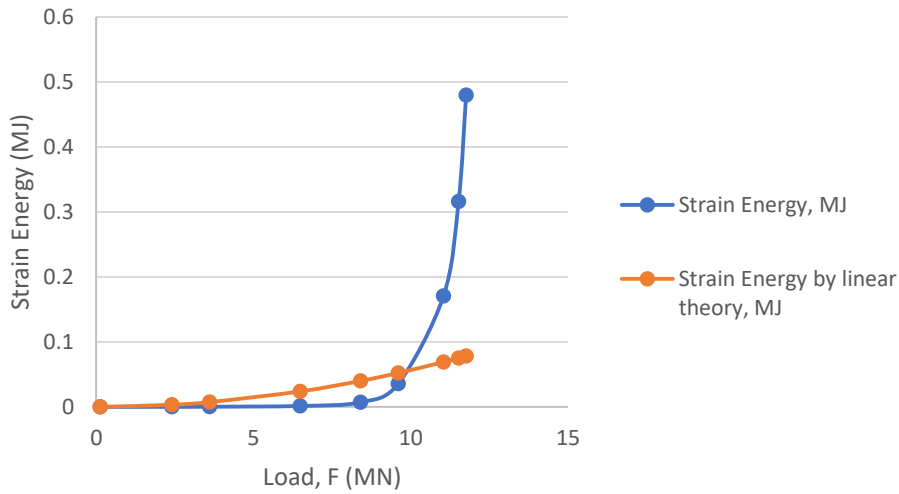


Figure 16: Bending energy stored in the beam for different magnitudes of loads.

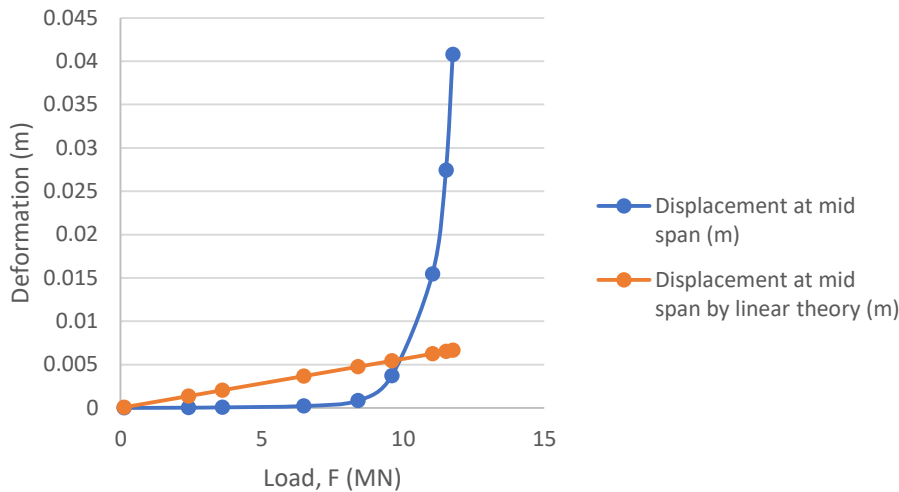


Figure 17: Deflection at the mid span of the beam for different magnitudes of loads.

It is worthy to note that this approach of structural analysis has given very accurate results which resembles material behavior (stress-strain curve). In comparison to linear theory, which is valid until the linear range of the material behavior, this method is giving accurate results not only in linear range but in nonlinear range as well.

5.3. Plane strain axis-symmetric problem:

Consider a long cylinder of inner and outer radius as ‘a’ and ‘b’ respectively, which is subjected to internal pressure of ‘ p_i ’. Figure 18 shows a cross section of the cylinder in XY-plane in which length of the cylinder is along Z- axis. This is quite a practical problem simulating an infinitely long pipe line subjected to internal pressure from the liquid. This is a plane strain axis symmetric problem in which deformation is in radial direction only and is constant along the circumferential direction. From the force equilibrium equations, variation

of radial (σ_ρ) and hoop (σ_θ) stresses along the thickness of the cylinder are derived as follows [21].

$$\sigma_\rho = \frac{a^2 p_i}{b^2 - a^2} \left(1 - \frac{b^2}{\rho^2}\right) \quad (25)$$

$$\sigma_\theta = \frac{a^2 p_i}{b^2 - a^2} \left(1 + \frac{b^2}{\rho^2}\right) \quad (26)$$

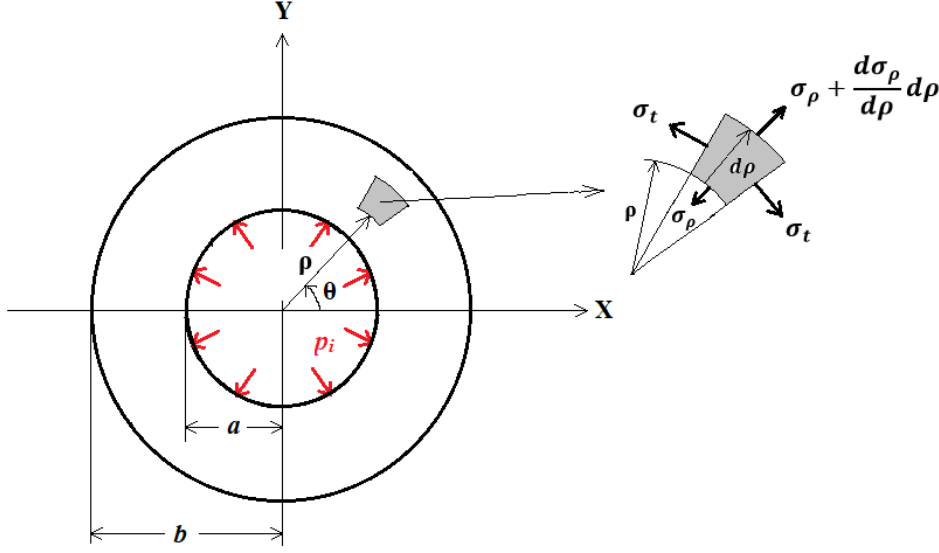


Figure 18: Section of the cylinder across Z- plane.

From equations (25) and (26), as there are two different stresses acting, strain energy stored in the cylindrical body can be evaluated as follows.

$$U = U_\rho + U_\theta \quad (27)$$

Here ' U_ρ ' is the strain energy stored because of radial stress and ' U_θ ' is the strain energy stored because of hoop stress. Strain energy stored per unit length of the cylinder because of radial stress only (U_ρ), can be evaluated as,

$$U_\rho = \int_A \sigma_\rho \varepsilon_\rho dA \quad (28)$$

As, stresses varies from inner surface to outer surface, this integration can be modified as,

$$U_\rho = \int_A dA \int_{r_1}^{r_2} \sigma_\rho \varepsilon_\rho d\rho \quad (29)$$

Here, r_1 and r_2 are the values of reference coordinate system at inner and outer layer of the cylinder which are obtained from the output of the neural network of equation (11) by input values of stress from equation (25) at inner and outer layer respectively. Further simplifying the expression of equation (29).

$$U_\rho = \pi(b^2 - a^2) \int_{r_1}^{r_2} \sigma_r \varepsilon_r dr \quad (30)$$

Similarly, strain energy stored because of hoop stress, per unit length of the cylinder, can be evaluated by the following expression.

$$U_{\theta} = \pi(b^2 - a^2) \int_{r_1}^{r_2} \sigma_{\theta} \varepsilon_{\theta} dr \quad (31)$$

Here, r_1 and r_2 can be obtained from the output of the neural network of equation (11) by input values of stress from equation (26) at inner and outer layer respectively. Numerical integration is performed in basic Python code. Once total strain energy of the system by equation (27) is evaluated, displacement in radial direction is evaluated by Castigliano's Theorem of equation (17). Exact radial deformation ' D_{ρ} ' of this problem at the inner layer by the linear theory is given by the expression [21].

$$(D)_{\rho=a} = \frac{ap_i}{E} \left(\frac{a^2+b^2}{b^2-a^2} + \nu \right) \quad (32)$$

Here 'E' is Young's modulus of linear theory which is evaluated from the stress-strain curve in linear range, ' ν ' is Poisson's ratio which is taken as 0.3 for mild steel. Taking inner radius 'a' as 10 cm and outer radius as 20 cm, total strain energy of the cylinder as well as displacement in radial direction at inner layer is evaluated by Castigliano's Theorem. Radial deformation at the inner layer of the cylinder is compared with linear theory. These results can be glanced in figure (19) and (20) with varying internal pressure.

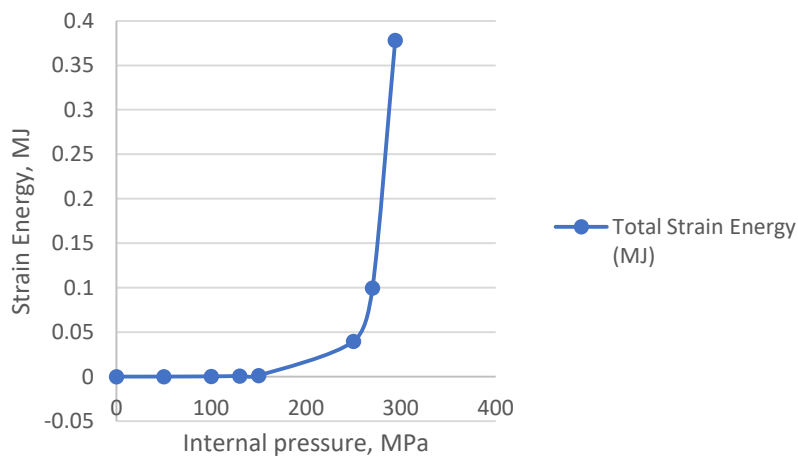


Figure 19: Variation of strain energy versus internal pressure inside the long cylinder.

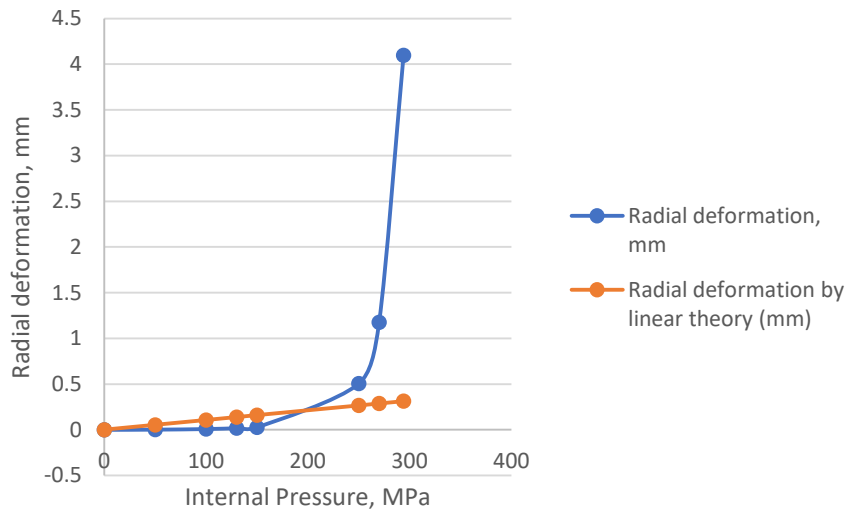


Figure 20: Variation of radial deformation as internal pressure increases inside the cylinder.

From the results of figure 20, it is important to notice the accuracy of the results which are matching with linear theory, in linear range of the material behaviour. In the plastic range, deformation explodes as per the material behaviour.

6. Conclusions:

The methodology solves the uniaxial bar, either linear or nonlinear range of the material behavior, with just one simple integration. It is important to notice the accuracy of the results which are matching with the linear theory in figures 11 and 12. The beam bending problem of simply supported beam is also solved by evaluating simple integral. Results in the linear range are in the good match with the linear theory (figures 16 & 17). From the results of the cylinder which is subjected to internal pressure only, radial deformation at the inner layer is found to be accurately matching with the results of linear theory (figure 20).

It should be noticed that finding the solution of these problem in plastic range is not a easy task. It requires step by step method in which secant modulus changes with each step, which is a long and cost consuming method. The analysis procedure also evaluates strain energy dissipated in plasticity by evaluating one simple integration (figure 11). From the results of uniaxial bar and plane strain axis symmetric problem, it is important to notice the accuracy of the results, which matched with the linear theory. This proves the validity and accuracy of the method.

Conflict of Interest: The authors declare that they have no conflict of interest.

References:

1. I.E. Lagaris, A. Likas, D.I. Fotiadis, Artificial neural networks for solving ordinary and partial differential equations, *IEEE Trans. Neural Netw.* 9 (1998) 987–1000.
2. M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* 378 (2019) 686–707.
3. E. Samaniego, C. Anitescu, S. Goswami, V.M. Nguyen-Thanh, H. Guo, K. Hamdia, X. Zhuang, T. Rabczuk, An energy approach to the solution of partial differential equations in computational mechanics via machine learning: Concepts, implementation and applications, *Comput. Methods Appl. Mech. Engrg.* 362 (2020) 112790.
4. Somdatta Goswami, Cosmin Anitescu, Souvik Chakraborty, Timon Rabczuk, Transfer learning enhanced physics informed neural network for phase-field modeling of fracture, *Theoretical and Applied Fracture Mechanics* 106 (2020) 102447.
5. Somdatta Goswami, Cosmin Anitescu, Timon Rabczuk, Adaptive fourth-order phase field analysis using deep energy minimization, *Theoretical and Applied Fracture Mechanics* 107 (2020) 102527.
6. Hongwei Guo, Xiaoying Zhuang and Timon Rabczuk, A Deep Collocation Method for the Bending Analysis of Kirchhoff Plate, *CMC*, vol.59, no.2, pp.433-456, 2019.
7. Bathe K. J., *Finite element procedures*, Second Edition, Massachusetts Institute of Technology, Cambridge, MA 02139, USA, 2014.
8. Timoshenko S. and Goodier G.N., *Theory of Elasticity*, 3rd Edition, Mcgraw Hill Education Private Limited, 1934.
9. Harrison Kinsley & Daniel Kukiela. *Neural Networks from Scratch (NNFS)*. <https://nnfs.io>
10. Kingma, D. P. & Ba, J. (2014), 'Adam: A Method for Stochastic Optimization' , cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.
11. Gaur H, Srivastav A, A novel formulation of material nonlinear analysis in structural mechanics, *Defence Technology*, <https://doi.org/10.1016/j.dt.2020.06.018>
12. Allen D.H. (2013) *Theory of Uniaxial Bars*. In: *Introduction to the Mechanics of Deformable Solids*. Springer, New York, NY. https://doi.org/10.1007/978-1-4614-4003-1_3.
13. Gaur, H., Dakssa, L., Dawood, M. et al. A novel stress-based formulation of finite element analysis. *J. Zhejiang Univ. Sci. A* 22, 481–491 (2021). <https://doi.org/10.1631/jzus.A2000397>
14. A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: *Advances in Neural Information Processing Systems*, 2012, pp.1097–1105.
15. B.M. Lake, R. Salakhutdinov, J.B. Tenenbaum, Human-level concept learning through probabilistic program induction, *Science* 350 (2015) 1332–1338.

16. B. Alipanahi, A. DeLong, M.T. Weirauch, B.J. Frey, Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning, *Nat. Biotech-nol.* 33 (2015) 831–838.
17. Lee, H. and Kang, I., Neural algorithms for solving differential equations, *Journal of Computational Physics*, vol. 91, pp. 110-117, 1990.
18. Meade Jr, A.J. and Fernadez, A.A., The numerical solution of Linear Ordinary Differential Equations by Feedforward Neural networks, *Math. Comput. Modelling*, vol. 19, no. 12, pp. 1-25, 1994.
19. Meade Jr, A.J. and Fernadez, A.A., Solution of Nonlinear Ordinary Differential Equations by Feedforward Neural networks, *Math. Comput. Modelling*, vol. 20, no. 9, pp. 19-44, 1994.
20. C. Anitescu, E. Atroshchenko, N. Alajlan, and T. Rabczuk: Artificial neural network methods for the solution of second order boundary value problems. *Computers, Materials and Continua*, 59(1):345-359, 2019.
21. Timoshenko S., *Strength of Materials*, Second edition, D. Van Nostrand Company, Inc. 1940.
22. Kumar P.: *Elements of Fracture Mechanics*, McGraw Hill Education Private Limited. P-24, Green Park Extension, New Delhi 110 016, India, 2014.
23. Abdel-Nasser Sharkawy, Principle of Neural Network and Its Main Types: Review, *Journal of Advances in Applied & Computational Mathematics*, 2020, 7, 8-19.
24. T.J. Hughes, *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Dover Publications, Mineola, NY, 2000.
25. A. Huerta, T. Belytschko, S. Fernández-Méndez, T. Rabczuk, X. Zhuang, M. Arroyo, Meshfree methods, *Encyclopedia Comput. Mech. Sec. Ed.* (2018) 1–38.
26. Nguyen V.P., Rabczuk T., Bordas S. Duot M.: Meshless methods: A review and computer implementation aspects, *Mathematics and Computers in Simulation*, 2008, 79, 763-813.
27. Nguyen-Thanh N., Zhou K., Zhuang X., Areias P., Nguyen-Xuan H., Bazilevs Y., Rabczuk T.: Isogeometric analysis of large-deformation thin shells using RHT splines for multiple-patch coupling, *Computer Methods in Applied Mechanics and Engineering*, 2017, 316, 1157-1178.
28. Zhuang, X., Guo, H., Alajlan, N., Zhu, H., Rabczuk, T., Deep autoencoder based energy method for the bending, vibration, and buckling analysis of Kirchhoff plates with transfer learning, *European Journal of Mechanics - A/Solids*, Volume 87, May–June 2021, 104225.
29. David J. Steigmann, “Invariants of the Stretch Tensors and their Application to Finite Elasticity Theory” *Mathematics and Mechanics of Solids*, Vol 7, Issue 4, 2002.
30. Rabczuk T., Zi G., Bordas S., Nguyen-Xuan H.: A geometrically non-linear three dimensional cohesive crack method for reinforced concrete structures, *Engineering Fracture Mechanics*, 2008, 75(16), 4740-4758.

31. Areias P., Reinoso J., Camanho P.P., Cesar de Sa J., Rabczuk T.: Effective 2D and 3D crack propagation with local mesh refinement and the screened Poisson equation, *Engineering Fracture Mechanics*, 2018, 189, 339-360.
32. Areias P., Rabczuk T.: Cesar de Sa J.: A novel two-stage discrete crack method based on the screened Poisson equation and local mesh refinement, *Computational Mechanics*, 2016, 58, 1003-1018.
33. Ghorashi S., Valizadeh N., Mohammadi S., Rabczuk T.: T-spline based XIGA for Fracture Analysis of Orthotropic Media, *Computers & Structures*, 2015, 147, 138-146.
34. Rabczuk T., Akkermann J., Eibl J.: A numerical model for reinforced concrete structures, *International Journal of Solids and Structures* 42 (2005) 1327–1354.
35. Rabczuk T., Belytschko T., A three-dimensional large deformation meshfree method for arbitrary evolving cracks, *Comput. Methods Appl. Mech. Engrg.* 196 (2007) 2777–2799.
36. Talebi H., Silani M., Bordas S., Kerfriden P., Rabczuk T.: A Computational Library for Multiscale Modelling of Material Failure, *Computational Mechanics*, 2014, 53(5), 1047-1071.
37. Nguyen-Thanh N., Zhou K., Zhuang X., Areias P., Nguyen-Xuan H., Bazilevs Y., Rabczuk T.: Isogeometric analysis of large-deformation thin shells using RHT splines for multiple-patch coupling, *Computer Methods in Applied Mechanics and Engineering*, 2017, 316, 1157-1178.
38. Nguyen-Thanh N., Kiendl J., Nguyen-Xuan H., Wüchner R., Bletzinger K.U., Bazilevs Y., Rabczuk T.: Rotation free isogeometric thin shell analysis using PHT-splines, *Computer Methods in Applied Mechanics and Engineering*, 2011, 200 (47- 48), 3410-3424.
39. Nguyen-Thanh N., Nguyen-Xuan H., Bordas S., Rabczuk T.: Isogeometric analysis using polynomial splines over hierarchical T-meshes for two-dimensional elastic solids, *Computer Methods in Applied Mechanics and Engineering*, 2011, 200(21-22), 1892-1908.
40. Nguyen-Thanh N., Valizadeh N., Nguyen M.N., Nguyen-Xuan H. , Zhuang X. , Areias P., Zi G., Bazilevs Y., De Lorenzis L., Rabczuk T.: An extended isogeometric thin shell analysis based on Kirchhoff-Love theory, *Computer Methods in Applied Mechanics and Engineering*, 2015, 284, 265 – 291.
41. Nguyen Huy B., Zhuang X., Rabczuk T.: NURBS-based formulation for nonlinear electro-gradient elasticity in semiconductors, *Computer Methods in Applied Mechanics and Engineering*, Volume 346, 1 April 2019, Pages 1074-1095.
42. Vu-Bac N., Duong T.X., Lahmer T., Zhuang X., Sauer R.A., Park H.S., Rabczuk T.: A NURBS-based Inverse Analysis for Reconstruction of Nonlinear Deformations in Thin Shell Structures, *Computer Methods in Applied Mechanics and Engineering*, 2018, 331, 427-455.