Machine learning for finite element analysis of holes under bi-axial load (MAE19047)

Nicholas Sung Wei Yong

Undergraduate, School of Mechanical and Aerospace engineering
Nanyang Technological University

Dr Chow Wai Tuck

School of Mechanical and Aerospace engineering
Nanyang Technological University

Abstract

Finite element analysis (FEA) is used to solve real life engineering problems because it can model complex geometry and capture local stress concentration effects. Given the developments in machine learning due to advancements in both computational hardware technology and software capabilities, it has produced reliable predictions based on patterns from existing data. This paper intends to analyse whether machine learning can increase the efficiency for the finite element analysis of rivet holes under biaxial load. This is done by training the neural network with the coarse mesh of a rivet hole under biaxial load against its analytical solution. The training is carried out using a small set of 125 data samples using 3D linear elements with variations in inner radius, outer radius, and the inner thickness. The neural networked will be trained based on three common backpropagation network algorithms. They are the Conjugate Gradient Descent, Levenberg-Marquardt algorithm, and Bayesian Regularization. In addition, tangent sigmoid and pure linear transfer functions will be evaluated. The trained neural networks are then applied to the rivet hole model and the target solution for the model is the maximum stress value for the fine mesh model. It is shown that the predictive errors are relatively low with errors generally below 5%. The Levenberg-Marquardt algorithm using Tangent Sigmoid transfer function was found to have the lowest errors. This indicates that supervised learning has predictive capabilities to aid in finite element analysis for stress concentrations.

Keywords Finite Element Analysis · Computational Structural Analysis · Supervised Learning · Computational Efficiency

1 INTRODUCTION

Rivets are permanent, one-piece fasteners that join parts together by fitting through a pre-drilled hole. They are the most widely used mechanical fasteners especially in aircraft fuselage structures. Aluminum and its alloys are difficult to solder. To make a good union and a strong joint, aluminum parts can be riveted together. Riveting provides both strength and neatness and is much easier to do than welding. It is the most common method used to fasten or join aluminum components in aircraft construction and repair. Under loading conditions, a countersunk rivet hole is often subjected to high stress concentration which may lead to crack initiation. This is the phenomenon of fatigue failure where metal parts fail due to repeated cyclic stress at lower values than which the part can normally withstand. Through cyclic loading, cracks will initially appear at the areas with high stress concentration and propagate until complete fracture of the part and fatigue failure occurs. [1] Hence, it is important that engineers calculate the stress concentration of their mechanical designs accurately.

To find the stress concentration Finite Element Analysis (FEA) is often used. FEA is a numerical technique used to sufficiently understand and quantify any physical phenomena. [2] To solve a problem, FEA subdivides a large system into smaller, simpler parts that are called finite elements. To attain more accurate results, more refined mesh models must be used where each finite element are smaller and more discretised. This however comes at the cost of greater computational power and longer processing time. To overcome this problem, sub-modelling is employed at area with high stress concentrations. [3] In comparison, a coarse mesh is used for the global model which is the entire design. Nonetheless, sub-modelling requires high amount of manual labour since each part usually consist of man delicate features. Besides the disadvantage of being highly laborious, 3D elements are often set to be linear due to computational limitations, leading to lower accuracy. Therefore, supervised machine learning through training the neural network is proposed to predict a more accurate solution for the maximum stress in a model and reduce the need for sub-modelling.

The objective of this project is to investigate if the neural network can use data parameters from a coarse mesh of a rivet radius to accurately predict the actual maximum stress value.

In this paper, three different backpropagation algorithms will be tested. They are the Scaled Conjugate Gradient (SCG), Levenberg-Marquardt algorithm (LM) and Bayesian Regularization (BR). The backpropagation algorithms will be used to train the neural network. In addition, pure linear and tangent sigmoid transfer functions will be used. A data set based on the coarse mesh of a rivet hole will be used to train the neural network. MATLAB's deep learning library, fitnet, will be used for the supervised learning process. The target output will be the analytical solution of the fine mesh hole under biaxial load.

2 LITERATURE REVIEW

2.1 Overview of Stress Concentration in a Rivet Radius

Riveted joints are areas of stress concentrations where crack initiation of an aircraft fuselage is tends to start. This is due to the rivet hole disrupting the uniform cross-sectional area of a surface resulting in stress concentration occurring when a load is applied. This can be explained using Figure 1 below.

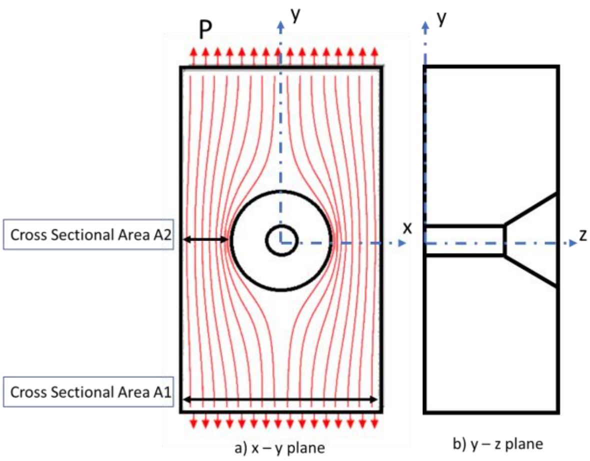

Figure 1: Stress Concentration in a Rivet Hole

In Figure 1, a rivet hole part is illustrated with tensile loads acting on both ends of the part. The stress, σ, is given by the load P divided by the cross-sectional area A. As such, since the cross-sectional area, A1 is bigger, hence a smaller value of stress σ_1 is experienced across part A1. In contrast, the section located about the rivet hole has a smaller cross-sectional area which results in a larger stress value σ_2. Due to the difference in the two stress values, the internal stress gets redistributed from the lower value to the higher one. The abrupt change in the cross-sectional area and the lack of space for the stress to redistribute itself evenly causes the stress at a particular point of the hole to exceed the stress of both σ_1 and σ_2.

2.2 Machine Learning using Neural Network

Machine learning is a form of Artificial Intelligence that provides systems the ability to learn from input data, identify patterns and provide results with minimal human intervention. [4] This is done by forming a trained neural network based on the input and output data which can predict different sets of data with the similar patterns. A neural network consists of connected neurons. A neuron is the basic unit of a neural network. It takes inputs, does some mathematical manipulations with them, and produces an output. The mathematical manipulations can include each input being multiplied by a weight, all the weighted inputs being added together with a bias and finally, the sum is passed through a transfer function which is used to turn an unbounded input into a predictable output.

Transfer functions refer to functions that calculates the weighted sum of inputs and biases within the neural network. This determines how significant the contribution from each neuron should be in computing the output. The purpose of the transfer function is to transform linear outputs into non-linear outputs so that the network can find patterns in the data. The transfer function is located just after the hidden layers, before the output layer, to transform the linear outputs from the hidden layer. [5] An illustration of a simple neural network is shown below in Figure 2.

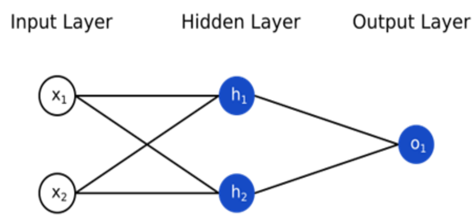

Figure 2: Simple Neural Network Architecture

As seen from Figure 2, the networks have two inputs (x_1 and x_2), a hidden layer with two neurons (h_1 and h_2) and an output layer of one neuron (o_1). The data that is entered into the input layer is multiplied by weights and added together with a bias in the hidden layer, summed together and passed through the transfer function to give the output neuron (o_1).

In this study, the training data first runs throughard through the neural network and the output data is then calculated. Using an iterative minimisation process based on different back propagation techniques, the weight and biases in the hidden layer is adjusted to minimise the difference between the predicted output and the calculated output. The backpropagation training algorithms are Conjugate Gradient Descent, Levenberg-Marquardt algorithm, and Bayesian Regularization and the transfer functions used will be tangent sigmoid and pure linear transfer functions. [6]

This study will make use of a shallow neural network that only has one hidden layer. This is to ensure that computational costs are low, and the training time is fast.

3 METHODOLOGY

3.1 Overview of Study

In this study, Ansys APDL is used to generate 3D models of rivet hole. The software uses FEA to solve and analyze problems involving structural mechanics and enables us to find the maximum stress in a structure. The first step was to create 125 variations of both coarse and fine models using APDL. Next, collect parameter data on the coarse finite width hole model as the training data set and the maximum stress on the fine mesh model as the target output.

The training coarse model is based on the section of the 3D rivet hole as shown in Figure 3 that is cut out and used as a training model.

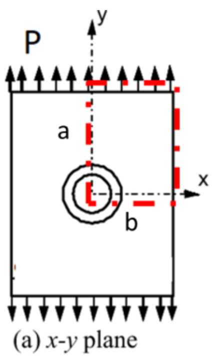

Figure 3: Section of Rivet Hole used as Training Model

The sides of the cutout, a and b are constrained in the direction of the tensile load and normal to the tensile load. Thereafter, input the training data set into the untrained neural network with the fine mesh data target output. Supervised learning will be used to “optimize” the coarse model. Lastly, the results obtained from the optimized coarse model will be compared to that of the fine model to ensure that it offers satisfactory accuracy.

3.1.1 Creating training data set

The creation of the training data sets begins with modelling a quarter hole on a 3D plate using APDL script functions. There are only two elements are the radius, this can be seen in Figure 4.

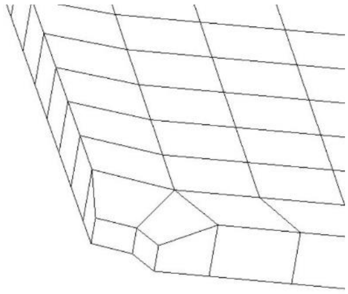

Figure 4: Coarse Mesh

For this study, the material property used are stated in Table 1.

Table 1: Material Property

Material	Aluminium
Modulus of Elasticity	10^7 psi
Poisson’s Ratio	0.3

Next, the element types are specified, and the model will be meshed. For the coarse meshed model in the training data set, linear elements like plane 182 and solid 185 were used. These elements have two degrees of freedom. The reason behind using linear elements is to reduce computational cost and processing time.

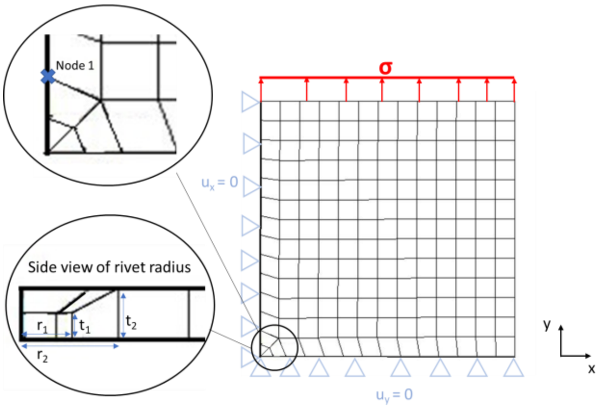

Figure 5: Top View of Coarse Mesh

With reference to Figure 5 and based on the countersunk rivet holes used in the industry, the inner rivet radius (r_1) is varied from 0.3 to 0.7 metres, the outer radius (r_2) is varied from 0.8 to 1.6 metres, inner rivet thickness (t_1) is varied from 0.5 to 0.9 metres, while the outer rivet thickness (t_2) is fixed at 1 metre. Based on this, 125 variations of different dimensions could be obtained. This variation is illustrated in Table 2 below.

Table 2: Variation of Normalised Parameter Values against outer thickness.

	Minimum	Maximum	Interval
Inner Radius (r_1)	0.3	0.7	0.1
Outer Radius (r_2)	0.8	1.6	0.2
Inner Thickness (t_1)	0.5	0.9	0.1

As shown in Figure 5, the mesh contains two elements along the rivet radius. There is a uniaxial tensile load of $\sigma = 1\,Pa$ acting at the top of the model and the boundary conditions of $u_x = 0$ is set on the left vertical boundary of the model while $u_y = 0$ is set on the bottom horizontal boundary of the model.

The four parameters used for training are shown in Figure 4. These parameters are all divided by outer rivet thickness (t_2) which is fixed at 1 metre, to create a unitless value. Parameter 1 is the inner rivet radius (r_1). Parameter 2 is radius (r_2) and Parameter 3 is inner rivet thickness (t_1). Parameter 4 is the y-displacement of node 1. In addition, Parameter 4 is scaled linearly by 0.7 and 1.5 to train the data to be more sensitive towards the change in y-displacement values. The sample of 125 variations are used to train the neural network.

3.1.2 Obtaining Target Data Set

For the fine mesh, instead of 2 elements are the radius, there will be 6 elements. The parameter in the target data set is the maximum stress value of the fine mesh model. This value is determined using the nodal solution provided from the highly refined mesh in ANSYS. One such fine mesh variation is shown below in Figure 6.

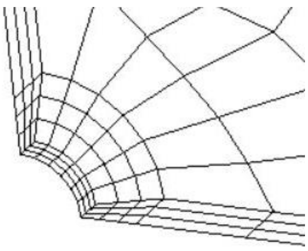

Figure 6: Fine Mesh

For more information on the scripting commands used to generate the coarse and fine mesh models, please refer to Appendix 1 and Appendix 2 respectively.

3.1.3 Training the Neural Network

After creating the training data set, the data can be input into a shallow neural network. Preprocessing of the data refers to the grouping of data and normalization of training variables. This will be done in MATLAB. In this study, MATLAB's library, fitnet, [7] is used for the neural network supervised training with 'trainscg' (Conjugate Gradient Descent), 'trainbr' (Bayesian Regularization) and 'trainlm' (Levenberg-Marquardt) for backpropagation.

When using the SCG and LM methods, 70% of the training sample size is used for training of neural network, 15% for validation of the neural network model and the remaining 15% for testing of the accuracy of the model. For the BR method, since it is a self-validating algorithm, 85% of the training sample size is used for training and 15% for testing.

The transfer function of the first layer will be either tangent sigmoid or pure linear, while the transfer function of the output layer will be set to pure linear. Other parameters such as training variables, number of neurons and weight initialization methods will be set before training the neural network. For the training, the target output will be the maximum stress for the fine mesh model. Finally, post processing of training will be done. Post processing includes obtaining the Root Mean Squared Errors (RMSE). Post processing is important because it validates that training is successful.

3.1.4 Determining Results

The accuracy of the result is determined from the root mean squared error (RMSE) between the neural network output versus the fine mesh solution for the maximum stress value in the model. The highest possible RMSE value is found to ensure that the worst-case scenario for the accuracy is found. To ensure this, each network is trained with 300 trials and the RMSE of the output of each of the 300 networks is determined. The highest value of the RMSE out of all these 300 trials is obtained as the output of the MATLAB code.

4 RESULTS

The accuracy of the results was determined from the root mean squared error (RMSE) between the neural network output versus the fine mesh solution for the maximum stress. Table 2 below shows the RMSE for the predictions of the test set in the training data. The training parameters used is described in section 3.1.1 and the target solution parameter used is the maximum stress as shown in section 3.1.2.

Table 2: Root Mean Squared Error (RMSE) of the test set with various algorithms and transfer functions

Algorithm	Maximum RMSE (%)* for test set	
	Pure Linear	Tangent Sigmoid
LM	4.6384	3.9073
BR	4.7849	4.5068
SCG	4.6293	5.2831

*Maximum RMSE based on 300 trials

From Table 2, the prediction errors for the test set for all the different algorithms and transfer functions are good, with most of the RMSEs being less than 5%.

5 DISCUSSION

From the prediction errors obtained, the most optimal neural networks are trained using the LM algorithm with the tangent sigmoid transfer function.

The scope of the project is limited in terms of it being self-validating and not using the trained neural network to test other variations of countersunk rivet holes and other types of rivet holes. In additions, this study only trains the neural network based on 3D coarse rivet hole. There are still many other types of holes such as blind holes which does not extend completely through an object, counterbored holes and other types of riveted holes. The stress concentration of these holes may differ from that of through holes. For future studies, the trained neural network can be tested against these different types of 3D holes. This way, supervised learning could be applied to more realistic and more general problems.

6 CONCLUSION

The aim of the project was to determine the effectiveness of supervised learning in determining the stress concentration in a rivet hole under a tensile load. The training data and the target data was obtained using Finite Element Analysis in ANSYS for a simple 3D rivet hole model. The networks were trained using three algorithms (Scaled Conjugate Gradient, Bayesian Regularization, and Levenberg-Marquardt) and two different transfer functions (pure linear and tangent sigmoid). 15% of the training and target data samples were set aside to test the effectiveness of the network predictions. It was found that the root mean squared error of this sample was low across the different algorithms and transfer functions. This shows using data parameters from a coarse mesh, we can accurately predict the actual maximum stress value.

ACKNOWLEDGMENT

I would like to express my sincere thanks and deepest gratitude to Dr Chow Wai Tuck for his guidance and assistance provided. His unwavering support and dedication have greatly benefitted me in my research. Dr Chow provided valuable advice and encouragement throughout this journey to improve on this project and me.

I would also like to thank my FYP senior Lau Jia Tai for their assistance in the use of ANSYS and MATLAB software.

Lastly, I would like to acknowledge the funding support from Nanyang Technological University – URECA Undergraduate Research Programme for the support provided in this research project.

REFERENCES

[1] Engineers Edge, L. (n.d.). Fatigue - Strength (Mechanics) of Materials. Available: https://www.engineersedge.com/material_science/fatigue_failure.htm [Accessed 21 June 2020].

[2] Harish, A. (2020, March 10). What Is FEM and FEA Explained: Finite Element Method. Available:

https://www.simscale.com/blog/2016/10/what-is-finite-element-method/ [Accessed 21 June 2020].

[3] E. Narvydas and N. Puodziuniene, “Applications of Sub-modeling in Structural Mechanics,” in *Mechanika*, Lithuania, 2014.

[4] E. (2020, May 29). What is Machine Learning? A definition. Available:

https://expertsystem.com/machine-learning-definition/ [Accessed 21 June 2020].

[5] Wiki, A. (2020). Weights and Biases. Available:

https://docs.paperspace.com/machine-learning/wiki/weights-and-biases [Accessed 21 June 2020].

[6] M. (2014). Sigmoid Transfer Function. Available: https://www.sciencedirect.com/topics/engineering/sigmoid-transfer-function [Accessed 21 June 2020].

[7] Mathworks, “Statistics and Machine Learning Toolbox,” [Online]. Available: https://www.mathworks.com/help/stats/. [Accessed 21 June 2020].

APPENDIX

Appendix 1

Appendix 1a – ANSYS main macro for training data set

```
/cle
/prep7
*dim,nd,,2                              ! create array to inout nd as variable
*cfopen,radius.txt                      !opens a 'command' file called radius.txt
*vwrite,'par1','par2','par3','u1'       !write data to the file in a formatted sequence

(1a12,6a16)                             !formatting for txt file
fmat                                    !run macro 'fmat'


ez = 0 !set value of z-strain
*do,iradius,0.3,0.7,0.1                 !iterating to get variations
    *do,jradius,0.8,1.6,0.2
        *do,ithickness,0.5,0.9,0.1
            functionmesh                !run macro 'functionmesh'
!write values to txt file
*vwrite,par1,par2,par3,uy(nd(1))

(1f12.4,6g16.8,' ')                     !formatting for txt file
eplot
        *enddo
    *enddo
*enddo

*cfclose                                !close the file radius.txt

/SHOW,CLOSE
/DEVICE,VECTOR,0
```

Appendix 1b – ‘functionmesh’ macro referenced in main macro

```
/prep7
vclear,all
vdel,all,,,1
! creates a rectangular
BLC4, , ,10,10
! create a circle
CYL4, , ,jradius
! subtracts the circle from the rectangular
ASBA,       1,       2
! picks something
FLST,2,1,5,ORDE,1
FITEM,2,3
! EXTRUDES VOLUME
VEXT,3, , ,0,0,1,,,,
! defines a keypoint
k,50,iradius
k,51,iradius,,ithickness
defines an area by connecting the keypoint
a,50,5,7,51
! defines a key point
k,100
k,101,,,1
! defines line between two keypoints
1,100,101
! rotate about the axis
VROTAT,8, , , , , ,100,101,90,
! merges
nummrg,all
! element type
ET,1,SOLID185

! elements size
esize,.5
! Fills an existing unmeshed volume with elements by sweeping the mesh from an adjacent area through the volume.
vsweep,2
esize,0.7
vsweep,1
! slect
asel,s,loc,x,0
! deifne degree of freedom
DA,all,SYMM
asel,s,loc,y,0
DA,all,SYMM
asel,s,loc,z,0
DA,all,SYMM
asel,s,loc,y,10
SFA,all,1,PRES,-1
allsel,all
allsel


!Assign three keypoints near the max stress point to node array
nd(1) = node(0,iradius,ithickness)
! Creates the parameters
par1 = jradius
par2 = iradius
par3 = ithickness

/solu
solve
/post1
set,last
plns,s,y
*get,maxStr3,plnsol,0,max
```

Appendix 2

Appendix 2a - ANSYS main macro for target data set

```
/cle
/prep7

*cfopen,radius.txt                      !opens a 'command' file called radius.txt
*vwrite,'target'                        !write data to the file in a formatted sequence
(A6)

fmat                                    !run macro 'fmat'


ez = 0 !set value of z-strain
*do,iradius,0.3,0.7,0.1                 !iterating to get variations
    *do,jradius,0.8,1.6,0.2
        *do,ithickness,0.5,0.9,0.1
            functionmesh

*vwrite,maxStr3                         !write values to txt file
(F16.14)
eplot
        *enddo
    *enddo
*enddo

*cfclose                                !close the file radius.txt

/SHOW,CLOSE
/DEVICE,VECTOR,0
```

Appendix 2b - ‘functionmesh’ macro referenced in main macro

```
/prep7
vclear,all
vdel,all,,,1

! creates a rectangular
BLC4, , ,10,10
! create a circle
CYL4, , ,jradius
! subtracts the circle from the rectangular
ASBA,       1,       2
! picks something
FLST,2,1,5,ORDE,1
FITEM,2,3
! EXTRUDES VOLUME
VEXT,3, , ,0,0,1,,,,
! defines a keypoint
k,50,iradius
k,51,iradius,,ithickness
defines an area by connecting the keypoint
a,50,5,7,51
! defines a key point
k,100
k,101,,,1
! defines line between two keypoints
1,100,101
! rotate about the axis
VROTAT,8, , , , , ,100,101,90,
! merges
nummrg,all

! element type
ET,1,SOLID185
! elements size
esize,.5
! Fills an existing unmeshed volume with elements by sweeping the mesh from an adjacent area through the volume.
vsweep,2
esize,0.7
vsweep,1
! slect
asel,s,loc,x,0
! deifne degree of freedom
DA,all,SYMM
asel,s,loc,y,0
DA,all,SYMM
asel,s,loc,z,0
DA,all,SYMM
asel,s,loc,y,10
SFA,all,1,PRES,-1
allsel,all
allsel


!solve for solution
/solu
solv
/post1
plns,s,1
!plnsol,u,y
*get,maxStr3,plnsol,0,max
```