Artificial Neural Network Modeling Approach for Elastic Plastic Stress and Strain Computation for Notched Components

Mehrnoosh Kazeruni

A Thesis in the Department of

Mechanical and Industrial Engineering

Presented in Partial Fulfillment of the Requirements for the

Degree of Master of Applied Science (Mechanical Engineering)

at Concordia University

Montréal, Québec, Canada

February 2023

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis proposal prepared

By: Mehrnoosh Kazeruni

Entitled: Artificial Neural Network Modeling Approach for Elastic Plastic Stress and Strain Computation for Notched Components

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Mechanical Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Dr. S. Narayanswamy, Chair

_____ Dr. S. Narayanswamy, Examiner

_____ Dr. E. Erkmen, External Examiner

_____ Dr. A. Ince, Supervisor

Approved by

Dr. M . Pugh

Chair of the Department of Mechanical, Industrial, and Aerospace Engineering

Mourad Debbabi

Dean of the Faculty of Engineering and Computer Science

Abstract

Artificial Neural Network Modeling Approach for Elastic Plastic Stress and Strain Computation for Notched Components

Mehrnoosh Kazeruni

Fatigue assessment of notched components requires knowledge of elastic-plastic stress-strain responses at notch areas. Traditional elastic-plastic finite element analysis (FEA) is not computationally efficient, and approximation methods are not highly accurate. Therefore, the present study uses the integration of an artificial neural network (ANN) and finite element (FE) to predict elastic-plastic stress and strains at notch locations on the basis of an elastic FEA solution. To this end, two different Finite Element (FE) models were developed to generate hypothetical elastic and elastic-plastic stress and strain datasets for different hardening materials and load levels. The first FE model was based on an elastic deformation state, while the second one was under an elastic-plastic deformation state. The elastic stress data obtained from the elastic FE model was used as the input data, and the elastic-plastic stress-strain data from the nonlinear elastic-plastic FE model was utilized as the output data. Subsequently, the ANN was trained for the dataset to establish a relationship between the input and output data. The dataset fed to the ANN model was then divided into three groups: training, verification, and testing. The training data was used for the ANN learning algorithm to attain the desired accuracy by obtaining the hyperparameters of the model. Subsequently, the verification data was employed to evaluate chosen hyperparameters and make adjustments of the hyperparameters to determine optimum

results, and in the last stage, the generalizability of the model was examined by the testing data. The predicted stress-strain results showed that the developed ANN model is able to accurately and efficiently predict elastic-plastic stress and strain for the notched body using only the elastic FEA solution. The developed ANN-FE methodology could efficiently estimate elastic-plastic stresses and strains for notch bodies with varying material hardening properties and load levels.

Acknowledgments

I would like to thank my supervisor, Prof. Ayhan Ince, for giving me the opportunity to be a part of his research group and for sharing his valuable experience in research. His manners taught me to be organized and work hard to succeed.

I would also like to thank Ali, Navid, and Farshid, my colleagues, for their comments during my research.

Table of Contents

Table of Figures

List of Tables

Nomenclature

FEA	Finite Element Analysis
ANN	Artificial Neural Network
ESED	Equivalent Strain Energy Density
FE	Finite Element
NN	Neural Network
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
APDL	Ansys Parametric Design Language
ReLU	Rectified Linear Unit
Tanh	hyperbolic tangent function
MSE	Mean square error
σ^e	Elastic stress
ε^e	Elastic strain
σ^a	Actual elastic-plastic stress
ε^a	Actual elastic-plastic strain
σ_{eq}^e	Equivalent elastic stress
ε_{eq}^e	Equivalent elastic strain
σ_{eq}^a	Equivalent elastic-plastic stress
ε_{eq}^a	Equivalent elastic-plastic strain
x_i	neurons
w_{ij}	weight

b_j	Bias
$F(x)$	Activation function
n	The number of neurons
$\hat{y}_i$	Output data
y_i	Predicted data
M	Material properties
E	Young's modulus
k'	cyclic hardening coefficient
n'	cyclic hardening exponent
X_{sclaed}	Normalized data
S_y^{el}	Elastic stress in Y direction
S_z^{el}	Elastic normal stress in Z direction
S_{yz}^{el}	Elastic shear stress
S_y^a	Elastic-plastic stress in Y direction
S_z^a	Elastic-plastic stress in Z direction
S_{yz}^a	Elastic-plastic shear stress
ε_y^a	Elastic-plastic strain in Y direction
ε_z^a	Elastic-plastic strain in Z direction
ε_{yz}^a	Elastic-plastic strain in YZ direction
ε_x^a	Elastic-plastic strain in X direction
S_{FEM}	Calculated stress based on finite element model
S_{ANN}	Calculated stress based on the artificial neural network model

To

My dear family

In the memory of all the brave Iranian women

who fought for their basic rights in my country, Iran

Chapter 1

Introduction

1.1 Background

Many mechanical components contain notches due to their design requirements. High stress and strain concentrated locally at notches induce local plastic deformations, which can lead to fatigue failure. As a result, precise stress and strain estimations in the vicinity of notch locations are required for reliable fatigue life analyses of notched components. Many notched components are subject to multiaxial loadings (combined tensile and torsional loadings) to induce complex elastic-plastic stress and strain responses at notches [1]. Finite Element Analysis (FEA) is widely used as a numerical approach to compute stress and strain responses for notch components under an elastic-plastic state. However, FEA is considered a computationally intensive and time-consuming method for structural components under long load histories. Therefore, notch approximation approaches have been developed to determine elastic-plastic stress and strain around the notch root [2].

The Neuber rule is a well-known analytical notch approximation [1]. Based on the Neuber rule, the total fictitious strain energy density under an elastic state of the notched root is equal to the total strain energy density under an elastic-plastic state, as schematically shown in Figure 1(a). The total strain energy includes the strain energy and the complementary strain energy density. The formulation of the Neuber rule for a notched body under the uniaxial loading can be expressed as Eq. (1) [3]:

$$\sigma^e \varepsilon^e = \sigma^a \varepsilon^a \tag{1}$$

where σ^e and ε^e are stress and strain at the notch root, respectively, assuming that the material behavior is hypothetically elastic. σ^a and ε^a are corresponding to actual elastic-plastic stress and strain. The Neuber rule was extended by Topper et al. [4] to predict stress-strain for different notched geometries subject to cyclic uniaxial loading. Molski and Glinka [5] presented an equivalent strain energy density (ESED) method as formulated in Eq. (2) to estimate the elastic-plastic stress and strain at the notch root. The ESED method suggested that the strain energy density computed using elastic assumption equals the actual elastic-plastic strain energy density under the same loading conditions as illustrated in Figure 1(b). Glinka [6] later improved the ESED method for cyclic uniaxial loading states.

$$\frac{1}{2}\sigma^e \varepsilon^e = \int_0^{\varepsilon^a} \sigma^a d\varepsilon^a \tag{2}$$

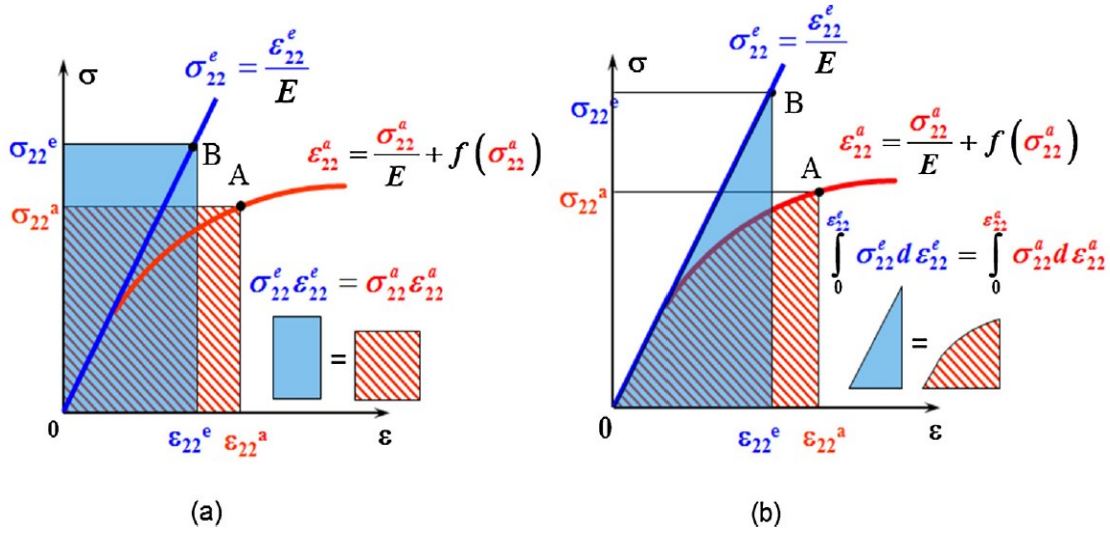


Figure 1. Graphical representation of (a) the Neuber rule and (b) the ESED method [1]

Sharpe et al. [7] evaluated the Neuber rule and the ESED method for various notched components. The authors suggested that the Neuber rule could be practical for notched geometries under a plane stress state. However, in case of a large constraint in the boundary condition, the ESED approach should be employed.

Hoffmann and Seeger [8] presented a new notch approximation method under multiaxial loading conditions by taking equivalent stress and strain into account as expressed in Eq. (3).

$$\sigma_{eq}^e \varepsilon_{eq}^e = \sigma_{eq}^a \varepsilon_{eq}^a \tag{3}$$

where σ_{eq}^e and ε_{eq}^e are equivalent elastic stress-strain responses and σ_{eq}^a and ε_{eq}^a are the actual equivalents of elastic-plastic stress and strain.

Singh [9] suggested two different methods to calculate the stress and strain of notched components under non-proportional loadings based on an incremental elastic-plastic relationship of the ESED method and the Neuber rule. Singh assumed that increments of the strain energy density of notched components under the elastic-plastic state are equal to increments of strain energy under a fictitious elastic state. The author presented estimated stress and strain results based on the increments of strain energy and concluded the terms were in fair agreement with FEM and experimental data.

Ye et al. [10] modified the ESED method by taking the stored and heat energy into account. In the modified form of the ESED method, the heat energy was regarded as dissipation, and the stored energy was defined as a contribution to stress-strain results under monotonic and cyclic loading.

Later, stress-strain notch approximation methods have been proposed on the basis of the notch-plasticity correction approaches in order to predict stress and strain for notched components subject to multiaxial loadings [11-14]. Ince and Glinka [15-17] proposed a stress-strain notch approximation model using the incremental deviatoric for the modified Neuber rule under multiaxial loadings. The predicted results were compared with FEA numerical datasets to evaluate the accuracy of the model. The results showed that the presented model for determining elastic-plastic data and the fatigue damage parameter were in good agreement with the experimental data for notched shafts under proportional and non-proportional loadings.

A number of stress-strain notch approximation models have been proposed in recent years as an alternative to the computationally intensive FEA; however, those approximation methods have not been proven to be successfully applied for different materials and load

cases. Therefore, a more generalizable approach is required to efficiently predict local stress-strain for notched bodies made of different materials and subject to varying loading conditions. The present study proposes a generalizable efficient computational methodology based on the integration of FEM and Artificial Neural Networks (ANNs) to accurately estimate the elastic-plastic notch stress and strain fields on the basis of the elastic FEA solution. However, the proposed approach includes some limitations. The developed ANN model can not predict elastic-plastic stresses and strains for the notched component subjected to cyclic loadings, additionally, the presented ANN model is not a robust model in case of extrapolation.

1.2 Artificial Neural Network Approach

Neural Networks (NNs) are one of the most frequently used algorithms to recognize patterns in data. Well-designed NN algorithms including Artificial Neural Networks (ANNs), Recurrent Neural Networks (RNN), and Convolutional Neural Networks (CNN) are employed to make function approximations and classifications based on the usage of data. The ANN is a computational approach inspired by the function of the human brain. This approach is mainly beneficial for addressing regression problems and function approximations in cases where there are nonlinear relationships in datasets [18]. The CNN is an algorithm mainly used for image classification, computer vision, and object recognition [19]. CNN is built based on the image-based data, in which image pixels can be considered as tensors, and changed to numerical data to maintain the spatial structure of datasets [20]. CNN algorithms are comprehensively applied in the field of medicine; however, they have received significant attention in various areas outside of medicine as well [21]. The RNN is the most advanced type of neural network algorithm, equipped with memory and the capacity

for self-learning [22]. In the present study, an ANN approach has been adopted due to the regression nature of the problem [23].

The ANN is a robust algorithm for finding complex patterns between numerical input and corresponding output data. The schematic view of an ANN model with one input layer, two hidden layers, and one output layer is illustrated in Figure 2. Each layer may have a different number of neurons (x_i) and each neuron in the previous layer is interconnected with the neurons of the next layer. Each connection between neurons is assigned a weight (w_{ij}), which is a positive value between 0 and 1. Choosing a large value for weight causes high activation of the neuron and leads to an increase in the influence of that neuron on output[23]. The output of a specific neuron is calculated after multiplying each input with its corresponding weight ($x_i w_{ij}$). In the next stage, the sum of activated neurons is added to the bias ($x_i w_{ij} + b_j$). Then, the calculated value is fed to the neuron in the first hidden layer as the input. The process continues for the next layer(s) to transfer the determined value to the output. The initial step in training a model is the preparation of data. All data fed to the ANN is normalized to arrange between 0 and 1. Normalization of data avoids the problem of dominating the training process with a large dataset value. In the ANN structure, the data is divided into three groups: training data, verification data, and testing data. Training data is used in the abovementioned process to achieve desired values for hyperparameters (e.g., weights and biases). Verification data is employed to determine whether the chosen weights and biases can account for verification of data. Subsequently, the generalization of the model must be considered using unseen new data (i.e., testing data); in other words, data is not used for training or verification of the ANN model.

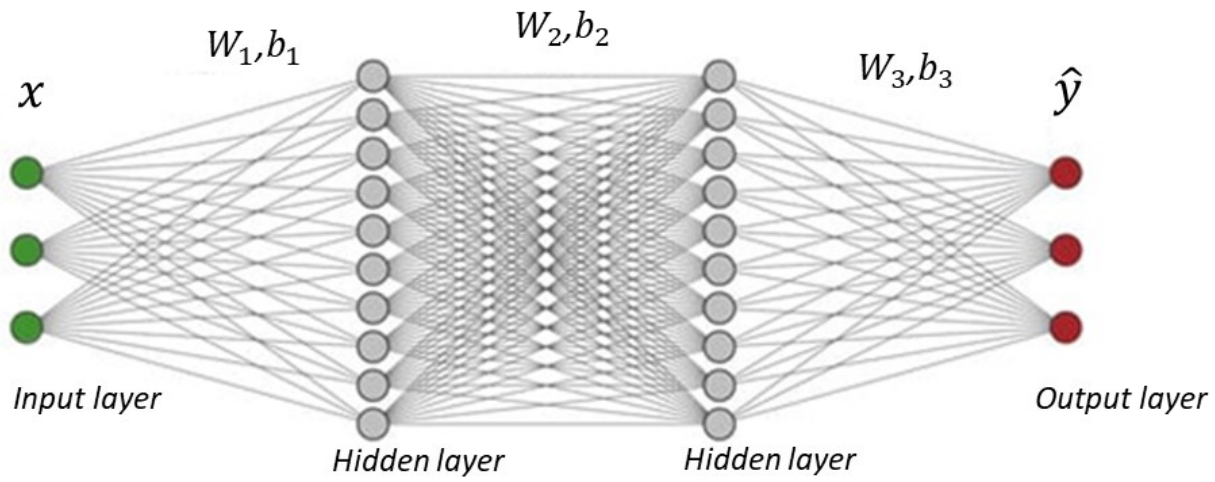


Figure 2. A schematic view of a general multilayer ANN

Hyperparameters such as the number of neurons, the number of hidden layers, activation functions, epochs, and learning rate must be carefully selected to produce a well-trained ANN model. Possible activation functions include linear function, Rectified Linear Unit (ReLU), sigmoid function, and Tanh.

- Linear function

A linear function (Eq. (4)) is an inappropriate choice for an ANN model as the ANN cannot improve a loss function. This is because the same gradient constant is used in each iteration. Linear functions can be used for simple tasks in which there are no complex relationships between the input and output data.

$$F(x) = a\mathrm{x} \tag{4}$$

- Rectified linear unit function:

One of the most popular activation functions is the rectified linear unit (ReLU), created from two linear functions. A key feature of ReLU is that a neuron is activated when the output of linear transformation is more than zero, and deactivation occurs for a value less than zero.

$$F(x) = \max(0, x) \tag{5}$$

- Sigmoid function

The Sigmoid function is an S-shaped nonlinear function that maps the range of values between 0 and 1, as shown in Eq. (6) [24].

$$F(x) = \frac{1}{e^{-x}} \tag{6}$$

- Hyperbolic tangent function

The hyperbolic tangent function (Tanh) is similar to the Sigmoid function as shown in Eq. (7); however, the slope is steep, and the output range is between -1 and 1. The Tanh function is preferred over the Sigmoid function because negative inputs are transformed into negative values [14].

$$F(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{7}$$

In the present study, ReLU is a chosen activation as it is non-linear and has the solving potential in the case of regression problems. ReLU contains a simple formulation that allows for less intensive computation and drives it to an efficient activation function. The ANN is trained on fed data, and the hyperparameters are adjusted during training; however, the accuracy of the model must be evaluated. Accuracy is assessed by calculating the differences between the predicted data and the output data with a loss function. Mean square error (MSE) is a widespread error function for assessing the performance accuracy of the ANN model, as expressed in Eq. (8).

$$\text{MSE}=\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2 \tag{8}$$

Subsequently, weights and biases are evaluated and updated on the basis of error gradients if the calculated MSE is not satisfactory. In other words, adjustment of weights and biases is repeated iteratively to minimize MSE function. The learning rate is another hyperparameter to control the training speed, and ranges between 0 and 1. If the selected learning rate is too low, the ANN model cannot converge to the optimal point, and once it is assumed to be a large number, gradient exploding occurs [20,23]. Therefore, the learning rate should be chosen carefully.

1.2.1 The application of ANN

The ANN is an efficient algorithm for extracting complex patterns between inputs and outputs and establishing full relationships between nonlinear data. The ANN is employed in

different fields such as healthcare, stock market, and defense, in addition to various engineering applications. As for solid mechanic applications, Haque and Sudhakar [18] employed an ANN to estimate the fracture toughness of micro-alloy steel on the basis of experimental data. Kusiak and Kuziak [25] developed an ANN model to predict the mechanical properties and microstructure of steel. The model was trained on experimental data, and the proposed results were verified with the experimental data to validate the accuracy of the model.

In recent years, the ANN has become one of the most promising machine learning algorithms to predict fatigue life. This is attributed to the fact that the ANN is a powerful approach to non-linear problems, and nonlinearity is a noticeable fatigue feature. ANN models can be developed to predict fatigue parameters in different cases, such as fatigue load, fatigue life prediction, and fatigue crack [26]. The ANN model can determine fatigue life directly and indirectly [27], which means that fatigue life can be defined as a variable in the ANN structure, or it can be calculated using defined fatigue life parameters in the ANN structure. Maleki et al. [28] presented an ANN model to predict the fatigue life of carbon steel in order to assess the effect of shot peening treatments on fatigue life. Shot peening parameters were considered as the input data, and fatigue life was defined as the output variable. Yen et al. [29] also directly used the fatigue life in an ANN model structure to predict an asphalt mixture fatigue life by defining it as an output variable in the data structure. Researchers have also developed ANN models to predict fatigue life indirectly based on the defined parameters in the ANN data structure. Genel [30] investigated the ANN model to indirectly estimate the fatigue life of steel by employing material properties.

Fatigue crack is another subset of fatigue that can be calculated by employing ANN algorithms. Researchers [31-32] developed ANN models to estimate the opening load in a crack. Mortazavi and Ince [23,33-34] used the ANN approach to obtain fatigue crack growth rates in short and long cracks. The authors also determined the J-integral by developing a robust ANN model. Actual elastic-plastic stress and strain were predicted around the crack under monotonic loading, and subsequently, J-integral was calculated by extracting data from the predicted results.

The ANN approach is also used in predicting the fatigue life of composite material. Bezazi et al. [35] employed an ANN to predict the fatigue life of a sandwich composite, and obtained the accuracy of the model by employing unseen data. El Kadi and Al-Assaf [36] used strain energy in input data to predict the fatigue life of a component made of composite. The authors evaluated the accuracy of the approach by comparing the predicted results with another ANN model in which strain energy was calculated indirectly based on the defined variables in the ANN model. Results were more accurate when the strain energy was used as the variable in the model.

1.3 The neural network limitation

The ANN is developed to extract the interrelated patterns between input and output data; however, it has certain limitations. Depending on the problem to be solved, its limitations can influence the predicted results [37]. Poor extrapolation, overfitting, and data-hungriness are examples of such limitations [22].

The ANN is an effective method in the case of interpolation within the training data. However, extrapolation implementation is inefficient outside the fed data [38]. Mortazavi and Ince [23] presented poor extrapolation capability of the ANN model to predict short fatigue crack propagations. Other researchers have also reported that the ANN model performs weakly outside the training range [39-42]. However, the ANN estimation can be improved through utilizing a large number of experimental data in the structure of datasets. In a study conducted by Christiansen et al. [38], the training data was determined to include a wide range of data. This extraction of input and output data assures that they represent quantifying nonlinearity. Another approach to enhance the extrapolation prediction capability of ANN models is the integration of a physics-based model and numerical data [42]. In order to do so, both experimental and analytical data are used to obtain a well-structured dataset [43-44].

Overfitting and underfitting are regarded as challenging problems for the development of ANN models [45]. When overfitting occurs, the model predicts well for the training data, but a poor estimation performance is provided for the test data. Poor generalization causes overfitting and underfitting [46]. In other words, Overfitting happens when the difference between the training and test errors is too high. Figure 3 shows three training cases in which cases one and two are related to poor learning, while case three illustrates a well-trained model.

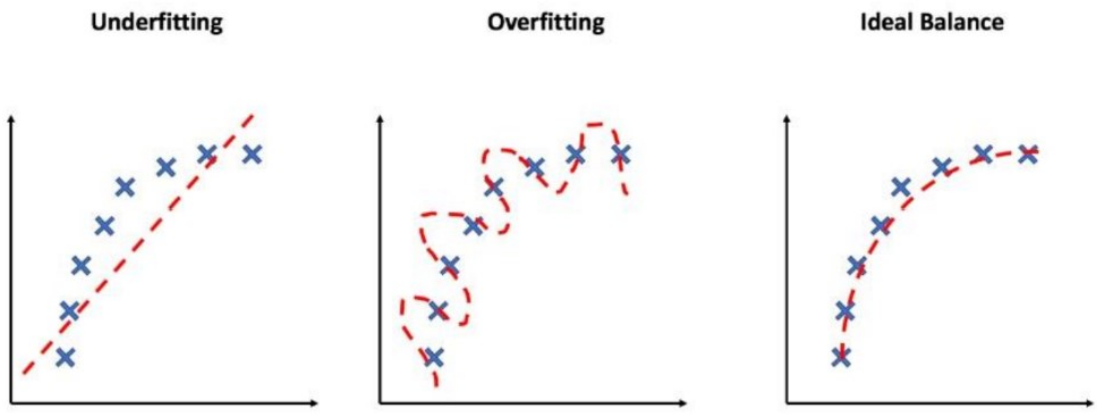


Figure 3. Visual demonstration of underfitting, overfitting, and ideal balance [47]

One method to avoid overfitting in the ANN model is dividing data into training, verification, and testing data sets [48-51]. The MSE function is then employed to assess training and verification of data accurately. The value of MSE for training and verification data should decrease to avoid overfitting.

Figure 4. Schematic view of a dataset divided into training, verification, and testing data sets

Another method used to avoid overfitting is cross-validation. It involves dividing the data into groups and repeating the training phase, with each group serving as a validation set. The best hyperparameter combination is determined by averaging the results of all trainings. Cross-validation is an effective method for small datasets, but it is rarely used in deep learning due to its high computation complexity[20].

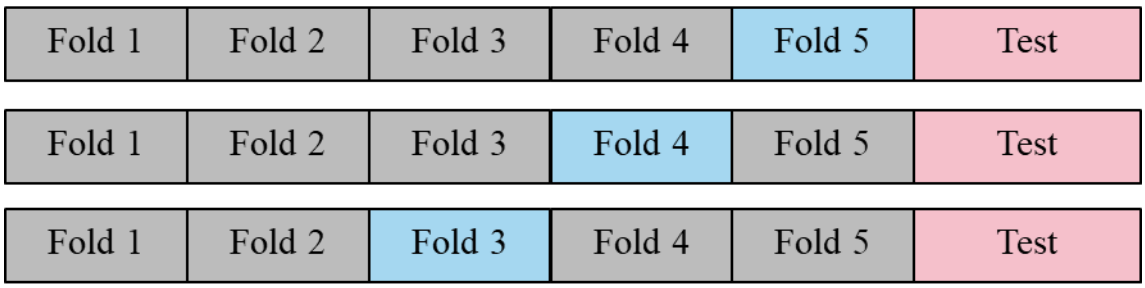


Figure 5. A visual example of cross-validation

1.4 Outline

This research hypothesized that the ANN is a powerful developer in finding nonlinear patterns between elastic stress data and elastic-plastic stress-strain data subject to multiaxial loadings. This approach is inspired by the Neuber rule to predict nonlinearity behavior at the notch root solely based on the elastic stress state. However, in the Neuber rule, both elastic stress and strain were used to estimate elastic-plastic stress and strain.

In the present study, the integration of FE-ANN is used for predicting the local stress-strain of a notched component subject to multiaxial loadings. The ANN is employed to develop a function that can predict the stress and strain at the notch surface. The ANN is a data-based approach that requires adequate data for finding algorithms in order to map elastic stress data to elastic-plastic stress-strain responses. Therefore, two different FEMs are defined to feed the ANN model. The FEM package of ANSYS APDL is employed to extract FE data. The first FE model is under elastic deformation state, and the second FE model is subject to elastic-plastic deformation state. Elastic stress data is extracted from the first model and fed to the inputs of the ANN model. Subsequently, elastic-plastic data obtained from the second FE model is used as the output. The data is split into three groups: training, verification, and testing data. The ANN model is trained on the training data to obtain the

desired combination of hyperparameters for the ANN model. The verification data is employed to ensure that the combination of hyperparameters is correctly chosen. In the last stage, the model is tested with unseen data to assess the model for generalization. Three test configurations defined for the test data involve materials with different hardenings and various loadings on the basis of interpolation and extrapolation of data. The predicted results for three configurations are compared with the elastic-plastic FE data under the elastic-plastic state.

1.5 Thesis layout

This thesis is investigated in manuscript style in accordance with "Thesis Preparation and Thesis Examination Regulations (version-2022) for Manuscript-based Thesis" published by the Concordia University School of Graduate Studies.

Chapter 1 introduces the problem, including background, literature review, and problem definition.

Chapter 2 includes the Artificial Neural Network Modeling Approach for Elastic-Plastic Stress and Strain Computation for Notched Components.

Chapter 3 discusses the conclusion of the study and proposes future research.

Chapter 2

Artificial Neural Network Modeling Approach for Elastic Plastic Stress and Strain Computation for Notched Components

2.1 Introduction

Fatigue failure is one of the most common failures encountered in many different industries, such as automotive, aerospace, defense, and construction industries [52]. Stress and strain determination is essential to assessing fatigue performance and residual life of structural components under service loading conditions. Additionally, various notch shapes emerge in the geometry of engineering components due to different design features and applications. Those components are mainly subjected to cyclic multiaxial loadings. Notched components under multiaxial loading experience complex stress and strain responses at those locations. Subsequently, local plastic deformation could occur due to high stress concentration at the notch area. Therefore, the non-linear elastic-plastic stress and strain responses of components at the notch area and its vicinity need to be determined so as to assess the structural integrity of notched components. The Finite Element Method (FEM) can be a strong numerical method for calculating elastic-plastic stress and strain of notch components. However, the FEM is an inefficient approach for complicated engineering problems in case of long time histories, load sequences, and design complexity. Accordingly, practical approximation methods are required to calculate stresses and strains at the notch area. [1].

The Neuber rule is a well-known approximation method for calculating elastic stresses and strains of notch components under uniaxial loading [1]. Based on the Neuber rule, the total strain energy density of the elastic-plastic component is equal to the total strain energy density of components in which the behavior of the material remains elastic beyond the yield point [3]. The Neuber rule was extended by Topper et al. [4], and the modified approach supports various notch geometries under cyclic uniaxial loadings. The equivalent strain energy density (ESED) method was proposed by Glinka and Molski [5]. Based on the ESED method, the actual elastic-plastic strain energy and pseudo strain energy are equal. Later, the ESED method was extended by Glinka [6] to account for cyclic uniaxial loadings. The Neuber rule was compared with the ESED method by Sharpe et al. [7]. Based on the comparison, the Neuber rule is efficient if there is plane stress, and the ESED method is efficient if there are considerable constraints in boundary conditions. Hoffman and Seeger [8,53] suggested using the equivalent stress in the Neuber rule to be applicable for multiaxial proportional loadings. Barkey [54] employed anisotropic plasticity theory to suggest a structural yield surface method for a notched bar under proportional and nonproportional multiaxial loading. A structural yield surface method was later improved by Koettgen et al. [55] by integrating the assessment of material governing equations in their calculation. Singh [9] proposed modifying the Neuber rule and the ESED method based on an incremental relationship between the strain energy of the elastic-plastic component and the strain energy of the fictitious elastic component under monotonic nonproportional loadings. Moftakhar [56] improved the Neuber rule and the ESED method for proportional loading by determining the upper limitation. A good agreement was observed for predicted strains and stresses with experimental and FEM data at the root of the notch. Researchers also used a

deviatoric interpretation of the Neuber rule to increase the accuracy of predicted results. Ince and Glinka [15-17] suggested a stress-strain notch approximation model based on the incremental deviatoric for the modified Neuber rule for notched shafts under proportional and non-proportional loadings. Predicted results were compared to FEA numerical datasets, revealing that the presented model for determining elastic-plastic data and the fatigue damage parameter agree well with experimental data.

Artificial Neural Networks (ANNs) are computational machine learning algorithms. Relationships among nonlinear datasets are established by developing ANNs. ANNs have been applied by researchers in diverse fields, including stock market estimation, healthcare, material science [57-58,30], fracture mechanics, and fatigue failure. ANNs were employed to predict mechanical material properties [59] and the stress-strain curve of materials [60] in the field of material science. Anijdan et al. [61] used the integration of ANN and genetic algorithms to investigate the optimal model of Al-Si casting alloy with the minimum porosity percentage. Hajializadeh and Ince [62] used an ANN to predict the residual stress field of additively manufactured components.

ANNs were used in various subsets of fatigue failure [63-68]. Artymiak et al. [58] employed the ANN approach to predict the S-N curve which determines a number of cycles to failure. Eventually, the authors estimated the fatigue limit of materials using the ANN approach. In their approach, the ANN model was trained with efficient data from a large assortment of materials. Genel [30] proposed an ANN model to predict strain life fatigue properties using fatigue parameters. The author used a wide range of materials to obtain a well-structured dataset. Mortazavi et al. [33-34] developed ANN models to predict fatigue crack growth rates for short and long-crack regimes. Also, Mortazavi and Ince [23]

developed an ANN model to predict elastic-plastic stress, strain, and deformation fields around the crack tip in order to calculate the J-integral. El Kadi and Al-Assaf [36] proposed an ANN model to predict the fatigue life of composite laminates in different conditions, such as various stress ratios and angles of laminated composite.

Furthermore, Convolutional Neural Network (CNN) algorithms have received remarkable attention as machine learning algorithms. CNNs are precisely used for image recognition and processing pixel data [69], which means that CNNs are employed mainly when images serve as the input and output data. These images are analyzed and transformed into a numerical format using robust CNN algorithms. CNNs are, therefore, quite useful in various fields of solid mechanics. Zhan et al. [70] developed a model of life component prediction under three different conditions: creep, fatigue, and creep-fatigue. Deep CNN was also employed by Kamiyama et al. [71] to predict the microscopic image of fatigue cracks after low-cycle fatigue. A Recurrent Neural Network (RNN) is another machine learning algorithm in which sequential data is used in the structure of a dataset. RNN is mainly used when problems are affected by the time [25]. It means that the RNN performance depends on the time histories of variables in order to determine the relationships among inputs at any given time. Subsequently, nonlinear relationships are established between input and output data sequences [39]. In the present study, an ANN is selected as a potential machine learning algorithm for numerical input and output data. However, a CNN could be used in future studies by changing the form of data to stress and strain distribution contour images in the vicinity of the notch. RNN can also be employed in future studies by changing monotonic loading to cyclic loading.

This paper proposes a new notch elastic-plastic stress and strain approximation method by integrating the FEM and ANN models on the basis of the linear elastic FE solution. The integrated FE-ANN methodology is developed to find a relationship between elastic and elastic-plastic responses of a notched shaft under multiaxial loadings. Based on the results of the present study, the fictitious stress of elastic components and material properties can be employed as variables in the input data structure to efficiently predict the actual stress and strain of elastic-plastic components. Therefore, two different FE models are developed under elastic and elastic-plastic solution states. Stress and stress-strain data obtained from the elastic and elastic-plastic FE models, respectively are fed into the ANN to predict actual stress-strain components. The presented approach is able to predict stress-strain results for notched components with various material hardenings and varying loadings.

2.2 Modeling Methodology

In the present project, an ANN approach can be used as a prediction function to determine the relationship between elastic and elastic-plastic solutions of the FE models as expressed in Eq. (9) [2]. The ANN model will use both FE elastic stresses at notch root and material properties as input data to predict elastic-plastic stresses and strains at notch area. Vector M in Eq. (9) involves material properties for the elastic-plastic material.

$$\left[\sigma_{el,pl}\ \varepsilon_{el,pl}\right] = f(\sigma_{el}, M) \tag{9}$$

Neuber considered a notched component subject to shear stress to determine whether the total strain energy density of a notched component under an elastic-plastic state is equal to the total energy of the same component under an elastic state [3]:

$$\sigma_{el,pl}\,\varepsilon_{el,pl} = \sigma_{el},\varepsilon_{el} \tag{10}$$

Comparing Eq. (9) and Eq. (10) illustrates that the Neuber rule can be defined by Eq. (9). The presented ANN model can predict elastic-plastic stress-strain results of a notched component under varying multiaxial loadings based on elastic stress data. Using material properties in variables of the presented study improves the ANN model applications; however, the Neuber rule is limited to one material and case of loading. The proposed ANN model is inspired by notch approximation methods, while it is not limited to a specific loading and material.

Figure 6 illustrates the schematic view of transforming elastic data to elastic-plastic data. The ANN is able to find algorithms between elastic data and elastic-plastic data. Therefore, the ANN is trained on data to predict elastic-plastic stress-strain data based on elastic solutions.

Two discrete FE models are developed in the initial step to generate stress and strain data under elastic and elastic-plastic states. Subsequently, stress-strain analysis data from the notch area is extracted to obtain the input and output data of the ANN model. The FEA datasets are generated under different loading conditions and various material hardening cases to provide sufficient datasets for the development of the generalized ANN model to account for the variation of multiaxial loads and material properties. In the next step, the dataset is split into training, verification, and testing data. Training data and verification data

are used to train the model and achieve the desired combination of hyperparameters in the ANN model. Testing data, on the other hand, is employed to examine the proposed ANN model in case of generalization and accuracy of predicted elastic-plastic stresses and strains.

The obtained stress data (σ_{el}) of the elastic FE model and vector (M) are fed into the ANN model as inputs. Stress-strain data ($\sigma_{el,pl}$ $\varepsilon_{el,pl}$) under an elastic-plastic state are used as outputs. Vector M contains two variables of k' and n' extracted from the Ramberg-Osgood equation:

$$\varepsilon_{el,pl} = \frac{\sigma_{el,pl}}{E} + \left(\frac{\sigma_{el,pl}}{K'}\right)^{n'} \tag{11}$$

where k' and n' are the cyclic hardening coefficient and cyclic hardening exponent, respectively, and E is Young's modulus [72].

The ANN model is trained, and eventually, the generalizability of the model is found by using verification and testing data. In the last step, the predicted elastic-plastic stresses and strains are compared with elastic-plastic FE data to examine the accuracy of the ANN model.

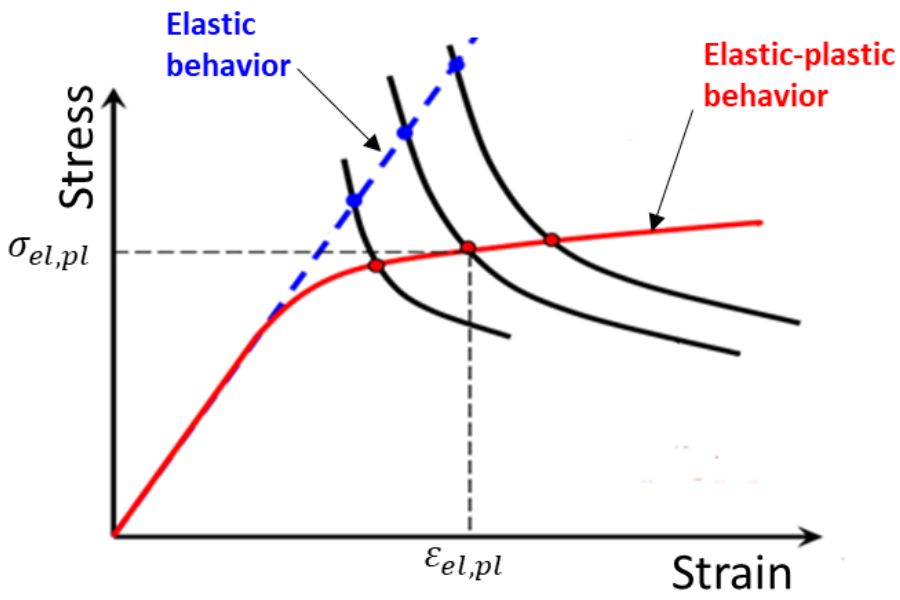


Figure 6. Mapping elastic data to elastic-plastic data

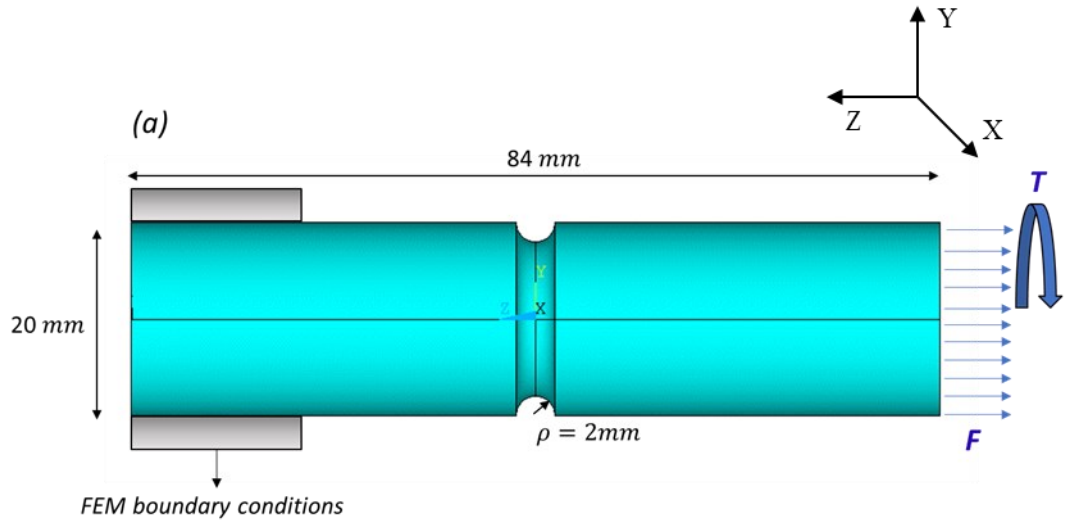
2.2.1 Finite Element modeling

ANSYS Parametric Design Language (APDL) is employed to extract elastic and elastic-plastic stress-strain analysis data around the notch area in order to feed the ANN model. APDL codes are used to create FE models, run FE analyses, and extract data for the ANN model. The code files include design parameters such as geometrical dimensions, mesh size, material properties, and loading information. The codes involve post-processing coding to extract stress and strain data around the notch root.

A notched shaft made of SAE 1045 steel under monotonic multiaxial loading is modeled. Two different types of loading are applied to the notched component: tension force and torque. In the first FE model, the material has a linear behavior based on Hook's law

[73], whereas, in the elastic-plastic FE model, the material obeys Ramberg-Osgood's law. M-2 is the material chosen to determine stress and strain, as illustrated in Figure 9.

3-D 8-node solid elements were used to mesh the notched shaft. Additionally, fixed support was selected for the boundary condition. The dimensions of the model, boundary condition, loadings, and generated mesh are shown in Figure 7. The radius of notch is 2 mm and the diameter of the shaft is 20 mm. The most common type of notch size is selected[16-17]. The size of mesh in the body of the shaft is 1 mm however it decreased to 0.125 mm at the surface of the notch to improve the accuracy of FEM results.



(b)

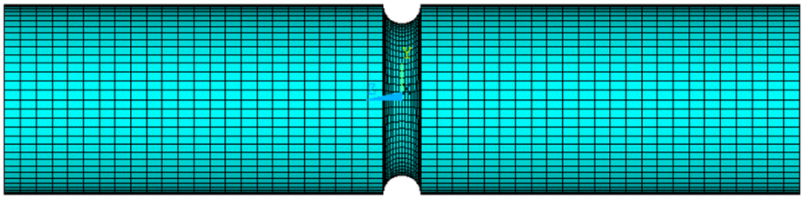


Figure 7. (a) Schematic geometry of the notched bar FE model subject to tension and torsion loadings; (b) FE model mesh

The notched shaft has a 2D stress state under multiaxial applied loadings. Plane stress is present owing to the traction-free surface at the notch shown in Figure 8. Accordingly, there are three stress components and four strain components on the notch surface. The stress state includes two normal stresses and one shear stress. Subsequently, the strain state involves three normal strains and one shear strain [1].

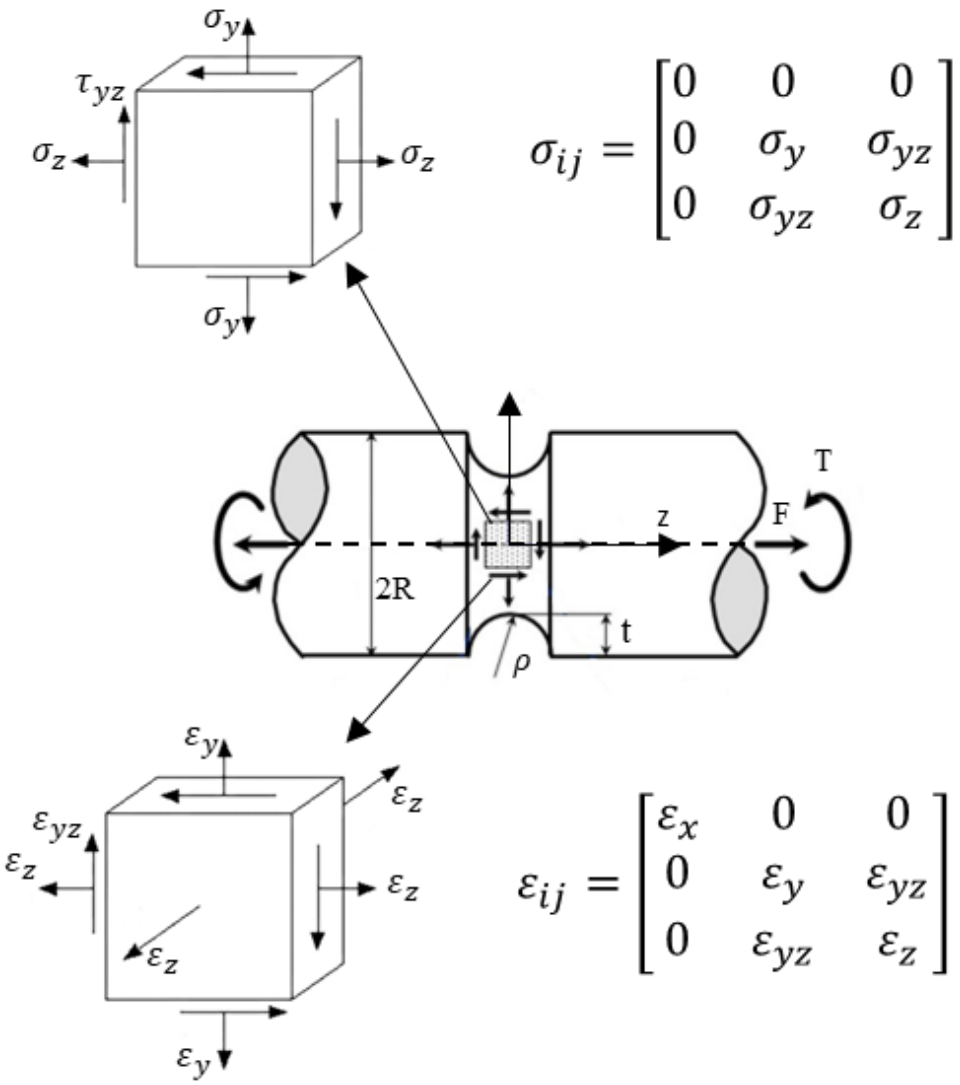


Figure 8. Multiaxial stress and strain state at the notch root for the component under multiaxial loading

In the present study, SAE 1045 was used as a base material (M-2) to build the FE model as illustrated in Figure 9. Then, Ramberg-Osgood constants (k' and n') of the base material (M-2) were varied to obtain different hardening behaviors of synthetic materials. Nine fictitious materials were defined in the present study based on Eq. (11). In order to do so, k' (1258 MPa) and n' (0.3) of (M-2) were transformed to new values to generate sufficient

materials to feed the ANN model. The mechanical properties of generated materials were used in the elastic-plastic FE model to determine the output data of the ANN model. Materials obtained by Eq. (10) are shown in Figure 9, and their respective mechanical properties are illustrated in Table 1.

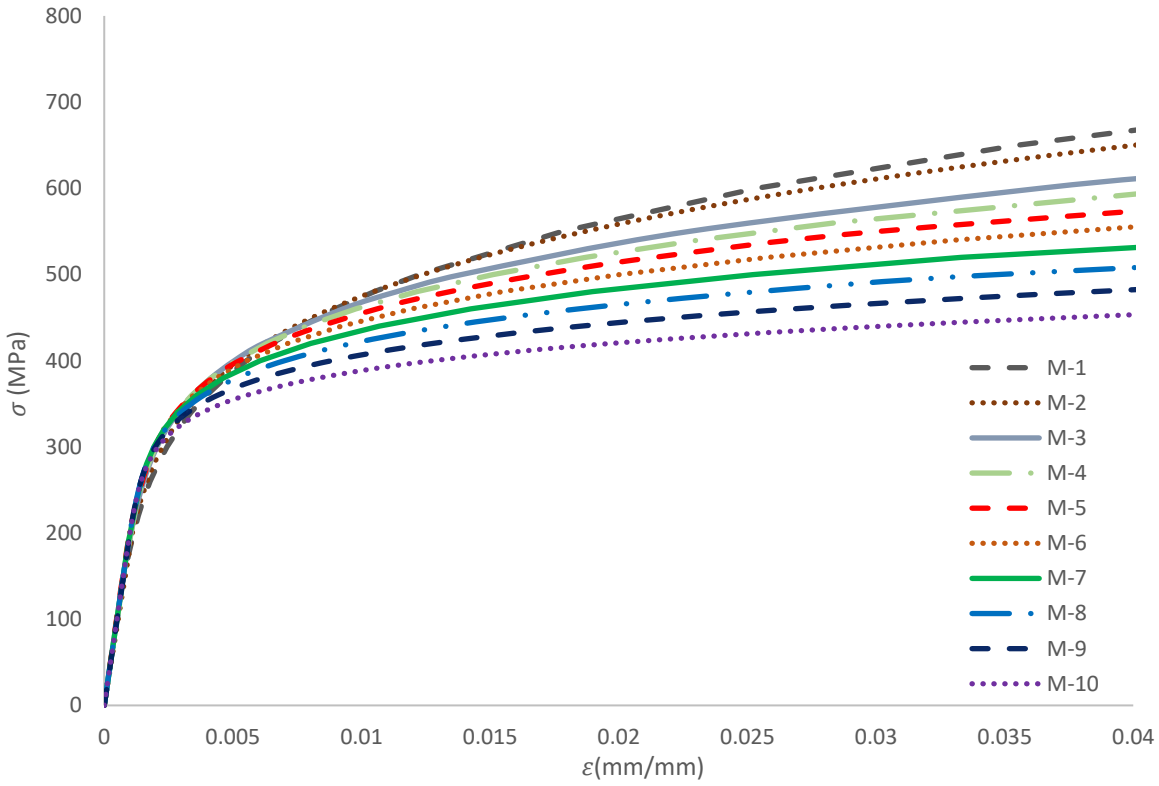


Figure 9. Cyclic σ-ε curve of materials based on Ramberg-Osgood law

Table 1. Monotonic and cyclic material properties of synthetic materials.

	Young's modulus, E (MPa)	Poisson's ratio, υ	Cyclic strength coefficient, K'(MPa)	Cyclic hardening exponent, n'
M-1	202,000	0.3	1383	0.22
M-2	202,000	0.3	1258	0.20
M-3	202,000	0.3	1069	0.17
M-4	202,000	0.3	1006	0.16
M-5	202,000	0.3	943	0.15
M-6	202,000	0.3	880	0.14
M-7	202,000	0.3	817	0.13
M-8	202,000	0.3	725	0.12
M-9	202,000	0.3	691	0.11
M-10	202,000	0.3	629	0.10

Nine load cases (force and torque) were defined based on monotonic loads, as shown in Table 2. The assumption to select loadings was that around 30% of equivalent stresses exceed the yield point under applied loadings in FE models.

Table 2. Load cases

loading	Load case 1	Load case 2	Load case 3	Load case 4	Load case 5	Load case 6	Load case 7	Load case 8	Load case 9
Force (N)	40000	41600	43200	44800	46400	48000	49600	40800	51200
Torque (N.mm)	150000	156000	162000	168000	174000	180000	186000	153000	192000

The elastic stress and elastic-plastic stress-strain data were extracted from FE models. The integration of FEM and ANN was used to predict actual stress-strain at the notch surface. Therefore, FE data of the notch region was extracted from a quarter of the notch surface because of a symmetric geometry used in the study (Figure 10).

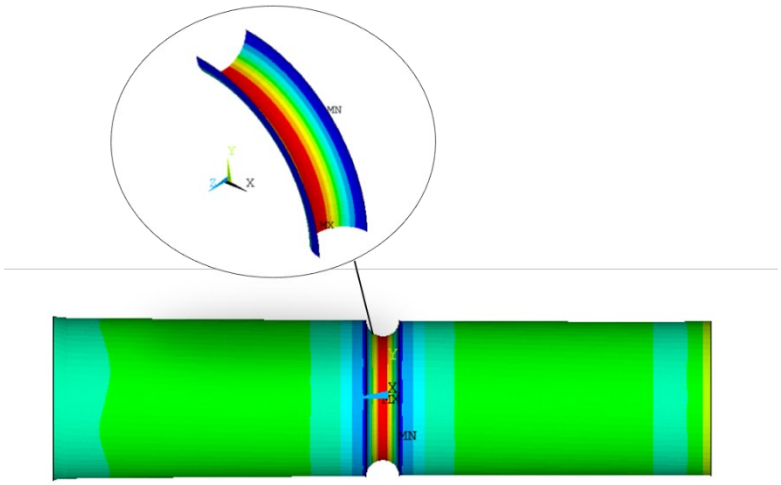


Figure 10. A quarter of the notch surface

2.2.2 The Artificial Neural Network Structure

ANN is developed to establish nonlinear relationships between the input data of hypothetical elastic stresses and the output data of actual elastic-plastic stresses and strains for different materials and various multiaxial load levels. Well-structured datasets are required to feed the ANN model in order to accurately determine the relationships among given data. The schematic view of the ANN model with one input layer, two hidden layers, and one output layer is shown in Figure 11.

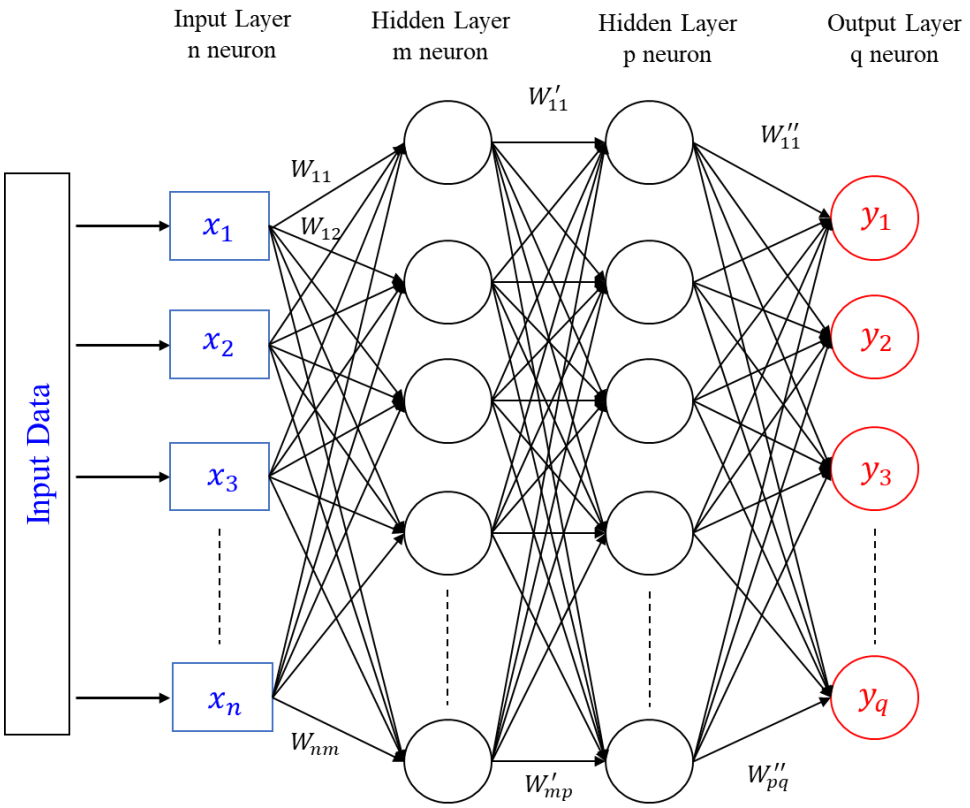


Figure 11. Structure of ANN

The training process begins by feeding the input layer with prepared input data. Assuming that the input data is introduced as x_j, the input layer contains n neurons which are multiplied by a particular weight (w_{ij}). Weights are numbers ranging from 0 to 1. A higher weight value indicates that a specific neuron is highly activated and significantly affects the output. Bias (b_i^{hid}) is added to the summation of $w_{ij}x_j$ to modify generated values on the basis of Eq. (12). Each layer has an activation function and is chosen based on the proposed problem. The final output of each layer is activated with the new weight, and the process continues for each hidden layer to transfer data to the output layer.

$$y_i = f(\sum_{j=1}^{n} w_{ij} x_j + b_i^{hid}) \quad (12)$$

In the present paper, input layer contains five variables therefore there are five neurons in the input layer: three elastic stress components (σ_{yy}^{el} , σ_{zz}^{el}, and σ_{yz}^{el}) and two materials properties (k' and n') extracted from Eq. (11)

$$x_j = (\sigma_{yy}^{el}, \sigma_{zz}^{el}, \sigma_{yz}^{el}, k', n') \quad (13)$$

Subsequently, each neuron (y_i) in the output layer has seven variables: three elastic-plastic stress components (σ_{yy}^{a} , σ_{zz}^{a} , and σ_{yz}^{a}) and four elastic-plastic strain components (ε_{yy}^{a}, ε_{zz}^{a}, ε_{xx}^{a} , and ε_{yz}^{a}).

$$y_j = (\sigma_{yy}^{a}, \sigma_{zz}^{a}, \sigma_{yz}^{a}, \varepsilon_{yy}^{a}, \varepsilon_{zz}^{a}, \varepsilon_{yz}^{a}, \varepsilon_{xx}^{a}) \quad (14)$$

Prior to starting the training of the model, it is necessary to normalize the data to account for multivariables with different ranges [23]. The variables with a larger scale potentially dominate the algorithm and negatively affect the learning performance of the ANN model. Therefore, data should be normalized to omit any potential problems. “Min-Max” normalization is the common method for scaling datasets between 0 and 1. The Min-Max function is shown in Eq. (15).

$$\frac{x - x_{max}}{x_{max} - x_{min}} \tag{15}$$

where x is the data, x_{min} is the minimum magnitude, and x_{max} is the maximum magnitude [74].

Hyperparameters that define the ANN structure should be chosen to build an appropriate ANN model [23]. Hyperparameters include weight, bias, the number of hidden layers, the number of neurons in each hidden layer, activation functions, and epochs. The rectified linear activation unit (ReLU) is the activation function used in hidden layers in the present study. For positive values, ReLU responds as a linear function, while deactivating neurons with negative values [24]. Therefore, ReLU can model nonlinear behavior, while remaining close to linearity, therefore allowing networks to retain some properties of linear functions that make the training process more manageable [20]. This feature makes ReLU a practical activation function for the ANN model.

Mean square error (MSE) is used to check the accuracy of the prediction. MSE is the basic and the most commonly employed loss function, and is defined as the mean or average of the square of the difference between actual and predicted results [75]. The gradient descent approach with backpropagation is employed to obtain the loss function accordingly to weights and biases. Then, weights and biases are updated based on the loss gradient (calculated by the backpropagation) and a hyperparameter called "learning rate." In the first step, the initial values of weights and biases are obtained. Following that, the inputs are fed in a forward direction. Subsequently, the outputs are computed based on the forward

propagation of the inputs. The MSE is employed to obtain the error, and the backward pass is performed by using backpropagation while modifying weights and biases. The process will repeat by passing a new forward propagation so as to improve and update hyperparameters to attain the required accuracy[20]. Eventually, an integration of hyperparameters is examined by inserting verification data. The hyperparameter combination must change if the MSE is acceptable for the training data but not for the verification data. In the last stage, the proposed ANN model must be examined by testing data to evaluate the generalization of the model. In order to do so, a new set of data that is not used to train and verify the approach is introduced to the model. Taylor expansion is employed to determine the gradient descent[76].

$$\begin{gathered} f(\vec{x}+h\vec{s}) = f(\vec{x}) + (\nabla f)^T \vec{s} h + o(h^2) \\ df = f(\vec{x}+h\vec{s}) - f(\vec{x}) \approx (\nabla f)^T \vec{s} h \\ \vec{s} = \nabla f \text{ as the optimal step direction} \end{gathered} \tag{16}$$

The loss function is represented by f , h shows the learning rate, and $\vec{s}$ is the derivative of the loss function with respect to the input data.

In the present paper, the ANN model was generated by Keras with TensorFlow, and the ANN code was written in Python.

In order to present the matrices of data, $\sigma_{rs,t}^{k,m,n,o}$, $\varepsilon_{rs,t}^{k,m,n,o}$ and M_k were chosen as notations. Both "r" and "s" show stress-strain component locations in the stress and strain tensor, and the subscript "t" shows elastic and actual (elastic-plastic) states. Subsequently, "m" and "n" refer to the number of loading case and time increments, respectively. The superscript "o" shows the number of nodes on the surface of the notch. The notched shaft surface is symmetric. Therefore, the elastic and elastic-plastic FEM responses are extracted

from a quarter of the notch surface, which contains 357 nodes as shown in Figure 10.The component M_k represents the material and "k" indicates the number of the material. In total, eight materials, seven loading sets, and 25 load increments are used. Each load case is divided into 25 load segments to account for the non-linear effect and the FE analysis convergence. Input and output structures are shown in Figures 12-13.

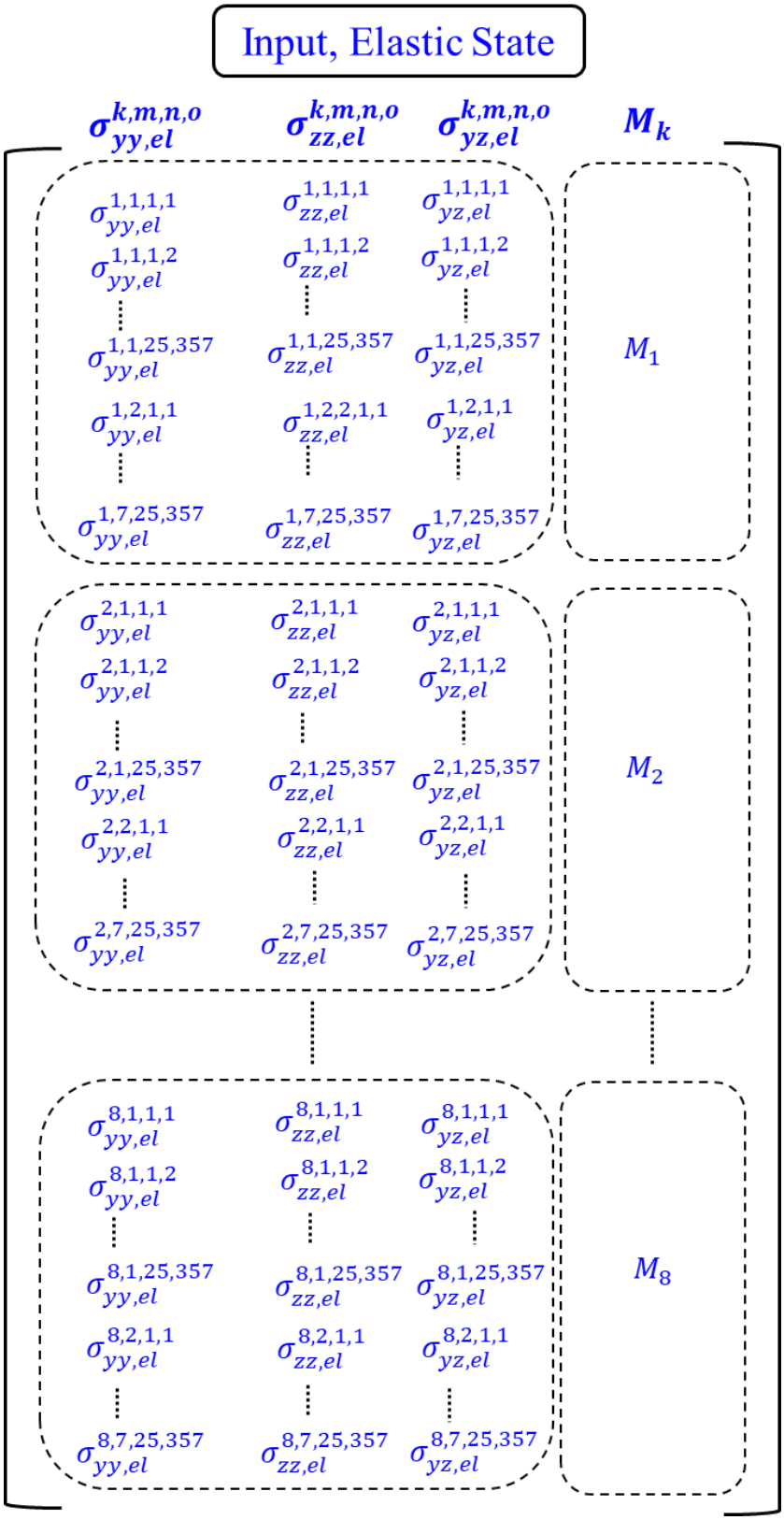


Figure 12. Input data in the ANN approach

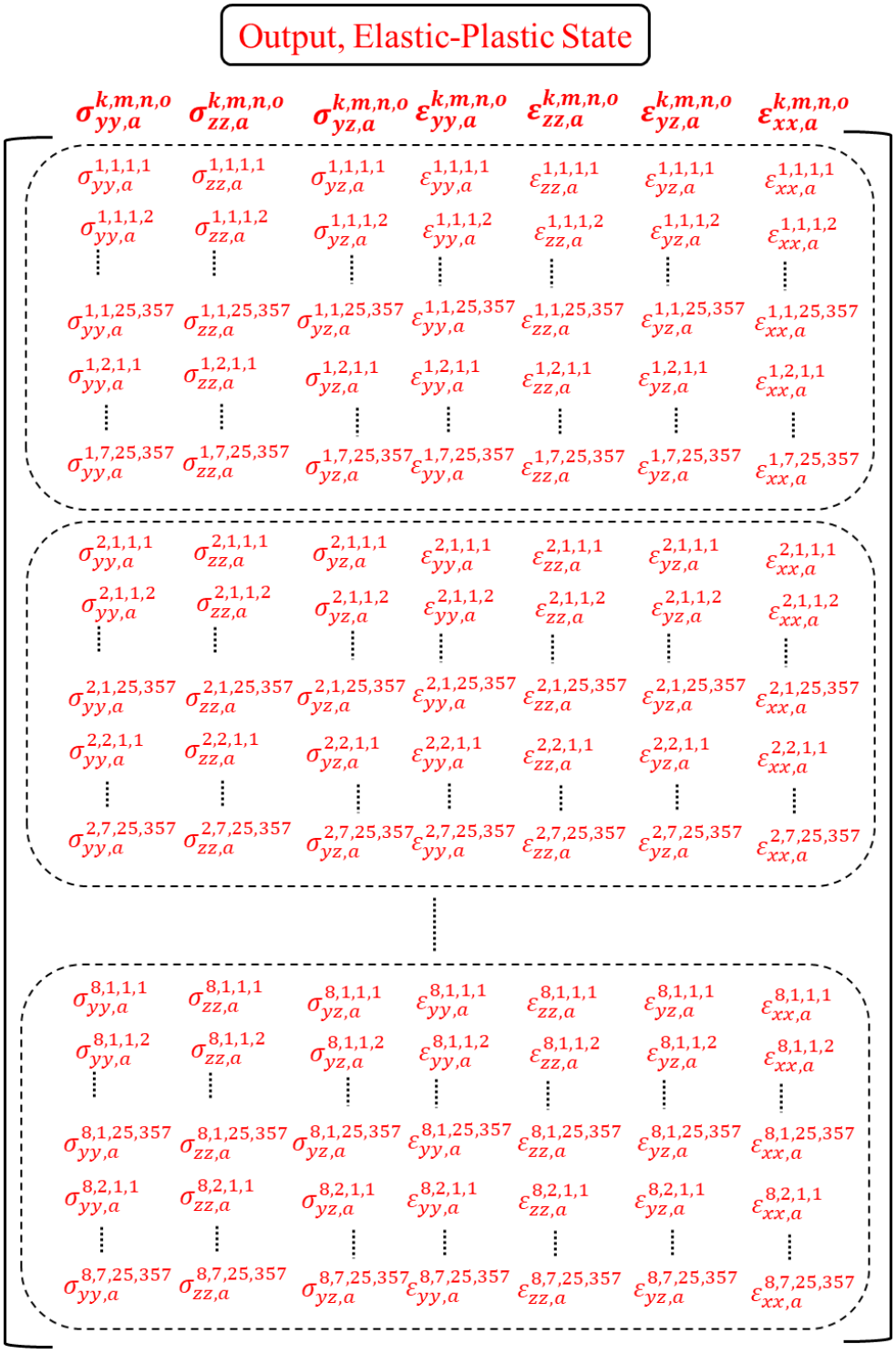


Figure 13. Output data in the ANN approach

The total number of input data is $8 \times 7 \times 25 \times 357 \times 5$ (number of materials $\times$ number of load cases $\times$ number of load increments $\times$ number of nodes $\times$ features) $= 2{,}499{,}000$, of which 72% is used as training data and the remainder as verification data. In outputs, there are three stress and four strain components. Then, the size of output data is $8 \times 7 \times 25 \times 357 \times 7 = 3{,}498{,}600$. The percentage of training data in the initial split of datasets was 80%, then decreases to 72% to improve the accuracy of predicted data for the verification data. Since the ANN model requires the same level of accuracy for the training data and the verification data.

Hyperparameters defining the number of hidden layers, the number of neurons, epochs, the number of variables, and activation function are shown in Table 3. There are two hidden layers in the present study. A trial and error process is applied to obtain the number of neurons and hidden layers [20]. The initial number of neurons obtained was 8 in each layer; then the accuracy of predicted results was obtained. After that, the number of neurons increased to 256 and 128 in the first and second layers, respectively to achieve an acceptable accuracy for the predicted results. During the optimization algorithm, the learning rate is utilized as a tuning hyperparameter that specifies the step size at each epoch, while attempting to minimize the loss function. Using a large number as the learning rate causes an increase in the learning time. The learning rate is set between 0 and 1 [23], and the learning rate selected is 1E-5.

Table 3. Hyperparameters used to train the ANN model

Number of hidden layers	2
Number of neurons in the first hidden layer	256
Number of neurons in the second hidden layer	128
Epochs	300-350
Learning rate	1e-5

2.3 Results and discussion

Data is divided into training and verification data so that overfitting and underfitting issues can be assessed. The performance of the ANN can be determined by comparing the error function and the accuracy for the training and verification. Although overfitting problem occurs when the error functions for the training reduce, the error function for the validation does not decrease at the same level. Figure 14 shows that the ANN model is not overfitted, and the number of epochs is selected correctly.

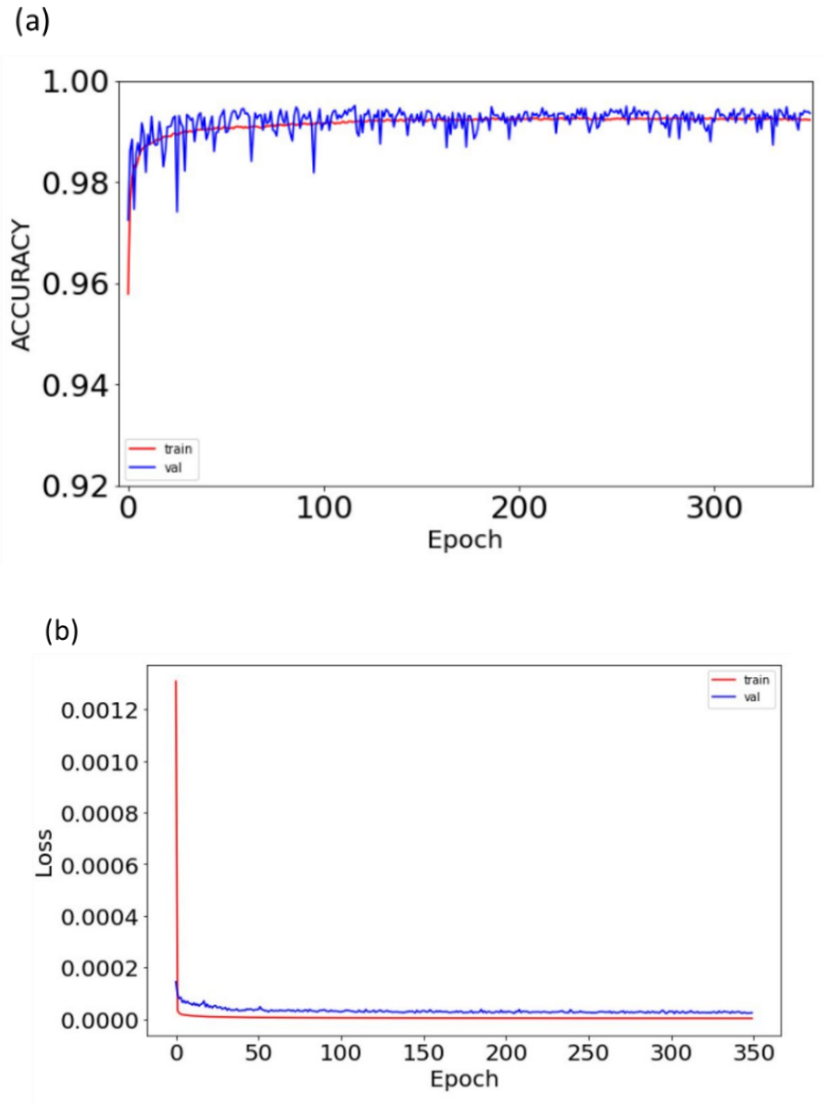


Figure 14. (a) The accuracy trend for training and verification of data (b) MSE for training and verification of data

The generalizability of the ANN model has been tested in respect of its interpolation and extrapolation prediction capability. The ANN is trained and verified based on a set of data limited to a data range called the interpolation zone, while all data outside this range is defined as an extrapolation zone. Figure 15 and Table 5 present the configurations of loading and material to be used for training, verification, and testing data. (M-5), (M-7), and (M-9) corresponding FE data are split into two groups to be used for training and verification data. Three configurations are considered to evaluate the generalization of the ANN model, as shown in Table 4. Generalizability refers to the ability of the ANN model to achieve the desired accuracy for unseen data based on the defined hyperparameters used to train the model.

Table 4. test configuration for examining the ANN model

	Material	Load Case
Test Configuration 1	M-3	Load Case 8
Test Configuration 2	M-6	Load Case 8
Test Configuration 3	M-1	Load Case 9

Predicted data of three nodal points are used to assess the accuracy of predicted elastic-plastic results and to compare full stress and strain curves between ANN and FEA. The nodes selected are located at the notched surface as shown in Figure 16, and their corresponding coordinates are presented in Table 6. Nodal points of α, β , and γ have been defined as high, intermediate, and low stress points in the notch area. ANN predicted

stress-strain curves for 25 load increments are compared against the elastic-plastic FEA results to assess the ANN prediction performance. The stress contour of the elastic-plastic FE model, ANN model, and error percentage contour are shown for all nodal points of quarter surface area. The error percentage is defined based on Eq. (17).

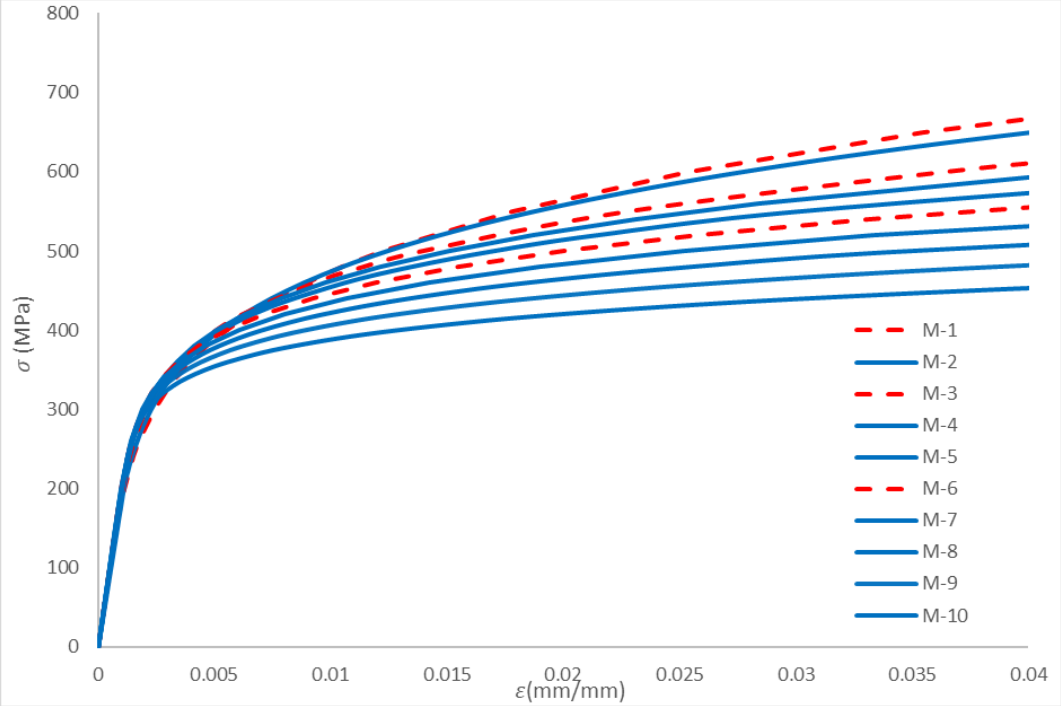


Figure 15. Material used to test the ANN model

Table 5. Material and loadings used for training, verification, and testing data

Loading	Material									
	M-1	M-2	M-3	M-4	M-5	M-6	M-7	M-8	M-9	M-10
Load case1	-	T	T	T	T	-	T	T	T	T
Load case 2	-	T	T	T	T	-	T	T	T	T
Load case 3	-	T	T	T	T/V	-	T/V	T	T/V	T
Load case 4	-	T	T	T	T/V	-	T/V	T	T/V	T
Load case 5	-	T	T	T	T/V	-	T/V	T	T/V	T
Load case 6	-	T	T	T	T/V	-	T/V	T	T/V	T
Load case 7	-	T	T	T	T/V	-	T/V	T	T/V	T
Load case 8	-	-	Testing	-	-	Testing	-	-	-	-
Load case 9	Testing	-	-	-	-	-	-	-	-	-

*T = training, V = verification

$$\text{Error percentage} = \frac{\sigma_{FEM} - \sigma_{ANN}}{\sigma_{FEM}} \times 100 \quad (17)$$

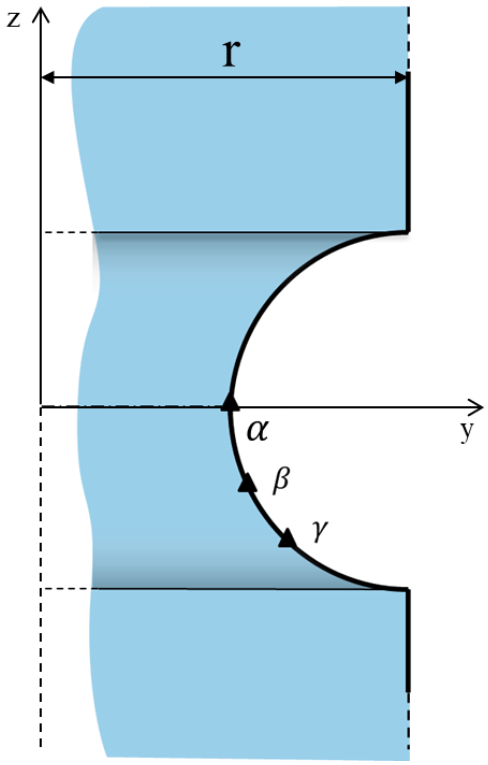


Figure 16. Locations of points at the notch root

Table 6. Node coordinates at the notch root

Node	Y(mm)	Z(mm)
α	8	0
β	8.0979	0.62
γ	8.218	0.9

- Test Configuration 1 (Interpolation)

The first case is an interpolation because the load case and material are obtained based on the interpolating of the training data. The values of K' and n' are 1069 MPa and 0.17, respectively. Load case 8 is applied on the notched shaft. As shown in Figures 17-19, the ANN model accurately predicts stress and strain values at $\alpha, \beta,$ and γ nodal points.

Figure 17(a) illustrates that the stress-strain curve of point α based on the ANN model agrees well with the stress-strain curve on the basis of the FE model, however, overpredicting can be seen in the last time increments. In order to evaluate the normal stress-strain results in the Z direction (σ_z and ε_z) for the corresponding time increments, the maximum FE elastic-plastic stress values in the datasets are compared to predicted stress-strain results. Based on the FE model, the maximum value for σ_z is 298 MPa while the predicted σ_z is 290 MPa. In other words, the overpredicted value for σ_z is less than the maximum stress magnitude used in datasets; therefore, the predicted result for σ_z is acceptable. Additionally, the error of ε_z in the last time increment is 3% for point α. Figure 17(b) illustrates that predicted results agree well with the FEM data. Figure 17(c) shows that the predicted results for shear stress (σ_{yz}) are under-predicted. The error percentages for the predicted strain and stress (σ_{yz} and ε_{yz}) in the last time increment are only 5.5% and 6%, respectively. Figures 18-19 illustrate that the ANN approach can predict elastic-plastic stress-strain results accurately.

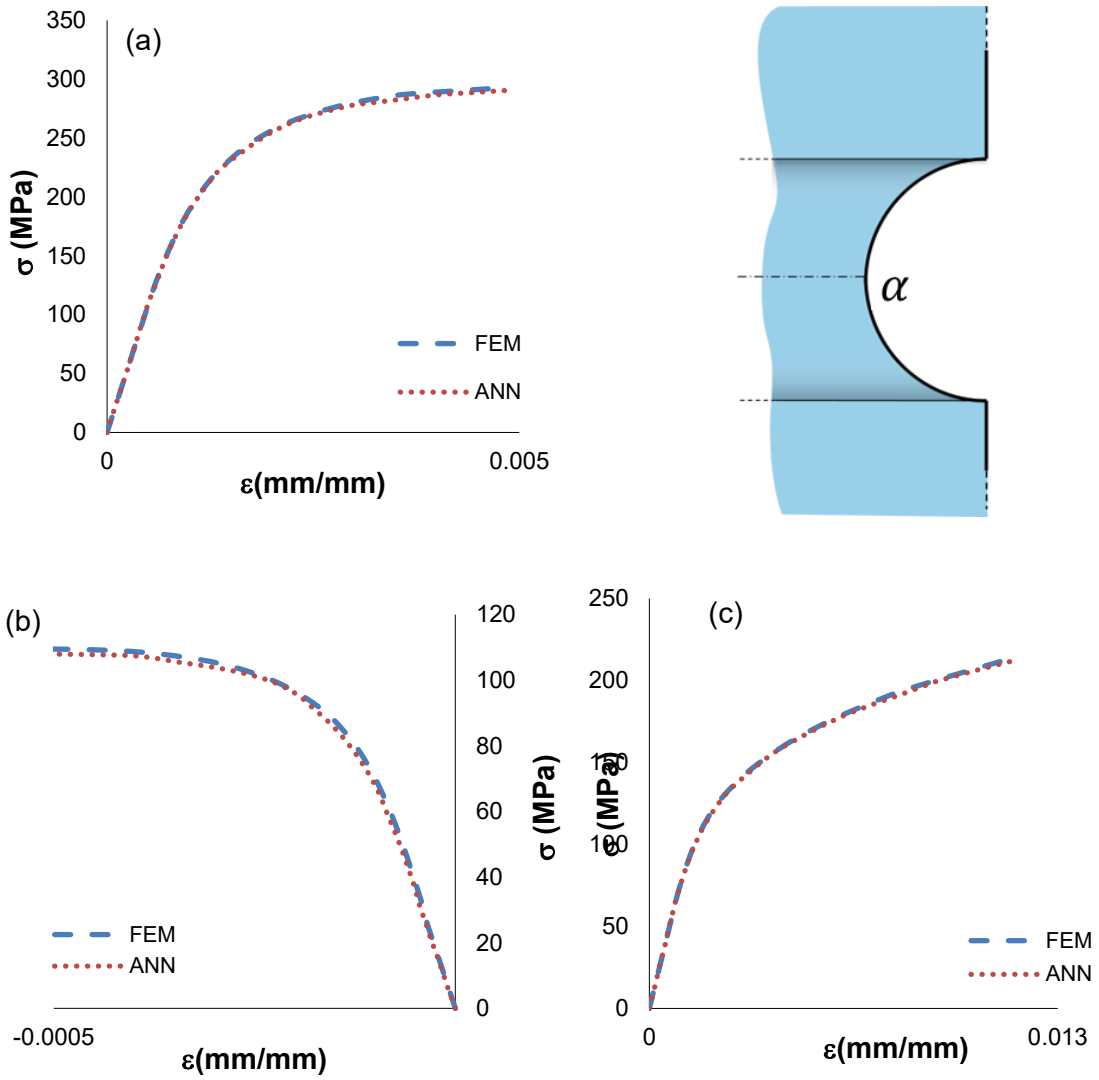


Figure 17. The σ-ε curve of α stress point for (a) normal stress in the Z direction, (b) normal stress in the Y direction, and (c) shear stress in the YZ direction

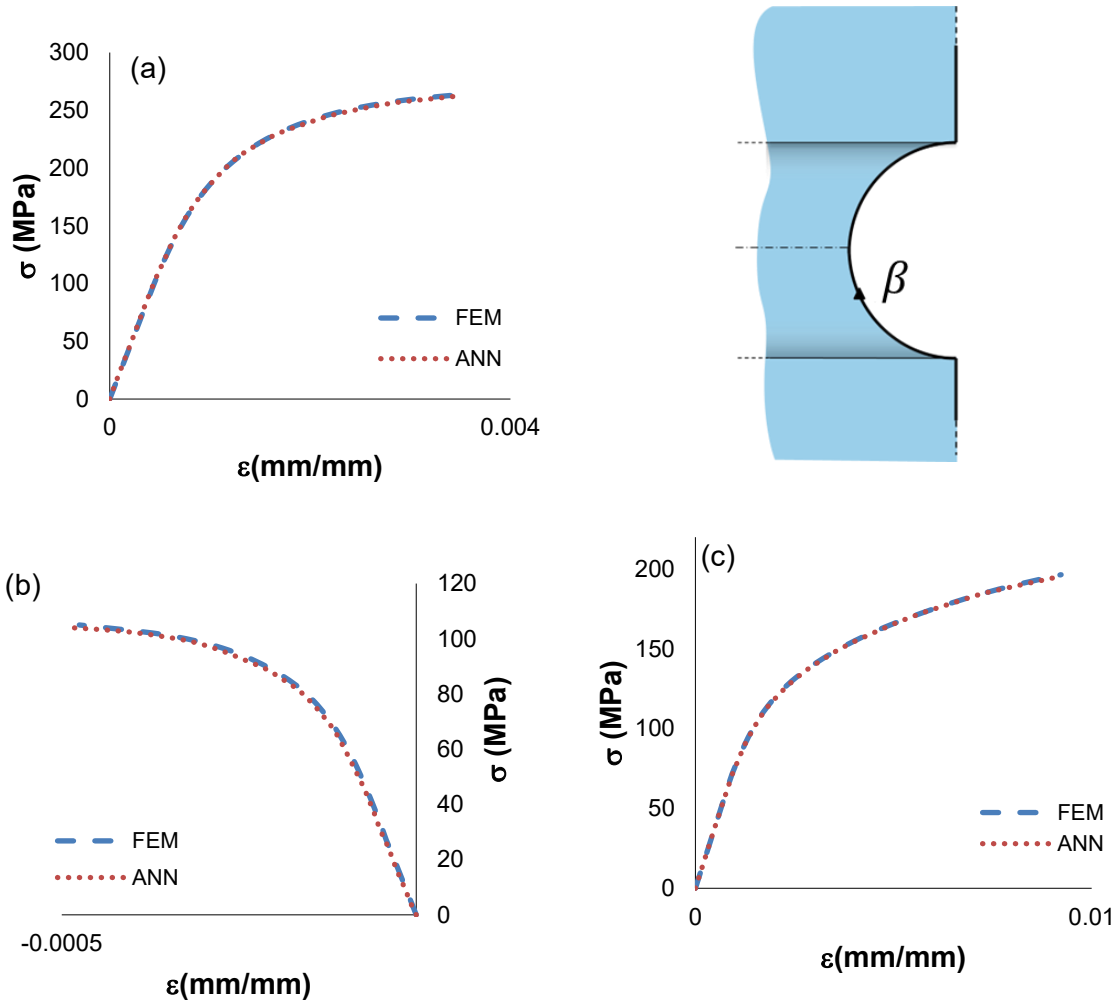


Figure 18. The σ-ε curve of stress point β for (a) normal stress in the Z direction, (b) normal stress in the Y direction, and (c) shear stress in the YZ direction

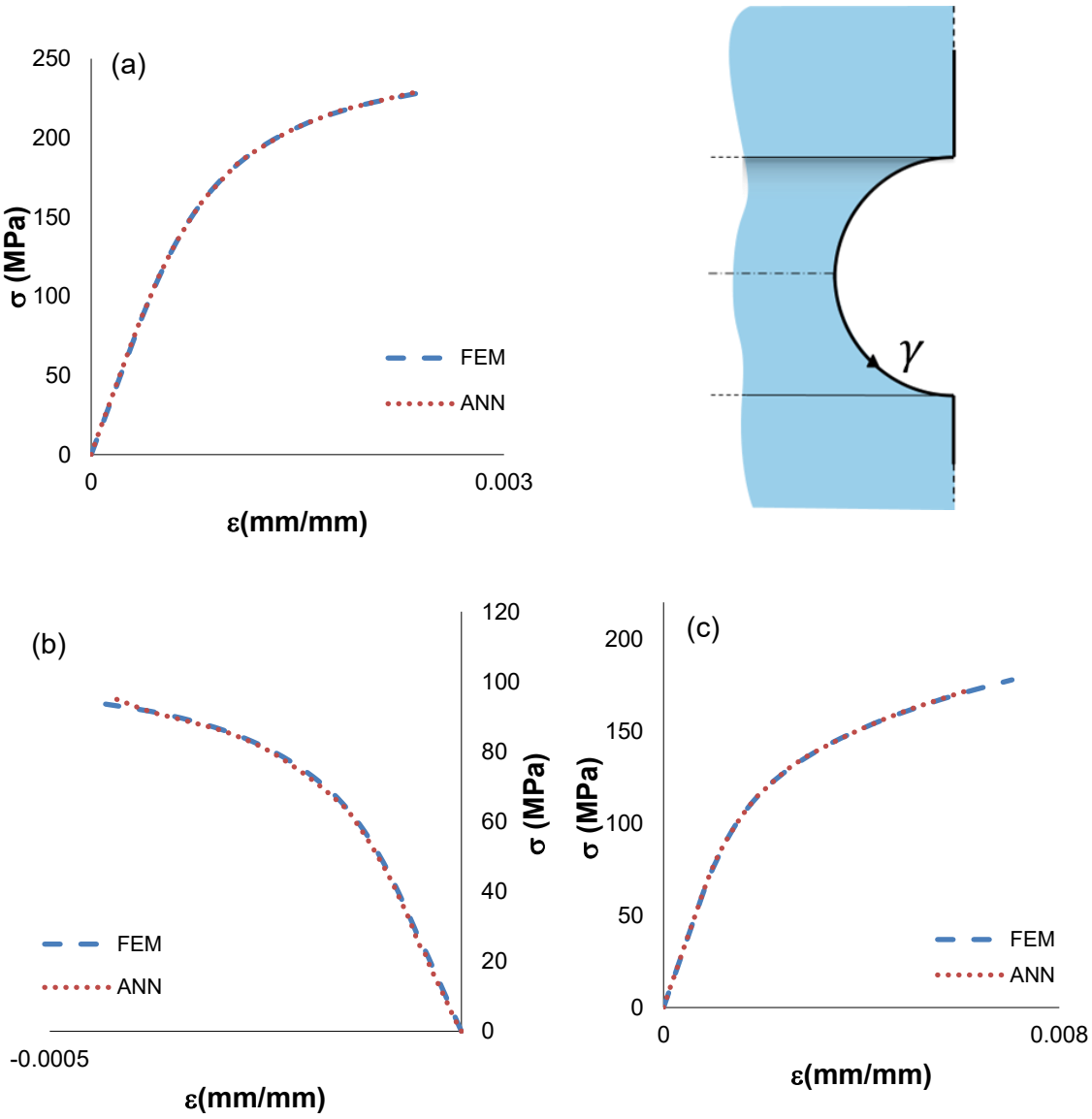


Figure 19. The σ-ε curve stress of point γ for (a) normal stress in the Z direction, (b) normal stress in the Y direction, and (c) shear stress in the YZ direction

Figure 20(a-b) shows the stress contour for normal stress (σ_z) for all nodal points at the notch surface based on the FE model and the ANN model, respectively. The error percentage of σ_z changes between 0 and 10%, and the maximum error values belong to nodes around the edge of the notch as shown in Figure 20(c). Based on Figure 20(b), the value of σ_z is less than 50 MPa at the edge of the notch; therefore, a 10% error is insignificant. The normal stress in the Y (σ_y) direction changes between -50 and 100 MPa. The error percentage contour shows that the majority of points have a 5% error at the notch vicinity, while at the edge, it increases to 25%, as illustrated in Figure 21(c). The error percentage for shear stress (σ_{yz}) at the root of the notch is 1%, and it rises to 3% at the edges, as shown in Figure 22.

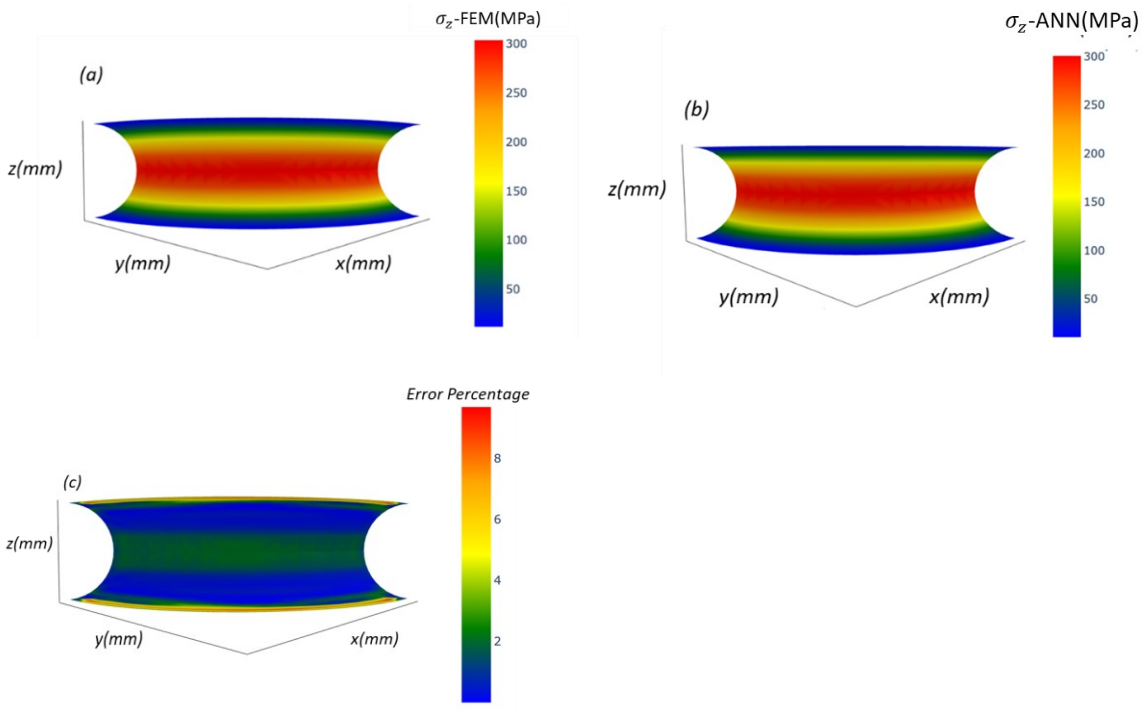


Figure 20. Test configuration 1: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage counters of σ_z

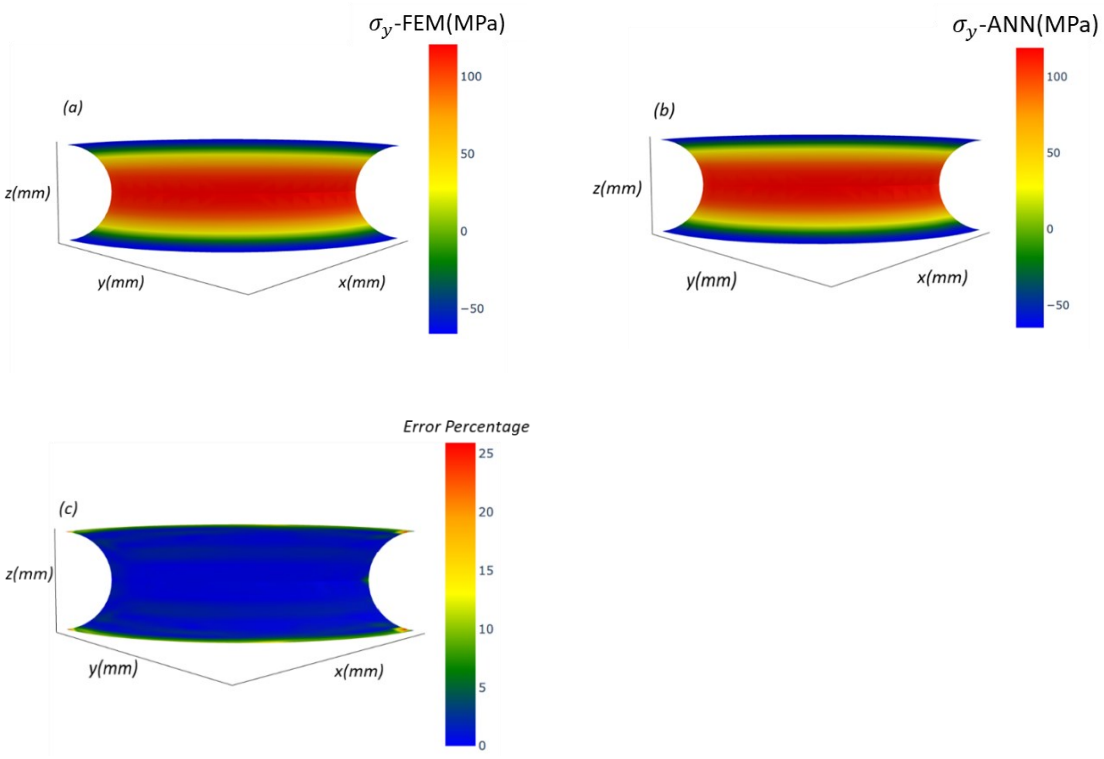


Figure 21. Test configuration 1: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage of σ_y

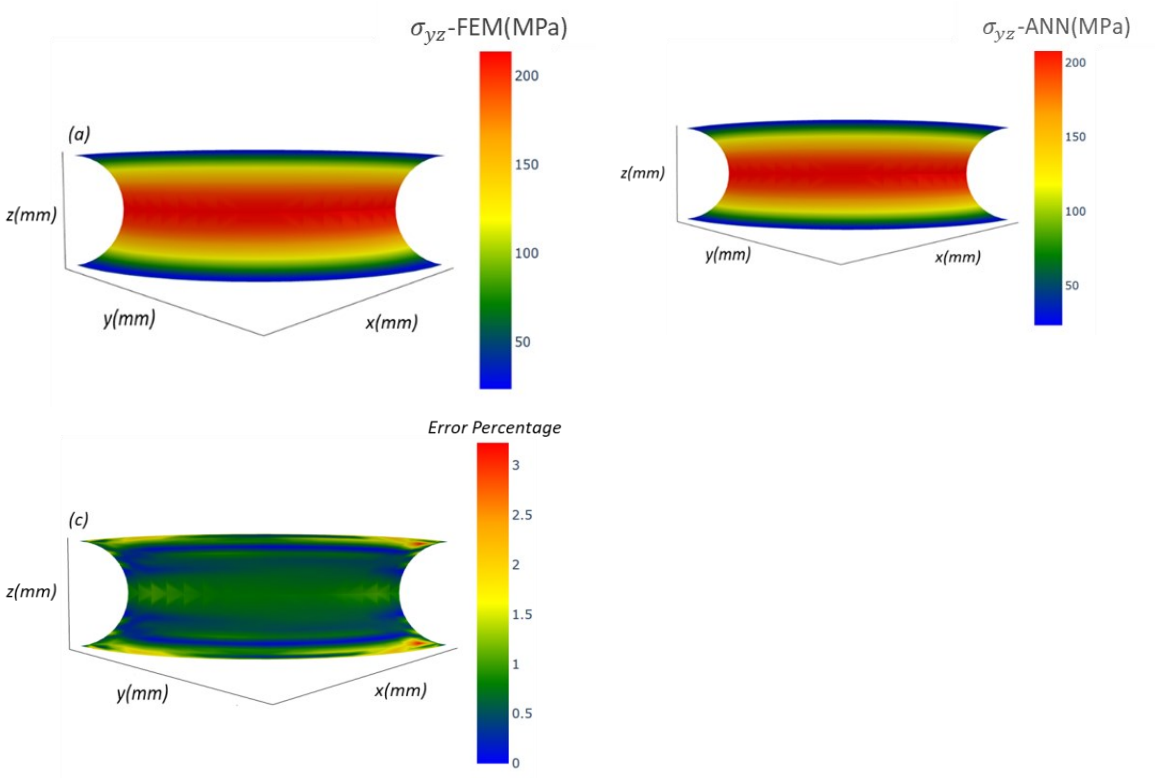


Figure 22. Test configuration 1: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage contours of σ_{yz}

- Test Configuration 2 (Interpolation)

Case two includes the elastic input data of the notched bar under new loadings (Load case-8), and Ramberg-Osgood constants of a new material (M-6) obtained based on interpolating. Figures 23-25 show stress-strain curves based on the FE model and the ANN model for nodal points α, β and γ. The predicted stress-strain results before the yield point involve high accuracy. The error percentage of normal stress (σ_z) in the last increment is 6%. The predicted σ_z in the last time increment is 254 MPa, while corresponding FEM elastic-plastic stress for node α equals 274 MPA, as represented in Figure 23(a).

Additionally, the error percentage for σ_y in the last increment for node α is 5%. Figure 24(a) illustrates that the predicted normal stress (σ_z) for point β is 227 MPa, and corresponding FEM value is 242 MPa, which shows only 6% percentage error.

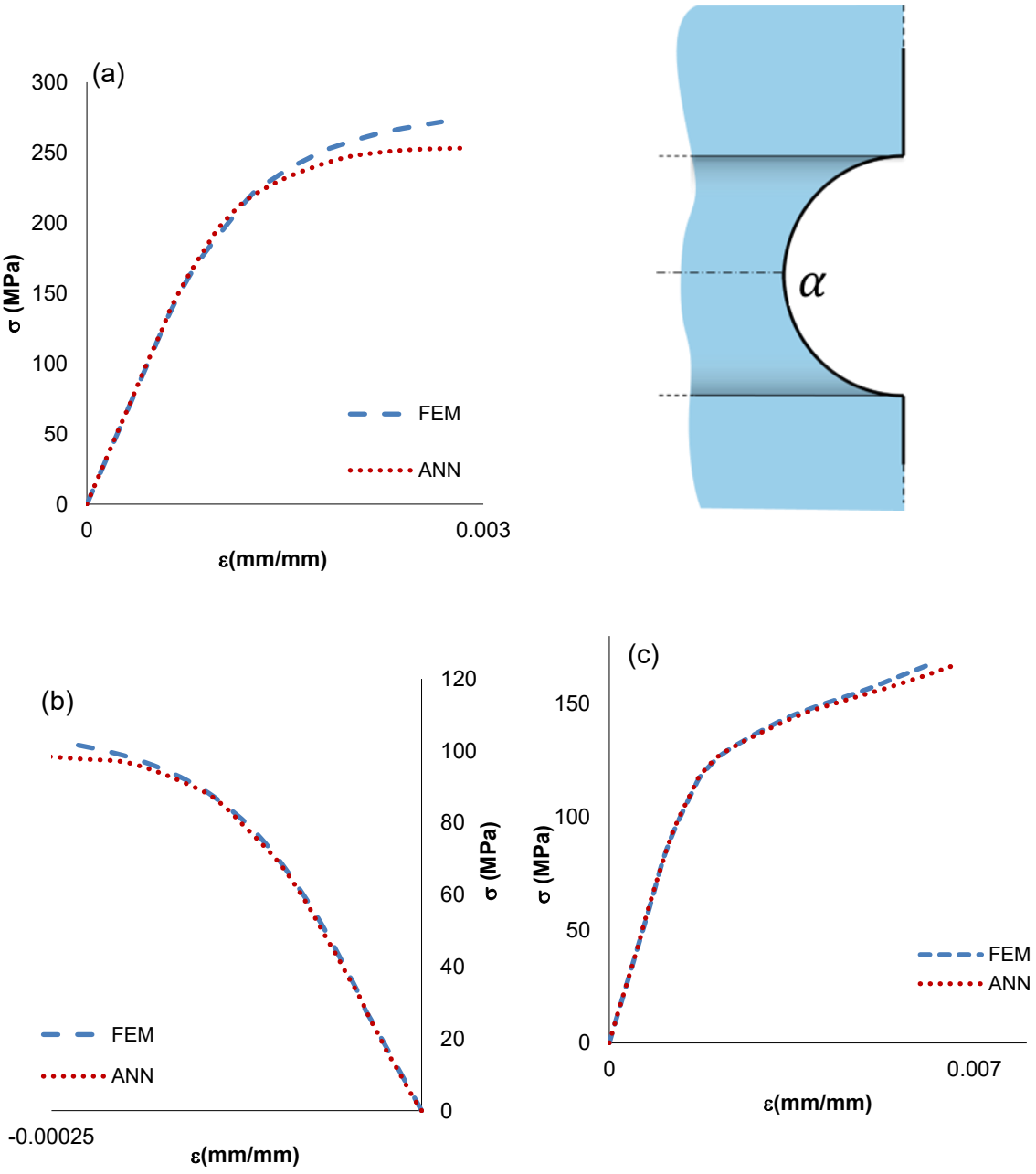


Figure 23. The σ-ε curve of stress point α for (a) normal Stress in the Z direction, (b) normal Stress in the Y direction, and (c) shear stress in the YZ direction

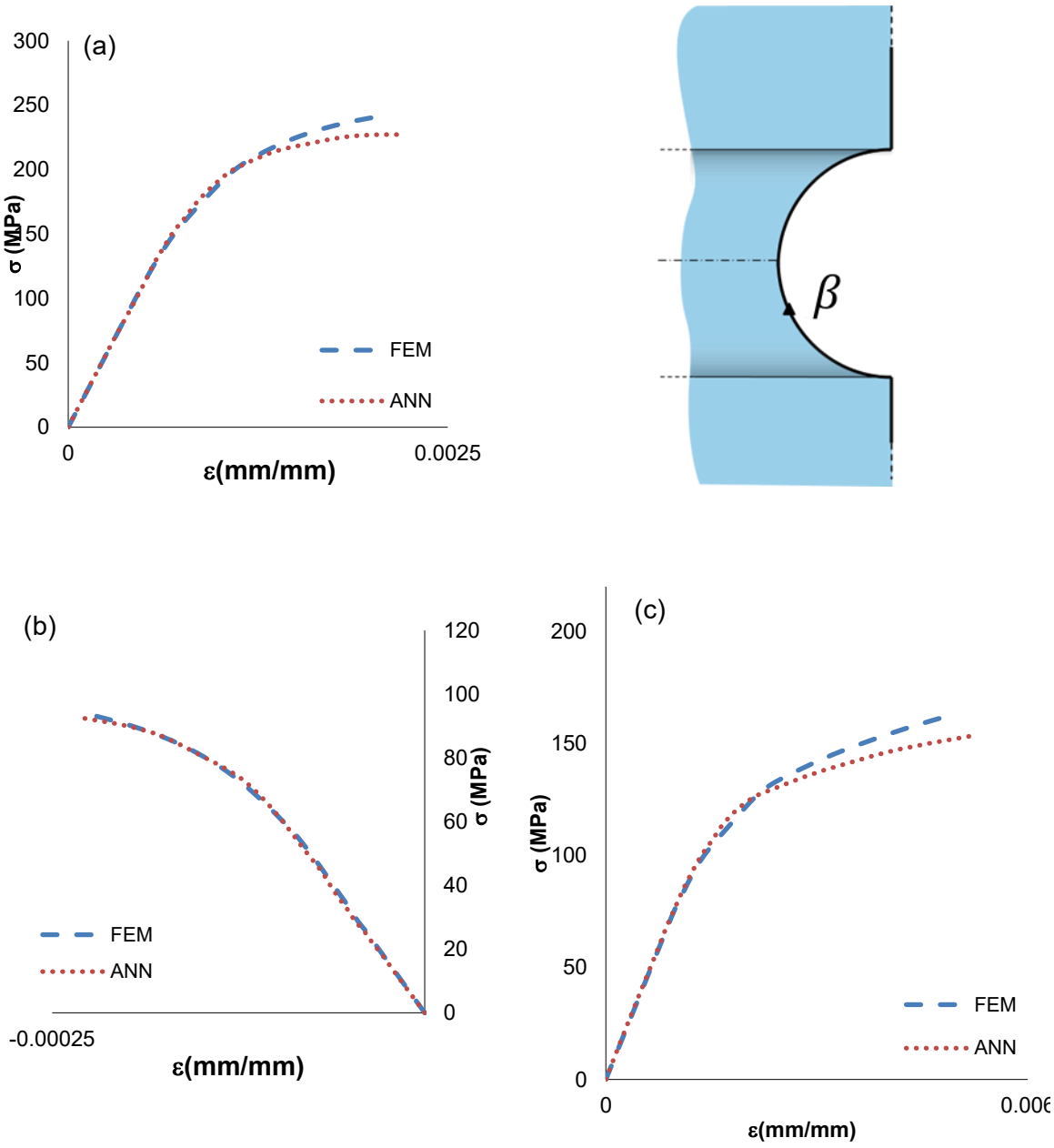


Figure 24. The σ-ε curve of stress point β for (a) normal stress in the Z direction, (b) normal stress in the Y direction, and (c) shear stress in the YZ direction

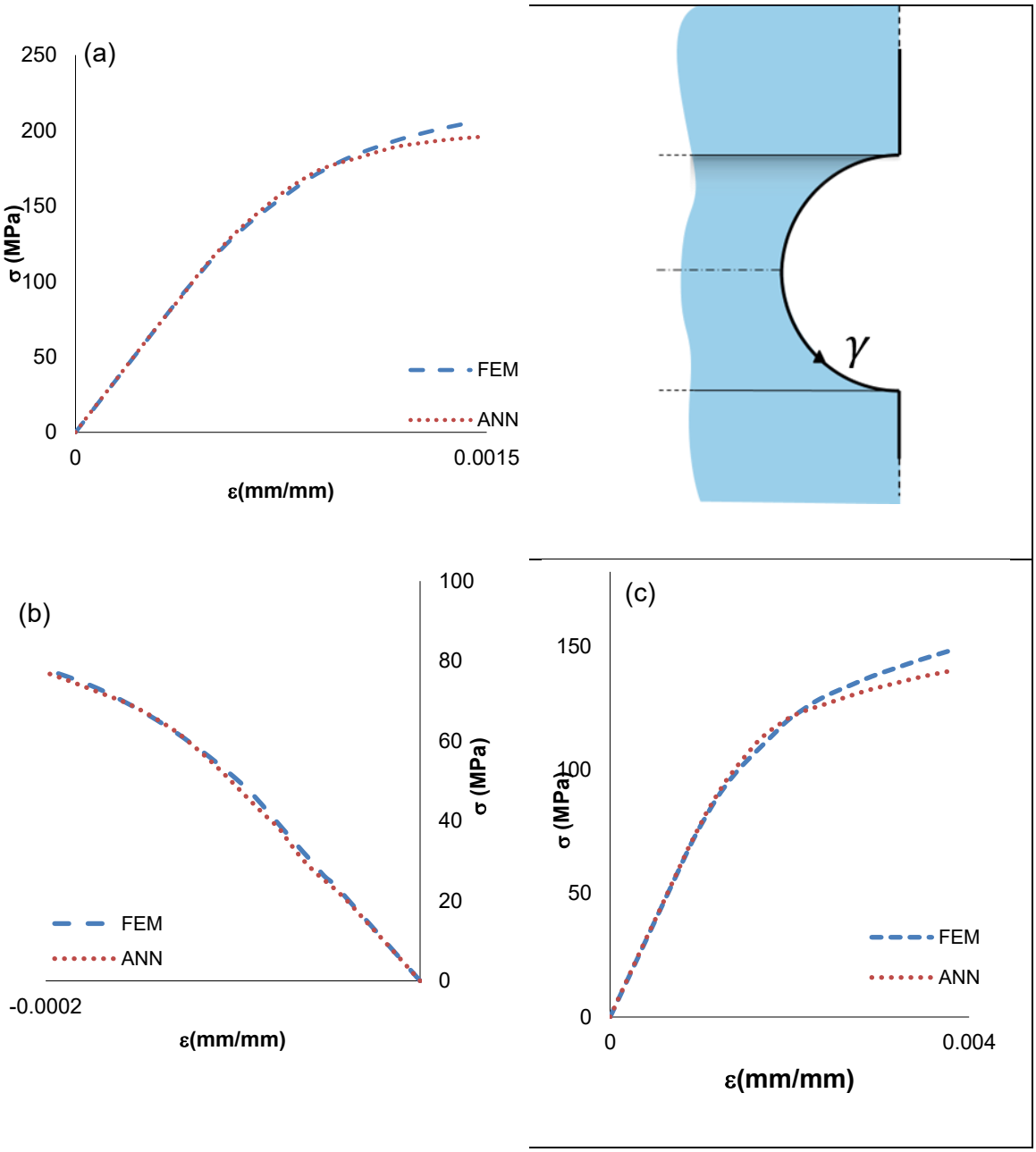


Figure 25. The σ-ε curve of stress point γ for (a) normal stress in the Z direction, (b) normal stress in the Y direction, and (c) shear stress in the YZ direction

Figure 26(c) shows that, the error percentage for σ_z is less than 4% however, for limited nodal points located at the edge of the notch it increases to 12%. Subsequently, the error percentage for σ_y is less than 10% and for the minority of nodes is 40% which the elastic-plastic stress of the corresponding error percentage is 50 MPa as shown in Figure 27. The error percentage for shear stress (σ_{yz}) is between 1 and 6 %, which means that the ANN model accurately predicts elastic-plastic results (Figure 28).

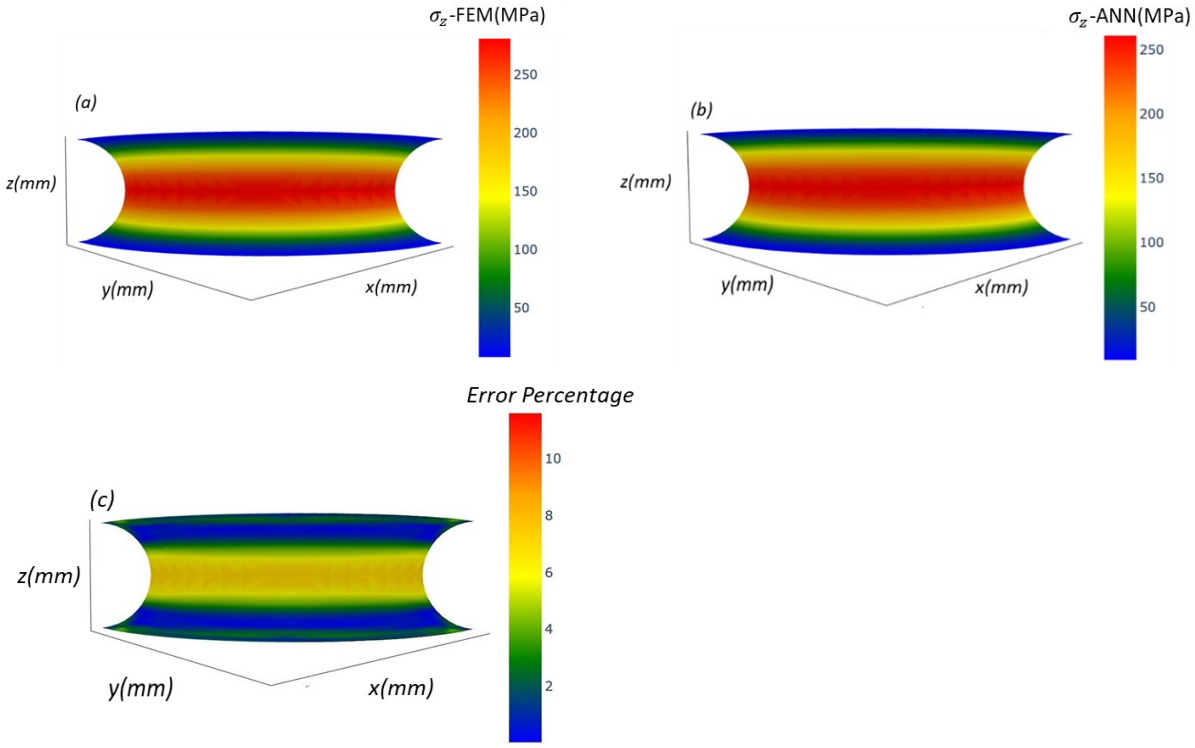


Figure 26. Test configuration 2: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage contours of σ_z

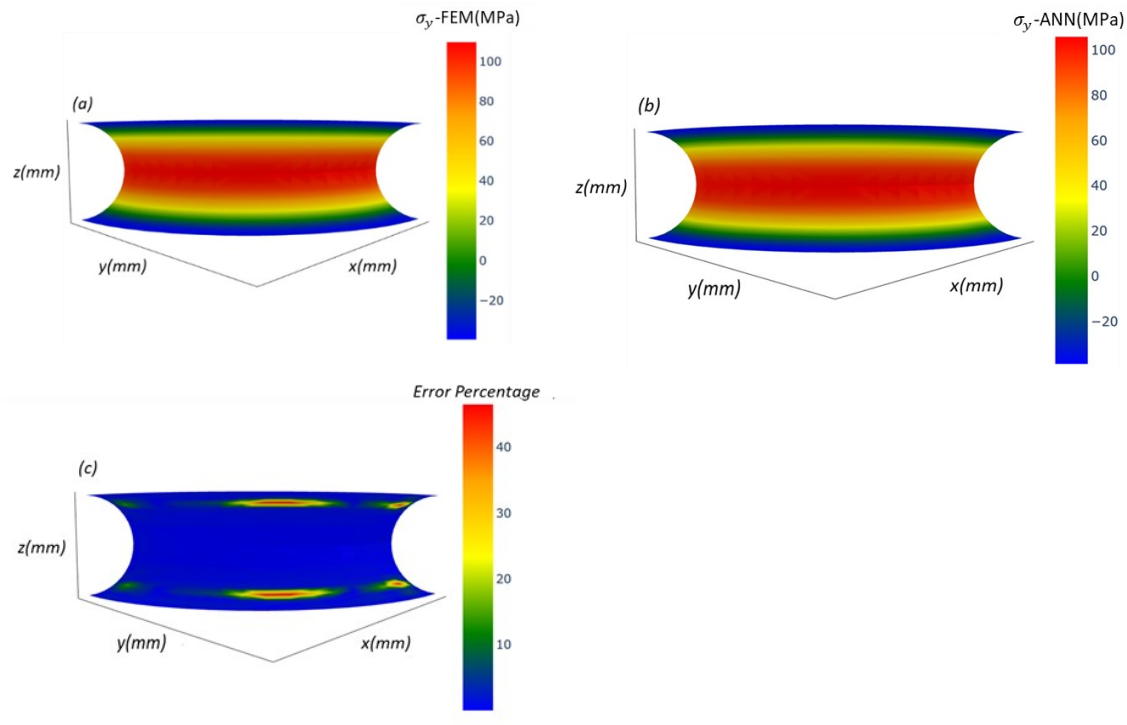


Figure 27. Test configuration 2: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage contours of σ_y

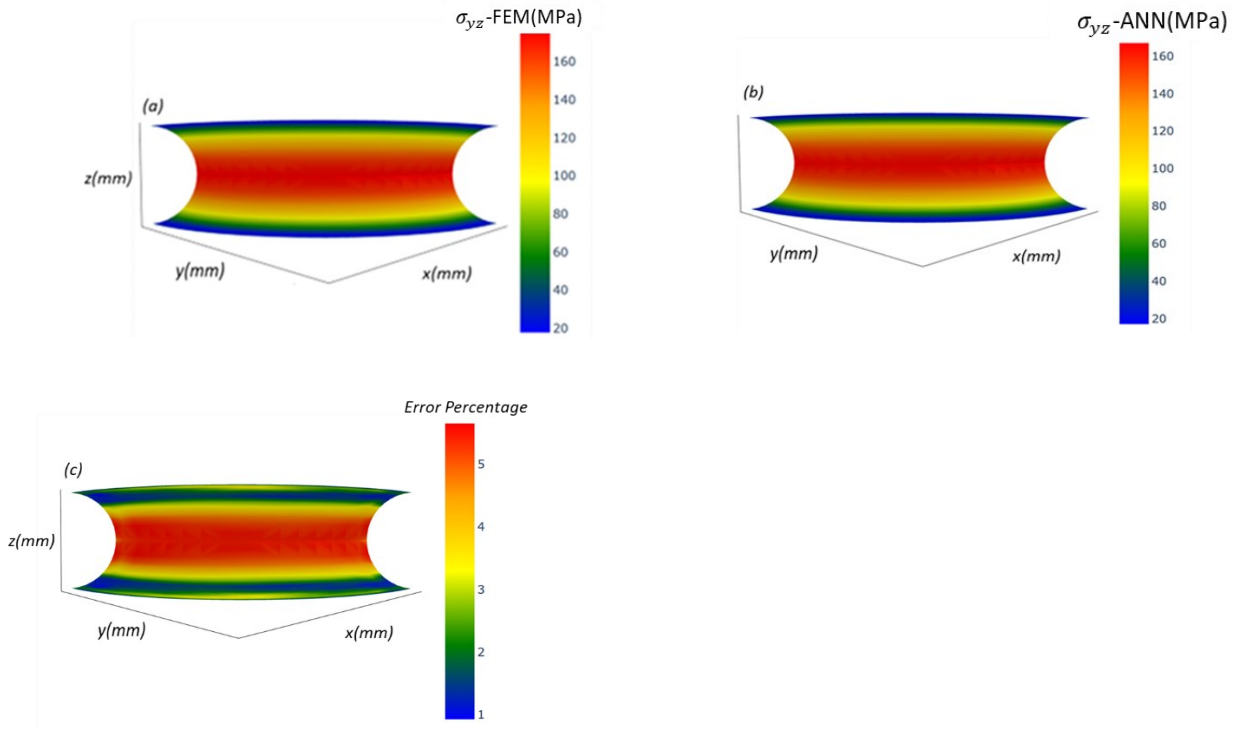


Figure 28. Test configuration 2: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage contours of σ_{yz}

- Test Configuration 3 (Extrapolation)

An artificial neural network is not a strong approach in the case of extrapolation. Therefore, choosing materials' hardening and loading for test data in case of extrapolation requires more consideration. In the present study, the values obtained for the extrapolation case are slightly higher than the values in the interpolation zone. M-1 and load case 9 are employed for the test configuration 3. σ_y changes between 0 and 60 MPa approximately before the yield point in the datasets, while it changes between 0 and 80 MPa for test configuration 3 (Figures 29–31(b)) based on the employed loadings and material. Therefore, the ANN results are not highly accurate before the yield point because of the configuration

of inputs and outputs. It can be seen that the maximum error percentage of σ_z and σ_{yz} has increased to 20% (Figures 32 and 33), while for interpolation cases, the maximum error percentage of σ_z is 12%, and shear stress error percentage is less than 6%.

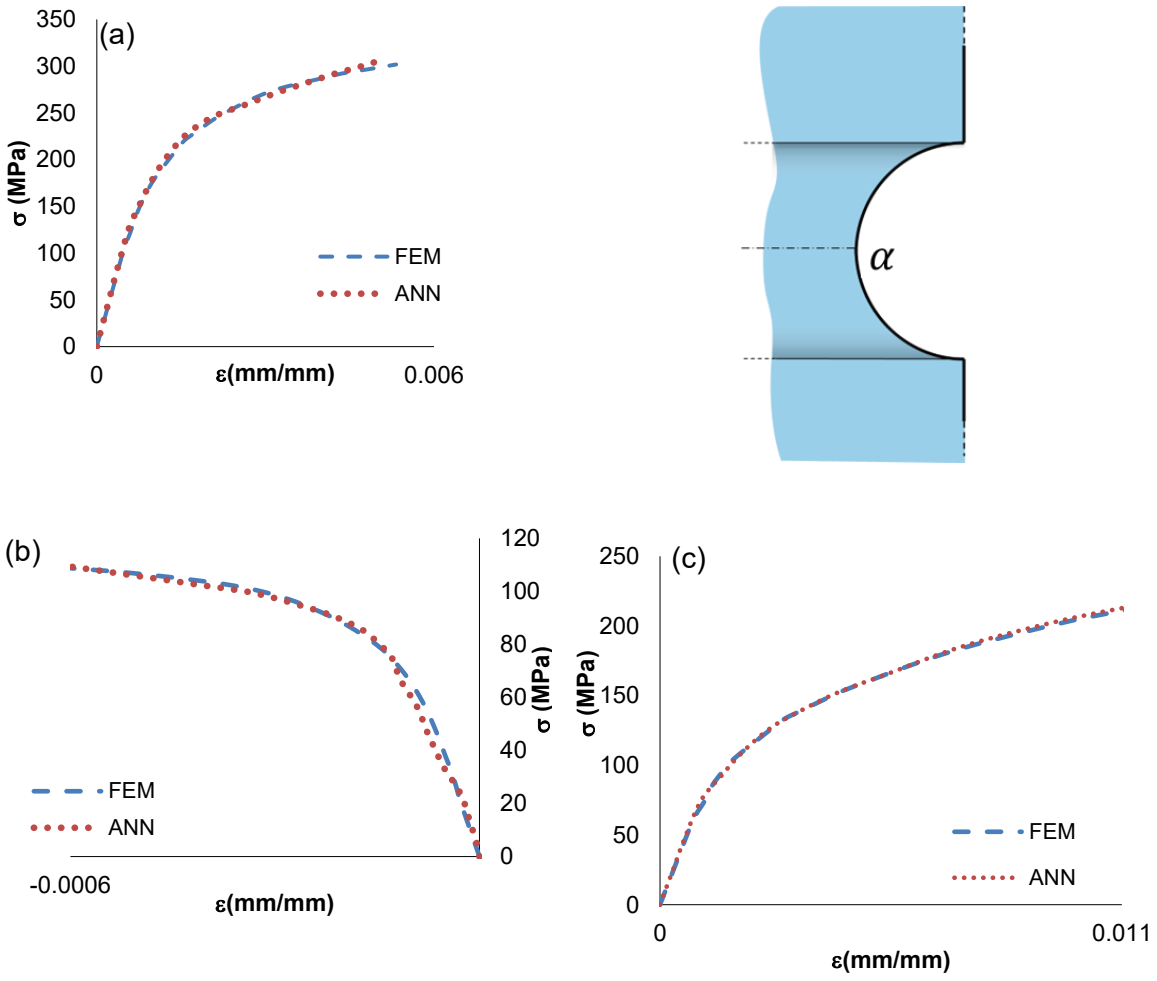


Figure 29. The σ-ε curve of stress point α for (a) normal stress in the Z direction, (b) normal stress in the Y direction, and (c) shear stress in the YZ direction

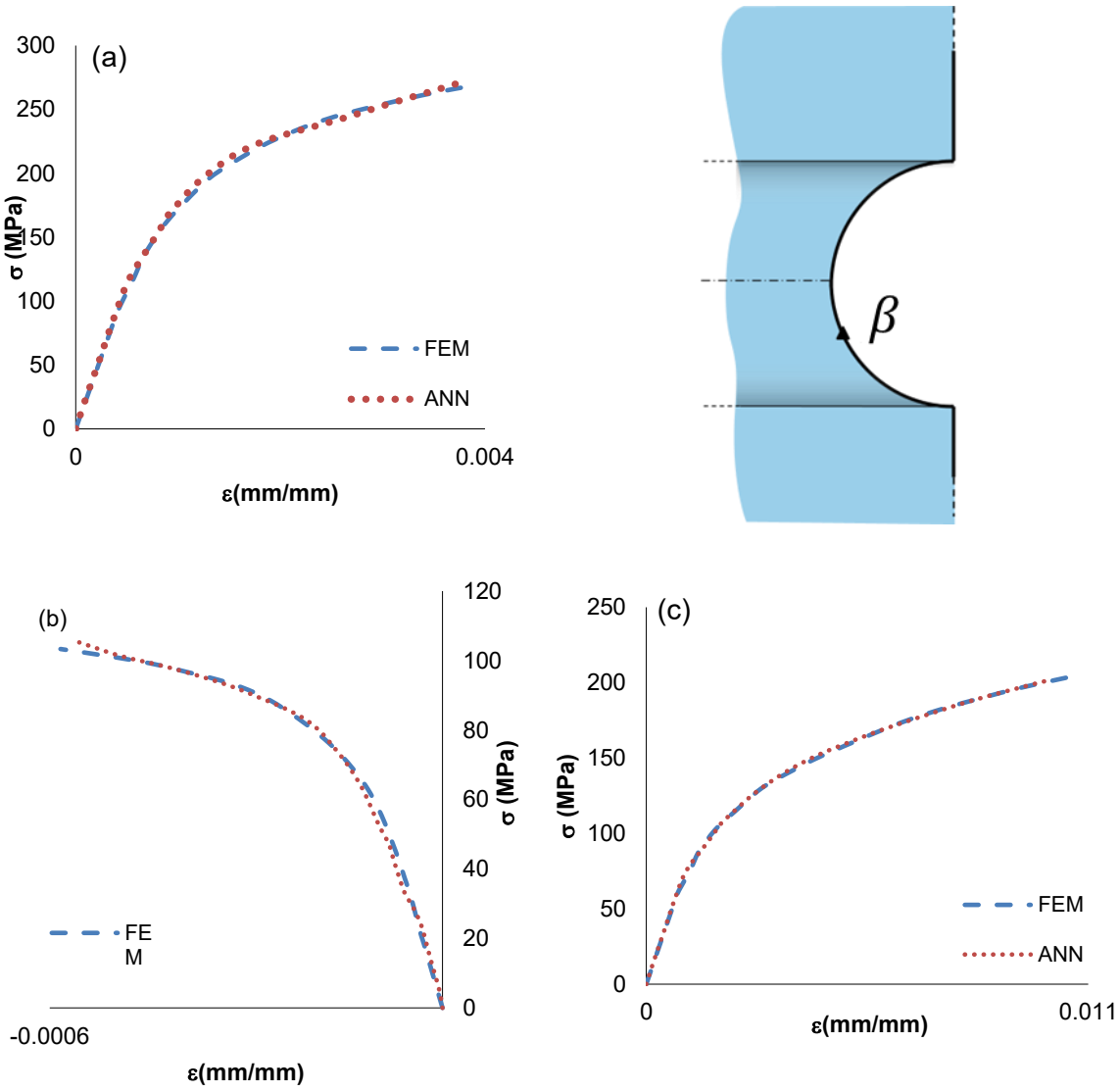


Figure 30. The σ-ε curve of stress point β for (a) normal stress in the Z direction, (b) normal stress in the Y direction, and (c) shear stress in the YZ direction

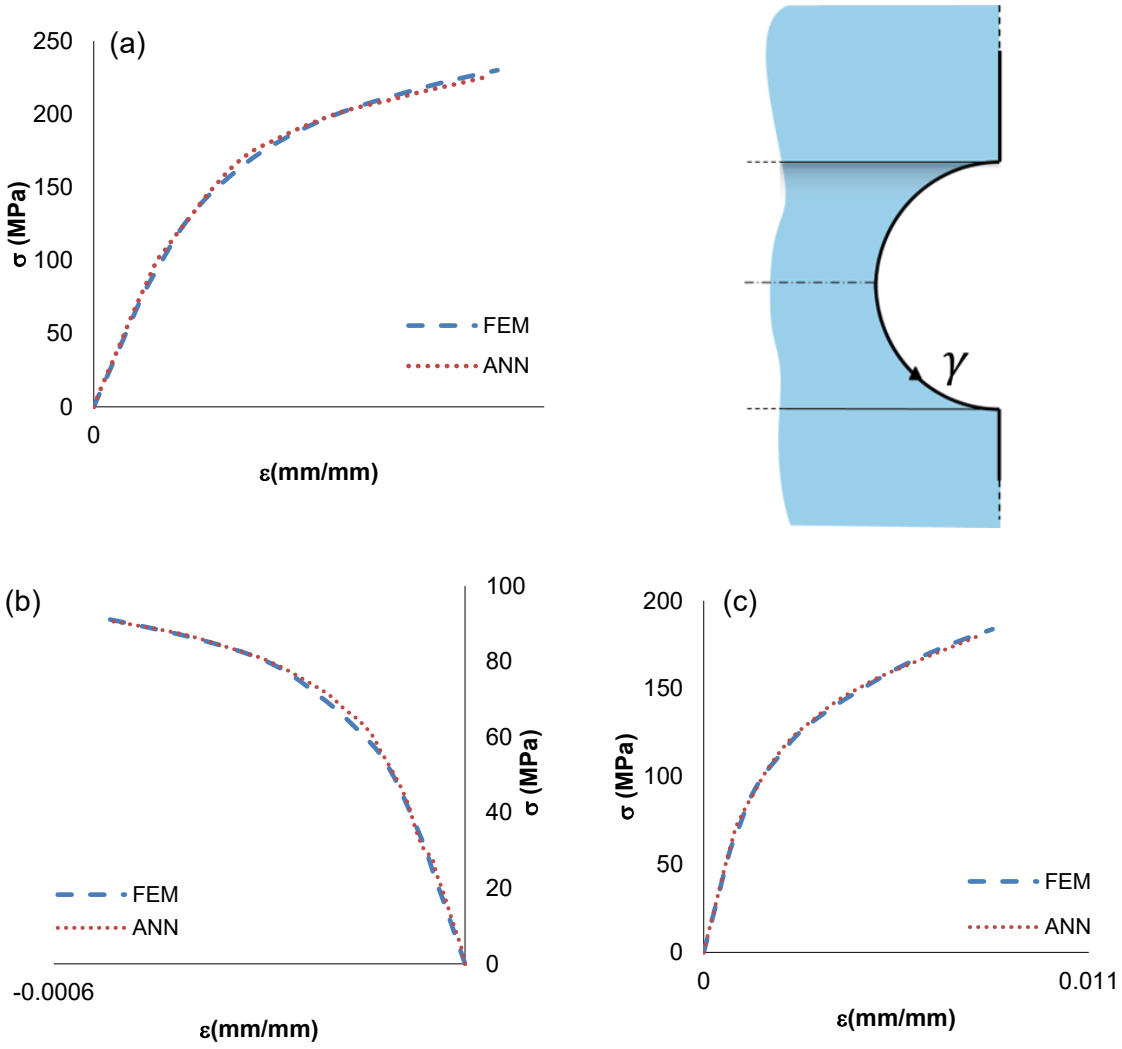


Figure 31. The σ-ε curve of stress point γ for (a) normal stress in the Z direction, (b) normal stress in the Y direction, and (c) shear stress in the YZ direction

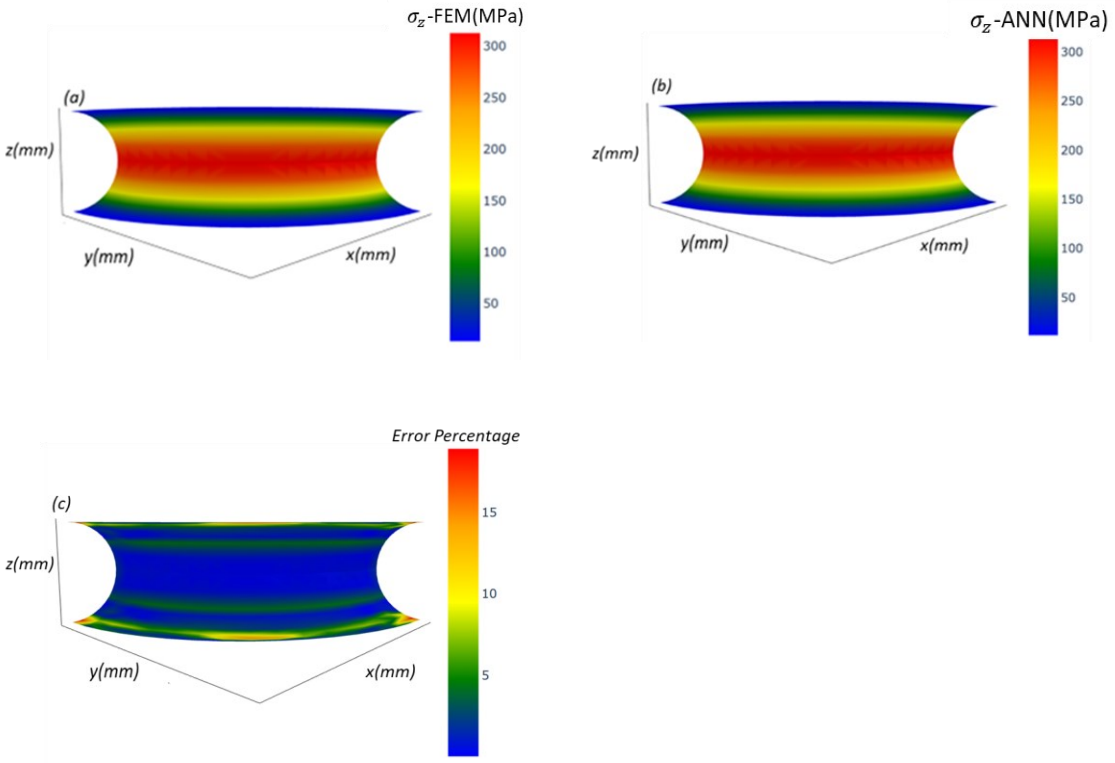


Figure 32. Test configuration 3: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage contours of σ_z.

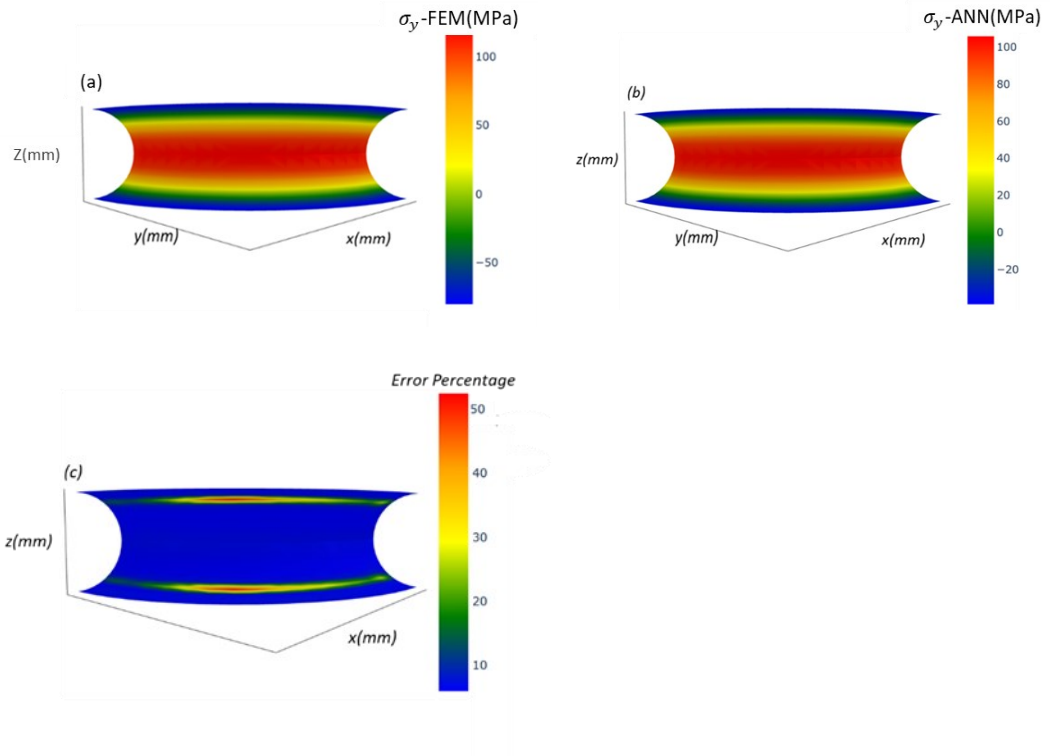


Figure 33. Test configuration 3: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage contours of σ_y.

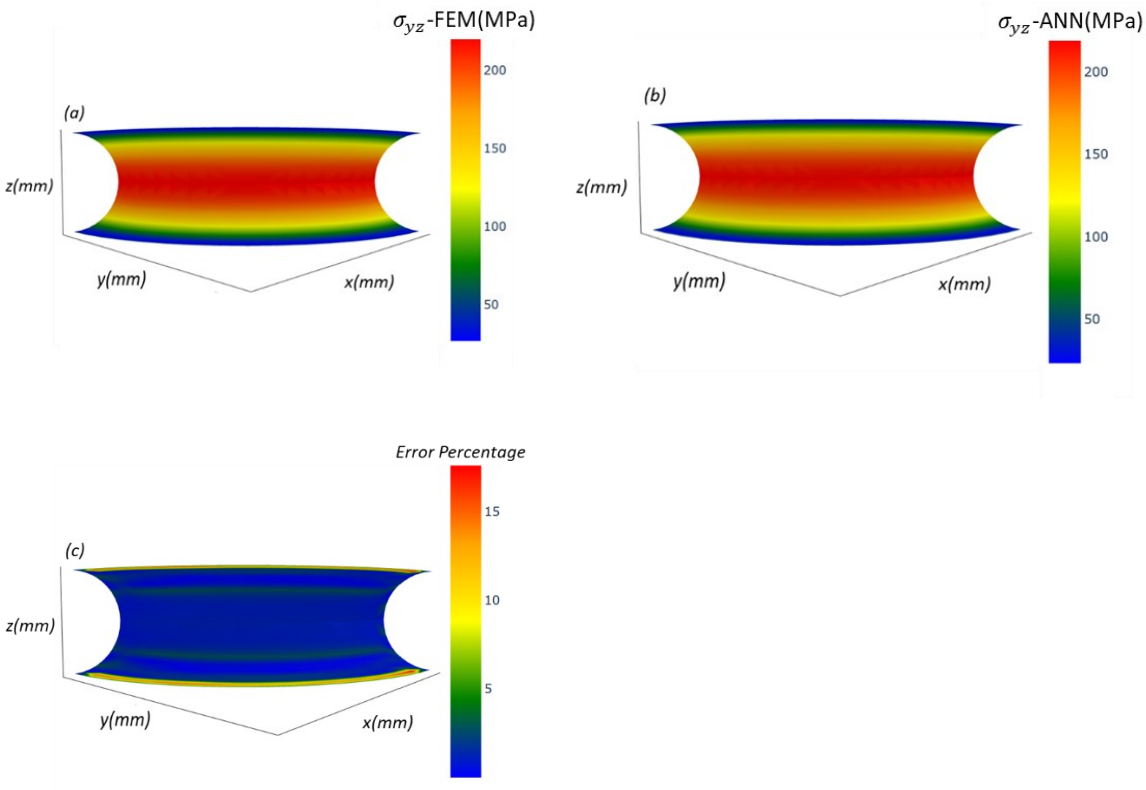


Figure 34. Test configuration 3: (a) FEM stress, (b) predicted ANN stress, and (c) error percentage contours of σ_{yz}.

Figure 35 presents the normal stress (σ_z) of nodes located at x=0 for three test configurations. It is evident that the predicted ANN results for test configuration 1 are highly accurate, while in test configuration 2, the accuracy decreases. The accuracy of predicted stress of test configuration 2 for the point located at Y=2 mm is 92%, whereas, for test configuration 1, it is 99%.

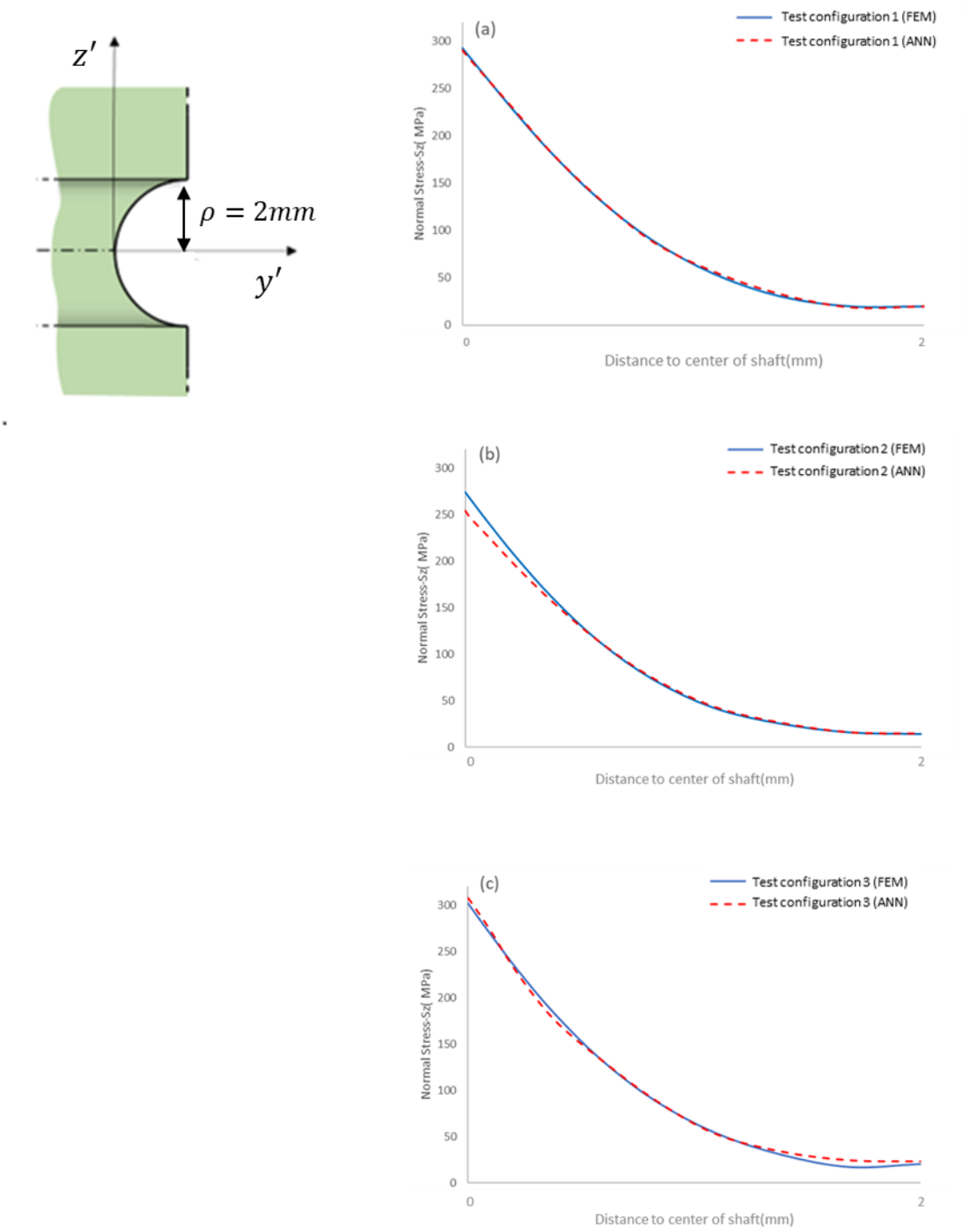


Figure 35. Predicted and FEM stress values for points located in X=0 (a) test configuration 1, (b) test configuration 2, and (c) test configuration3

2.4 Conclusion

In this paper, an ANN model was developed based on two FE models to predict stresses and strains at the surface of a notched shaft under monotonic multiaxial loading. The ANN is a data-based modeling approach; therefore, sufficient data need to be generated to achieve the desired accuracy for the approached model. Nine fictitious materials were defined based on the Ramberg-Osgood law, and the elastic-plastic FE model generated sufficient data for various loadings. The generalizability of the model was evaluated by examining the model with the testing data not used in the training and verification stages. The accuracy of the predicted results for the testing stage was taken into account by comparing results with elastic-plastic stress and strain generated by the elastic-plastic FE model.

The results show that the ANN approach can accurately determine elastic-plastic stress and strain fields based on linear elastic stress fields. In addition, using an ANN to predict elastic-plastic behavior provides significant savings in computational time over complex non-linear FEM methods. Generalizing the model by interpolating the input data is possible, and the ANN model is robust. However, although the ANN model accurately predicted the results for extrapolation cases, it cannot be used vastly.

Chapter 3

Conclusion and Future Research

3.1 Conclusion

In this thesis, based on the FE model, an ANN model was developed to predict the stress and strain states of a notched shaft subject to multiaxial loading. In order to do so, two different FE models were defined to feed the ANN model. Well-structured data is required to establish a relationship between stress FE analysis data under the elastic state and elastic-plastic stress-strain FE data. Therefore, adequate inputs and outputs were obtained to train and verify the ANN model. The model was tested with a new dataset and the predicted results were verified by stress-strain FE analysis data under an elastic-plastic state. As a result, the predicted results are in fair agreement with elastic-plastic FEM data. The error percentage of predicted results is less than 5% for the majority of nodal stresses. The error percentage increased for limited nodal stresses in Y direction. The reason is the low range of stress in Y direction. The proposed approach for estimating elastic-plastic results for a notched model under multiaxial loading is simpler in comparison to FE and analytical methods. The FE method is considered computationally expensive for complex models in the case of geometry, and the analytical solution is not generalizable. However, the proposed ANN model is able to predict elastic-plastic stress-strain results for notch components made of different materials and subject to various loadings. Notably, ANN is not a robust approach in the case of extrapolating data. The main reason is that the ANN approach is data-based; therefore, predicting results for extrapolated inputs can be complex for the ANN model. In

addition, there are limitations to the proposed model. The developed ANN approach is not able to predict results for a notched component under cyclic stress because the ANN model is trained on the FE data under monotonic loading.

3.2 Future research

In the future, the proposed ANN model can be improved in several ways. Adding efficient variables to the structure of inputs and outputs significantly impacts the generalization of the proposed model. Geometry features can be considered and added to variables in order to be able to predict results for different notch sizes and shapes such as semicircular and V-shapes.

In addition, cyclic loading is an important issue in different engineering industries with respect to fatigue life. Therefore, the model can be updated by determining cyclic loading; however, the data structure must change to predict results for cyclic loading.

Bibliography

[1] Ince A, Bang D, "Deviatoric Neuber method for stress and strain analysis at notches under multiaxial loadings," *Int. J. Fatigue*, 2017; 102: 229-240.

[2] Burghardt R, Wächter M, Masendorf L, Esderts A, "Estimation of elastic-plastic notch strains and stresses using artificial neural network," *Fatigue & Fracture of Engineering Materials & structures*, 2021; 44: 1-18.

[3] Neuber H, "Theory of stress concentration for shear strained prismatic bodies with arbitrary stress-strain law," *J Appl Mech*, 1961; 28: 544–50.

[4] Topper TH, Wetzel RM, Morrow JD, "Neuber's rule applied to fatigue of notched specimens," *J Mater*, 1969; 1: 200–9.

[5] Molski K, Glinka G, "A method of elastic-plastic stress and strain calculation at a notch root," *Mater Sci Eng*, 1981; 50: 93–100.

[6] Glinka G, "Energy density approach to calculation of inelastic strain–stress near notches and cracks," *Eng. Fract Mech*,1985; 22: 485–508.

[7] Sharpe WN, Yang CH, Tregoning RL, "An evaluation of the Neuber and Glinka relations for monotonic loading," *J Appl Mech, ASME*, 1992; 59: S50–6.

[8] Hoffmann M, Seeger T, "A generalized method for estimating multiaxial elastic plastic notch stresses and strains, part I: theory," *J Eng. Mater Technol*, 1985; 107: 250–4.

[9] Singh MNK, "Notch tip stress-strain analysis in bodies subjected to nonproportional cyclic loads," [Ph.D. Dissertation], University of Waterloo (ON, Canada): Dept. Mech. Eng.,1998.

[10] Ye D, Matsuoka S, Suzuki N, Maeda Y, "Further investigation of Neuber's rule and the equivalent strain energy density (ESED) method," *Int J Fatigue*, 2004; 26: 447-455.

[11] Jianhui L, Shengnan W, Wuyin J, Wen G, "A modified method for calculating notch-root stresses and strains under multiaxial loading," *Adv Mech Eng*, 2014; 6: 513804.

[12] Firat M, "Cyclic plasticity modeling and finite element analyzes of circumferentially notched round bar under combined axial and torsion loadings," *Mater Des*, 2012; 34: 842–52.

[13] Gao Z, Qiu B, Wang X, Jiang Y, "An investigation of fatigue of a notched member," *Int J Fatigue*, 2010; 32: 1960–9.

[14] Marangon C, Campagnolo A, Berto F, "Three-dimensional effects at the tip of rounded notches subjected to mode-I loading under cyclic plasticity," *J Strain Anal Eng Des*, 2015; 50: 299–313.

[15] Ince A, Glinka G, Buczynski A, "Computational modelling of multiaxial elasto-plastic stress-strain response for notched components under non-proportional loading," *Int J Fatigue*, 2014; 62: 42-52.

[16] Ince A, "Numerical validation of computational stress and strain analysis model for notched components subject to nonproportional loadings," *Theor Appl Fract Mech*, 2016; 84: 26-37.

[17] Ince A, Glinka G, "Innovative computational modelling of multiaxial fatigue analysis for notched components," *Int J Fatigue*, 2016; 82: 134-145.

[18] Haque, M.E, Sudhakar, K.V, "Ann backpropagation prediction model for fracture toughness in microalloy steel," *International Journal of Fatigue,* 2002; 24: 1003-1010.

[19] Pyoa J, Duan H, Baek S, Kim MS, Jeon T, Kwon JS, Lee H, Choa KHA, "convolutional neural network regression for quantifying cyanobacteria using hyperspectral imagery," *Remote Sens Environ*, 2019; 233: 111350.

[20] Goodfellow J, Bengio Y, Courville A, "Deep learning," *MIT Press*, 2016.

[21] Sarvamangala DR, Kulkarni RV"Convolutional neural networks in medical image understanding: a survey," *Evolutionary Intelligence,* 2022;15: 1-22.

[22] Chen J, Liu Yongming, "Fatigue modeling using neural networks: A comprehensive review," *Fatigue and Fracture of Engineering Materials & Structures*, 2022; 45: 945-979.

[23] Mortazavi S.N.S, Ince A, "Artificial neural networks-based J-integral prediction for cracked bodies under elasto-plastic deformation state-monotonic loading", *International Journal of Fatigue,* 2022; 167: 107311.

[24] Sharma A, Sharma S, Athaiya A, "Activation function in neural network", *International Journal of Engineering applied Science and Technology*, 2020; 4: 2455-2143.

[25] Chen J, Liu Yongming, "Fatigue modeling using neural networks: A comprehensive review", *Fatigue and Fracture of Engineering Materials & Structures*, 2022; 45:945-979.

[26] Pujol JCF, Pinto JMA, "A neural network approach to fatigue life prediction", *Int J Fatigue*, 2011; 33:313-322.

[27] Maleki E, Unal O, Reza Kashyzadeh K, "Fatigue behavior prediction and analysis of shot peened mild carbon steels", *Int J Fatigue*, 2018; 116:48-67.

[28] Kusiak J, Kuziak R, "Modelling of microstructure and mechanical properties of steel using the artificial neural network", Journal *of Materials Processing Technology*, 2002; 127:115-121.

[29] Yan C, Gao R, Huang W, "Asphalt mixture fatigue life prediction model based on neural network", *CICTP*, 2018:1292-1299.

[30] Genel K, "Application of artificial neural network for predicting strain-life fatigue properties of steels on the basis of tensile tests", *Int J Fatigue*, 2004; 26:1027-1035.

[31] Kang JY, Song JH, “Neural network applications in determining the fatigue crack opening load”, *Int J Fatigue*, 1998; 20: 57-69.

[32] Xie L, Yang Y, Zhou Z, Zheng J, Tao M, Man Z, “Dynamic neural modeling of fatigue crack growth process in ductile alloys”, *Inf Sci (Ny)*, 2016; 364–365:167-183.

[33] Mortazavi S.N.S, Ince A, “A radial basis function artificial neural network methodology for short and long fatigue crack propagation,” *Proc Can Soc Mech Eng. Int Congr 2021*, 2021.

[34] Mortazavi SNS, Ince A, “An artificial neural network modeling approach for short and long fatigue crack propagation,” *Comput Mater Sci*, 2020; 185: 109962.

[35] Bezazi A, Pierce SG, Worden K, Harkati EH, “Fatigue life prediction of sandwich composite materials under flexural tests using a Bayesian trained artificial neural network,” *Int J Fatigue*, 2007; 29: 738-747.

[36] El Kadi H, Al-Assaf Y, “Energy-based fatigue life prediction of fiberglass/epoxy composites using modular neural networks,” *Compos Struct*, 2002; 57: 85–9.

[37] Bucˇar T, Nagode M, Fajdiga M, “A neural network approach to describing the scatter of S–N curves,” *Int J Fatigue*, 2006; 28: 311-323.

[38] Christiansen N.H, Voie P.E.T, Høgsberg J, Sødahl N, “Efficient mooring line fatigue analysis using a hybrid method time domain simulation scheme”, *In Proceedings of the ASME 2013 32nd International Conference on Ocean, Offshore and Arctic Engineering, Nantes*, 2013: V001T01A035.

[39] Yang J, Kang G, Liu Y, Kan Q, “A novel method of multiaxial fatigue life prediction based on deep learning”, *Int J Fatigue*, 2021; 151:106356.

[40] Mohanty R, Verma BB, Ray PK, Parhi DRK, “Application of artificial neural network for fatigue life prediction under interspersed mode-I spike overload”, *J Test Eval*, 2010; 38: 177-187.

[41] Abambres M, Lantsoght EOL, “ANN-based fatigue strength of concrete under compression”, *Materials (Basel)*, 2019; 12:26-34.

[42] Guo S, Li C, Shi J, Luan F, Song X, “Effect of quenching media and tempering temperature on fatigue property and fatigue life estimation based on RBF neural network of 0.44% carbon steel”, *Mech Sci*, 2019;10: 273–286.

[43] Belfiore NP, Ianniello F, Stocchi D, et al, “A hybrid approach to the development of a multilayer neural network for wear and fatigue prediction in metal forming,” *Tribol Int*, 2007;40: 1705–1717.

[44] Ramasubramanian K, Moolayil J, “Applied supervised learning with R,” *Packt*, 2019.

[45] Mahamad AK, Saon S, Hiyama T, “Predicting remaining useful life of rotating machinery based artificial neural network,” *Comput Math Appl*, 2010; 60: 1078-1087.

[46] Dresia K, Waxenegger-Wilfing G, Riccius J, Deeken J, Oschwald M, “Numerically efficient fatigue life prediction of rocket combustion chambers using artificial neural networks Networks,” *EUCASS2019*, 2019: 264.

[47] Ramasubramanian K, Moolayil J, “Applied supervised learning with R,” *Packt*, 2019.

[48] Zhan Z, Li H, “A novel approach based on the elastoplastic fatigue damage and machine learning models for life prediction of aerospace alloy parts fabricated by additive manufacturing,” *Int J Fatigue*, 2021,145: 106089.

[49] Eftekhar Azam S, Rageh A, Linzell D, “Damage detection in structural systems utilizing artificial neural networks and proper orthogonal decomposition,” *Struct Control HealthMonit*, 2019; 26:1-24.

[50] Sbarufatti C, Manes A, Giglio M, “Performance optimization of a diagnostic system based upon a simulated strain field for fatigue damage characterization,” *Mech Syst Signal Process*, 2013; 40:667-690.

[51] Lotfi B, Beiss P, “Application of neural networking for fatigue limit prediction of powder metallurgy steel parts,” *Mater Des*, 2013; 50: 440-445.

[52] Zhang W, Bao Z, Jiang S, He J, “An artificial neural network-based algorithm for evaluation of fatigue crack propagation considering nonlinear damage accumulation,” *Materials*, 2016; 9: 483.

[53] Hoffmann M, Seeger T, “Stress-strain analysis and life predictions of a notched shaft under multiaxial loading,” In: Leese GE, Socie D, editors. Multiaxial fatigue: analysis and experiments AE-14. Warrendale (PA): Society of Automotive Engineers, 1989: 81–101.

[54] Barkey ME, “Calculation of notch strains under multiaxial nominal loading,” [Ph.D. Dissertation], University of Illinois at Urbana-Champaign, 1993.

[55] Koettgen VB, Schoen M, and Seeger T, “Application of a multiaxial load-notch strain approximation procedure to autofrettage of pressurized components,” *ASTM SPECIAL TECHNICAL PUBLICATION*, 1993: 375-375.

[56] Moftakhar A, “Calculation of time independent and time-dependent strains and stresses in notches,” [Ph.D. Dissertation], Waterloo (ON, Canada): University of Waterloo, Department of Mechanical Engineering, 1994.

[57] Albuquerque VHCD, Tavares JMRS, Cortez PC, “Quantification of the microstructures of hypoeutectic white cast iron using mathematical morphology and an artificial neural network,” *Int J Microstruct Mater Prop*, 2010; 5: 52.

[58] Artymiak P, Bukowski L, Feliks J, Narberhaus S, Zenner H, “Determination of SN curves with the application of artificial neural networks,” *Fatigue Fract Eng Mater Struct,* 1999; 22: 723–8.

[59] Ostad Shabani M, Mazahery A, "The ANN application in FEM modeling of mechanical properties of Al-Si alloy," *Applied Mathematical Modeling*, 2011; 35: 5707-5713.

[60] Setti Srinivasu G, Nawal Rao R, "Artificial neural network approach for prediction of stress-strain curve of near β titanium alloy," *Rare Met,* 2014; 33: 249-257.

[61] Mousavi Anijdan SH, Bahrami A, Madaah Hosseini HR, Shafyei A, "Using genetic algorithm and artificial neural network analyses to design an Al-Si casting alloy of minimum porosity," Material*& Design*, 2006; 27: 605-609.

[62] Hajializade F, Ince A, "Integration of artificial neural network with finite element analysis for residual stress prediction of the direct metal deposition process," *Materials Today COMMUNICATION*, 2021; 27:102197.

[63] Venkatesh V, Rack HJ, "A neural network approach to elevated temperature creep-fatigue life of carbon and low-alloy steels," *Nucl Eng Des,* 2000; 197:1-12.

[64] Lee JA, Almond DP, Harris B, "The use of neural networks for the prediction of fatigue lives of composite materials," *Compos Part A: Appl Sci Manufacturing*,1999; 30: 1159-69.

[65] Ince R, "Prediction of fracture parameters of concrete by Artificial Neural Networks," *Engineering Fracture Mechanics*, 2003;71: 2143-2159.

[66] Fathi A, Aghakouchak A.A, "Prediction of Fatigue crack growth rate in welded tubular joints neural network," *International Journal of Fatigue*, 2007; 29: 261-275.

[67] Alqahtani H, Bharadwaj S, Ray A, "Classification of fatigue crack damage in polycrystalline alloy structures using convolutional neural networks," *Engineering Failure Analysis*, 2021; 119:104908.

[68] Srinivasan V.S, Valsan M, Bhanu Sankara Roa K, mannan S.L, raj B, "Low cycle fatigue and creep-fatigue interaction behavior of 316L(N) stainless steel and life prediction by artificial neural network approach," International *Journal of Fatigue*, 2003; 25:1327-1338.

[69] Thai HT, "Machine learning for structural engineering a state-of-the-art review," *Structures*, 2022; 38:448–91.

[70] Zhang XC, Gong JG, Xuan FZ, "A deep learning based life prediction method for components under creep, fatigue and creep-fatigue conditions," *Int J Fatigue,* 2021; 148:106236.

[71] Kamiyama M, Shimizu K, Akiniwa Y, "Prediction of low-cycle fatigue crack development of sputtered Cu thin film using deep convolutional neural network," *Int J Fatigue*, 2022; 162:106998.

[72] Ramberg W, Osgood WR, "Description of stress-strain curves by three parameters," *(No. NACA-TN-902)*, 1943.

[73] Wang CT, "Applied elasticity," New York: McGraw-Hill Book Company,1953.

[74] Aksu G, Güzeller CM, Eser MT, "The Effect of the Normalization Method Used in Different Sample Sizes on the Success of Artificial Neural Network Model," *International Journal of Assessment Tools in Education*, 2019; 6:170-192.

[75] H. Christiansen N, Totbergsen Voie P.E, Winther O, Høgsberg J, "Comparison of Neural Network Error Measures for Simulation of Slender Marine Structures," *Journal of Applied Mathematics*, 2014; 4:1-11.

[76] Gulikers T, "An Integrated Machine Learning and Finite Element Analysis Framework," Applied to Composite Substructures including Damage. 2018.

Appendix A : Developed Ansys Parametric Design Language scrips

```
FINISH

/CLEAR,NOSTART

! start the preprocessor

/prep7

NOM_R=10

NOM_L=40

NOTCH_R=2

TOLL=0.01

TOTAL_L=NOTCH_R+NOM_L

! Meshing size parameters

DIV_NM_R=16

DIV_NM_C=10

DIV_NM_L=24

DIV_CS_L=16

k,1,0,0

k,2,0,NOM_R-NOTCH_R

k,3,0,NOM_R
```

```
k,4,NOTCH_R,NOM_R

k,5,NOTCH_R+NOM_L,NOM_R

k,6,NOTCH_R+NOM_L,0

k,7,NOTCH_R+NOTCH_R,0

l,1,2       !1

larc,2,4,3,NOTCH_R    !2

l,4,5       !3

l,5,6       !4

l,6,7       !5

l,7,4       !6

l,7,1       !7

allsel

al,1,2,6,7          !1

al,6,3,4,5          !2

lsel,s,,,1

lesize,all,,,DIV_CS_L,1.0/8.0,

lsel,s,,,6

lesize,all,,,DIV_CS_L,1.0/1.0,

lsel,s,,,4
```

```
lesize,all,,,DIV_CS_L,1,

lsel,all,all

lsel,s,,,2

lesize,all,,,DIV_NM_C,1,

lsel,all,all

lsel,s,,,7

lesize,all,,,DIV_NM_C,1,

lsel,all,all

lsel,s,,,3

lsel,a,,,5

lesize,all,,,DIV_NM_L,1,

allsel

ARSYM,X,1,2 ,1 , ,0,0

NUMMRG,KP,0.001, , ,LOW

vrotat,all,,,,,,1,6,360,,

ET,1,SOLID45

TYPE,  1 $ REAL,1 $ MAT,1

vmesh,all

!!ET,1,SOLID45
```

```
!TYPE,  1 $ REAL,1 $ MAT,1

WPROTA,0,-90,0

CSWPLA,11,CYLIN

CSYS,11

VGEN,2,All,,,0,90,0,,0,1

CSYS,1

WPCSYS,1,1

NROTAT, All

ALLSEL

NSEL,ALL

NSEL,R,LOC,Z,+TOTAL_L-TOLL,+TOTAL_L+TOLL

CM,FNODES,NODE

ALLSEL

LSEL,S, , , 27

LSEL,A, , , 41

LSEL,A, , , 55

LSEL,A, , , 60

NSLL,S,1

CM,TNODES,NODE
```

```
/TITLE, SAE1045 NOTCHED BAR LINEAR ANALYSIS

/PREP7

ALLSEL

MP,EX,1,202E3    ! Elastic

MP,NUXY,1,0.3  ! poisson

MAT,1

TB,KINH,1,1,20,0

TBPT,,0.000940,190

TBPT,,0.0011,220

TBPT,,0.0013,240

TBPT,,0.0015,260

TBPT,,0.0017,280

TBPT,,0.0020,300

TBPT,,0.0023,320

TBPT,,0.0028,340

TBPT,,0.0034,360
```

TBPT,,0.0042,380

TBPT,,0.0053,400

TBPT,,0.0066,420

TBPT,,0.0084,440

TBPT,,0.0106,460

TBPT,,0.0134,480

TBPT,,0.0170,500

TBPT,,0.0214,520

TBPT,,0.0269,540

TBPT,,0.0336,560

TBPT,,0.0519,600

/PREP7

*DIM,FORCE,TABLE,1,1,1, , ,

*DIM,TORQUE,TABLE,1,1,1, , ,

!*

*SET,FORCE(0,1,1) , 1

*SET,TORQUE(0,1,1) , 1

*TREAD,FORCE,'force','dat'

*TREAD,TORQUE,'torque','dat'

```
*SET,FORCE(0,1,1) , 1

*VEDIT,,1

*SET,TORQUE(0,1,1) , 1

*VEDIT,,1

/PREP7

/SOLU

! Elast0-plastic FEA

! enter the solver, apply contraints and loads

! apply the fixed contraint

ALLSEL

NSEL,R,LOC,Z,-TOTAL_L-TOLL,-TOTAL_L+TOLL

D,ALL,ALL

ALLSEL

!ANTYPE,TRANS

!TRNOPT,FULL

!NLGEOM,ON

!SSTIF,ON

!EQSLV,PCG,1E-8

!TIMINT,ON
```

```
!  Nonlinear Options:

!CNVTOL,F, ,0.1,2, ,

!NEQIT=20

!NCNV,,,...

PRED,ON

NS=25

NT=1

TM_ST=1

TM_END=1

TM_INC=1

*DO,TM,TM_ST,TM_END,TM_INC

! AXIAL LOADING...

CMSEL,S,FNODES,NODE

*GET,NODECNT1,NODE,,COUNT

F,ALL,FZ,FORCE(TM)/NODECNT1

! ALLSEL

! TORQUE LOADING

CMSEL,S,TNODES,NODE

*GET,NODECNT2,NODE,,COUNT
```

F,ALL,FY,TORQUE(TM)/NODECNT2

!ALLSEL

OUTPR,ALL,ALL

OUTRES,ALL,ALL

TIME,TM

NSUBST,NS,25,25

CUTCONTROL,PLSLIMIT,10

ALLSEL

NSEL,ALL

SOLVE

*ENDDO

/POST1

/GRAPHICS, FULL

ALLSEL

RSYS,1

AllSEL

NSEL,ALL

ASEL,s,,,6 !Select the area you want (e.g. area or 45)

ASEL,a,,,13

```
NSLA,s,1 !Select nodes associated to this area

CM,NNODE,NODE

/POST1

SET,1,1

ALLSEL

CMSEL,S,NNODE,NODE

*VGET,NODELIST,NODE,NNODE,NLIST,,,,4

!*VGET,NODELIST,NODE,1,NSEL

!Count number of nodes selected

*GET,QtyNodes,NODE,0,COUNT

*DIM,Stress,TABLE,QtyNodes,4

SET,1,1,,,1

!Stress Results

*Do,J,1,QtyNodes,1

Stress(J,1) = NODELIST(J)

*GET,Stress(J,2),NODE,NODELIST(J),S,Y

*GET,Stress(J,3),NODE,NODELIST(J),S,Z

*GET,Stress(J,4),NODE,NODELIST(J),S,YZ

*ENDDO
```

Appendix B : Develeoped Python-based scripts

```
 import os
import numpy as np
import pandas as pd
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Activation, Dense
from tensorflow.keras import layers
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

import matplotlib.pyplot as plt
print('Done')
```

```
INPUT = pd.read_csv()
OUTPUT = pd.read_csv()
```

```
from sklearn.preprocessing import MinMaxScaler
sc=MinMaxScaler()
X=sc.fit_transform(INPUT)
X=pd.DataFrame(X)
```

```
from sklearn.preprocessing import MinMaxScaler
sc=MinMaxScaler()
Y=sc.fit_transform(OUTPUT)
Y=pd.DataFrame(Y)
```

```
X_trn = X.loc[]
frame_1= []
X_trn= pd.concat(frame_1)
```

```
X_tst=X.loc[]
```

```
Y_trn_1 = Y.loc[]
Y_tst=Y.loc[]
```

```
model  = Sequential()
model .add(layers.Dense(256, activation="relu", name="layer1"))
model .add(layers.Dense(128, activation="relu", name="layer2"))
model .add(layers.Dense(7 , activation="linear" ,name="layer3")) #We use a sigmoid on the output layer to ensure our network output is between 0 and 1 and easy to map
model.layers
```

```
model.compile(optimizer='adam', loss='mse',metrics=['accuracy'])
keras.optimizers.Adam(lr=0.00001)
```

```
ann_history=model.fit(X_trn,Y_trn,validation_data=(X_tst,Y_tst),
epochs=350,batch_size=100)
fig, ax = plt.subplots(1, 1, figsize=(10,6))
ax.plot((ann_history.history['loss']), 'r', label='train')
ax.plot((ann_history.history['val_loss']), 'b' ,label='val')
ax.set_xlabel(r'Epoch', fontsize=20)
ax.set_ylabel(r'Loss', fontsize=20)
ax.legend()
ax.tick_params(labelsize=20)
```

```
Ypredict_tst=model.predict(X_test_data,batch_size=10)
```

```
Ypredict_tst=pd.DataFrame(Ypredict_tst)
Ypredict_tst
fig, ax = plt.subplots(1, 1, figsize=(10,6))
ax.plot((ann_history.history['accuracy']), 'r', label='train')
ax.plot((ann_history.history['val_accuracy']), 'b' ,label='val')
ax.axis([-5, 350, 0.92, 1]) # [xmin, xmax, ymin, ymax]
ax.set_xlabel(r'Epoch', fontsize=20)
ax.set_ylabel(r'ACCURACY', fontsize=20)
ax.legend()
ax.tick_params(labelsize=25)
```