ESTIMATION OF STRESS INTENSITY FACTORS FOR CRACKED FINITE PLATES USING A HYBRID MODEL OF ARTIFICIAL NEURAL NETWORK AND FINITE ELEMENT METHOD (ANSYS)

YAPAY SİNİR AĞLARI VE SONLU ELEMANLAR METODUNDAN (ANSYS) OLUŞAN KARMA BİR MODELİ KULLANARAK ÇATLAK İÇEREN SONLU PLAKALARIN GERİLME YOĞUNLUK FAKTÖRLERİNİN TAHMİNİ

YUSUF YABİR

PROF. DR. BORA YILDIRIM

Supervisor

Submitted to

Graduate School of Science and Engineering of Hacettepe University

as a Partial Fulfillment to the Requirments

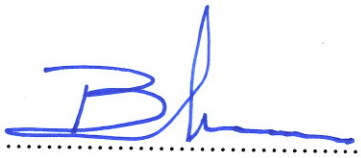
for the Award of the Degree of Master of Science

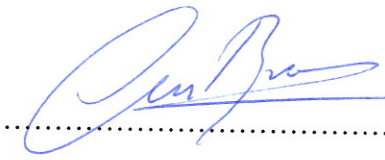
in Mechanical Engineering

2019

This work titled " Estimation of Stress Intensity Factors for Cracked Finite Plates Using a Hybrid Model of Artificial Neural Network and Finite Element Method (ANSYS) " by YUSUF YABİR has been approved as a thesis for the Degree of Master of Science in Mechanical Engineering by the Examining Committee Members mentioned below.

Assoc. Prof. Dr. Barış SABUNCUOĞLU
Head .....

Prof. Dr. Bora YILDIRIM
Supervisor ..... 

Assoc. Prof. Dr. Can Ulaş DOĞRUER
Member ..... 

Assist. Prof. Dr. Hasan Basri ULAŞ
Member .....

Assist. Prof. Dr. Okan GÖRTAN
Member .....

This thesis has been approved as a thesis for the Degree of **Master of Science** in **Mechanical Engineering** by Board of Directors of the Institute of Graduate School of Science and Engineering on …/…/2019

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU
Director of the Institute of
Graduate School of Science and Engineering

ETHICS

In this thesis study, prepared in accordance with the spelling rules of Institute of Graduate School of Science and Engineering of Hacettepe University.

I declared that

- all the information and documents have been obtained in the base of the academic rules
- all audio-visual and written information and results have been presented according to the rules of scientific ethics.
- in case of using other works, related studies have been cited in accordance with the scientific standarts
- all cited studies have been fully referenced
- I did not do any distortion in the data set
- and any part of this thesis has not been presented as another thesis study at this or any other university.

27/05/2019

YUSUF YABİR

YAYIMLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamı veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarda kullanıma açma iznini Hacettepe Üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetki sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan ***"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"*** kapsamında tezim aşağıda belirtilen koşullar haricinde YÖK Ulusal Tez Merkezi / H. Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- ☐ Enstitü / Fakülte yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir.

- ☐ Enstitü / Fakülte yönetim kurulu gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren .... ay ertelenmiştir.

- ☐ Tezim ile ilgili gizlilik kararı verilmiştir.

27/05/2019

YUSUF YABİR

ÖZET

YAPAY SİNİR AĞLARI VE SONLU ELEMANLAR METODUNDAN (ANSYS) OLUŞAN KARMA BİR MODELİ KULLANARAK ÇATLAK İÇEREN SONLU PLAKALARIN GERİLME YOĞUNLUK FAKTÖRLERİNİN TAHMİNİ

Yusuf YABİR

Yüksek Lisans, Makina Mühendisliği Bölümü
Tez Danışmanı: Prof. Dr. Bora YILDIRIM
Mayıs 2019, 94 sayfa

Bu çalışmanın amacı gerilme yoğunluk faktörü formülü bulunmayan yarı eliptik yüzey çatlağı içeren 2 farklı tipteki plakanın gerilme yoğunluk faktörlerini tahmin etmek için yapay sinir ağı modeli geliştirmekti. Bu plakalardan biri ön ve arka yüzünde birer adet yarı eliptik yüzey çatlağı içermektedir. Diğerinin ise ön ve arka yüzünde birbirine paralel 2 adet yarı eliptik yüzey çatlağı bulunmaktadır. İlk olarak sinir ağı modelinin eğitilmesi için gerekli olan veriler sonlu elemanlar metodu (Ansys) yardımıyla oluşturulmuştur. Birinci durum için (her iki yüzeyde de 1 tane yarı eliptik yüzey çatlağı içeren) minör yarıçapın (a), minör yarıçapın majör yarıçapa oranının (a/c) ve minör yarıçapının plaka kalınlığına oranının (a/t) farklı değerleri kullanılarak toplamda 179 Ansys simülasyonu yapılmıştır. Birinci durum için bu simülasyonlar sonucunda 8234 adet veri üretilmiş (her simülasyon için 46 farklı parametrik açı değeri) ve bu verilerin 1061 tanesi yapay sinir ağı modelinin eğitilmesinde kullanılmıştır. Bunun yanı sıra ikinci durum için (her iki yüzeyde birbirine paralel 2 adet yarı eliptik yüzey çatlağı içeren) a, a/c, a/t ve h'ın (iki paralel çatlak arasındaki dikey uzaklık) farklı değerleri kullanılarak 523 Ansys simülasyonu yapılmıştır. Bu simülasyonlardan sonra 24058 tane veri elde edilmiş ve 4248

tanesinden ağın eğitim aşamasında yararlanılmıştır. Yapay sinir ağı modellemesinde Matlab sinir ağı modülü (nntool) kullanılmıştır. Eğitim aşamasında farklı tipte ağ yapıları kullanılmıştır. Eğitilen sinir ağlarının doğruluğu, en iyi ağ modelini belirleyebilmek için Ansys'te üretilen 760 (birinci durum) ve 1139 (ikinci durum) tane yeni veri kullanılarak test edilmiştir. Sonuç olarak birinci durum için minimum sapma değeri 2 tane gizli katman ve her bir gizli katmanda 15 nöron içeren ağ modelinde elde edilmiştir ve minimum sapma değeri % 0.32 olarak hesaplanmıştır. Benzer şekilde ikinci durum için minimum sapma değeri % 0.49 olarak hesaplanmış ve bu model 3 gizli katman ve her bir gizli katman için 14 nöron içermektedir. Bu çalışma sonucunda herhangi bir simülasyon / analiz yapmadan 2 farklı durum için gerilme yoğunluk faktörü değerlerini tahmin edebilme potansiyeline sahip 2 tane yapay sinir ağı modeli elde edilmiştir.

Anahtar Kelimeler: Yapay Sinir Ağları, Sonlu Elemanlar Metodu, Kırılma, Yarı Eliptik Yüzey Çatlağı, Gerilme Yoğunluk Faktörü

ABSTRACT

ESTIMATION OF STRESS INTENSITY FACTORS FOR CRACKED FINITE PLATES USING A HYBRID MODEL OF ARTIFICIAL NEURAL NETWORK AND FINITE ELEMENT METHOD (ANSYS)

Yusuf YABİR

Master of Science, Department of Mechanical Engineering
Supervisor: Prof. Dr. Bora YILDIRIM
May 2019, 94 pages

The aim of this study was to develop an artificial neural network model in order to estimate stress intensity factor values of two different types of semi elliptical surface cracked plates which have no explicit stress intensity factor formula. One of these plates contains one semi elliptical surface crack at both sides, front side and back side. The other one contains two parallel semi elliptical surface cracks at the front and back side. First of all, data which were needed for neural network model training were generated in aid of finite element method (Ansys). In the first case (one semi elliptical surface crack at both sides), 179 Ansys simulations were done in total using different values of minor radius (a), the ratio of minor radius to major radius (a/c) and the ratio of minor radius to plate thickness (a/t). As a result of these simulations in case 1, 8234 data were generated (46 different parametric angles for each simulation) and 1061 of these data were used to train the artificial neural network model. Besides, 523 Ansys simulations were done using different values of a, a/c, a/t and h (vertical distance between two parallel cracks) for the second case (two parallel semi elliptical surface cracks at both sides). 24058 data were obtained after these simulations and 4248 of them were utilized for the network training

process. Matlab neural network module (nntool) was used for artificial neural network modelling. Different types of network structures were used in the training process. The accuracy of the trained neural networks for the first and second case were tested using 760 and 1139 new data respectively, which were generated via Ansys so as to determine the best network model. Consequently, minimum deviation value (difference between Ansys and Matlab neural network result) of case 1 was obtained for the network model that has 2 hidden layers and 15 neurons for each hidden layer and minimum deviation was calculated as 0.32%. Similarly, minimum deviation value of the model for the second case was calculated as 0.49% and this model has 3 hidden layers and 14 neurons for each hidden layer. As a result of this study, for two different types of cases, two artificial neural network models which have the potential to estimate stress intensity factor values without doing any simulations, were obtained.

Keywords: Artificial Neural Network, Finite Element Method, Fracture, Semi Elliptical Surface Crack, Stress Intensity Factor

ACKNOWLEDGEMENTS

Firstly, I would like to express my profound thankfulness to my respectable supervisor, Prof. Dr. Bora Yıldırım, for his encouragement, guidance and supervision. I completed this study with the help of him.

Most of all, I would like to give thanks to my wife for her moral support, patience and endless love.

Yusuf YABİR
May 2019, Ankara

TABLE OF CONTENTS

ÖZET……..i
ABSTRACT…….iii
ACKNOWLEDGEMENTS……..... v
TABLE OF CONTENTS…….....vi
LIST OF TABLES…….. vii
LIST OF FIGURES……..ix
LIST OF SYMBOLS AND ABBREVIATIONS……...…xiii
1. INTRODUCTION……1
1.1. Aim and Scope of the Study……2
1.2. Literature Survey……4
2. THEORY, MODELLING AND METHODS……...7
2.1. Theory……..7
2.1.1. Fracture Theories and Stress Intensity Factor Formulation……...7
2.1.2. Finite Element Method……12
2.1.3. Artificial Neural Network……15
2.2. Modelling and Methods…….23
2.2.1. Finite Element Model Establishment……..24
2.2.2. Mesh Convergence Study……27
2.2.3. Verification of the Finite Element Model with Formula…….33
2.2.4. Artificial Neural Network Model Development……..36
2.2.5. Data Generation with Finite Element Analysis (Ansys)…….43
2.2.6. Training of Artificial Neural Network Model…….50
2.2.7. Testing of Artificial Neural Network Model……...53
3. ANALYSIS AND RESULTS…….55
3.1. Analysis and Results for Case 1……56
3.2. Analysis and Results for Case 2……69
3.3. Conclusions and Recommendations……..……87
REFERENCES……90
APPENDIX…….93

LIST OF TABLES

Table 2.1. Main parts of biological human neuron and artificial neural network…...17

Table 2.2. Convergence study for main mesh size……..29

Table 2.3. Ansys result for main mesh size 3, circumferential division 64, crack front division 45 and mesh contour 10……29

Table 2.4. Convergence study for number of mesh contours……..30

Table 2.5. Convergence study for number of circumferential division……...30

Table 2.6. Stress intensity factor calculation result for circumferential division 96, main mesh size 3, crack front division 45 and mesh contour 10……31

Table 2.7. Verification of finite element model with Newman Raju Equation……...34

Table 2.8. An example of mean error calculation for verification process…….34

Table 2.9. Verification of the Ansys model with Newman Raju equation (h=20 cm, a/t= 0.05)……...36

Table 2.10. Values of input variables used in Ansys for case 1 (179 simulations)…...43

Table 2.11. Values of input variables used in Ansys for case 2 (523 simulations)……44

Table 2.12. A sample study for determination of plate dimension……49

Table 2.13. Simulations done for testing the trained model (case 1)…….……53

Table 2.14. Simulations done for testing the trained model (case 2)……..…….54

Table 3.1. Deviation values of 760 test data for different types of network structures (case 1)……...57

Table 3.2. Minimum and maximum values needed for normalization (case 1)……..61

Table 3.3. Weights of the trained neural network for case 1 – between input neurons (4) and first hidden layer neurons (15)……...61

Table 3.4. Weights of the trained neural network for case 1 – between five of first hidden layer neurons (1-5) and second hidden layer neurons (15)……62

Table 3.5. Weights of the trained neural network for case 1 – between five of first hidden layer neurons (6-10) and second hidden layer neurons (15)……..62

Table 3.6. Weights of the trained neural network for case 1 – between five of first hidden layer neurons (11-15) and second hidden layer neurons (15)……63

Table 3.7. Weights of the trained neural network for case 1 – between second hidden layer neurons (15) and output neuron (1)……...63

Table 3.8. Bias weights of the trained neural network for case 1……64

Table 3.9. Deviation values of 1139 test data for different types of ANN in case 2…71
Table 3.10. Minimum and maximum values needed for normalization (case 2)……..74
Table 3.11. Weights of the trained neural network for case 2 – between input neurons (5) and first hidden layer neurons (14)……...74
Table 3.12. Weights of the trained neural network for case 2 – between five of first hidden layer neurons (1-5) and second hidden layer neurons (14)……75
Table 3.13. Weights of the trained neural network for case 2 – between five of first hidden layer neurons (6-10) and second hidden layer neurons (14)……..75
Table 3.14. Weights of the trained neural network for case 2 – between four of first hidden layer neurons (11-14) and second hidden layer neurons (14)……76
Table 3.15. Weights of the trained neural network for case 2 – between five of second hidden layer neurons (1-5) and third hidden layer neurons (14)……76
Table 3.16. Weights of the trained neural network for case 2 – between five of second hidden layer neurons (6-10) and third hidden layer neurons (14)……..77
Table 3.17. Weights of the trained neural network for case 2 – between five of second hidden layer neurons (11-14) and third hidden layer neurons (14)……77
Table 3.18. Weights of the trained neural network for case 2 – between third hidden layer neurons (14) and output neuron (1)……...78
Table 3.19. Bias weights of the trained neural network for case 2……78

LIST OF FIGURES

Figure 1.1. First case - two semi elliptical surface cracked plate - (a) front and back sides and (c) right hand side of the plate……2

Figure 1.2. Second case - four semi elliptical surface cracked plate – (a) front and back sides and (b) right hand side of the plate……3

Figure 2.1. Schematic representation of Griffith theory [12]……...7

Figure 2.2. (a) and (b) Schematic representation of stress state around the crack tip……...8

Figure 2.3. Fracture modes – opening, shearing and tearing mode respectively [13]…….....9

Figure 2.4. Variation of geometry correction factor for edge and center crack [15]…10

Figure 2.5. Mode I stress intensity factor calculation of a plate with center crack [16]……...10

Figure 2.6. Mode I stress intensity factor calculation of a plate with double crack [16]……...11

Figure 2.7. Mode I stress intensity factor calculation of a plate with single edge crack [16]……...11

Figure 2.8. Mode I stress intensity factor calculation of single semi elliptical surface cracked body [17]……11

Figure 2.9. Representation of minor, major radius and parametric angle in semi elliptical surface crack [18]……..12

Figure 2.10. A simple finite element model [19]……13

Figure 2.11. Types of elements used in finite element analysis……...13

Figure 2.12. Finite element method flowchart [20]……..14

Figure 2.13. Schematic representation of a biological human neuron [21]…….....15

Figure 2.14. A simple artificial neural network model……16

Figure 2.15. Summation and activation function of an artificial neural network model [22]……...17

Figure 2.16. A multi layered artificial neural network (input, hidden and output layer) [23]……...18

Figure 2.17. Effect of the input weights to the corresponding output of the input…...19

Figure 2.18. Effect of bias to the output of the network……...19

Figure 2.19. Some of the activation / transfer functions (a) hard limit (b) linear (c) logistic sigmoid (d) tangent hyperbolic……...20
Figure 2.20. Schematic representation of gradient descent backpropagation function……22
Figure 2.21. Front and back side of the semi elliptical surface cracked bodies considered in the thesis (a) case 1 (b) case 2……..24
Figure 2.22. Fracture module in Ansys and an example for the plate used for analyzes……25
Figure 2.23. Creating semi elliptical crack using fracture module……...25
Figure 2.24. Symmetry conditions for case 1 (a) and case 2 (b)……..26
Figure 2.25. Details of a semi elliptical crack in fracture module in Ansys……27
Figure 2.26. Schematic representation of a semi elliptical crack [25]…….27
Figure 2.27. Graphical representation of the convergence study for mesh contour….30
Figure 2.28. Graphical representation of convergence study for circumferential division…….31
Figure 2.29. Representation of the crack used in Ansys (general view)……..…32
Figure 2.30. Detailed view (a) and meshed model of the semi elliptical crack used in Ansys……33
Figure 2.31. Basic model for a multi layered artificial neural network……37
Figure 2.32. ANN model for the first problem at the first step of modelling……38
Figure 2.33. ANN model for the second case at the first step of modelling…….38
Figure 2.34. Feed forward back propagation neural network……...39
Figure 2.35. Levenberg – Marquardt training function flowchart [28]……40
Figure 2.36. Difference between good fit (a) and overfitted (b) ANN model [33]…...51
Figure 2.37. An underfitted model [33]……51
Figure 2.38. Variation of mean errors for training and test data [34]……...51
Figure 2.39. Graph of the tangent hyperbolic function……52
Figure 3.1. Matlab nntool module……..55
Figure 3.2. Training parameters used in the training process…….56
Figure 3.3. Variation of stress intensity factor for different values of a/t (a/c=0.5)...56
Figure 3.4. Variation of stress intensity factor for different values of a/t (a/c=1)…..57
Figure 3.5. Schematic representation of the trained model for case 1……58
Figure 3.6. Output screen at the end of training process for case 1 (4 input neurons, 2 hidden layers, 15 neurons for each hidden layers and 1 output neuron)..58

Figure 3.7. Performance graph of the training process (case 1)…….59
Figure 3.8. Change in the value of gradient, damping factor and number of validation check during training (case 1)……..59
Figure 3.9. Correlation coefficient for training, validation and test data (case 1)…..60
Figure 3.10. Variation of SIF along the crack front for case 1 (a=0.0442 m, a/c=1.3, a/t=0.33, deviation=1.32%)…..……...64
Figure 3.11. Variation of SIF along the crack front for case 1 - (a) Ansys result and (b) Comparative graph (a=0.0425 m, a/c=2.5, a/t=0.41, deviation= 0.49%)……..65
Figure 3.12. Variation of SIF along the crack front for case 1 - (a) Ansys result and (b) Comparative graph (a=0.0088 m, a/c=1.1, a/t=0.15, deviation= 0.08%)…….....66
Figure 3.13. Variation of SIF along the crack front for case 1 – (a) Ansys screen and (b) Comparative graph (a=0.003125 m, a/c=0.25, a/t=0.44, deviation= 0.38%)…….……67
Figure 3.14. Variation of SIF along the crack front for case 1 (a=0.012 m, a/c=0.4, a/t=0.47, deviation=0.65%)……..……...68
Figure 3.15. Variation of SIF along the crack front for case 1 (a=0.05 m, a/c=2, a/t=0.25, deviation=0.2%)……..…….68
Figure 3.16. Variation of SIF along the crack front for case 1 – (a) Ansys result and (b) Comparative graph (a=0.0189 m, a/c= 0.7, a/t= 0.38, deviation= 0.13%)……..69
Figure 3.17. Variation of stress intensity factor for different values of a/h (a/c=0.3, a/t=0.2)…….70
Figure 3.18. Variation of stress intensity factor for different values of a/h (a/c= 1, a/t= 0.4)……...70
Figure 3.19. Schematic representation of the trained model for case 2……71
Figure 3.20. Output screen at the end of training process for case 2 (5 input neurons, 3 hidden layers, 14 neurons for each hidden layers and 1 output neuron)..72
Figure 3.21. Performance graph of the training process (case 2)…….72
Figure 3.22. Change in the value of gradient, damping factor and number of validation check during training (case 2)……..73
Figure 3.23. Correlation coefficient for training, validation and test data (case 2)…..73

Figure 3.24. Variation of SIF along the crack front for case 2 (a=0.0168 m, a/c=0.8, h=0.05 m / a/h=0.336, a/t=0.27, deviation=0.21%)……..……..79
Figure 3.25. Variation of SIF along the crack front for case 2 (a=0.005 m, a/c=1, h= 0.003 m / a/h=1.67, a/t=0.35, deviation =0.19%)…….……...79
Figure 3.26. Variation of SIF along the crack front for case 2 – (a) Ansys result and (b) Comparative graph (a=0.01125 m, a/c=2.25, h=0.015 m / a/h=0.75, a/t=0.25, deviation =0.43%)……..……...80
Figure 3.27. Variation of SIF along the crack front for case 2 – (a) Ansys result and (b) Comparative graph (a=0.003 m, a/c=0.3, h=0.015 m / a/h=0.2, a/t=0.25, deviation =0.15%)……..……...81
Figure 3.28. Variation of SIF along the crack front for case 2 – (a) Ansys result and (b) Comparative graph (a=0.0081 m, a/c=0.9, h=0.013 m / a/h=0.623, a/t=0.33, deviation =0.39%)……..……...82
Figure 3.29. Variation of SIF along the crack front for case 2 – (a) Ansys result and (b) Comparative result (a=0.0054 m, a/c=1.8, h=0.025 m / a/h=0.216, a/t=0.3, deviation =1.25%)……..…….83
Figure 3.30. Variation of SIF along the crack front for case 2 (a=0.014 m, a/c=0.5, h=0.05 m / a/h=0.28, a/t=0.32, deviation =0.93%)…..……84
Figure 3.31. Variation of SIF along the crack front for case 2 (a=0.024 m, a/c=1.2, h=0.15 m / a/h=0.16, a/t=0.36, deviation =0.46%)……..……84
Figure 3.32. Variation of SIF along the crack front for case 2 (a) Ansys result (b) Comparative graph (a=0.0036 m, a/c=1.2, h=0.02 m / a/h=0.18, a/t=0.2, deviation =0.85%)……..……..85
Figure 3.33. Variation of SIF along the crack front for case 2 (a=0.008 m, a/c=0.5, h=0.005 m / a/h=1.6, a/t=0.45, deviation =0.17%)…..……86
Figure 3.34. Variation of SIF along the crack front for case 2 (a=0.01 m, a/c=2, h=0.008 m a/h=1.25, a/t=0.35, deviation =0.28%)……..……86
Figure 3.35. Variation of SIF along the crack front for case 2 (a=0.006 m, a/c=1.5, h=0.01 m / a/h=0.6, a/t=0.42, deviation =0.5%)……..……87

LIST OF SYMBOLS AND ABBREVIATIONS

SIF	Stress Intensity Factor
ANN	Artificial Neural Network
FEA	Finite Element Analysis
E	Elastic modulus
a	Minor radius of the semi elliptical crack
c	Major radius of the semi elliptical crack
t	Thickness of the cracked plate / body
Θ	Parametric angle
h	Distance between two parallel cracks
b	Width of the plate
γ	Specific surface energy
G_c	Strain energy release rate
σ	Applied stress
K_I	Stress intensity factor in Mode 1
K_{II}	Stress intensity factor in Mode 2
K_{III}	Stress intensity factor in Mode 3
f	Geometry correction factor
K_{Ic}	Critical stress intensity factor in Mode 1
K_{IIc}	Critical stress intensity factor in Mode 2
K_{IIIc}	Critical stress intensity factor in Mode 3
w	Weight in the artificial neural network model
δ	Difference between target and output in ANN
η	Learning rate
J	Jacobian matrix
μ	Damping factor

1. INTRODUCTION

In mechanical engineering, stress distribution over the component is very critical for the design of any mechanical part. So stress analysis is done in order to determine critical points of the component and specify the shape and material of the mechanical part.

Fracture mechanics is one of the branches of solid mechanics and it deals with the behaviour of cracks which are in the materials [1]. In this field, there are some theories which can be used to identify the behaviour of the material with discontinuity [2]. Discontinuity is a very important phenomenon in engineering design of materials because all of the materials / machine elements have discontinuities such as notches, flaws, cracks, etc. and these discontinuities effect the strength of the materials and cause crack propagation.

Crack propagation in the structures may lead to failure. In fracture mechanics field, a concept which is called stress intensity factor (SIF) is used to determine the stress intensity near the crack tip (magnitude of the singularity at the crack) and to guess a crack starts to grow or not. Therefore the calculation of SIF is very important for anti breaking design of materials and it has a significant role in determining the crack propagation.

Crack propagation can be predicted by comparing the critical SIF value and calculated SIF. If calculated SIF value is greater than the critical value, propagation starts and fracture occurs. The value of SIF can be calculated using some formulas (analytical method), numerical methods, finite element method (FEM) etc. Some formulas can be used for determination of SIF (for known cracked bodies) but there is no explicit formula for complicated bodies in general. For these situations, FEM can be easily used and very accurate results can be obtained. But if it is needed to perform so many simulations, SIF calculation with FEM procedure may be time consuming. Therefore, for these cases, an artificial neural network (ANN) model may be utilized to generate an explicit formula or a network, which calculates SIF values for different cases accurately and fast.

ANN is a very powerful computational tool and it is used to estimate the result of any problem or case utilizing some input values and relations / weights. Actually, ANN is a simulation of the human brain. It is very common all over the world in recent years. This tool can be used for pattern recognition / classification or function approximation

problems and very accurate results can be obtained. It analyzes the input and output data given to the network like a human brain, then forms a network which consists of a lot of weights, adjusts these weights and finally trained neural network is utilized to estimate / calculate the values of outputs. So ANN is easier and faster than FEM approach.

1.1. Aim and Scope of the Study

In fracture mechanics, there are some formulas in order to calculate the SIF. These formulas can be obtained from some handbooks, but they are only valid for some common bodies, so, for complicated cases, there is no explicit formula. Therefore, it is needed a new approach to calculate these SIF values.

In this thesis, the aim is to form an ANN structure for two different bodies / plates so as to estimate SIF values precisely. There is no formula for these two different cases / plates in literature. So, the formed ANN structure can be directly used as a formula in aid of a simple code or Matlab neural network module. In the first case, the plate has two semi elliptical surface cracks, one of them is at the front side and the other one is at the back side of the plate and these cracks are symmetric to each other as shown in Figure 1.1.

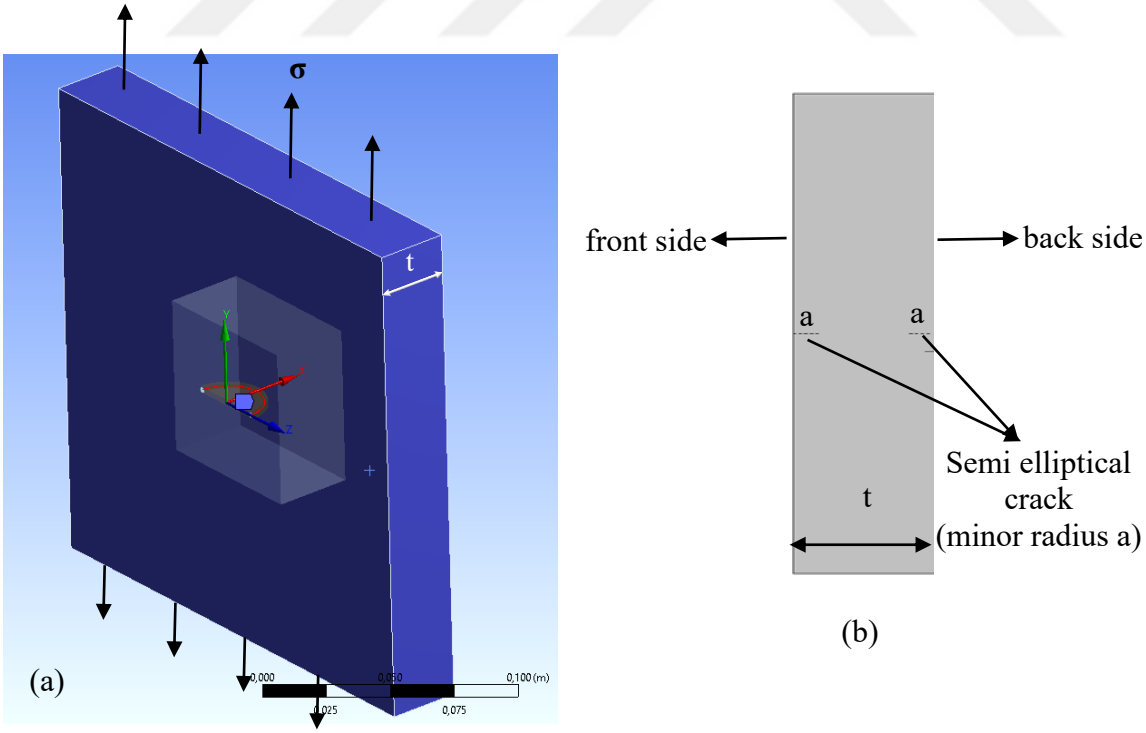


Figure 1.1. First case - two semi elliptical surface cracked plate - (a) front and back sides and (c) right hand side of the plate

In the second case, the plate has four semi elliptical surface cracks, two of them are at the front side (parallel to each other – vertical distance between the cracks is h), the other ones are at the back side (parallel to each other - vertical distance between the cracks is h) and again these cracks are symmetric to each other as shown in Figure 1.2.

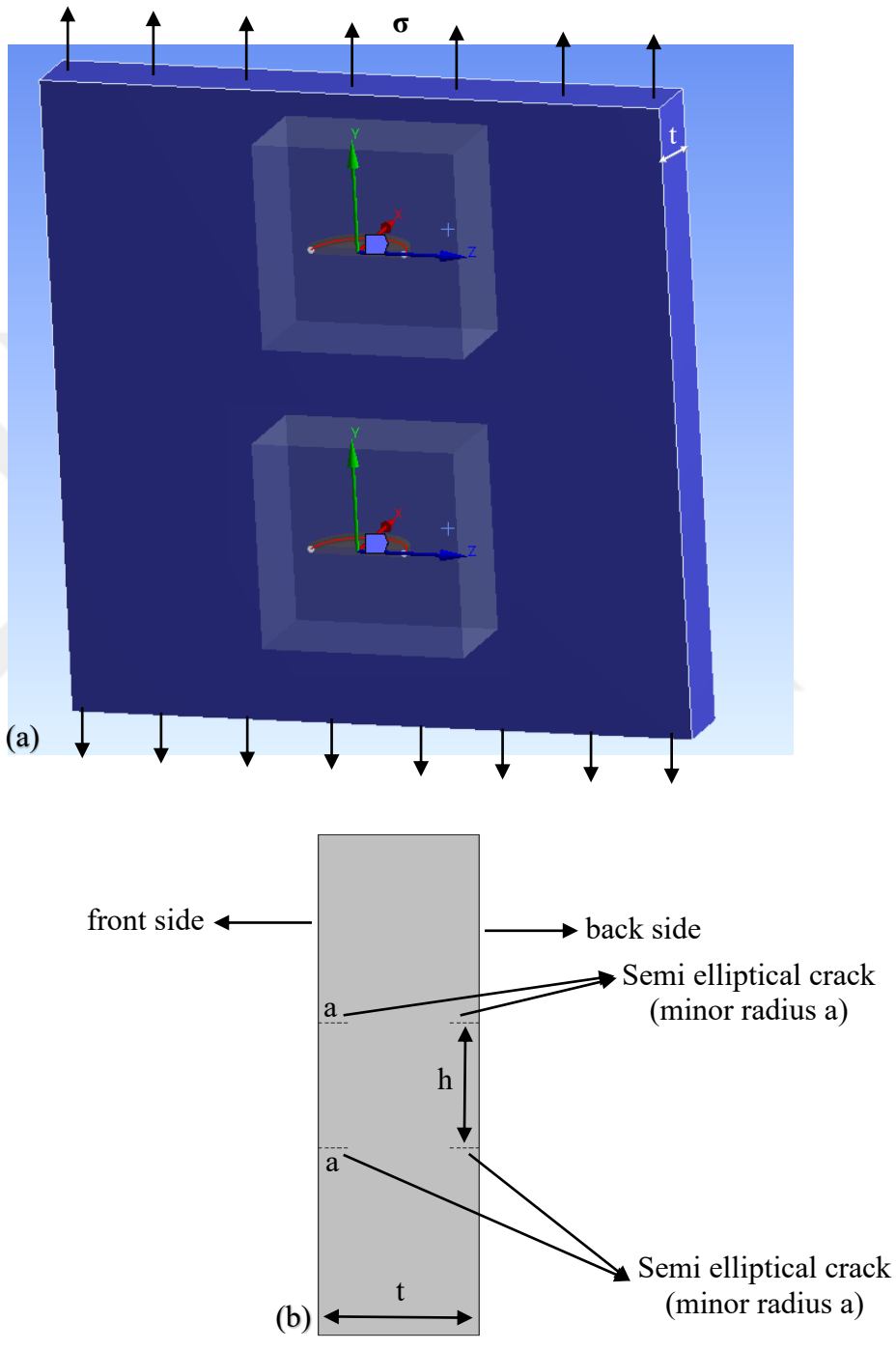


Figure 1.2. Second case - four semi elliptical surface cracked plate – (a) front and back sides and (b) right hand side of the plate

In this study, firstly, finite element analysis (FEA) software Ansys will be used to calculate SIF of various cases. As it was stated before, there are two different plates in this thesis. In the first case (only one semi elliptical surface crack in both sides of the plate), variables are minor and major radius of the crack and plate thickness. In the second case (two semi elliptical surface cracks in both sides), variables are minor and major radius of the crack, plate thickness and distance between two parallel cracks. Using different values of these variables, hundreds of simulations will be done in Ansys. Then, using these data, ANN will be trained. Training is adjusting the network weights so as to get the best output results. Matlab neural network tool (nntool) will be utilized for neural network analysis. After successfully training, the network will be tested again with new data and results obtained with Matlab nntool will be compared with Ansys outputs. After obtaining deviation good enough for the test data, this trained neural network structure (weights, number of layers, etc.) will be presented in order to estimate SIF of any other cases without using FEM. Thus, developed ANN will be used as an explicit formula. By means of ANN, it will take less time to calculate SIF and there will be no need to do any simulations in Ansys.

1.2. Literature Survey

In literature, there are some studies which use the finite element method (Ansys, Abaqus, LS Dyna, etc.) and artificial neural network as a hybrid model to compute/predict some outputs like stress intensity factor, maximum stress, maximum strain, maximum deformation, etc. In these studies, in general, when the difference between finite element analysis results and trained neural network results was less than 5% more or less, trained network was considered as a successful model. It can be deduced from these papers that if there is no explicit formula for the output parameter and calculation of the output of the problem using some software packages is tedious or forming the experimental set up and doing the experiment is troublesome, using a neural network is very advantageous and useful due to its speed and accuracy.

Rusia & Pathak [3] studied on the calculation of maximum equivalent von Mises stress, strain and directional deformation for a hexagonal plate with central hole using Ansys workbench and ANN. In this study, input variables were edge length, hole diameter, plate thickness and applied stress. It was carried out 81 different Ansys simulations and using these data, ANN was trained. Then accuracy of the network was tested with 4 new cases

and it was found that deviation for maximum equivalent von Mises stress, strain and directional deformation were 3.605%, 3.921% and 7.705% respectively. Rusia & Pathak [4] also followed the same procedure for a triangular plate with a central hole and deviations were 3.85% (maximum equivalent von Mises stress), 4.2% (maximum equivalent strain) and 3.98% maximum directional deformation) for this case.

In Nicholas, Padmanaban, Vasudevan & Selvaraj's study [5], buckling strength of a laminated composite plate with a central circular hole was predicted in aid of ANN. Different values of thickness, fiber orientation, material and stacking sequence were selected and FEA was utilized to obtain the data. Then, using these data, an ANN was constructed. As a result of this study, it was proposed to take advantage of this ANN model to estimate the buckling strength of the composite plate.

Ali et al. [6] predicted the SIF for different single edge crack positions of a plate. Abaqus and Matlab were utilized to create a prediction tool. In this study, plate size and crack length were constant and only changing parameter was crack position along the y axis. It was shown that the SIF value of the crack which was close to the middle of the plate was lower than the other ones.

Kutuk, Atmaca & Guzelbey [7] formed an explicit formula for three types of cracks using FEM – Ansys and ANN. Types of cracks were center crack, single edge crack and double edge crack. Variables which were used to obtain input data via Ansys for ANN were type of the crack, crack length, width of the plate and applied stress. ANN was trained with these data and an explicit formulation was created in aid of weights of the ANN structure. As a conclusion, the results of new formulation and FEM were in good agreement.

Jabur & Mohsin's study [8] was about the variation of SIF and effect of crack position, crack orientation, etc. There were five different cases in this study. These were double edge cracked plate (one crack for each side), double edge cracked plate with different positions along y axis, four edge cracked plate with different positions along y axis (two cracks for each side), double edge cracked plate with different crack orientation and double edge cracked plate with different crack orientation and kinked. In this study, it was seen that SIF values increased linearly with relative crack length and applied stress in case 1, SIF values increased exponentially when cracks were close to the upper or lower side of the plate for case 2, SIF values were increased exponentially when the

distance between cracks increased in case 3. Also, it was shown that decrease in SIF values for case 3 were higher than case 2 when crack was close to the middle of the edge (mutual shielding effect).

Rubio, Abella & Rubio [9] studied on estimation of SIF of semi elliptical cracked rotating shaft subjected to bending. In this study, crack was in the middle of the shaft and Abaqus was used to carry out finite element simulations. Variables were crack depth ratio, crack shape ratio, position of the crack and rotation angle. An ANN model was formed using input and output (SIF) data. It was shown that the ANN approach was successful to predict SIF of semi elliptical cracked rotating shaft subjected to bending. It was proposed to use ANN because of its efficiency, ease of use and low computational costs.

Kilic, Ekici & Hartomacioglu [10] developed an ANN model to estimate the ballistic penetration depth of a bullet fast and accurately. Firstly, real tests/experiments were conducted for different speed range of a bullet and these data were used to validate LS Dyna model. Then, data obtained via LS Dyna were given to the neural network and this neural network was trained. ANN model's input variables were impact velocity and thickness of the armor. Finally, the ANN model was formed and its prediction accuracy was 95%.

2. THEORY, MODELLING AND METHODS

2.1. Theory

2.1.1. Fracture Theories and Stress Intensity Factor Formulation

Fracture is the separation of the material into two or more parts because of the stresses and it is induced by crack initiation and crack propagation [11]. Crack initiation is the first part and crack propagation is the second part of the fracture. There are some reasons for crack initiation / crack propagation like creep, fatigue, impact, thermal stresses etc.

In literature, there are some theories to explain crack growth and fracture of the material. Some of them are Griffith, Orowan and Irwin theories.

According to Griffith theory, materials always have a preexisting crack and this crack grows if the elastic energy release is greater than the work which is necessary to form new fracture surfaces. It is only valid for brittle materials. This theory is shown schematically in Figure 2.1.

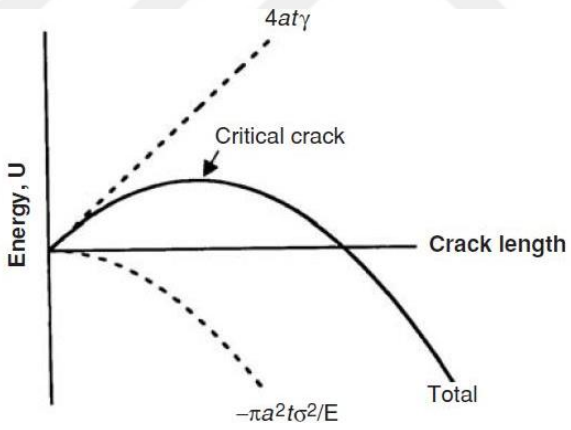


Figure 2.1. Schematic representation of Griffith theory [12]

There are two formulas in Figure 2.1. First one is surface energy and other one is elastic energy decrease in the material. As it can be seen from the figure above, total energy increases, but after a certain point total energy starts to decrease. This point is known as the critical crack length.

$$\Delta U_{elastic} = -\frac{\pi a^2 t \sigma^2}{E} \quad (2.1)$$

$$\Delta U_{surface} = 4at\gamma \quad (2.2)$$

$$\sigma = \sqrt{\frac{2E\gamma}{\pi a}} \qquad (2.3)$$

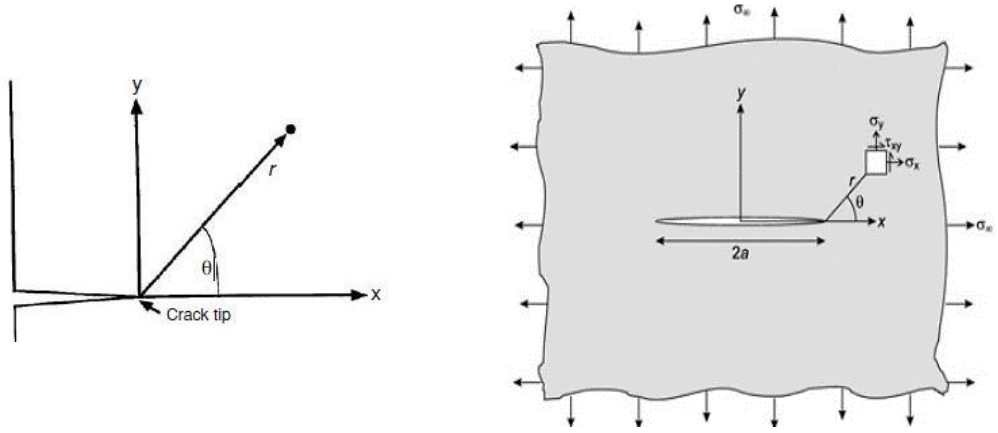
Related formulas for Griffith theory are given above. Equation (2.1) is the formula for decrease of elastic energy. Equation (2.2) is the formula for energy needed to form new fracture surfaces and last formula (2.3) is known as Griffith criterion. In these equations, parameter γ represents the specific surface energy (J/m^2). According to this formula if the length of the preexisting internal crack is greater than 2a (a is calculated with Eq. (2.3)), crack grows and fracture occurs.

Another approach is Orowan theory. This approach is very similar to the Griffith's theory. In this theory, plastic work is also included. Griffith's approach does not include plastic work and is not valid for metals. Orowan's theory's formula is given below.

$$\sigma = \sqrt{\frac{EG_c}{\pi a}} \quad (2.4)$$

As it can be seen from Eq. (2.4), Orowan's and Griffith's formulas are very similar. The only difference is 2γ and G_c. G_c is known as the strain energy release rate and includes plastic work [12].

The last approach is Irwin's theory. In this theory, stress state in the vicinity of the crack tip is very critical. Schematic representation of stress state around crack tip is given in Figure 2.2. Also related equations for Irwin's approach are given below.



(a) (b)

Figure 2.2. (a) and (b) Schematic representation of stress state around the crack tip

For a body which is under tension (Figure 2.2), normal and shear stress distribution formulas in the vicinity of sharp crack are shown below.

$$\sigma_x = K_I/(2\pi r)^{\frac{1}{2}} cos(\frac{\theta}{2})[1 - sin\left(\frac{\theta}{2}\right) sin\left(\frac{3\theta}{2}\right)] \quad (2.5)$$

$$\sigma_y = K_I/(2\pi r)^{\frac{1}{2}} cos(\frac{\theta}{2})[1 + sin\left(\frac{\theta}{2}\right) sin\left(\frac{3\theta}{2}\right)] \quad (2.6)$$

$$\tau_{xy} = K_I/(2\pi r)^{\frac{1}{2}} cos(\frac{\theta}{2}) sin\left(\frac{\theta}{2}\right) cos\left(\frac{3\theta}{2}\right)] \quad (2.7)$$

In the equations (2.5), (2.6) and (2.7) K_I is the SIF value. Subscript I indicates that SIF value is valid for opening / tension mode. There are three types of fracture modes. Figure 2.3 shows these types of modes.

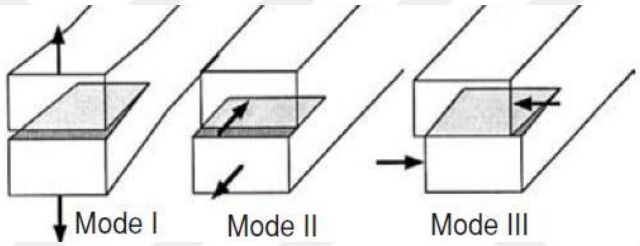


Figure 2.3. Fracture modes – opening, shearing and tearing mode respectively [13]

As illustrated in Figure 2.3 in mode I, plane of fracture is perpendicular to the load direction, in mode II, direction of fracture is same with load and in mode III, propagation of fracture plane is perpendicular to the shear force.

Each material has a different value of K_I, K_{II} and K_{III}. Because the strain energy release rate is different for each mode. In this study, mode I, opening mode calculations will be considered. Because opening mode is the most important and widespread mode [14].

Formulation of SIF of a semiinfinite body for mode I is given below Eq. (2.8) [15].

$$K_I = \sigma\sqrt{\pi a} \quad (2.8)$$

For finite bodies, this formula is multiplied by f which is a geometry factor as seen in Eq. (2.9). Geometry factor varies with the shape of the model.

$$K_I = f\sigma\sqrt{\pi a} \quad (2.9)$$

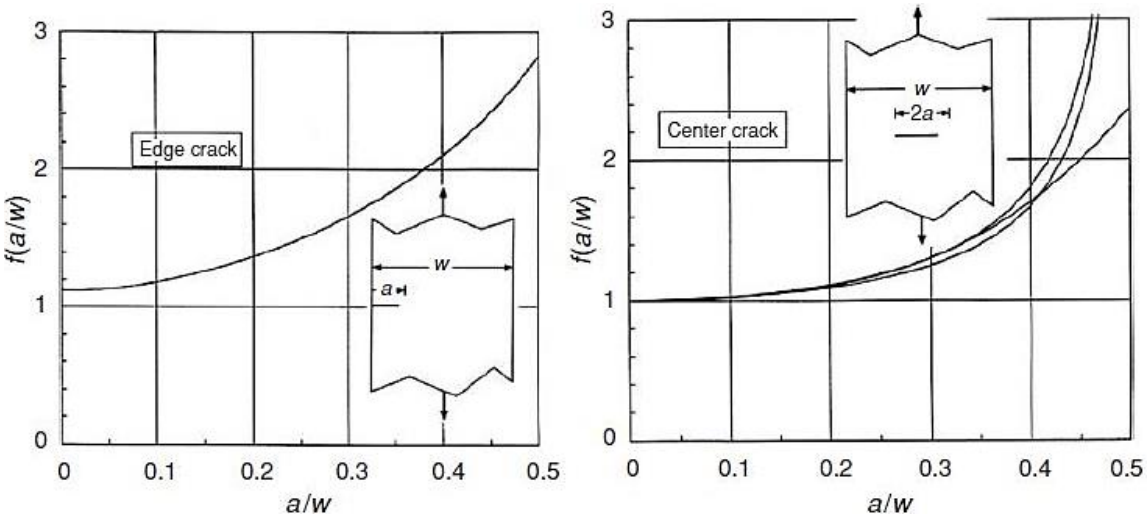
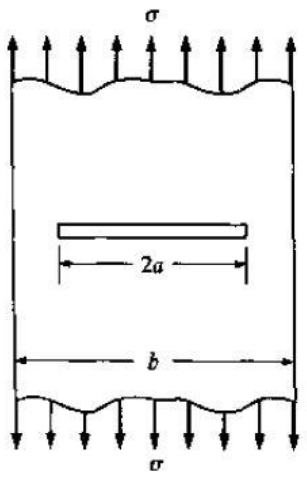


Figure 2.4. Variation of geometry correction factor for edge and center crack [15]

In Irwin's approach, if the SIF value of any fracture mode reaches a certain value, crack grows and fracture occurs. This certain value is known as critical SIF value (K_{Ic}-K_{IIc}-K_{IIIc}). Critical SIF value is a material property and it has different values for different types of modes. It also depends on temperature, plane stress, strain condition, loading rate etc.

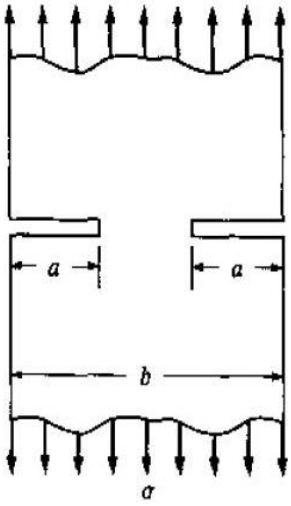
In literature, there are some formulas for common geometries in order to calculate mode I SIF values. Some of these geometries and corresponding formulas are illustrated below.



$$K_I = f\sigma\sqrt{\pi a}$$

$$f = (1 - 0.1\left(\frac{a}{b}\right)^2 + 0.96\left(\frac{a}{b}\right)^4)\sqrt{sec(\frac{\pi a}{b})}$$

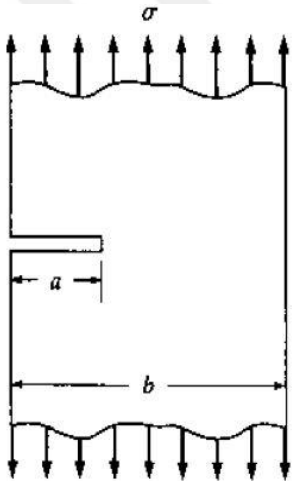
Figure 2.5. Mode I stress intensity factor calculation of a plate with center crack [16]



$$K_I = f\sigma\sqrt{\pi a}$$

$$f = [1 + 0.122cos^4\left(\frac{\pi a}{b}\right)]\sqrt{tan\left(\frac{\pi a}{b}\right)/(\frac{\pi a}{b})}$$

Figure 2.6. Mode I stress intensity factor calculation of a plate with double crack [16]

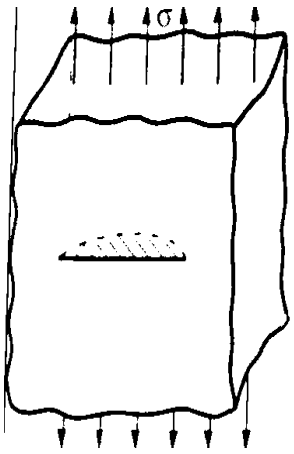


$$K_I = f\sigma\sqrt{\pi a}$$

$$f = [(0.752 + \frac{2.02a}{b} + 0.37\,(1$$

$$- sin\left(\frac{\pi a}{2b}\right))^3)/(cos(\frac{\pi a}{2b})]\sqrt{\frac{2b}{\pi a}tan\left(\frac{\pi a}{2b}\right)}$$

Figure 2.7. Mode I stress intensity factor calculation of a plate with single edge crack [16]



$$K_I = a_s\sigma\sqrt{\frac{\pi a}{Q}}f(\emptyset)$$

$$Q = 1 + 1.464(\frac{a}{c})^{1.65}$$

$$a_s = [1.13 - 0.09(a/c)][1 + 0.1(1 - sin(\emptyset))^2]$$

$$f(\emptyset) = [sin^2\emptyset + \left(\frac{a}{c}\right)^2 cos^2\emptyset]^{1/4}$$

Figure 2.8. Mode I stress intensity factor calculation of single semi elliptical surface cracked body [17]

Equations in Figure 2.8 are known as Newman-Raju equations. In these equations, a, c and ∅ / Θ represent minor radius, major radius and parametric angle respectively. These parameters are shown schematically in Figure 2.9 below.

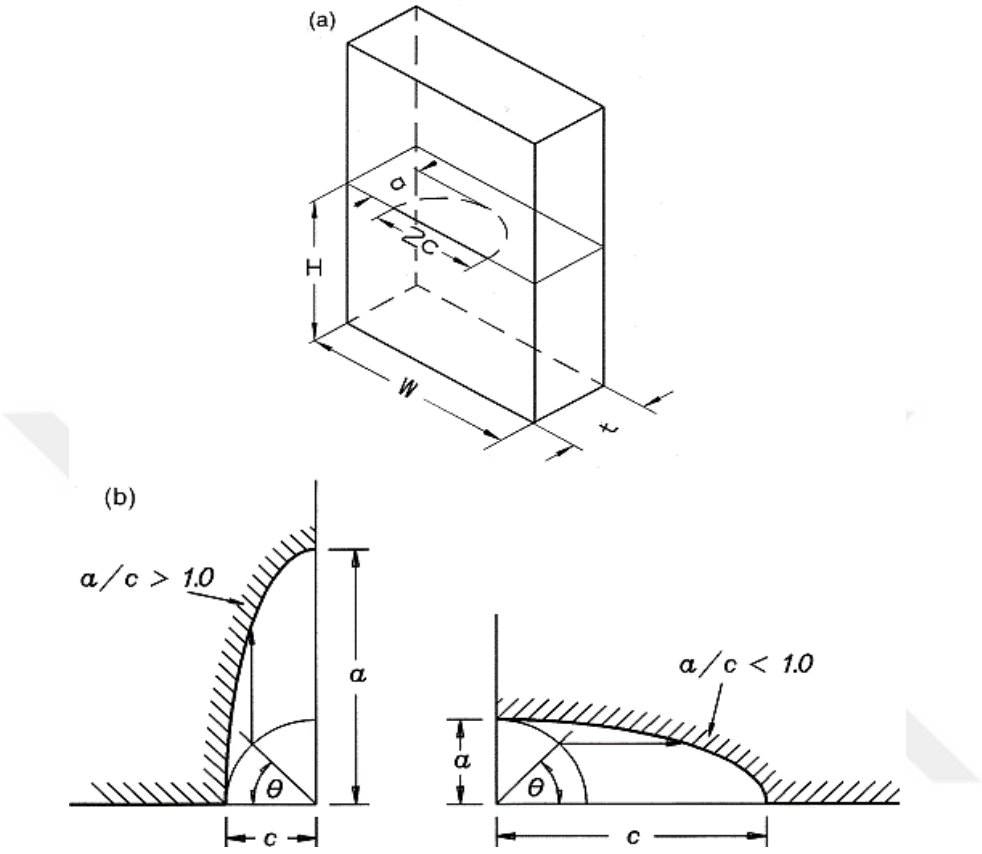


Figure 2.9. Representation of minor, major radius and parametric angle in semi elliptical surface crack [18]

In aid of these SIF equations, SIF values of some geometries can be calculated easily. But for complicated geometries, there is no explicit formula to determine SIF values analytically. So in these cases, numerical calculations like FEM approach can be used.

2.1.2. Finite Element Method

FEM or FEA is a numerical method which is used to solve engineering problems. As it was said in the previous section, if the model or problem can not be solved or too hard to solve analytically, FEM is a very good choise. It is suitable for linear or nonlinear problems.

Basic principles for FEA is quite simple. In this approach, complex geometry is divided into smaller bodies which are called as element or finite element. All of these elements

are connected to each other. Connection points of the elements to the each others are known as nodes. A simple model of element and node is illustrated in figure 2.9.

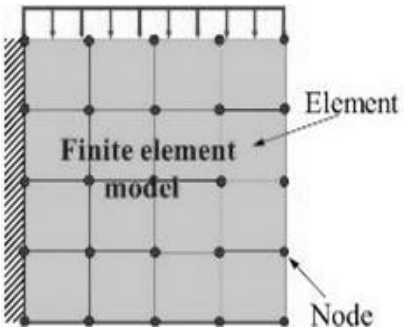


Figure 2.10. A simple finite element model [19]

There are different types of elements like line (1D), plane (2D) and solid (3D) elements. These elements are also divided into subcategories like for two dimensional elements triangular, quad, etc. and for three dimensional elements tetrahedron, hexahedron quadratic, etc. Some of these types of elements are shown below in Figure 2.11.

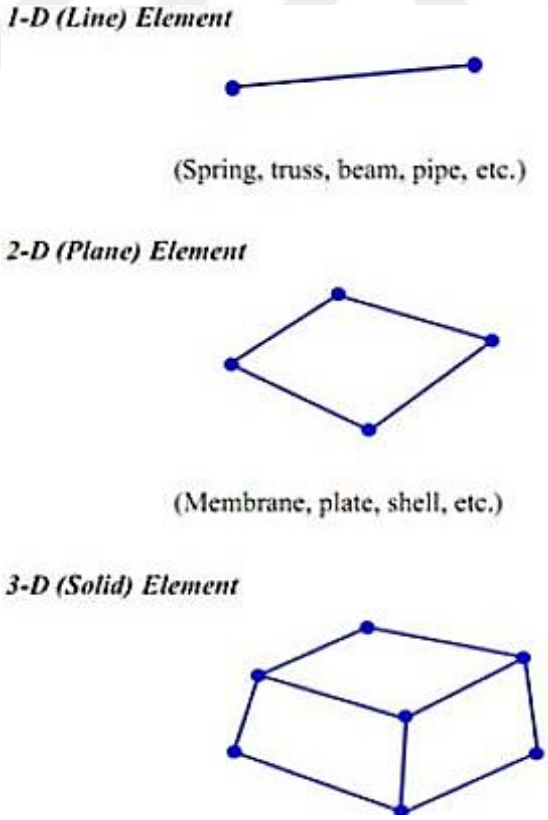


Figure 2.11. Types of elements used in finite element analysis

In FEA, equations are formed for each element and then whole system equations are formed using matrices. Finally, these equations are solved in aid of appropriate boundary conditions. Number of equations are too large for more complicated problems, so, computers and packaged softwares like Ansys, Abaqus etc. are used to get approximate solution for these problems.

Flowchart which illustrates the steps in FEM calculation is given below in Figure 2.12.

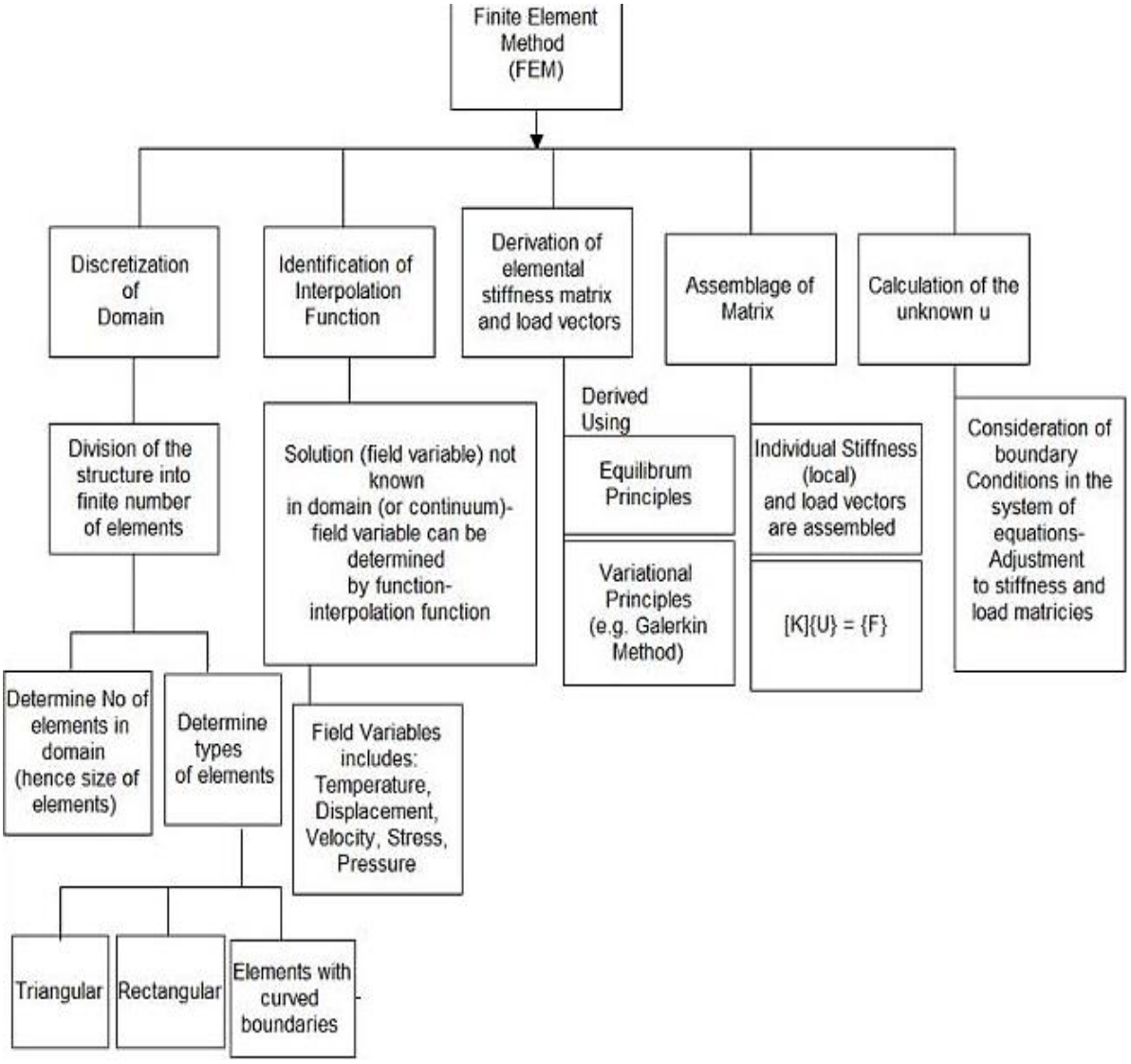


Figure 2.12. Finite element method flowchart [20]

In Ansys, there are two different types of SIF calculation method. These are displacement extrapolation method and interaction integral method. Interaction integral method is used commonly for SIF calculation and it is very similar to the J integral method. Corresponding equations for J integral and interaction integral are given as follows.

$$I=-\int_v q_{ij}[\sigma_{k,l}\varepsilon^{aux}_{k,l}\delta_{i,j}-\sigma^{aux}_{k,j}u_{k,i}-\sigma_{k,j}u^{aux}_{k,i}]dv/\int_s \delta q_n ds \quad (2.10)$$

In the equation (2.10), σ, ε and u are stress, strain and displacement respectively.

$$\sigma^{aux}_{ij}=(K^{aux}_{I}/\sqrt{2\pi r})f^{I}_{ij}(\Theta)+(K^{aux}_{II}/\sqrt{2\pi r})f^{II}_{ij}(\Theta)+(K^{aux}_{III}/\sqrt{2\pi r})f^{III}_{ij}(\Theta) \quad (2.11)$$

$$u^{aux}_{ij}=(K^{aux}_{I}/2\mu)(\sqrt{r/2\pi})g^{I}_{j}(\Theta,v)+(K^{aux}_{II}/2\mu)(\sqrt{r/2\pi})g^{II}_{j}(\Theta,v)+(K^{aux}_{III}/2\mu)(\sqrt{r/2\pi})g^{III}_{j}(\Theta,v)$$

$$\varepsilon^{aux}_{i,j}=1/2(u^{aux}_{i,j}+u^{aux}_{j,i}) \quad (2.12)$$

$$J=[(K_I^2+K_{II}^2)/(1-v^2)]/E+[K_{III}^2(1+v)]/E \quad (2.13)$$

$$J=[(K_I+K_I^{aux})^2+(K_{II}+K_{II}^{aux})^2](1-v^2)/E+(K_{III}^2+K_{III}^{aux})(1+v)/E \quad (2.14)$$

$$J=J+J^{aux}+I \quad (2.15)$$

$$I=[2(1-v^2)/E](K_IK_I^{aux}+K_{II}K_{II}^{aux})+(1/\mu)K_{III}K_{III}^{aux} \quad (2.16)$$

In these equations (Eq (2.11), (2.12), (2.13), (2.14), (2.15), (2.16)), J represents J integral and I represents interaction integral. Also μ is shear modulus.

2.1.3. Artificial Neural Network

ANN is a tool used in machine learning. It is a computational model and it is used to form or generate a relationship between the given inputs and outputs. This tool is utilized for function approximation and pattern recognition problems. It is a very powerful tool for problems which involve complicated relationship between input and output data. It is an imitation of biological human neuron. Representation of a biological human neuron is shown in Figure 2.13.

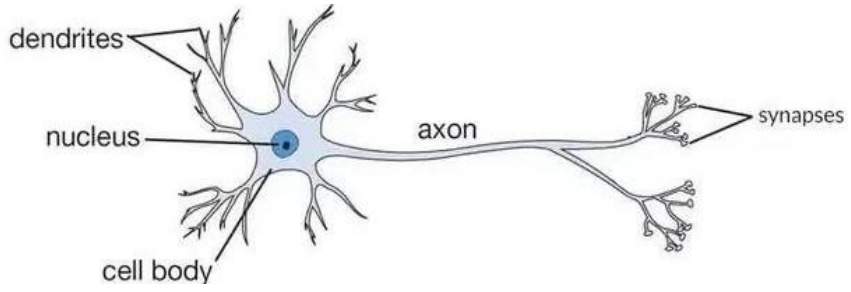


Figure 2.13. Schematic representation of a biological human neuron [21]

As it can be seen from Figure 2.13, a human neuron consists of dendrites, cell body, nucleus, axon and synapses. Dendrites receive signals from other neurons and they transmit these signals to the nucleus. Some of these signals are dominant or have larger magnitude and some of them are not dominant. So effect of these messages coming from other neuron is not the same. Then the nucleus sums these messages and transmits them to the axon. Axon process these signals and in aid of synapses (contact points), new messages are transmitted to the other neuron's dendrites.

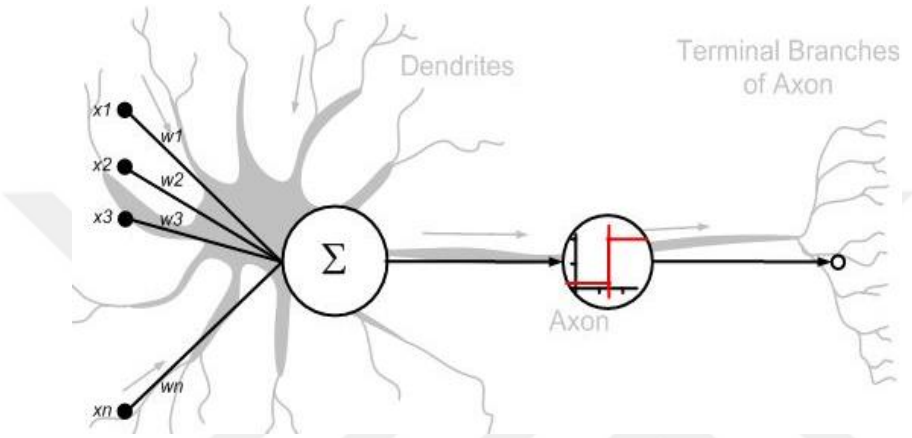


Figure 2.14. A simple artificial neural network model

In figure 2.14 above, it can be noticed that ANN is very similar with the biological human neuron. Parameters x1, x2, x3, etc. are inputs for ANN and w1, w2, w3, etc. are the weights of these inputs.

In ANN, input and output data are given to the model, this model compares these data and tries to form best relationship between these inputs and outputs executing some mathematical manipulations. Basically, the main principle of ANN is to adjust the weights of the inputs and to obtain minimum error and the best fit to the output. Adjusting the weights so as to get the best relation between input and output data is called training the network.

Main training or learning procedure of a network is as follows. Every input of the model is multiplied by its own weight and all of these multiplication results are summed. This part is known as summation part and it is similar with the nucleus or soma part of the biological human neuron. After multiplication and summation process, data goes to activation / transfer function part.

The transfer function part is again similar with the axon part of the biological human neuron. Finally processed data are compared with the target data which is given to the model as an output, error is calculated and then the same procedure is executed again to minimize the error using new input weights.

Table 2.1. Main parts of biological human neuron and artificial neural network

Biological Human Neuron	Articial Neural Network
Soma / Nucleus	Node / Summation Function
Dendrite	Input
Synapse	Weight
Axon	Transfer / Activation Function

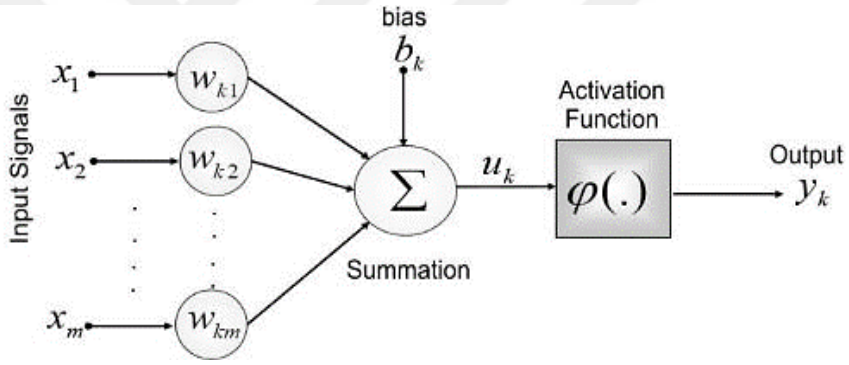


Figure 2.15. Summation and activation function of an artificial neural network model [22]

Summation and activation function parts can be seen in Figure 2.15 above. This is a simple ANN model and there are two layers, input and output. If data are linearly separable, two layers, input and output, are enough for the best fit and this model is known as perceptron. But if there is nonlinearity, at least one hidden layer must be used in the model and it is called as multilayer perceptron. Hidden layer is an intermediate layer.

Actually, having multiple hidden layer makes the model more flexible and it helps the model to learn more complex relationships. An example of multi layered model is shown below in Figure 2.16.

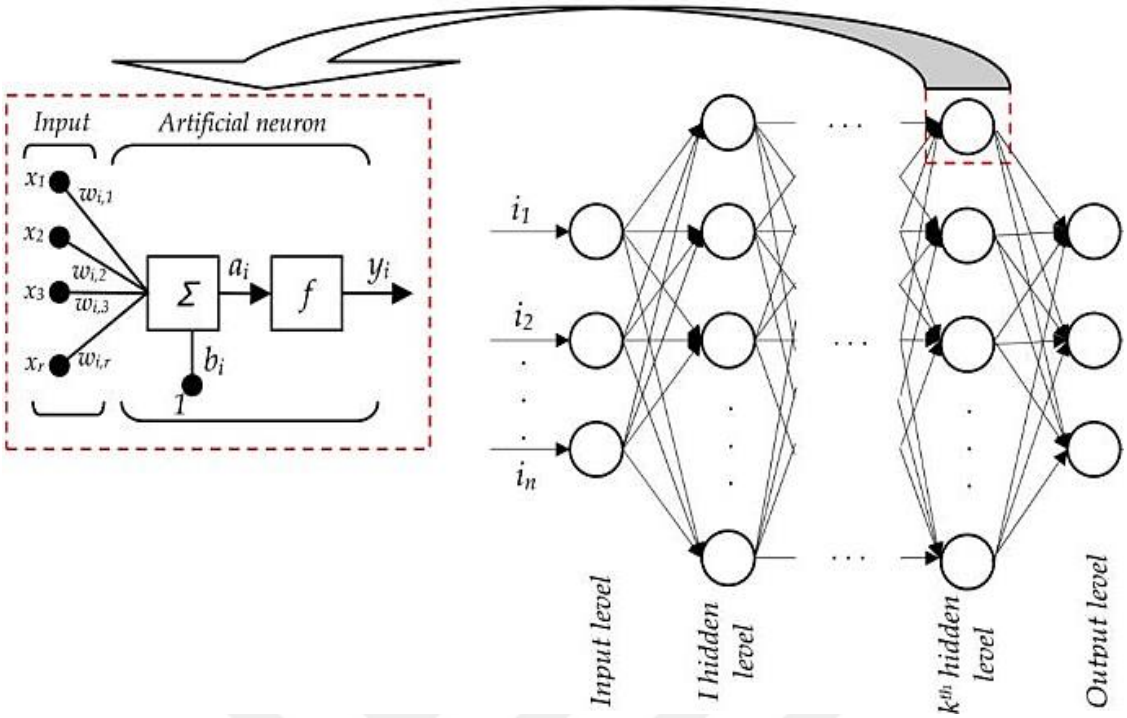


Figure 2.16. A multi layered artificial neural network (input, hidden and output layer) [23]

Moreover, there is a terminology which is called bias in ANN. It can be seen in the network models in figure 2.15 and 2.16. This is an extra node or neuron used in the model. It is a constant valued node, 1 or another constant value. ANN model changes the value of bias for every hidden and output layer node during training. It is used to shift the transfer function of the node to the left or right. So it increases the flexibility of the network and thus model fits better.

It is said that bias increases the flexibility of the model. Let's consider two cases for the logistic sigmoid activation function. One of them is the case that the weight of the input changes (Figure 2.17). Other one is the case that weight of the input is constant, but bias of the neuron or node changes (Figure 2.18).

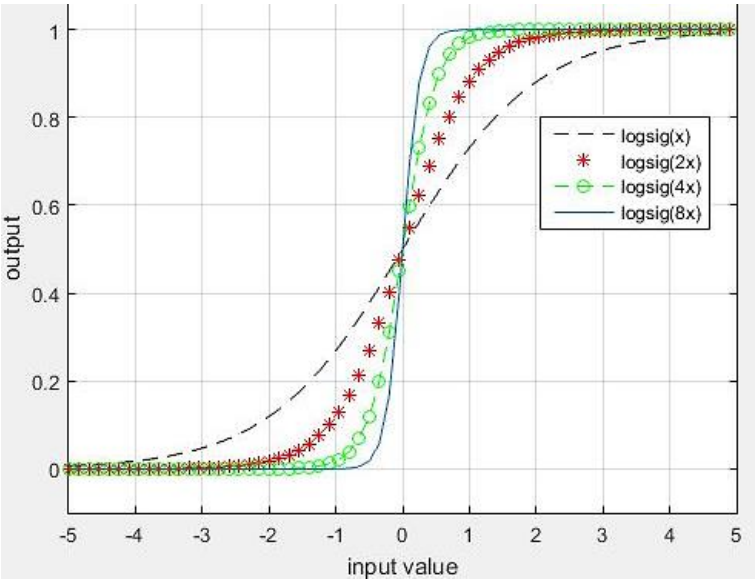


Figure 2.17. Effect of the input weights to the corresponding output of the input

As it can be noticed in figure 2.17 that weight of the input changes from 1 to 8 and it only changes the steepness of the graph.

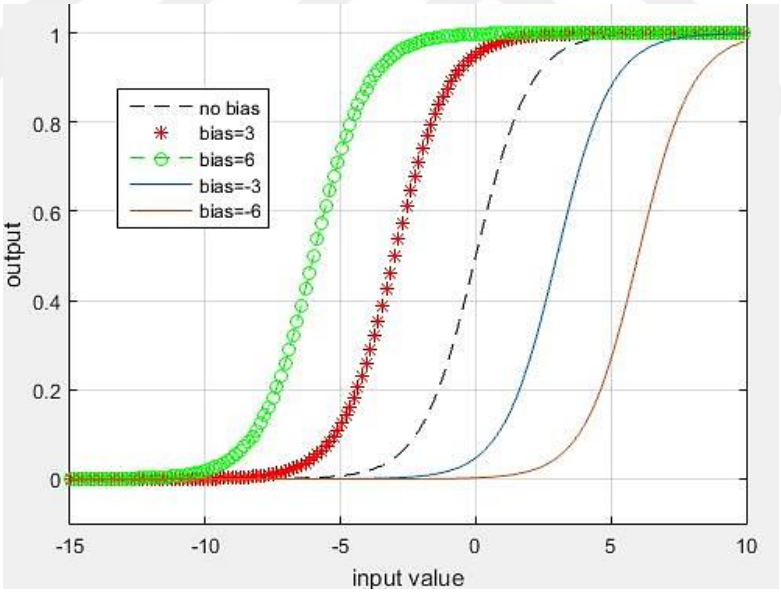


Figure 2.18. Effect of bias to the output of the network

As shown in figure 2.18, the bias value shifts the activation function and so, the estimation capacity of the model increases. This is the main advantage of the bias node in ANN. In figure 2.15 and 2.16, it can be seen that after the summation function, there is a part which is called activation function or another words, transfer function. It is very important for ANN because in aid of activation or transfer functions, ANN can easily learn complex

and nonlinear models. So selecting the type of the activation function is an important subject. In an ANN model, every input is multiplied by their weights, then multiplied inputs go to the summation function. After the summation function, all of the data go to activation fiunction and it processes these data. There are different types of activation functions. Some of them are linear and some of them are nonlinear. If the linear activation function is used, ANN loses its flexibility and it becomes a linear estimation model. But nearly all of engineering problems include nonlinearity, so using nonlinear activation function makes the network model more flexible. Some of the activation functions used in ANN model are given in Figure 2.19.

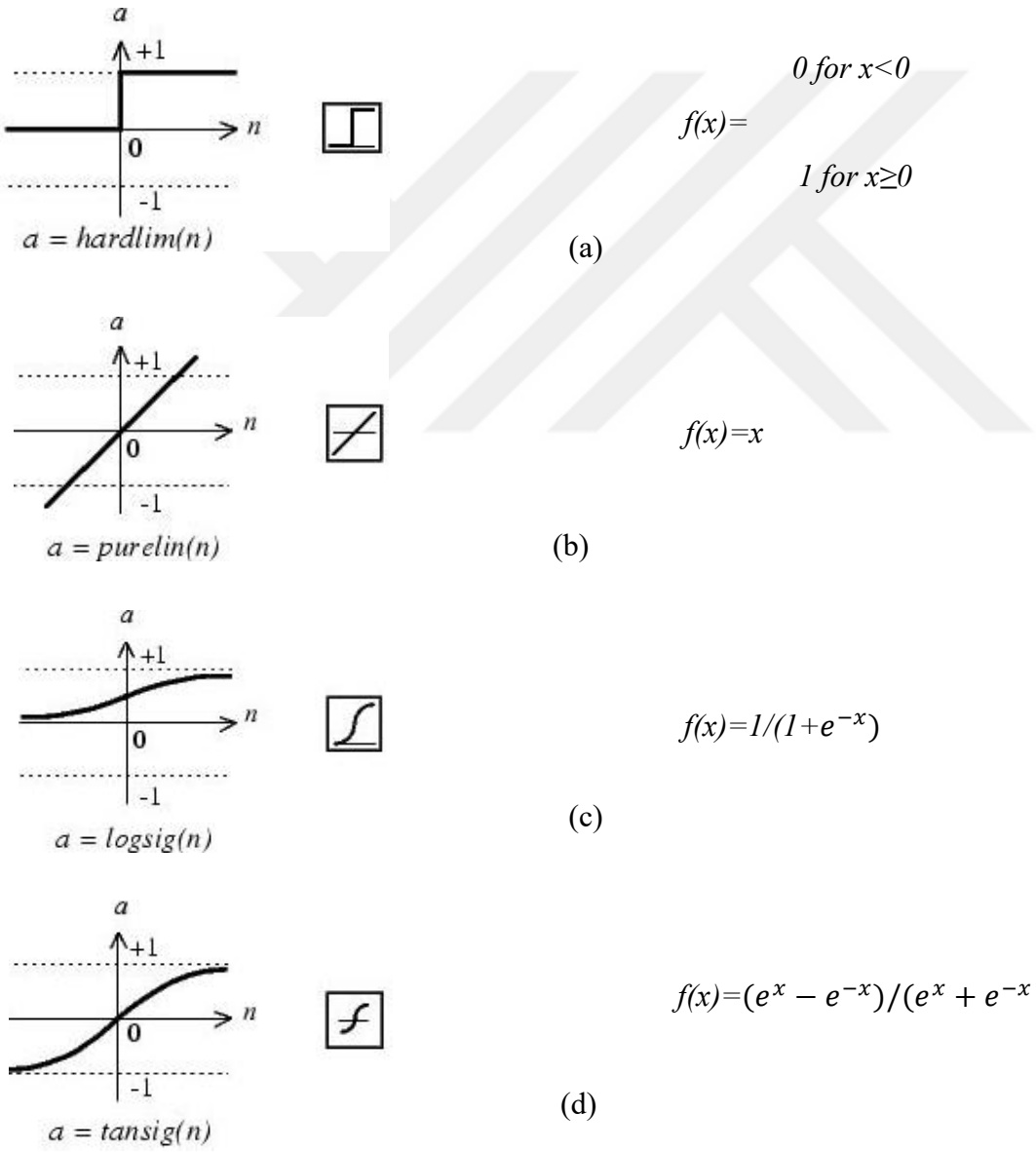


Figure 2.19. Some of the activation / transfer functions (a) hard limit (b) linear (c) logistic sigmoid (d) tangent hyperbolic

Hard limit activation function is a threshold function. The purelin activation function is linear and tangent hyperbolic and logistic sigmoid activation functions are nonlinear activation functions.

In ANN, some training or learning algorithms are used to obtain the best fit for inputs and outputs. These algorithms are divided into two categories. One of these is incremental training and the other one is batch training.

In batch training, weights of the inputs or the hidden layer are updated when all of the inputs of the network are given to the network.

In incremental training, weights of the inputs are updated for every input and network constantly updates the weights. Therefore, batch training takes less time. Incremental training is also known as online training. Most of the algorithms are batch training algorithms.

Basically, in incremental training weights are updated for each datum, in batch training weights are updated for one epoch. Epoch means one set of updates of the network weights for all of the inputs. It is a kind of iteration.

Some of the training / learning functions used in ANN are presented below.

- Gradient descent backpropagation
- Gradient descent with momentum backpropagation
- Gradient descent with adaptive learning rate backpropagation
- Gradient descent with momentum and adaptive learning rate backpropagation
- Resilient Backpropagation
- Gauss Newton
- Scaled Conjugate Gradient
- Levenberg-Marquardt

The main purpose of all of these training functions is to minimize the output error and obtain the best relationship between input and output data. But intermediate steps differ from each other and these steps specify training function’s speed, learning capacity, computer memory usage, etc.

Let's consider a simple training function – gradient descent backpropagation example and learn some of the basic concepts of ANN. For computation of output error, there is a terminology which is error function. It evaluates the difference between the target value and output value.

Target value is desired value which is given to the network by the user and the output value is the value which ANN computes using its weights and activation functions.

There are different types of error functions like mean square error, sum square error, etc. In this example sum square error is used and its equation is given in Eq. (2.17).

$$E(w)=1/2[\ (y_{target}-y_{output})^{2}]\quad (2.17)$$

In gradient descent backpropagation approach, aim is to approach the minimum value of error function using the derivative of the error function.

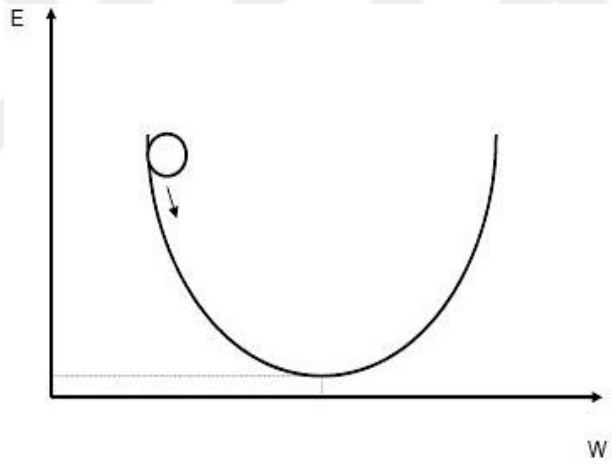


Figure 2.20. Schematic representation of gradient descent backpropagation function

Derivative of the error function with respect to the weight is calculated and it gives information about the direction which it is needed to move.

$$\frac{\partial \mathrm{E}}{\partial \mathrm{w}}=\frac{\partial \mathrm{E}}{\partial \mathrm{y}_{\mathrm{output}}}\frac{\partial \mathrm{y}_{\mathrm{output}}}{\partial \mathrm{w}}\qquad (2.18)$$

y in equation (2.18) represents activation function. In this example, it is assumed that activation function is a linear function.

$$y = f(x)\sum(wx) \quad \text{(Linear activation function)} \quad (2.19)$$

$$\frac{\partial \mathrm{E}}{\partial \mathrm{w}} = \frac{\partial \mathrm{E}}{\partial \mathrm{y_{output}}} \frac{\partial \mathrm{y_{output}}}{\partial \mathrm{w}} = -(y_{\mathrm{target}} - y_{output})x = -\delta x \quad (2.20)$$

Equation (2.19) and (2.20) show the calculation the derivative of error function. The term δ represents the difference between target value and output value. Derivation of error function is negative, so in order to decrease error, weight of the input is moved to the negative direction of derivative as shown in Eq. (2.21). (Figure 2.20)

$$-(-\delta x) = \delta x \quad (2.21)$$

Final step is to calculate the new value of weight using the gradient of the error function and follow the same procedure to obtain error small enough. After some epochs, value of the difference between the new weight and old weight becomes so small, nearly zero. It means the error is very small.

$$w_{new} = w_{old} + \eta\, \delta x \quad (2.22)$$

Term η in Eq. (2.22) is known as the learning rate of the model. The value of learning rate is very important for training procedure. If it is too low, training function takes too much time to converge. Also, if it is too high, training function becomes unstable. For gradient descent backpropagation algorithm, learning rate is constant, but for some types of algorithms, learning rate changes during the learning process and it improves performance of the training function.

Another parameter used in some of the training functions like gradient descent with momentum backpropagation, gradient descent with momentum and adaptive learning rate backpropagation etc. is momentum coefficient (Eq. 2.23).

$$w_{new} = w_{old} + \eta\, \delta x + momentum\ coeffcient * \Delta w \quad (2.23)$$

Momentum coefficient in improves the stability of the network model. Also in aid of momentum coefficient, it can be avoided from converging to a local minimum value [24].

2.2. Modelling and Methods

As stated in previous chapters, in this study, the aim is to estimate SIF values of two different semi elliptical surface cracked bodies. One of these bodies has two semi elliptical surface cracks (first case), one of them is at the front side and the other one is at

the back side. In the second case, the body has four semi elliptical surface cracks, two of them is at the front side and two of them is at the back side.

In this study, firstly, finite element models for both cases were established, then data were generated using these finite element models. After FEA step, ANN models were established for both cases and these models were trained with data obtained with FEM. Finally, two succesfully trained ANN models were developed for calculation of SIF values without any simulation or analytical calculation.

2.2.1. Finite Element Model Establishment

In this study, for FEA, Ansys Workbench was used. Figure 2.21 shows the front side of the cases considered in this thesis. Back sides of the bodies are same with the front sides.

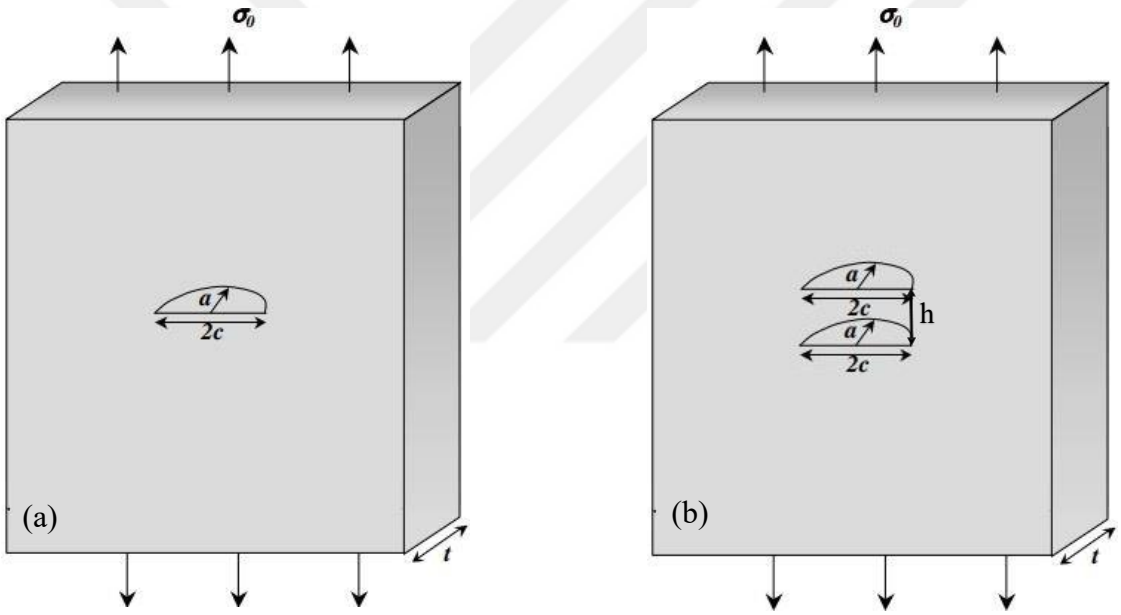


Figure 2.21. Front and back side of the semi elliptical surface cracked bodies considered in the thesis (a) case 1 (b) case 2

Ansys Static Structural module was used in this thesis. Plates used in Ansys were square. It was considered that the dimensions of the semi elliptical cracks in the plates were so small compared with the dimensions of the plates. So in the FEA, appropriate dimension for the plates was used. Firstly, plate was created using geometry section in static structural module, then model section was utilized for FEA.

In model section, crack was created using fracture module. Figure 2.22 shows the created plate and fracture module.

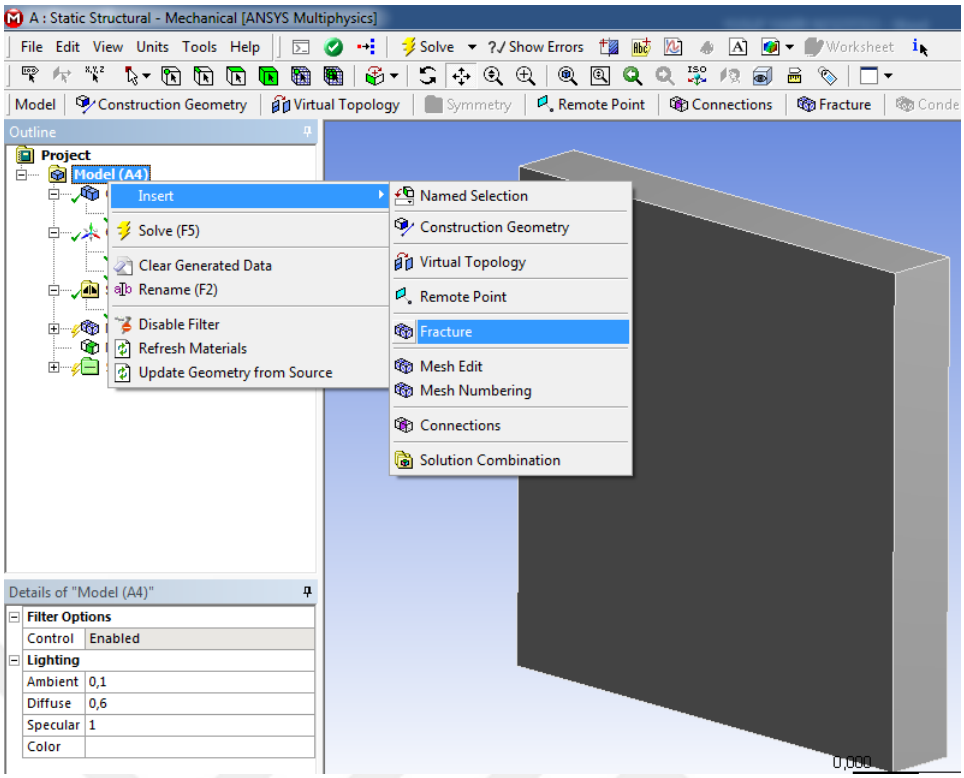


Figure 2.22. Fracture module in Ansys and an example for the plate used for analyzes

Different types of cracks can be created using fracture module. In this study, as stated before, semi elliptical crack was used. Figure 2.23 shows the types of the cracks in fracture module and formation of a semi elliptical crack.

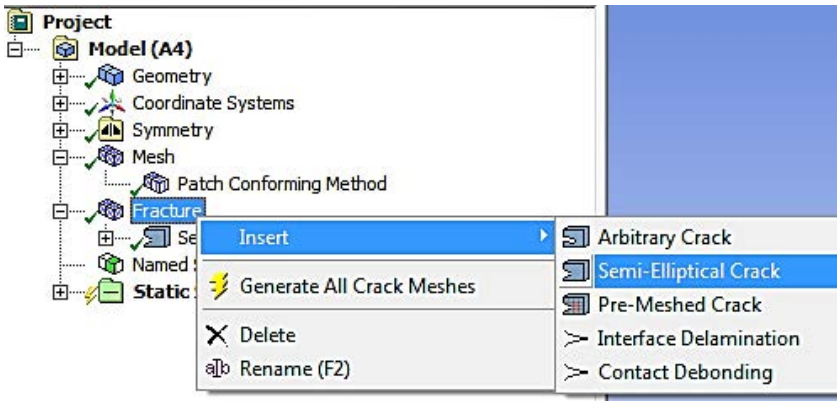
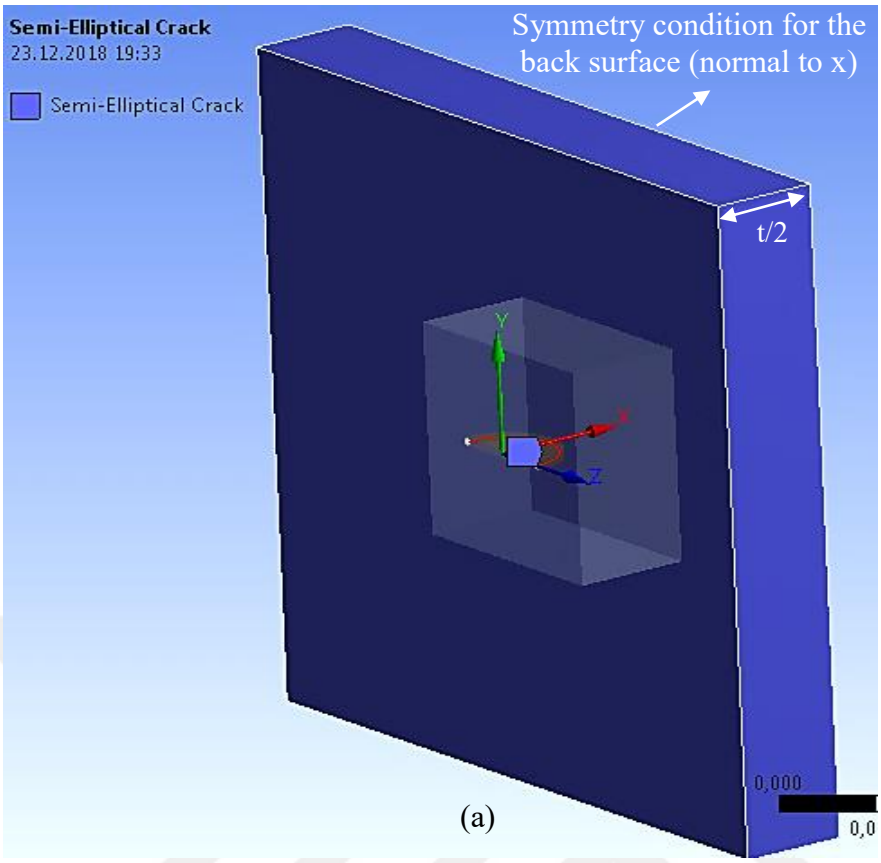
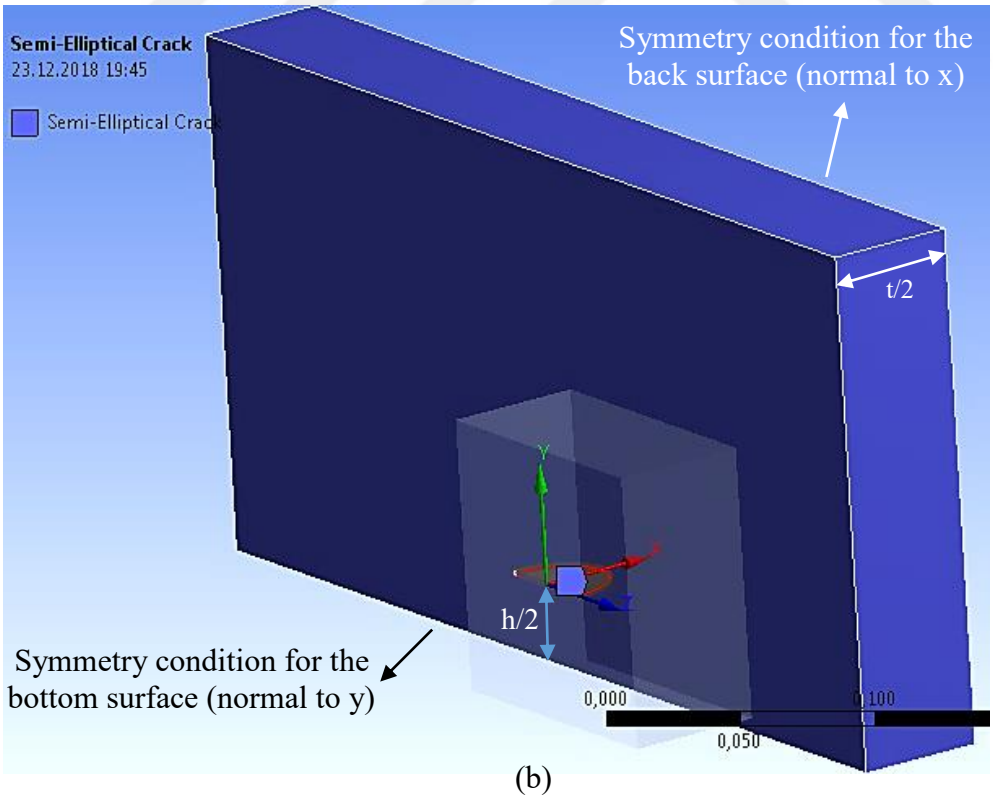


Figure 2.23. Creating semi elliptical crack using fracture module

In FEM, using symmetry conditions makes the model simpler and reduces time required to solve the problem. So in this study, symmetry conditions were used. In the first case, half of the plate and in the second case, quarter of the plate was modeled. These models are shown in figure 2.24.



(a)



(b)

Figure 2.24. Symmetry conditions for case 1 (a) and case 2 (b)

2.2.2. Mesh Convergence Study

In FEA, mesh convergence study is very important. Because it must be provided that mesh size is small enough to obtain the accurate result. In most of finite element studies, only parameter which effects the result accuracy is main mesh size. So in these studies, firstly bigger element size is used, then it is decreased gradually. After a certain point, change in the mesh size does not change the result considerably. This point is considered as the ideal mesh size for analysis.

In this study, there were some parameters which have a significant role on the accuracy of the SIF value. These were main mesh size, circumferential division, crack front division and mesh contour. As shown in figure 2.25 and 2.26, three of these parameters, circumferential division, crack front division and mesh contour, are related with semi elliptical crack.

Details of "Semi-Elliptical Crack"	
Source	Analytical Crack
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Coordinate System	Coordinate System
Align with Face Normal	Yes
Project to Nearest Surface	Yes
Crack Shape	Semi-Elliptical
--Major Radius	5,e-003 m
--Minor Radius	6,e-003 m
Mesh Method	Hex Dominant
Largest Contour Radius	1,e-003 m
Crack Front Divisions	45
Fracture Affected Zone	Program Controlled
Fracture Affected Zone Height	2,477e-003 m
Circumferential Divisions	96
Mesh Contours	10

Figure 2.25. Details of a semi elliptical crack in fracture module in Ansys

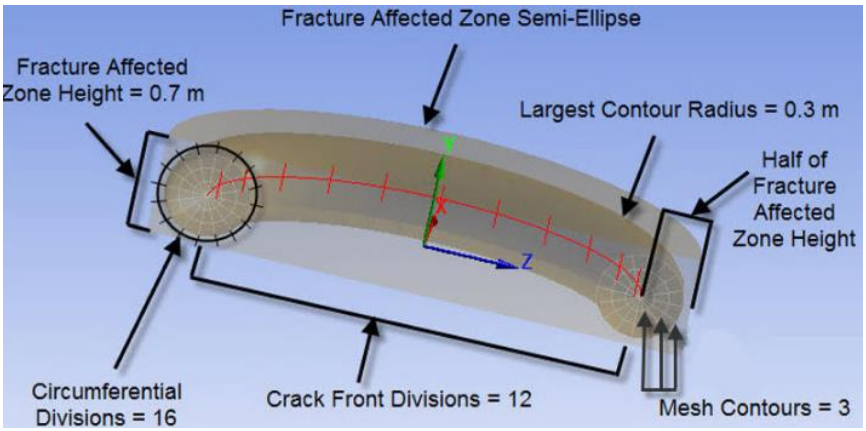


Figure 2.26. Schematic representation of a semi elliptical crack [25]

Schematic representation of a semi elliptical crack and its defining parameters are given in figure 2.26. As seen in figure 2.26, there are some parameters used to form a semi elliptical crack in Ansys. Mesh contour is the circles around the crack tip line and contour nearest to the crack tip line has largest contour number. For instance, in figure 2.26, the number of mesh contour is 3 and innermost circle is defined as the contour 3. Largest contour radius is the radius of mesh contour 1 which is the outmost circle. Circumferential division is the division number of the mesh contours and its value must be multiples of 8. The last parameter, crack front division is the division number of the crack tip line. Almost all of these parameters effect accuracy of the SIF. So mesh convergence study was done to determine the ideal values of these parameters and main mesh size in this thesis. Another parameter which effects the accuracy of the result is dimension of the body / plate. Because in this study, as stated before, it was considered that the SIF value of the semi elliptical crack was not dependent to the length / width of the plate. So determination of optimum value for length and width of the plate is also important. Procedure which was followed to determine the optimum value of the dimension of the plate was explained in the results section.

Meshing around the crack line is much more important for the SIF calculation of semi elliptical surface cracked bodies. Because maximum deformation occurs at these points. So semi elliptical crack dependent parameters like mesh contour, circumferential division, etc. play much more critical role than the mesh size for the accuracy of the SIF value. In mesh convergence study, the results of FEM, Ansys, were compared with the Newman Raju equation (figure 2.8) which is valid for semi elliptical surface cracked plate. (only one crack at the front side of the body) Different values of main mesh size, circumferential division, crack front division and mesh contour were used in FEA. Firstly, three of these parameters were taken as constant and value of one of these parameters was changed. Obtained result was compared with the equation result and optimum value was determined. The same procedure was followed for each of these parameters.

In mesh convergence analysis, it was considered that value of applied stress, largest contour radius, major radius (c) and minor radius (a) were 1 MPa, 1 mm, 10 mm and 5 mm respectively. 200 mm cube was used.

- Convergence study for main mesh size
    - Other parameters (constant parameters) were considered as

- Circumferential division 64
- Mesh contour 10
- Crack front division 45

Results of the convergence study for main mesh size are given in table 2.2 below. As seen in this table, optimum value for main mesh size is 3. Table 2.3 shows the result of the analysis for main mesh size 3 mm.

Table 2.2. Convergence study for main mesh size

Main Mesh Size (mm)	6	4	3	2.5	2
Mean Error %	1.196	1.177	1.151	1.15	1.15

Table 2.3. Ansys result for main mesh size 3, circumferential division 64, crack front division 45 and mesh contour 10

64-3-45-10				K_I Values (Pa√m)				
a (m)	c (m)	a/c	Angle (°)	FEM	$\frac{KI_{FEM}}{\sigma\sqrt{(\pi a/Q)}}$	Analytical	$\frac{KI_{analytic}}{\sigma\sqrt{(\pi a/Q)}}$	Error %
0,005	0,01	0,5	0	86534	0,836	87343,0	0,844	0,92619
0,005	0,01	0,5	6,13	86298	0,834	86461,7	0,835	0,18937
0,005	0,01	0,5	12,07	85943	0,830	87009,7	0,841	1,22597
0,005	0,01	0,5	22,98	89181	0,862	90482,6	0,874	1,43854
0,005	0,01	0,5	32,58	93650	0,905	94829,1	0,916	1,2434
0,005	0,01	0,5	41,13	97803	0,945	98907,4	0,956	1,11663
0,005	0,01	0,5	52,62	102810	0,993	104003,7	1,005	1,14776
0,005	0,01	0,5	63,05	106450	1,029	107819,7	1,042	1,27036
0,005	0,01	0,5	76,01	109460	1,058	111050,8	1,073	1,43254
0,005	0,01	0,5	90	110580	1,068	112292,4	1,085	1,52492
				Mean error for optimum mesh size				1,151

- Convergence study for mesh contour
  - Other parameters (constant parameters) were considered as
    - Circumferential division 64
    - Main mesh size 3 mm
    - Crack front division 45

Error values for 7 different cases are shown in table 2.4 and as it is seen in figure 2.27, the optimum value of mesh contour is 10.

Table 2.4. Convergence study for number of mesh contours

Number of Mesh Contours	2	3	4	5	8	10	12
Mean Error %	1.33	1.278	1.235	1.197	1.165	1.151	1.15

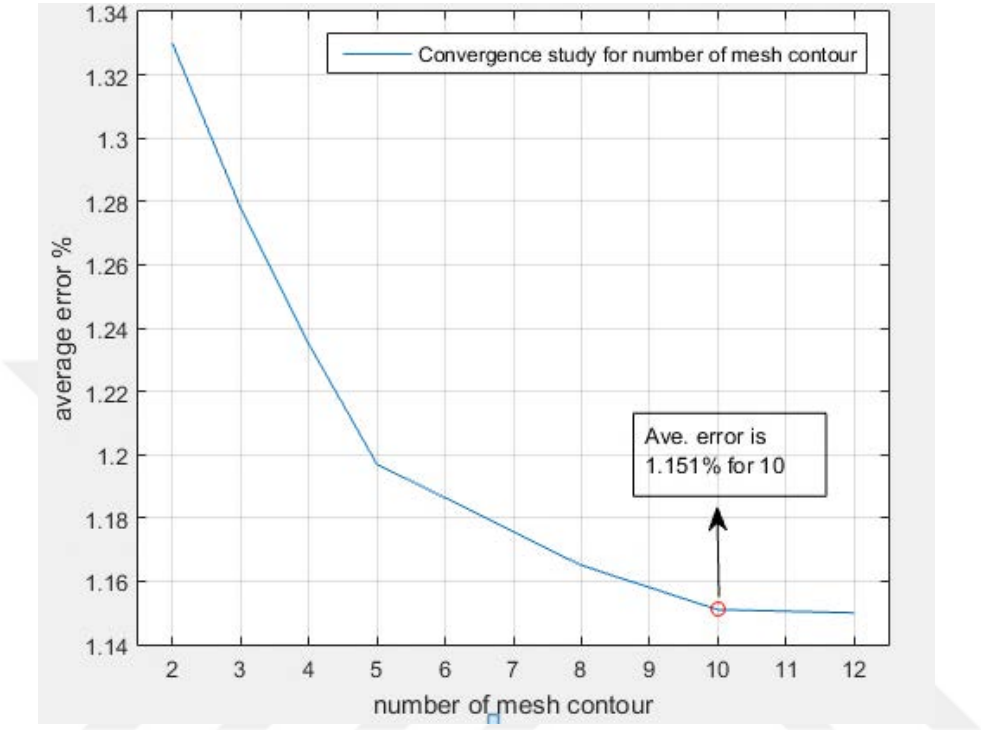


Figure 2.27. Graphical representation of the convergence study for mesh contour

- Convergence study for circumferential division
  - Other parameters (constant parameters) were considered as
    - Mesh contour 10
    - Main mesh size 3 mm
    - Crack front division 45

Results of the convergence study for the number of circumferential division are presented in table 2.5. As it is understood from figure 2.28, ideal value is 96.

Table 2.5. Convergence study for number of circumferential division

Circum. Division	8	16	32	48	64	72	80	88	96	104
Mean Error %	1.836	1.546	1.332	1.23	1.151	1.134	1.121	1.092	1.08	1.076

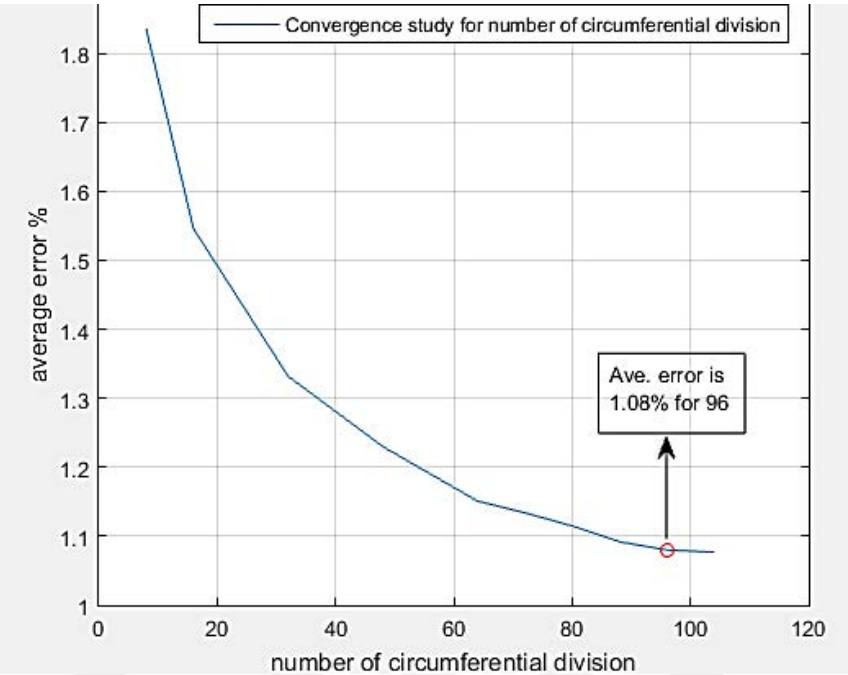


Figure 2.28. Graphical representation of convergence study for circumferential division

Table 2.6. Stress intensity factor calculation results for circumferential division 96, main mesh size 3, crack front division 45 and mesh contour 10

96-3-45-10				K_I Values (Pa√m)				
a (m)	c (m)	a/c	Angle (°)	FEM	$\frac{KI_{FEM}}{\sigma\sqrt{(\pi a/Q)}}$	Analytical	$\frac{KI_{analytic}}{\sigma\sqrt{(\pi a/Q)}}$	Error %
0,005	0,01	0,5	0	86523	0,836	87343,0	0,844	0,93879
0,005	0,01	0,5	6,13	86324	0,834	86461,7	0,835	0,1593
0,005	0,01	0,5	12,07	86015	0,831	87009,7	0,841	1,14322
0,005	0,01	0,5	22,98	89276	0,863	90482,6	0,874	1,33355
0,005	0,01	0,5	32,58	93725	0,906	94829,1	0,916	1,16432
0,005	0,01	0,5	41,13	97848	0,945	98907,4	0,956	1,07113
0,005	0,01	0,5	52,62	102910	0,994	104003,7	1,005	1,05161
0,005	0,01	0,5	63,05	106540	1,029	107819,7	1,042	1,18689
0,005	0,01	0,5	76,01	109580	1,059	111050,8	1,073	1,32448
0,005	0,01	0,5	90	110670	1,069	112292,4	1,085	1,44477
								1,08

Optimum values for circumferential division, mesh contour and main mesh size were determined as 96, 10 and 3 mm respectively. Last parameter which effects the accuracy of SIF value was crack front division. As seen in convergence studies above, crack front division value was considered as 45. In this thesis, the main purpose was to estimate SIF

value for case 1 and case 2 for different values of major radius, minor radius, parametric angle, etc. So it was needed so many angle values for each analysis in order to train the ANN model and estimate SIF value accurately at any angle. In the result of a simulation in Ansys, the program gives x, y and z coordinates of the crack front and corresponding SIF values. If the crack front division value is selected as 45, Ansys gives 90 different coordinates for SIF calculation, so it means, 90 different values are obtained in one analysis. Actually, 90 different values were enough for this study, but an extra analysis was done for crack front division value 90 (minor radius 2.5 mm and major radius 5 mm) and results were compared with each other to be sure for the value of 45. Mean error values of analysis for 96 (circumferential division), 3 (main mesh size), 90 (crack front division), 10 (mesh contour) and 96 (circumferential division), 3 (main mesh size), 45 (crack front division), 10 (mesh contour) were 1.35% and 1.36% respectively. Error values were so close to each other and therefore value of 45 for crack front division was acceptable. Besides, taking crack front division value as 90 made the analysis more tedious job. It took almost doubled the time compared to analysis using the value of 45.

In consequence of mesh convergence study, it was determined that the value of circumferential division, main mesh size, crack front division and mesh contour were 96, 3, 45 and 10 respectively.

Representation of crack in Ansys is given in figure 2.29 and meshed model of the crack is given in figure 2.30.

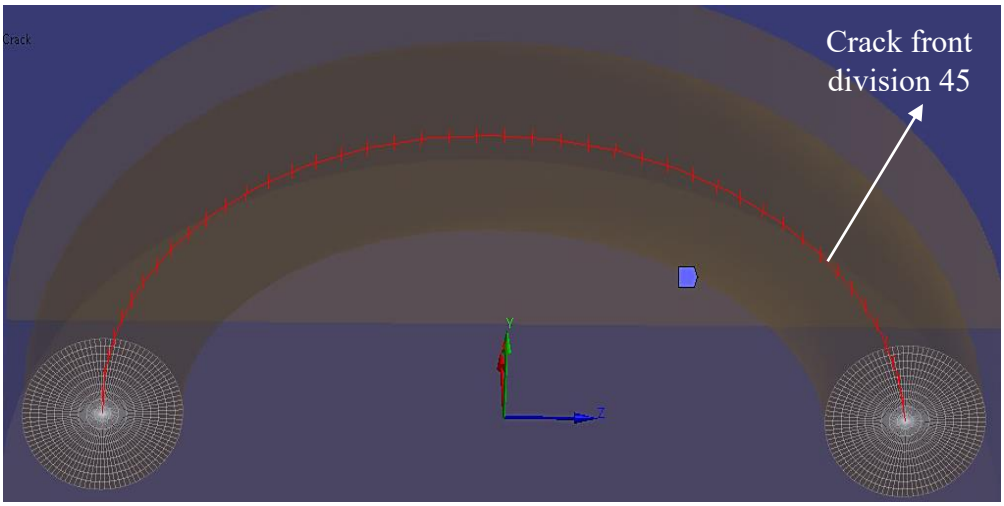


Figure 2.29. Representation of the crack used in Ansys (general view)

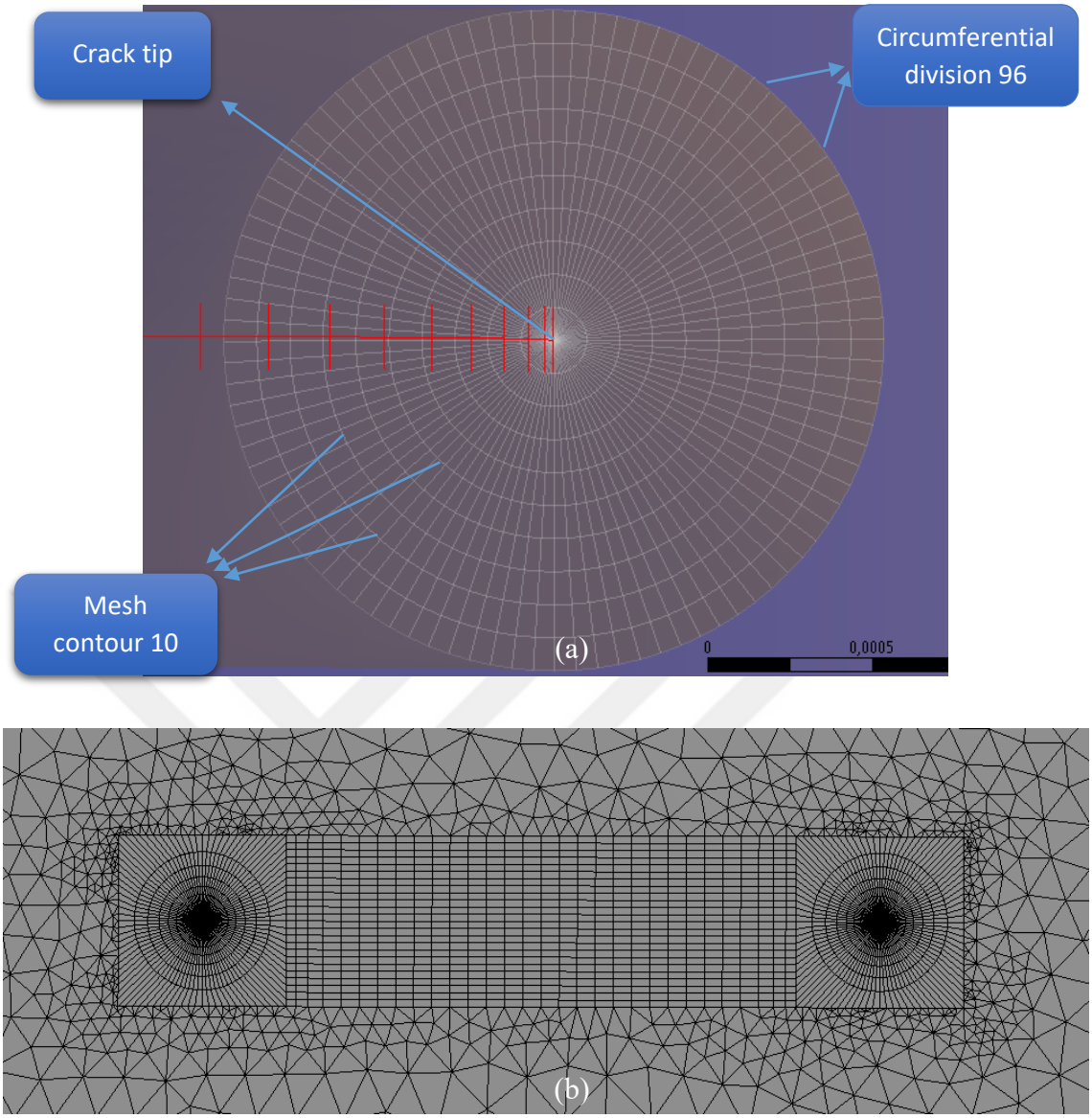


Figure 2.30. Detailed view (a) and meshed model of the semi elliptical crack used in Ansys

2.2.3. Verification of the Finite Element Model with Formula

As a result of mesh convergence study, optimum values of 4 different parameters were determined. But mesh convergence study was done for the model that minor and major radius was 5 mm and 10 mm and so mean error value was only valid for this model. So few Ansys simulations were done and mean error values were calculated so as to verify the model which was obtained in consequence of mesh convergence study. Table 2.7 shows the results of these simulations.

Table 2.7. Verification of finite element model with Newman Raju Equation

Circumferential Division : 96 Main Mesh Size : 3 mm Crack Front Division : 45 Mesh Contour : 10			
Major Radius c (m)	Minor Radius a (m)	a/c	Mean Error %
0.005	0.0025	0.5	1.35
0.005	0.004	0.8	0.65
0.005	0.005	1	0.23
0.005	0.006	1.2	0.5
0.01	0.003	0.3	1.3
0.01	0.005	0.5	1.08
0.01	0.008	0.8	0.6
0.01	0.01	1	0.38
Total mean error			**0.76**

As one can see in table 2.7 above, total mean error value (error between FEA-Ansys and Newman Raju equation) was calculated 0.76% as a result of 8 different cases. This value was well enough, so this model (96-3-45-10- circumferential division, main mesh size, crack front division and mesh contour) was used in the subsequent Ansys simulations. An example of mean error calculation in table 2.7 (c= 0.005 m, a= 0.005 m) is given in table 2.8 as follows.

Table 2.8. An example of mean error calculation for verification process

a (m)	c (m)	a/c	Angle (°)	FEM (Pa√m)	$\frac{KI_{FEM}}{\sigma\sqrt{(\pi a/Q)}}$	Analytical (Pa√m)	$\frac{KI_{analytic}}{\sigma\sqrt{(\pi a/Q)}}$	Error %
0,005	0,005	1	0,00	90272	1,131	91341,0	1,144	1,17031
0,005	0,005	1	2,00	91315	1,144	90771,9	1,137	0,5983
0,005	0,005	1	6,00	90088	1,128	89695,2	1,123	0,43791
0,005	0,005	1	8,00	89445	1,120	89190,3	1,117	0,28556
0,005	0,005	1	10,01	88682	1,111	88706,1	1,111	0,02711
0,005	0,005	1	12,00	88175	1,104	88247,1	1,105	0,08166
0,005	0,005	1	14,00	87602	1,097	87809,0	1,100	0,23575
0,005	0,005	1	15,99	87211	1,092	87397,2	1,095	0,2131
0,005	0,005	1	17,98	86753	1,087	87005,9	1,090	0,29061
0,005	0,005	1	19,99	86440	1,083	86634,2	1,085	0,22414
0,005	0,005	1	21,98	86062	1,078	86288,9	1,081	0,26298
0,005	0,005	1	23,99	85803	1,075	85962,5	1,077	0,18555
0,005	0,005	1	26,02	85481	1,071	85654,6	1,073	0,20266
0,005	0,005	1	28,03	85267	1,068	85372,1	1,069	0,1231
0,005	0,005	1	30,00	84993	1,064	85114,4	1,066	0,14259

0,005	0,005	1	32,00	84821	1,062	84873,4	1,063	0,06168
0,005	0,005	1	33,97	84588	1,059	84655,4	1,060	0,07964
0,005	0,005	1	35,97	84446	1,058	84453,0	1,058	0,00825
0,005	0,005	1	38,00	84243	1,055	84265,4	1,055	0,0266
0,005	0,005	1	40,00	84127	1,054	84098,4	1,053	0,03403
0,005	0,005	1	41,97	83951	1,051	83950,4	1,051	0,00076
0,005	0,005	1	43,97	83856	1,050	83814,9	1,050	0,04899
0,005	0,005	1	46,03	83699	1,048	83691,7	1,048	0,0087
0,005	0,005	1	47,96	83621	1,047	83588,9	1,047	0,03836
0,005	0,005	1	50,03	83483	1,046	83492,3	1,046	0,01118
0,005	0,005	1	51,99	83422	1,045	83413,1	1,045	0,01062
0,005	0,005	1	53,99	83300	1,043	83342,7	1,044	0,05122
0,005	0,005	1	55,96	83252	1,043	83283,3	1,043	0,03763
0,005	0,005	1	58,00	83141	1,041	83231,5	1,042	0,10871
0,005	0,005	1	60,00	83103	1,041	83188,8	1,042	0,10312
0,005	0,005	1	61,97	83003	1,040	83154,0	1,041	0,18162
0,005	0,005	1	64,02	82979	1,039	83124,7	1,041	0,17529
0,005	0,005	1	66,04	82893	1,038	83101,7	1,041	0,25109
0,005	0,005	1	68,03	82879	1,038	83083,8	1,041	0,24655
0,005	0,005	1	70,00	82805	1,037	83070,3	1,040	0,31942
0,005	0,005	1	71,95	82800	1,037	83060,4	1,040	0,31346
0,005	0,005	1	73,98	82733	1,036	83052,8	1,040	0,3851
0,005	0,005	1	76,00	82739	1,036	83047,7	1,040	0,37176
0,005	0,005	1	77,99	82681	1,036	83044,5	1,040	0,43766
0,005	0,005	1	79,98	82694	1,036	83042,5	1,040	0,41966
0,005	0,005	1	81,95	82646	1,035	83041,5	1,040	0,47621
0,005	0,005	1	84,03	82666	1,035	83041,0	1,040	0,45156
0,005	0,005	1	85,99	82623	1,035	83040,9	1,040	0,5032
0,005	0,005	1	88,05	82649	1,035	83040,9	1,040	0,47194
0,005	0,005	1	90,00	82613	1,035	83041,0	1,040	0,51538
							Mean Error %	0,23

Moreover, an extra study was done in order to check the accuracy of the Ansys model. In this study, 3 different simulations were done for case 2. The value of a/t (ratio of minor radius to thickness) was selected so small and the value of h (vertical distance between two parallel cracks) was selected so large in these simulations. Because it is expected that, if the thickness of the plate is much larger than the minor radius of the crack, nearly the same SIF results of the plate which has crack / cracks only in one side, are obtained and if the two parallel cracks are far away from each other (the value of h is so large), SIF results approach the SIF value of the plate that has one crack instead of two parallel cracks (no mutual shielding effect). The value of a/t was taken as 0.05 and the value of h was taken as 20 cm for this extra study. In the wake of simulations, as it was expected, SIF

results were so close to the results of one semi elliptical surface cracked plate. The results of this study are shown in table 2.9 below. As seen in table 2.7 and table 2.9, mean error values are so close to each other. (0.65% - 0.74%, 0.23% - 0.26%, 0.5% - 0.45%)

Table 2.9. Verification of the Ansys model with Newman Raju equation (h=20 cm, a/t= 0.05)

Major Radius c (m)	Minor Radius a (m)	Mean Error %
0.005	0.004	0.74
0.005	0.005	0.26
0.005	0.006	0.45

2.2.4. Artificial Neural Network Model Development

The main purpose of an ANN is to correlate input and corresponding output values successfully. In order to provide this condition, some basic parameters like network type, training function, transfer function, input layer, hidden layer, output layer etc. are used. Matlab Neural Network Module (nntool) was utilized in this study to model ANN and perform neural network analysis. Basically, there are some main steps to develop an efficient ANN model. These basic steps are given as follows.

Step #1

- Determining number of input and output neurons

Step #2

- Determining the network type

Step #3

- Selecting an appropriate training function

Step #4

- Designating the total number of hidden layers

Step #5

- Selecting optimum number of hidden layer neurons

Step #6

- Determining transfer functions for hidden layer / layers and output layer

At the beginning of ANN modelling part in this thesis, there was a main neural network model, but nearly all of the model parameters like number of input, hidden and output layer neurons, the number of hidden layers etc. were unknown as shown in figure 2.31 below.

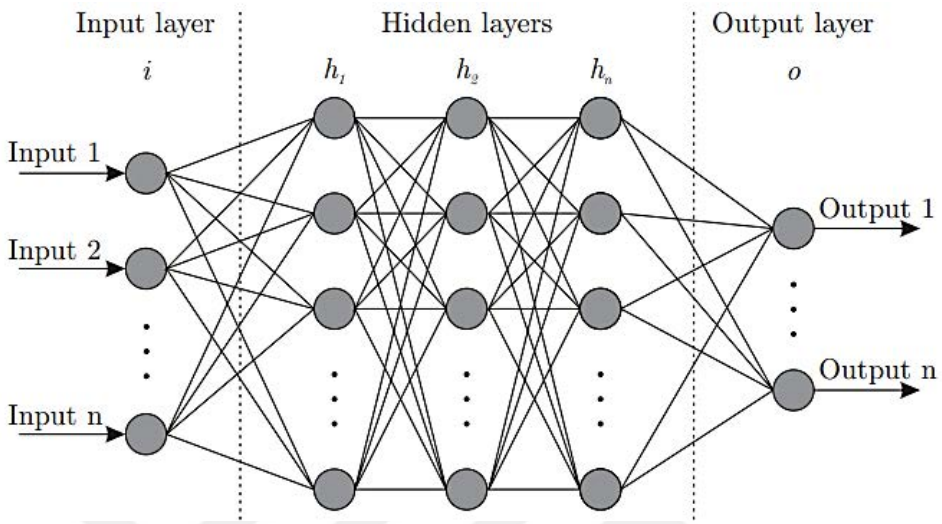


Figure 2.31. Basic model for a multi layered artificial neural network [26]

First step and the simplest step is related to number of input - output neurons. Number of input and output neurons depend on the problem. Number of input neuron is the variables used in the problem like thickness of the body, minor and major radius of the semi elliptical surface crack, etc. and number of output neuron is the parameters which was obtained at the end of the problem.

There were two different problems for this study. In the first case, there was a plate which had one semi elliptical surface crack at both sides, front side and back side. Variables in the first study were minor radius (a), ratio of minor radius to major radius (a/c), parametric angle (Θ) and ratio of minor radius to thickness of the plate (a/t). a/t was considered as a variable which effects SIF value. Because Jabur and Mohsin showed in their study [8] that SIF value for the plate which had two edge cracks in both sides (front and back) increased when thickness of the plate was decreased. Therefore, there were 4 variables and number of input neurons was 4 for the first case. Also number of output neuron was 1, since the only parameter which was desired to calculate was SIF value. Schematic representation of this model is shown in figure 2.32. But number of hidden layers and number of hidden layers' neurons were still unknown at this step.

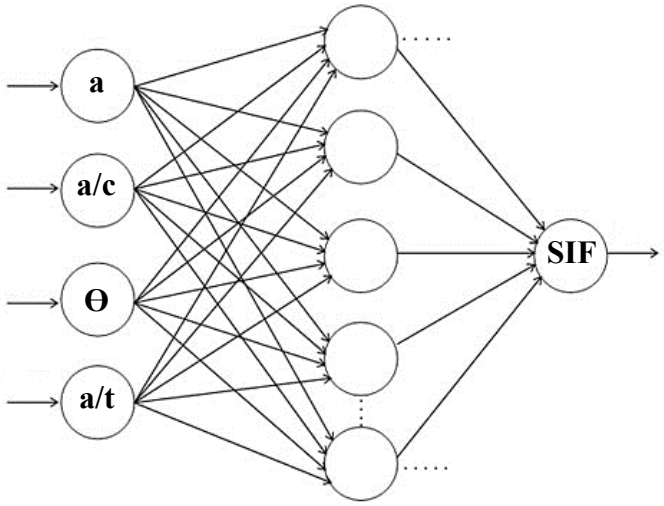


Figure 2.32. ANN model for the first problem at the first step of modelling

In the second case, there was a plate which had two semi elliptical cracks at both sides. In this case, four variables used in the first case were utilized again and an extra parameter was added to the input layer. This parameter was h (a/h was also a good choice but h was chosen) which was the vertical distance between two parallel cracks at the front or back side. It was added as a variable, because in Jabur and Mohsin's study [8], it was shown that in the case of parallel cracks, SIF reduced when h was decreased due to the mutual shielding effect. So, number of input layer neurons was 5 for the second case. Number of output neuron was again 1 like in the first case. As stated in the first case section, number of hidden layers and hidden layer neurons were still unknown at the first step of neural network modelling. Figure 2.33 shows the model used in the second case.

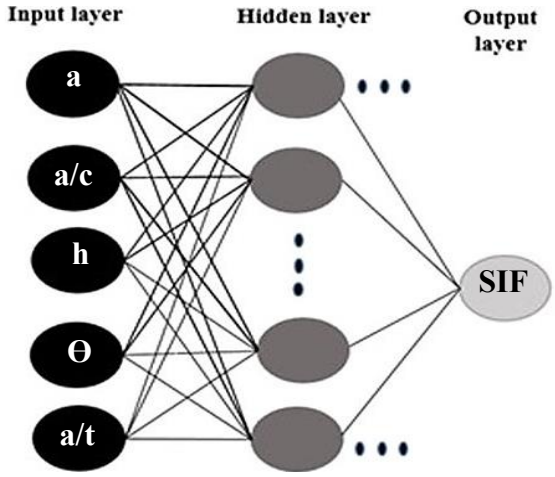


Figure 2.33. ANN model for the second case at the first step of modelling

The second step is determining the network type. There are different types of networks for ANN modelling. In this study, feed forward back propagation network type was used. In this type of network, first of all, feed forward process is executed, then back propagation operation is performed. This process array is repeated again and again for each epoch to obtain the best weights and best model. Feed forward network is formed of series of layers like input layer, hidden layer, output layer, etc. Layers have a connection to the next layer. Input data come to the first layer, then data go to next layer for instance hidden layer. Finally output layer generates the output of the network. In other words, data flows in one direction.

In backpropagation process, network error is calculated using the output of the network, which is the result of feed forward operation and the target value, then signal / data goes back to the input layer so as to adjust the weights of the input and hidden layers and reduce calculated error. This is the main procedure for feed forward back propagation network type. Figure 2.34 shows this procedure.

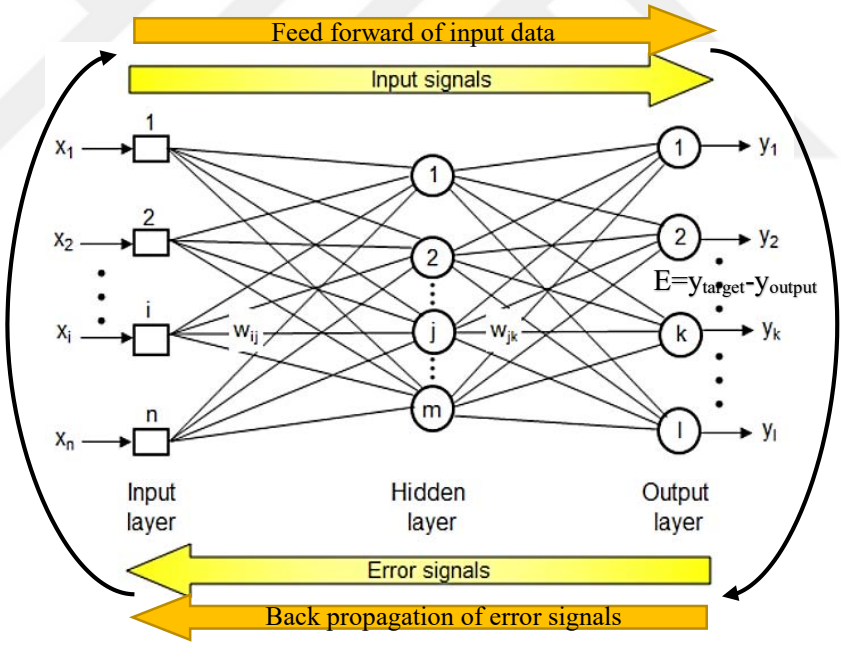


Figure 2.34. Feed forward back propagation neural network

The third step of neural network modelling is selecting the appropriate training function. For feed forward back propagation network type, there are different types of training functions like gradient descent with momentum back propagation, Levenberg Marquardt back propagation, gradient descent with momentum and adaptive learning rate backpropagation, resilient back propagation etc. It is not an easy job to know which

training function is appropriate for any study. This selection depends on some factors like problem complexity, number of data and corresponding weights and biases, accuracy of the result etc. Some of training functions are appropriate for pattern recognition problems and some of them are ideal for function approximation problems. Besides, some of training functions converge faster than others, get results more accurate than others, use less memory than others.

This study, estimation of SIF values of semi elliptical surfaces cracked plates, was a function approximation problem. So Levenberg Marquardt back propagation training function was used in neural network studies. Because usually for function approximation problems which have a few hundred total weights, Levenberg Marquardt back propagation type training function converges faster than other ones. Also Levenberg Marquardt back propagation algorithm gives the most accurate results. But this algorithm uses more memory than other ones. So if number of weights in the neural network model is so large, using Levenberg Marquardt back propagation algorithm necessitates more computer memory [27]. Main procedure which is executed in the background by Matlab neural network toolbox (nntool) for Levenberg Marquardt is shown in figure 2.35 below.

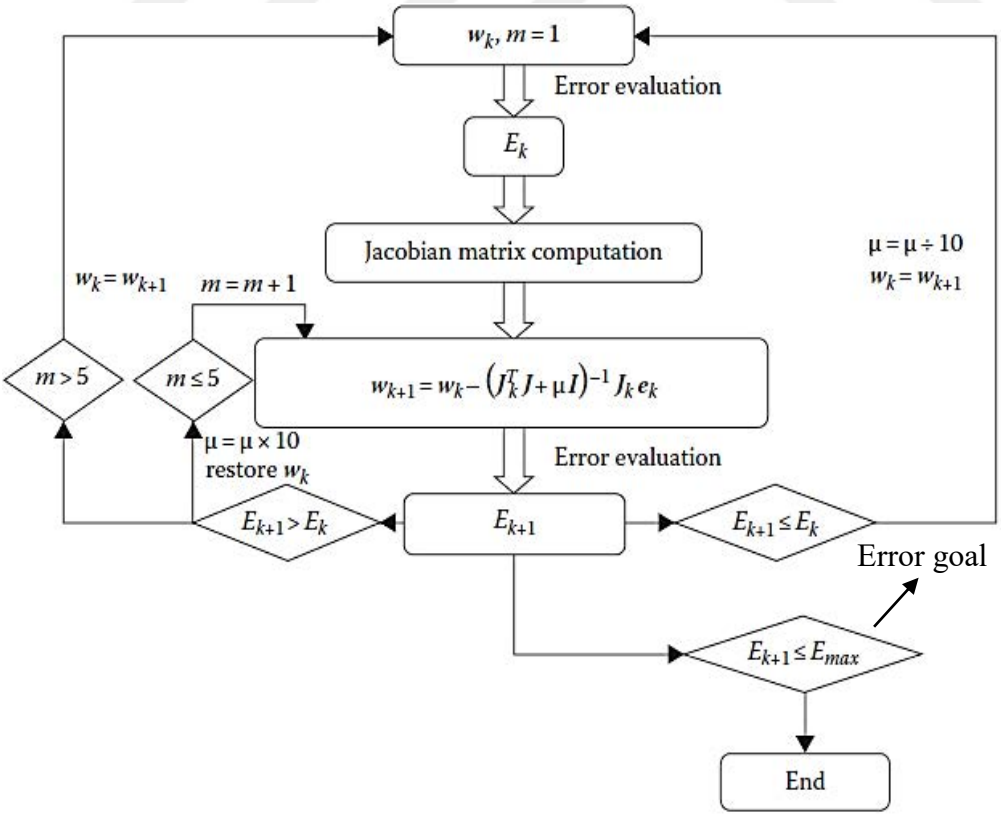


Figure 2.35. Levenberg – Marquardt training function flowchart [28]

In figure 2.35, w_k and w_{k+1} represent the current weight and the next weight, respectively. Also E_k and E_{k+1} represent the last total error and current total error of the network respectively. In Levenberg Marquardt back propagation algorithm, first of all, initial total error of the network is calculated using initial random weights. It is called as random weights because at the beginning of the training process, Matlab neural network toolbox assigns random values which are between 0 and 1 to all of the weights. As seen in flowchart above, then Jacobian matrix is formed. Jacobian matrix can be formed as follows.

$$J=\begin{bmatrix} \frac{\partial F(x_1,w)}{\partial w_1} & \cdots & \frac{\partial F(x_1,w)}{\partial w_W} \\ \vdots & \ddots & \vdots \\ \frac{\partial F(x_N,w)}{\partial w_1} & \cdots & \frac{\partial F(x_N,w)}{\partial w_W} \end{bmatrix}$$

This matrix is an A by B matrix. A is the number of input data set for training of the network and B is total number of weights. $F(x_i,w)$ is the network error function for i th input data vector. After forming Jacobian matrix, equation (2.24) is solved in order to calculate the weight update vector, δ and update all of the weights.

$$(J^TJ+\mu I)\delta=J^TE_k \qquad (2.24)$$

In this equation, parameter μ is known as damping factor or combination coefficient. It adjusts the step size in order to approach minimum error value. If its value is decreased, step size increases. Its default initial value is 0.001 in Matlab nntool and default value of its decrease and increase factor are 0.1 and 10 respectively. After updating weights, new total error is recalculated and new total error is compared with last total error value. If it is greater than last total error, damping factor is multiplied by increase factor, thus step size is decreased and weight update vector is calculated again. If new total error is smaller than last total error, damping factor is multiplied by decrease factor and new epoch starts.

Also in Matlab nntool, error goal value can be defined. As seen in figure 2.35, if new error value is smaller than error goal, training is stopped and this trained network is used for estimation of the desired output.

The forth step of neural network modelling study is designating the total number of hidden layers and the next step is selecting optimum number of hidden layer neurons.

Hidden layer increases the complexity of the model and it develops flexibility of the model. Besides, number of hidden layer neurons influences the model estimation capability of the network. But using many hidden layers or hidden layer neurons makes the analysis so difficult to solve and neural network analysis takes so much time. There is no certain rule for this determination process. So the best way to determine the number of hidden layers and corresponding neurons is the trial and error method. Neural network training is done using smaller number of hidden layer and hidden layer neuron and these values are increased gradually in this method. Then error values are compared with each other in order to determine the optimum number of hidden layers and corresponding neurons.

Training studies were done using different values of hidden layer and neuron. Optimum number of hidden layer - neuron was determined with regard to the total error values. Studies done in the forth and fifth steps of neural network modelling were explained in analysis and results section.

The final step for neural network modelling is determining the appropriate transfer function for hidden layer / layers and output layer. As explained in the previous chapters, there are different types of transfer functions, for instance, linear, logistic sigmoid, tangent hyperbolic transfer function, etc. Nonlinear transfer functions like logistic sigmoid or tangent hyperbolic are used for hidden layers in order to increase flexibility of the model. As seen in figure 2.19, in the theory of ANN section, the output of the logistic sigmoid (logsig) transfer function is between 0 and 1. If the input data for logistic sigmoid function is substantially negative, output of the hidden neuron which has logsig transfer function, become 0 and thus learning process almost stops [29]. But the output of tangent hyperbolic transfer function is between -1 and 1, this makes the model more balanced and it is reasonable to utilize tansig for all hidden layers [30]. Also tangent hyperbolic function has stronger gradient than logistic sigmoid function. So tangent hyperbolic transfer function was used for hidden layers of the model. But even so, logistic sigmoid could be a good option.

For output layer, linear activation function was used. Because linear output layer is commonly used for function approximation problems and there is no need to use a nonlinear activation function [30-31].

2.2.5. Data Generation with Finite Element Analysis (Ansys)

In this thesis, so many FEM studies were needed in order to train neural network successfully and estimate SIF values for case 1 (two semi elliptical cracks) and case 2 (four semi elliptical cracks) accurately. Because both cases were complicated for neural network modelling and training.

In the first case, variables of analysis were minor radius (a), ratio of minor radius to major radius (a/c) and ratio of minor radius to plate thickness (a/t). In the second case, these were minor radius (a), ratio of minor radius to major radius (a/c), ratio of minor radius to plate thickness (a/t) and vertical distance between the parallel cracks (h). 179 finite element simulations were done for case 1 and 523 simulations were done for case 2 using different values of foregoing variables. Table 2.10 and table 2.11 show the values of variables used in each analysis so as to generate training data for case 1 and case 2 respectively.

Table 2.10. Values of input variables used in Ansys for case 1 (179 simulations)

a(mm)	a/c	a/t
1,5	0,3	0,1
1,5	0,3	0,2
1,5	0,3	0,3
1,5	0,3	0,35
1,5	0,3	0,4
2,5	0,5	0,1
2,5	0,5	0,2
2,5	0,5	0,3
2,5	0,5	0,35
2,5	0,5	0,4
2,5	0,5	0,42
2,5	0,5	0,45
4	0,8	0,1
4	0,8	0,2
4	0,8	0,35
4	0,8	0,4
4	0,8	0,42
4	0,8	0,45
5	1	0,1
5	1	0,2
5	1	0,3
5	1	0,35
5	1	0,4
5	1	0,42

a(mm)	a/c	a/t
5	1	0,45
6	1,2	0,1
6	1,2	0,2
6	1,2	0,3
6	1,2	0,35
6	1,2	0,4
6	1,2	0,42
6	1,2	0,45
7,5	1,5	0,1
7,5	1,5	0,2
7,5	1,5	0,3
7,5	1,5	0,35
7,5	1,5	0,4
7,5	1,5	0,42
7,5	1,5	0,45
9	1,8	0,1
9	1,8	0,2
9	1,8	0,3
9	1,8	0,35
9	1,8	0,4
9	1,8	0,42
9	1,8	0,45
3	0,3	0,1
3	0,3	0,2

a(mm)	a/c	a/t
3	0,3	0,3
3	0,3	0,35
3	0,3	0,4
3	0,3	0,42
3	0,3	0,45
5	0,5	0,1
5	0,5	0,2
5	0,5	0,3
5	0,5	0,35
5	0,5	0,4
5	0,5	0,42
5	0,5	0,45
8	0,8	0,1
8	0,8	0,2
8	0,8	0,3
8	0,8	0,35
8	0,8	0,4
8	0,8	0,42
8	0,8	0,45
10	1	0,1
10	1	0,2
10	1	0,3
10	1	0,35
10	1	0,4

10	1	0,42
10	1	0,45
12	1,2	0,1
12	1,2	0,2
12	1,2	0,3
12	1,2	0,35
12	1,2	0,4
12	1,2	0,42
12	1,2	0,45
15	1,5	0,1
15	1,5	0,2
15	1,5	0,3
15	1,5	0,35
15	1,5	0,4
15	1,5	0,42
15	1,5	0,45
18	1,8	0,1
18	1,8	0,2
18	1,8	0,3
18	1,8	0,35
18	1,8	0,4
18	1,8	0,42
18	1,8	0,45
4,5	0,3	0,2
4,5	0,3	0,3
4,5	0,3	0,35
4,5	0,3	0,4
4,5	0,3	0,42
4,5	0,3	0,45
7,5	0,5	0,2
7,5	0,5	0,3
7,5	0,5	0,35
7,5	0,5	0,4
7,5	0,5	0,42
7,5	0,5	0,45
12	0,8	0,2
12	0,8	0,3
12	0,8	0,35
12	0,8	0,4
12	0,8	0,42
12	0,8	0,45
15	1	0,2
15	1	0,3
15	1	0,35
15	1	0,4
15	1	0,42
15	1	0,45
18	1,2	0,2
18	1,2	0,3
18	1,2	0,35
18	1,2	0,4
18	1,2	0,42
18	1,2	0,45
22,5	1,5	0,2
22,5	1,5	0,3
22,5	1,5	0,35
22,5	1,5	0,4
22,5	1,5	0,42
22,5	1,5	0,45
27	1,8	0,2
27	1,8	0,3
27	1,8	0,35
27	1,8	0,4
27	1,8	0,42
27	1,8	0,45
7,5	0,3	0,2
7,5	0,3	0,3
7,5	0,3	0,35
7,5	0,3	0,4
7,5	0,3	0,42
7,5	0,3	0,45
12,5	0,5	0,2
12,5	0,5	0,3
12,5	0,5	0,35
12,5	0,5	0,4
12,5	0,5	0,42
12,5	0,5	0,45
20	0,8	0,2
20	0,8	0,3
20	0,8	0,35
20	0,8	0,4
20	0,8	0,42
20	0,8	0,45
25	1	0,2
25	1	0,3
25	1	0,35
25	1	0,4
25	1	0,42
25	1	0,45
30	1,2	0,2
30	1,2	0,3
30	1,2	0,35
30	1,2	0,4
30	1,2	0,42
30	1,2	0,45
37,5	1,5	0,25
37,5	1,5	0,3
37,5	1,5	0,35
37,5	1,5	0,4
37,5	1,5	0,42
37,5	1,5	0,45
45	1,8	0,25
45	1,8	0,3
45	1,8	0,35
45	1,8	0,4
45	1,8	0,42
45	1,8	0,45

Table 2.11. Values of input variables used in Ansys for case 2 (523 simulations)

a mm	a/c	h - a/h mm		a/t
1,5	0,3	100	0,015	0,1
1,5	0,3	50	0,030	0,1
1,5	0,3	20	0,075	0,1
1,5	0,3	10	0,150	0,1
1,5	0,3	5	0,300	0,1
1,5	0,3	3	0,500	0,1
1,5	0,3	100	0,015	0,2
1,5	0,3	50	0,030	0,2
1,5	0,3	25	0,060	0,2
1,5	0,3	10	0,150	0,2
1,5	0,3	5	0,300	0,2
1,5	0,3	3	0,500	0,2
2,5	0,5	100	0,025	0,1
2,5	0,5	50	0,050	0,1
2,5	0,5	25	0,100	0,1
2,5	0,5	10	0,250	0,1
2,5	0,5	5	0,500	0,1
2,5	0,5	2,5	1,000	0,1

2,5	0,5	100	0,025	0,2
2,5	0,5	50	0,050	0,2
2,5	0,5	25	0,100	0,2
2,5	0,5	10	0,250	0,2
2,5	0,5	5	0,500	0,2
2,5	0,5	3	0,833	0,2
2,5	0,5	100	0,025	0,3
2,5	0,5	50	0,050	0,3
2,5	0,5	25	0,100	0,3
2,5	0,5	10	0,250	0,3
2,5	0,5	5	0,500	0,3
2,5	0,5	3	0,833	0,3
4	0,8	100	0,040	0,1
4	0,8	20	0,200	0,1
4	0,8	10	0,400	0,1
4	0,8	5	0,800	0,1
5	1	100	0,050	0,1
5	1	50	0,100	0,1
5	1	25	0,200	0,1
5	1	10	0,500	0,1
5	1	5	1,000	0,1
5	1	3	1,667	0,1
5	1	100	0,050	0,2
5	1	50	0,100	0,2
5	1	20	0,250	0,2
5	1	10	0,500	0,2
5	1	5	1,000	0,2
5	1	3	1,667	0,2
5	1	100	0,050	0,3
5	1	50	0,100	0,3
5	1	20	0,250	0,3
5	1	10	0,500	0,3
5	1	5	1,000	0,3
5	1	3	1,667	0,3
5	1	100	0,050	0,4
5	1	50	0,100	0,4
5	1	20	0,250	0,4
5	1	10	0,500	0,4
5	1	5	1,000	0,4
5	1	3	1,667	0,4
5	1	100	0,050	0,42
5	1	50	0,100	0,42
5	1	20	0,250	0,42
5	1	10	0,500	0,42
5	1	5	1,000	0,42
5	1	3	1,667	0,42
5	1	100	0,050	0,45
5	1	50	0,100	0,45

5	1	20	0,250	0,45
5	1	10	0,500	0,45
5	1	5	1,000	0,45
5	1	3	1,667	0,45
6	1,2	100	0,060	0,1
6	1,2	50	0,120	0,1
6	1,2	20	0,300	0,1
6	1,2	10	0,600	0,1
6	1,2	5	1,200	0,1
6	1,2	3	2,000	0,1
6	1,2	100	0,060	0,2
6	1,2	50	0,120	0,2
6	1,2	20	0,300	0,2
6	1,2	10	0,600	0,2
6	1,2	5	1,200	0,2
6	1,2	3	2,000	0,2
6	1,2	100	0,060	0,3
6	1,2	50	0,120	0,3
6	1,2	20	0,300	0,3
6	1,2	10	0,600	0,3
6	1,2	5	1,200	0,3
6	1,2	3	2,000	0,3
6	1,2	100	0,060	0,4
6	1,2	50	0,120	0,4
6	1,2	20	0,300	0,4
6	1,2	10	0,600	0,4
6	1,2	5	1,200	0,4
6	1,2	3	2,000	0,4
6	1,2	100	0,060	0,42
6	1,2	50	0,120	0,42
6	1,2	20	0,300	0,42
6	1,2	10	0,600	0,42
6	1,2	5	1,200	0,42
6	1,2	3	2,000	0,42
6	1,2	100	0,060	0,45
6	1,2	50	0,120	0,45
6	1,2	20	0,300	0,45
6	1,2	10	0,600	0,45
6	1,2	5	1,200	0,45
6	1,2	3	2,000	0,45
9	1,8	100	0,090	0,1
9	1,8	50	0,180	0,1
9	1,8	25	0,360	0,1
9	1,8	10	0,900	0,1
9	1,8	5	1,800	0,1
9	1,8	3	3,000	0,1
9	1,8	100	0,090	0,2
9	1,8	50	0,180	0,2

9	1,8	20	0,450	0,2
9	1,8	10	0,900	0,2
9	1,8	5	1,800	0,2
9	1,8	3	3,000	0,2
9	1,8	100	0,090	0,3
9	1,8	50	0,180	0,3
9	1,8	20	0,450	0,3
9	1,8	10	0,900	0,3
9	1,8	5	1,800	0,3
9	1,8	3	3,000	0,3
9	1,8	100	0,090	0,4
9	1,8	50	0,180	0,4
9	1,8	20	0,450	0,4
9	1,8	10	0,900	0,4
9	1,8	5	1,800	0,4
9	1,8	3	3,000	0,4
9	1,8	100	0,090	0,42
9	1,8	50	0,180	0,42
9	1,8	20	0,450	0,42
9	1,8	10	0,900	0,42
9	1,8	5	1,800	0,42
9	1,8	3	3,000	0,42
9	1,8	100	0,090	0,45
9	1,8	50	0,180	0,45
9	1,8	20	0,450	0,45
9	1,8	10	0,900	0,45
9	1,8	5	1,800	0,45
9	1,8	3	3,000	0,45
3	0,3	100	0,030	0,1
3	0,3	50	0,060	0,1
3	0,3	20	0,150	0,1
3	0,3	10	0,300	0,1
3	0,3	5	0,600	0,1
3	0,3	3	1,000	0,1
3	0,3	100	0,030	0,2
3	0,3	50	0,060	0,2
3	0,3	25	0,120	0,2
3	0,3	20	0,150	0,2
3	0,3	10	0,300	0,2
3	0,3	5	0,600	0,2
3	0,3	3	1,000	0,2
8	0,8	100	0,080	0,1
8	0,8	50	0,160	0,1
8	0,8	20	0,400	0,1
8	0,8	10	0,800	0,1
8	0,8	5	1,600	0,1
8	0,8	3	2,667	0,1
8	0,8	100	0,080	0,2

8	0,8	50	0,160	0,2
8	0,8	20	0,400	0,2
8	0,8	10	0,800	0,2
8	0,8	5	1,600	0,2
8	0,8	3	2,667	0,2
8	0,8	100	0,080	0,3
8	0,8	50	0,160	0,3
8	0,8	20	0,400	0,3
8	0,8	10	0,800	0,3
8	0,8	5	1,600	0,3
8	0,8	100	0,080	0,4
8	0,8	50	0,160	0,4
8	0,8	20	0,400	0,4
8	0,8	10	0,800	0,4
8	0,8	5	1,600	0,4
8	0,8	3	2,667	0,4
8	0,8	100	0,080	0,42
8	0,8	50	0,160	0,42
8	0,8	20	0,400	0,42
8	0,8	10	0,800	0,42
8	0,8	5	1,600	0,42
8	0,8	3	2,667	0,42
8	0,8	100	0,080	0,45
8	0,8	50	0,160	0,45
8	0,8	20	0,400	0,45
8	0,8	10	0,800	0,45
8	0,8	5	1,600	0,45
8	0,8	3	2,667	0,45
10	1	100	0,100	0,1
10	1	50	0,200	0,1
10	1	20	0,500	0,1
10	1	10	1,000	0,1
10	1	5	2,000	0,1
10	1	3	3,333	0,1
10	1	100	0,100	0,2
10	1	50	0,200	0,2
10	1	20	0,500	0,2
10	1	10	1,000	0,2
10	1	5	2,000	0,2
10	1	3	3,333	0,2
10	1	100	0,100	0,3
10	1	50	0,200	0,3
10	1	30	0,333	0,3
10	1	20	0,500	0,3
10	1	10	1,000	0,3
10	1	5	2,000	0,3
10	1	100	0,100	0,4
10	1	50	0,200	0,4

10	1	27,5	0,364	0,4
10	1	17,5	0,571	0,4
10	1	10	1,000	0,4
10	1	5	2,000	0,4
10	1	100	0,100	0,42
10	1	50	0,200	0,42
10	1	25	0,400	0,42
10	1	17,5	0,571	0,42
10	1	10	1,000	0,42
10	1	5	2,000	0,42
10	1	100	0,100	0,45
10	1	50	0,200	0,45
10	1	25	0,400	0,45
10	1	17,5	0,571	0,45
10	1	10	1,000	0,45
10	1	5	2,000	0,45
15	1,5	100	0,150	0,1
15	1,5	50	0,300	0,1
15	1,5	25	0,600	0,1
15	1,5	17,5	0,857	0,1
15	1,5	10	1,500	0,1
15	1,5	5	3,000	0,1
15	1,5	100	0,150	0,2
15	1,5	50	0,300	0,2
15	1,5	25	0,600	0,2
15	1,5	17,5	0,857	0,2
15	1,5	10	1,500	0,2
15	1,5	5	3,000	0,2
15	1,5	100	0,150	0,3
15	1,5	50	0,300	0,3
15	1,5	25	0,600	0,3
15	1,5	17,5	0,857	0,3
15	1,5	10	1,500	0,3
15	1,5	5	3,000	0,3
15	1,5	3	5,000	0,3
15	1,5	100	0,150	0,4
15	1,5	50	0,300	0,4
15	1,5	25	0,600	0,4
15	1,5	17,5	0,857	0,4
15	1,5	10	1,500	0,4
15	1,5	5	3,000	0,4
15	1,5	3	5,000	0,4
15	1,5	100	0,150	0,45
15	1,5	50	0,300	0,45
15	1,5	25	0,600	0,45
15	1,5	17,5	0,857	0,45
15	1,5	10	1,500	0,45
15	1,5	5	3,000	0,45

15	1,5	3	5,000	0,45
18	1,8	100	0,180	0,1
18	1,8	50	0,360	0,1
18	1,8	25	0,720	0,1
18	1,8	17,5	1,029	0,1
18	1,8	10	1,800	0,1
18	1,8	5	3,600	0,1
18	1,8	3	6,000	0,1
18	1,8	100	0,180	0,2
18	1,8	50	0,360	0,2
18	1,8	25	0,720	0,2
18	1,8	17,5	1,029	0,2
18	1,8	10	1,800	0,2
18	1,8	5	3,600	0,2
18	1,8	100	0,180	0,3
18	1,8	50	0,360	0,3
18	1,8	25	0,720	0,3
18	1,8	17,5	1,029	0,3
18	1,8	10	1,800	0,3
18	1,8	5	3,600	0,3
18	1,8	3	6,000	0,3
18	1,8	100	0,180	0,4
18	1,8	50	0,360	0,4
18	1,8	25	0,720	0,4
18	1,8	17,5	1,029	0,4
18	1,8	10	1,800	0,4
18	1,8	5	3,600	0,4
18	1,8	100	0,180	0,42
18	1,8	50	0,360	0,42
18	1,8	25	0,720	0,42
18	1,8	17,5	1,029	0,42
18	1,8	10	1,800	0,42
18	1,8	5	3,600	0,42
18	1,8	100	0,180	0,45
18	1,8	50	0,360	0,45
18	1,8	25	0,720	0,45
18	1,8	17,5	1,029	0,45
18	1,8	10	1,800	0,45
18	1,8	5	3,600	0,45
18	1,8	3	6,000	0,45
18	1,8	2,5	7,200	0,45
4,5	0,3	100	0,045	0,2
4,5	0,3	50	0,090	0,2
4,5	0,3	20	0,225	0,2
4,5	0,3	10	0,450	0,2
4,5	0,3	5	0,900	0,2
4,5	0,3	3	1,500	0,2
4,5	0,3	100	0,045	0,3

4,5	0,3	50	0,090	0,3
4,5	0,3	20	0,225	0,3
4,5	0,3	10	0,450	0,3
4,5	0,3	5	0,900	0,3
4,5	0,3	3	1,500	0,3
7,5	0,5	100	0,075	0,2
7,5	0,5	50	0,150	0,2
7,5	0,5	20	0,375	0,2
7,5	0,5	10	0,750	0,2
7,5	0,5	5	1,500	0,2
7,5	0,5	3	2,500	0,2
7,5	0,5	100	0,075	0,3
7,5	0,5	50	0,150	0,3
7,5	0,5	20	0,375	0,3
7,5	0,5	10	0,750	0,3
7,5	0,5	5	1,500	0,3
7,5	0,5	3	2,500	0,3
7,5	0,5	100	0,075	0,4
7,5	0,5	50	0,150	0,4
7,5	0,5	25	0,300	0,4
7,5	0,5	17,5	0,429	0,4
7,5	0,5	10	0,750	0,4
7,5	0,5	5	1,500	0,4
7,5	0,5	100	0,075	0,42
7,5	0,5	50	0,150	0,42
7,5	0,5	25	0,300	0,42
7,5	0,5	17,5	0,429	0,42
7,5	0,5	10	0,750	0,42
7,5	0,5	5	1,500	0,42
7,5	0,5	3	2,500	0,42
7,5	0,5	100	0,075	0,45
7,5	0,5	50	0,150	0,45
7,5	0,5	25	0,300	0,45
7,5	0,5	17,5	0,429	0,45
7,5	0,5	10	0,750	0,45
7,5	0,5	5	1,500	0,45
15	1	100	0,150	0,2
15	1	50	0,300	0,2
15	1	25	0,600	0,2
15	1	17,5	0,857	0,2
15	1	10	1,500	0,2
15	1	5	3,000	0,2
15	1	100	0,150	0,3
15	1	50	0,300	0,3
15	1	25	0,600	0,3
15	1	17,5	0,857	0,3
15	1	7,5	2,000	0,3
15	1	3	5,000	0,3
15	1	100	0,150	0,4
15	1	50	0,300	0,4
15	1	25	0,600	0,4
15	1	17,5	0,857	0,4
15	1	10	1,500	0,4
15	1	5	3,000	0,4
15	1	100	0,150	0,45
15	1	40	0,375	0,45
15	1	25	0,600	0,45
15	1	12,5	1,200	0,45
15	1	6	2,500	0,45
15	1	3	5,000	0,45
18	1,2	100	0,180	0,2
18	1,2	50	0,360	0,2
18	1,2	25	0,720	0,2
18	1,2	15	1,200	0,2
18	1,2	7,5	2,400	0,2
18	1,2	3	6,000	0,2
18	1,2	100	0,180	0,3
18	1,2	50	0,360	0,3
18	1,2	25	0,720	0,3
18	1,2	15	1,200	0,3
18	1,2	7,5	2,400	0,3
18	1,2	3	6,000	0,3
18	1,2	100	0,180	0,4
18	1,2	50	0,360	0,4
18	1,2	30	0,600	0,4
18	1,2	17,5	1,029	0,4
18	1,2	10	1,800	0,4
18	1,2	5	3,600	0,4
18	1,2	120	0,150	0,45
18	1,2	60	0,300	0,45
18	1,2	30	0,600	0,45
18	1,2	18	1,000	0,45
18	1,2	10	1,800	0,45
18	1,2	3	6,000	0,45
27	1,8	100	0,270	0,2
27	1,8	50	0,540	0,2
27	1,8	27,5	0,982	0,2
27	1,8	17,5	1,543	0,2
27	1,8	10	2,700	0,2
27	1,8	3	9,000	0,2
27	1,8	100	0,270	0,3
27	1,8	50	0,540	0,3
27	1,8	25	1,080	0,3
27	1,8	15	1,800	0,3
27	1,8	7,5	3,600	0,3
27	1,8	3	9,000	0,3
27	1,8	120	0,225	0,4
27	1,8	100	0,270	0,4
27	1,8	60	0,450	0,4
27	1,8	27,5	0,982	0,4
27	1,8	15	1,800	0,4
27	1,8	5	5,400	0,4
27	1,8	3	9,000	0,4
27	1,8	120	0,225	0,45
27	1,8	50	0,540	0,45
27	1,8	27,5	0,982	0,45
27	1,8	17,5	1,543	0,45
27	1,8	10	2,700	0,45
27	1,8	4	6,750	0,45
6	0,3	120	0,050	0,2
6	0,3	50	0,120	0,2
6	0,3	27,5	0,218	0,2
6	0,3	15	0,400	0,2
6	0,3	7,5	0,800	0,2
6	0,3	3	2,000	0,2
6	0,3	120	0,050	0,3
6	0,3	50	0,120	0,3
6	0,3	20	0,300	0,3
6	0,3	10	0,600	0,3
6	0,3	5	1,200	0,3
6	0,3	3	2,000	0,3
6	0,3	120	0,050	0,4
6	0,3	50	0,120	0,4
6	0,3	20	0,300	0,4
6	0,3	10	0,600	0,4
6	0,3	5	1,200	0,4
6	0,3	3	2,000	0,4
10	0,5	120	0,083	0,4
10	0,5	50	0,200	0,4
10	0,5	20	0,500	0,4
10	0,5	10	1,000	0,4
10	0,5	5	2,000	0,4
10	0,5	3	3,333	0,4
16	0,8	120	0,133	0,2
16	0,8	50	0,320	0,2
16	0,8	25	0,640	0,2
16	0,8	12,5	1,280	0,2
16	0,8	5	3,200	0,2
16	0,8	3	5,333	0,2
16	0,8	120	0,133	0,3
16	0,8	50	0,320	0,3
16	0,8	27,5	0,582	0,3
16	0,8	15	1,067	0,3
16	0,8	6	2,667	0,3

16	0,8	3	5,333	0,3
16	0,8	150	0,107	0,42
16	0,8	60	0,267	0,42
16	0,8	27,5	0,582	0,42
16	0,8	15	1,067	0,42
16	0,8	7,5	2,133	0,42
16	0,8	3	5,333	0,42
16	0,8	150	0,107	0,45
16	0,8	80	0,200	0,45
16	0,8	30	0,533	0,45
16	0,8	17,5	0,914	0,45
16	0,8	10	1,600	0,45
16	0,8	5	3,200	0,45
20	1	150	0,133	0,3
20	1	60	0,333	0,3
20	1	30	0,667	0,3
20	1	15	1,333	0,3
20	1	7,5	2,667	0,3
20	1	3	6,667	0,3
20	1	150	0,133	0,4
20	1	75	0,267	0,4
20	1	30	0,667	0,4
20	1	17,5	1,143	0,4
20	1	10	2,000	0,4
20	1	5	4,000	0,4

20	1	150	0,133	0,45
20	1	50	0,400	0,45
20	1	25	0,800	0,45
20	1	15	1,333	0,45
20	1	10	2,000	0,45
20	1	4	5,000	0,45
30	1,5	150	0,200	0,2
30	1,5	60	0,500	0,2
30	1,5	30	1,000	0,2
30	1,5	17,5	1,714	0,2
30	1,5	7,5	4,000	0,2
30	1,5	3	10,00	0,2
30	1,5	150	0,200	0,3
30	1,5	60	0,500	0,3
30	1,5	27,5	1,091	0,3
30	1,5	17,5	1,714	0,3
30	1,5	10	3,000	0,3
30	1,5	4	7,500	0,3
30	1,5	150	0,200	0,4
30	1,5	60	0,500	0,4
30	1,5	27,5	1,091	0,4
30	1,5	17,5	1,714	0,4
30	1,5	10	3,000	0,4
30	1,5	4	7,500	0,4
30	1,5	120	0,250	0,45

30	1,5	60	0,500	0,45
30	1,5	32,5	0,923	0,45
30	1,5	20	1,500	0,45
30	1,5	12	2,500	0,45
30	1,5	5	6,000	0,45
36	1,8	150	0,240	0,3
36	1,8	60	0,600	0,3
36	1,8	35	1,029	0,3
36	1,8	20	1,800	0,3
36	1,8	9	4,000	0,3
36	1,8	3	12,00	0,3
36	1,8	150	0,240	0,4
36	1,8	80	0,450	0,4
36	1,8	40	0,900	0,4
36	1,8	25	1,440	0,4
36	1,8	12,5	2,880	0,4
36	1,8	5	7,200	0,4
36	1,8	150	0,240	0,45
36	1,8	80	0,450	0,45
36	1,8	40	0,900	0,45
36	1,8	25	1,440	0,45
36	1,8	10	3,600	0,45
36	1,8	4	9,000	0,45

At the end of each analysis, 91 SIFs and corresponding coordinate values were obtained. Using x, y and z coordinate points, parametric angle was calculated for each coordinate and its value was 0 at the beginning of the semi elliptical crack tip and 180 at the end of the crack tip. Actually SIF value for 0° and 180°, 10° and 170° etc. were almost the same due to the symmetry. So the results of half of the crack (from 0° to 90° or 90° to 180°) were used in neural network process.

As mentioned before, in this thesis, it was considered that semi elliptical crack was so small compared to the plate dimensions. That is to say, SIF of the crack was not dependent to the dimensions of the plate. So in finite element studies, simulations were done using different values of plate dimensions (width and length) and results of these simulations were compared with each others. For instance, width and length of plate were taken as 20 cm. (In Ansys studies, the minimum value of the dimension was taken as 20 cm). Then same analysis was done using the value of width and length of plate as 30 cm and 50 cm. After these simulations, outputs of the simulations for 20 cm and 30 cm were checked against analysis for 50 cm. When the difference between analysis for 20 cm and 50 cm

was smaller than 0.3% more or less, value of width and length of the plate was considered as 20 cm or when the difference was greater than nearly 0.3%, check for 25, 30, 35, 40 cm etc. against 50 cm plate were done and the most reasonable value for plate dimension was selected. The reason why these comparisons were made was that using greater values for plate dimensions made the analysis more tedious job. For example, analysis for 40 cm or 50 cm took time three times more than analysis for 20 cm. So, a deviation threshold was determined (0.3% more or less) and optimum plate dimension was used in simulations. Table 2.12 shows an example for comparison study (a= 0,016 m, a/c= 0,8, a/t= 0,2). As a result of study in table 2.12, 30x30 cm plate was used in FEA.

Table 2.12. A sample study for determination of plate dimension

a/t=0,2				50x50 KI Pa√m	50x50 KI Norm.	Deviation %		25x25 KI Pa√m	25x25 KI Norm.	30x30 KI Pa√m	30x30 KI Norm.
a (m)	c (m)	a/c	Angle (°)			30x30 Plate	25x25 Plate				
0,016	0,020	0,8	0,00	161070	1,019	0,186	0,323	161590	1,023	161370	1,021
0,016	0,020	0,8	2,26	162410	1,028	0,185	0,320	162930	1,031	162710	1,030
0,016	0,020	0,8	4,51	162290	1,027	0,179	0,320	162810	1,030	162580	1,029
0,016	0,020	0,8	6,76	160890	1,018	0,168	0,311	161390	1,021	161160	1,020
0,016	0,020	0,8	9,01	160210	1,014	0,169	0,300	160690	1,017	160480	1,016
0,016	0,020	0,8	11,24	159020	1,006	0,182	0,302	159500	1,009	159310	1,008
0,016	0,020	0,8	13,47	158430	1,003	0,202	0,316	158930	1,006	158750	1,005
0,016	0,020	0,8	15,69	157580	0,997	0,222	0,324	158090	1,000	157930	0,999
0,016	0,020	0,8	17,90	157290	0,995	0,216	0,318	157790	0,999	157630	0,998
0,016	0,020	0,8	20,09	156800	0,992	0,210	0,300	157270	0,995	157130	0,994
0,016	0,020	0,8	22,27	156760	0,992	0,211	0,268	157180	0,995	157090	0,994
0,016	0,020	0,8	24,43	156530	0,991	0,204	0,236	156900	0,993	156850	0,993
0,016	0,020	0,8	26,58	156660	0,991	0,191	0,236	157030	0,994	156960	0,993
0,016	0,020	0,8	28,71	156600	0,991	0,172	0,249	156990	0,993	156870	0,993
0,016	0,020	0,8	30,83	156870	0,993	0,159	0,249	157260	0,995	157120	0,994
0,016	0,020	0,8	32,92	156970	0,993	0,147	0,248	157360	0,996	157200	0,995
0,016	0,020	0,8	35,00	157310	0,996	0,153	0,267	157730	0,998	157550	0,997
0,016	0,020	0,8	37,07	157480	0,997	0,165	0,286	157930	0,999	157740	0,998
0,016	0,020	0,8	39,12	157860	0,999	0,184	0,304	158340	1,002	158150	1,001
0,016	0,020	0,8	41,15	158080	1,000	0,190	0,310	158570	1,003	158380	1,002
0,016	0,020	0,8	43,17	158490	1,003	0,170	0,315	158990	1,006	158760	1,005
0,016	0,020	0,8	45,16	158760	1,005	0,145	0,321	159270	1,008	158990	1,006
0,016	0,020	0,8	47,15	159190	1,007	0,151	0,333	159720	1,011	159430	1,009
0,016	0,020	0,8	49,12	159500	1,009	0,144	0,332	160030	1,013	159730	1,011
0,016	0,020	0,8	51,08	159950	1,012	0,163	0,338	160490	1,016	160210	1,014
0,016	0,020	0,8	53,02	160270	1,014	0,175	0,343	160820	1,018	160550	1,016
0,016	0,020	0,8	54,95	160670	1,017	0,205	0,361	161250	1,020	161000	1,019
0,016	0,020	0,8	56,87	160950	1,019	0,217	0,367	161540	1,022	161300	1,021

0,016	0,020	0,8	58,77	161350	1,021	0,211	0,347	161910	1,025	161690	1,023
0,016	0,020	0,8	60,67	161630	1,023	0,204	0,328	162160	1,026	161960	1,025
0,016	0,020	0,8	62,55	162020	1,025	0,179	0,302	162510	1,028	162310	1,027
0,016	0,020	0,8	64,42	162290	1,027	0,166	0,290	162760	1,030	162560	1,029
0,016	0,020	0,8	66,29	162650	1,029	0,141	0,289	163120	1,032	162880	1,031
0,016	0,020	0,8	68,14	162900	1,031	0,123	0,289	163370	1,034	163100	1,032
0,016	0,020	0,8	70,00	163200	1,033	0,129	0,270	163640	1,036	163410	1,034
0,016	0,020	0,8	71,84	163380	1,034	0,135	0,251	163790	1,037	163600	1,035
0,016	0,020	0,8	73,68	163610	1,035	0,141	0,238	164000	1,038	163840	1,037
0,016	0,020	0,8	75,49	163720	1,036	0,165	0,250	164130	1,039	163990	1,038
0,016	0,020	0,8	77,32	163930	1,037	0,171	0,262	164360	1,040	164210	1,039
0,016	0,020	0,8	79,15	164020	1,038	0,171	0,287	164490	1,041	164300	1,040
0,016	0,020	0,8	80,95	164180	1,039	0,158	0,305	164680	1,042	164440	1,041
0,016	0,020	0,8	82,78	164230	1,039	0,146	0,311	164740	1,043	164470	1,041
0,016	0,020	0,8	84,56	164360	1,040	0,140	0,292	164840	1,043	164590	1,042
0,016	0,020	0,8	86,38	164380	1,040	0,140	0,280	164840	1,043	164610	1,042
0,016	0,020	0,8	88,19	164430	1,041	0,176	0,304	164930	1,044	164720	1,042
0,016	0,020	0,8	90,00	164390	1,040	0,195	0,316	164910	1,044	164710	1,042
Deviation (%) according to 50x50						0,173	0,298				

2.2.6. Training of Artificial Neural Network Model

Data generated using Ansys Static Structural were used to train the ANN model. As explained in ANN model development section, feed forward back propagation network type and Levenberg Marquardt training function were used in the model. Also tangent hyperbolic and linear transfer function were utilized for hidden and output layers respectively.

In case 1, 179 simulations were done in Ansys and 8234 different data were obtained since each analysis had 46 different parametric angles changing from 0 to 90°. Similarly, 523 simulations were done in Ansys for case 2 and 24058 different data were obtained to train the ANN model. Although there were 8234 data for case 1 and 24058 data for case 2, a certain part of these data were used for the training process. Because overfitting is an important problem in ANN training process and it must be taken into consideration. Overfitting is the case that trained model estimate the output, which is given to the network successfully, but when new data are given to the network, error value becomes so large. In other words, the ANN model memorizes the presented data and does not have the ability to generalize the new data [32]. The difference between overfitted and good fit model is shown in figure 2.36 as follows.

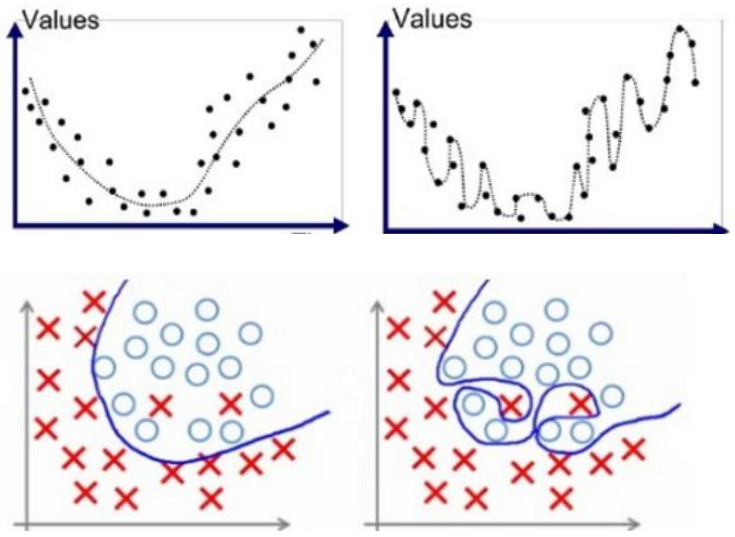


Figure 2.36. Difference between good fit (a) and overfitted (b) ANN model [33]

If too many same kinds of data are presented to the network in order to train it, overfitting problem may occur. Moreover, if very few data are presented to the network, a new problem which is called underfitting may occur. Underfitting can also occur if the model is not complex enough to learn the relationships successfully. Underfitted network can estimate neither the given data nor new data. Figure 2.37 shows an underfitted model.

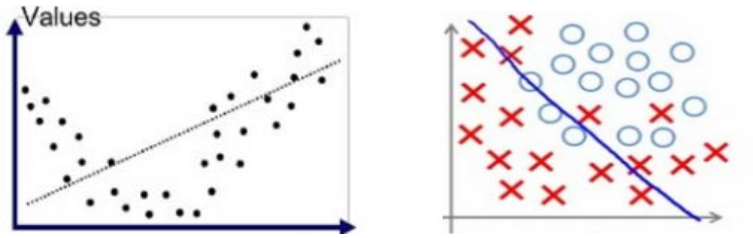


Figure 2.37. An underfitted model [33]

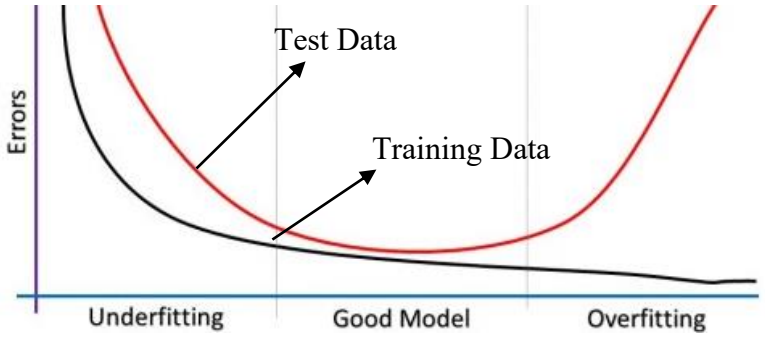


Figure 2.38. Variation of mean errors for training and test data [34]

In the light of information provided above, 1061 data were selected from 8234 data for case 1 and 4248 data were selected from 24058 data.

Before training process, all of these data must be scaled. This is known as normalization of the data. It is very important for the ANN learning process. It brings the data to a specific range like from 0 to 1, -1 to 1, -0.9 to 0.9 etc. in order to regulate importance of each input. It makes the training process faster, more efficient and reduces the complexity of the model since as a result of normalization, small values are used instead of large values. In general, simulation or experiment data range from value too small to value too large. So without normalization, larger values become dominant during training. Also in some cases, model cannot distinguish the effect of inputs on the output. For example, if one of the input is 10, the other one is 1000 and transfer function is tangent hyperbolic, results of both inputs become 1. So the network cannot see the difference between these data. In this thesis, tangent hyperbolic transfer function was used for hidden layers. A graph of the tangent hyperbolic function is given in figure 2.39.

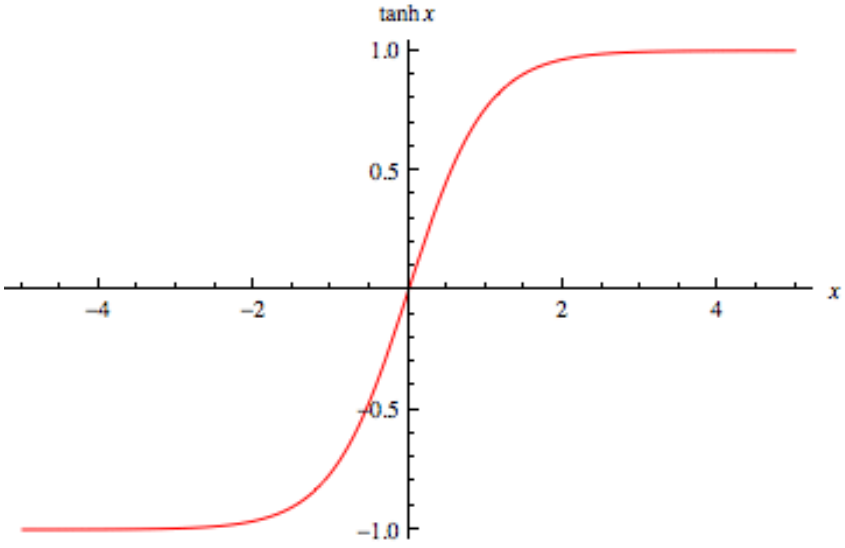


Figure 2.39. Graph of the tangent hyperbolic function

As seen in figure 2.39 above, tangent hyperbolic function produces the result ranges -1 to 1. Also when x is between nearly -0.9 and 0.9 or -1 and 1, change in the function is relatively high and it makes the training process faster and easier. Due to these reasons, all training data were normalized between -0.9 and 0.9 before training process in order to ease mathematical operations and get results faster. General equation for normalization and equation used in this thesis are given in Eq. (2.25) and Eq. (2.26) as follows.

$$x_{norm}=(b-a)(x-x_{min})/(x_{max}-x_{min})+a \quad (2.25)$$

$$x_{norm}=1.8(x-x_{min})/(x_{max}-x_{min})-0.9 \quad (2.26)$$

2.2.7. Testing of Artificial Neural Network Model

Testing is an important part of the ANN modelling process. So model obtained after training process must be checked using new data. Because in some cases, model estimates the desired value accurately for given values, but when it comes to new data, model does not perform good performance. Therefore, extra simulations were done to check the accuracy of the trained model for case 1 and case 2. In case 1, 118 extra simulations were done and accuracy of the trained model was checked using 760 new data (almost 7 angle for each analysis). Also in case 2, 25 extra simulations were done and 1139 new data were used for checking the model (almost 7 parametric angle for each analysis). Values of variables used for extra simulations are presented in table 2.13 and 2.14.

Table 2.13. Simulations done for testing the trained model (case 1)

a(mm)	a/c	a/t
1,5	0,5	0,2
42,5	2,5	0,41
44,2	1,3	0,33
12	0,4	0,47
8,8	1,1	0,15
3,125	0,25	0,44
50	2	0,25
18,9	0,7	0,38
1,5	0,5	0,25
1,5	0,5	0,3
1,5	0,5	0,35
1,5	0,5	0,4
1,5	0,5	0,42
1,5	0,5	0,45
2,4	0,8	0,2
2,4	0,8	0,25
2,4	0,8	0,3
2,4	0,8	0,35
2,4	0,8	0,4
2,4	0,8	0,42
2,4	0,8	0,45
3	1	0,2
3	1	0,25
3	1	0,3
3	1	0,35
3	1	0,4
3	1	0,42
3	1	0,45
3,6	1,2	0,2
3,6	1,2	0,25
3,6	1,2	0,3
3,6	1,2	0,35
3,6	1,2	0,4
3,6	1,2	0,42
3,6	1,2	0,45
4,5	1,5	0,2
4,5	1,5	0,25
4,5	1,5	0,3
4,5	1,5	0,35
4,5	1,5	0,4
4,5	1,5	0,42
4,5	1,5	0,45
5,4	1,8	0,2
5,4	1,8	0,25
5,4	1,8	0,3
5,4	1,8	0,35
5,4	1,8	0,4
5,4	1,8	0,42
5,4	1,8	0,45
1,5	0,3	0,25
2,5	0,5	0,25
4	0,8	0,25
5	1	0,25
7,5	1,5	0,25
3	0,3	0,25
5	0,5	0,25
10	1	0,25
12	1,2	0,25
18	1,8	0,25
4,5	0,3	0,25
12	0,8	0,25
15	1	0,25
27	1,8	0,25
6	0,3	0,2
6	0,3	0,25
6	0,3	0,3
6	0,3	0,35
6	0,3	0,4
6	0,3	0,42
6	0,3	0,45
10	0,5	0,2
10	0,5	0,25
10	0,5	0,3
10	0,5	0,35
10	0,5	0,4
10	0,5	0,42
10	0,5	0,45

16	0,8	0,2
16	0,8	0,25
16	0,8	0,3
16	0,8	0,35
16	0,8	0,4
16	0,8	0,42
16	0,8	0,45
20	1	0,2
20	1	0,25
20	1	0,3
20	1	0,35
20	1	0,4
20	1	0,42
20	1	0,45

24	1,2	0,2
24	1,2	0,25
24	1,2	0,3
24	1,2	0,35
24	1,2	0,4
24	1,2	0,42
24	1,2	0,45
30	1,5	0,2
30	1,5	0,25
30	1,5	0,3
30	1,5	0,35
30	1,5	0,4
30	1,5	0,42
30	1,5	0,45

36	1,8	0,2
36	1,8	0,25
36	1,8	0,3
36	1,8	0,35
36	1,8	0,4
36	1,8	0,42
36	1,8	0,45
7,5	0,3	0,25
20	0,8	0,25
25	1	0,25
37,5	1,5	0,25
45	1,8	0,25

Table 2.14. Simulations done for testing the trained model (case 2)

a (mm)	a/c	h(mm) - a/h		a/t
3,6	1,2	20	0,180	0,2
5,4	1,8	3	1,800	0,3
5,4	1,8	25	0,216	0,3
5	1	3	1,667	0,35
5	0,5	20	0,250	0,25
3	0,3	15	0,200	0,25
12,5	0,5	80	0,156	0,3
6	1,5	10	0,600	0,42
14,4	1,8	100	0,144	0,45
10,2	1,2	80	0,128	0,4
11,25	2,25	15	0,750	0,25
10	2	8	1,250	0,35
24	1,2	150	0,160	0,36
7,5	1	25	0,300	0,3
12	1	4	3,000	0,4
14	0,5	50	0,280	0,32
20,4	0,85	35	0,583	0,25
16,8	0,8	50	0,336	0,27
6	0,3	17,5	0,343	0,38
5,25	0,3	10	0,525	0,3
3,75	0,3	10	0,375	0,2
8	0,5	5	1,600	0,45
7,5	0,5	7,5	1,000	0,26
4,5	0,75	30	0,150	0,1
8,1	0,9	13	0,623	0,33

3. ANALYSIS AND RESULTS

As stated in the previous chapter, 1061 and 760 data were used in the ANN training and testing process respectively for case 1. Also 4248 and 1139 data were used in the training and testing process respectively for case 2. Normalized values of these data were used in all of the neural network simulations. First of all, training data were given to the network model, then this network model was trained and finally accuracy of the model was checked using test data. Finally, model which had minimum deviation was selected as the ultimate model. Figure 3.1 shows screenshot of the Matlab nntool module.

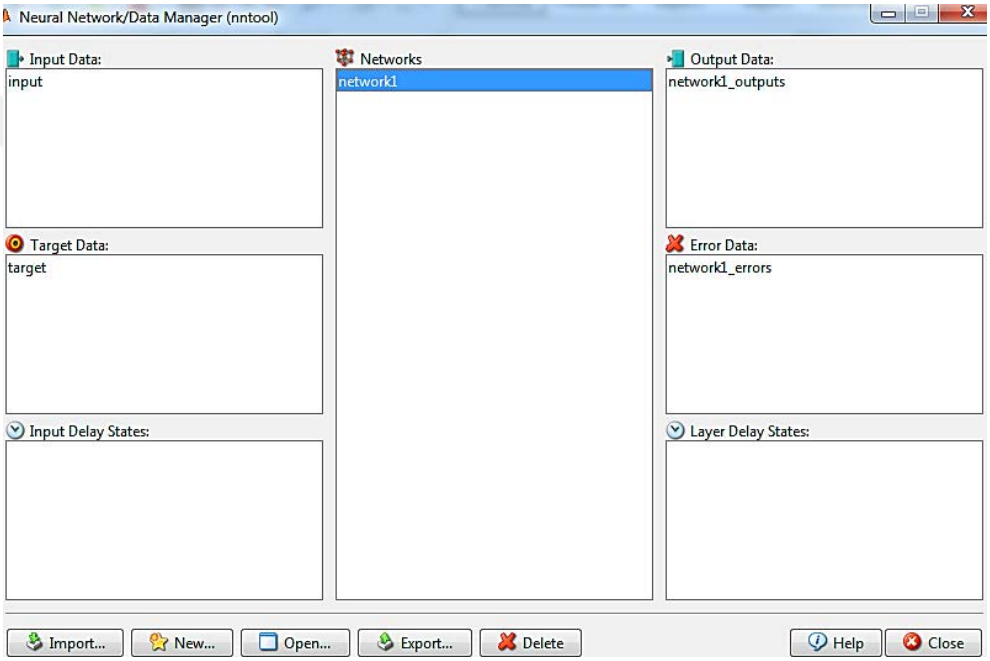


Figure 3.1. Matlab nntool module

In the training process, firstly, Matlab nntool module randomly splits the data into three parts, training (70%), validation (15%) and testing (15%) part. (Percentage values are default – dividerand command) Also initial values of the weights are randomly selected between 0 and 1. Since Matlab randomly splits the data and selects initial weights at the start of each training process, network models which have the same structure does not give the same results. The training data set is used to train the network. The validation data set is not directly used for training, it is used to control the performance of the model and overfitting. When the neural network model starts to overfit, value of validation data set error starts to increase. So Matlab checks validation data set error and stops training in the minimum value of validation data set error. This is known as early stopping. Best

point for training is this point. Finally test data set is used to check overall training performance of the model [35]. Most of default training parameters for Levenberg Marquardt training function were used for training the network. The only parameter which was changed was validation check (max_fail). Value of validation check was selected as 15 (default value 6). This is the number of epochs which validation data set error fails to decrease. Values of training parameters used in Matlab are shown in Figure 3.2 below.

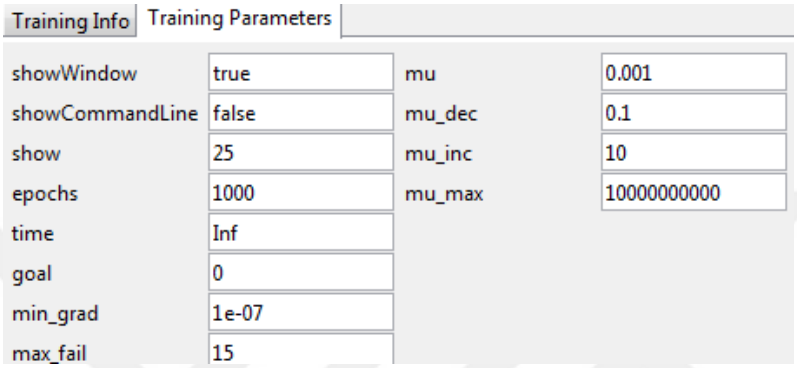


Figure 3.2. Training parameters used in the training process

3.1. Analysis and Results for Case 1

Some of the Ansys simulation results (normalized SIF distribution) for case 1 are given in figure 3.3 and figure 3.4.

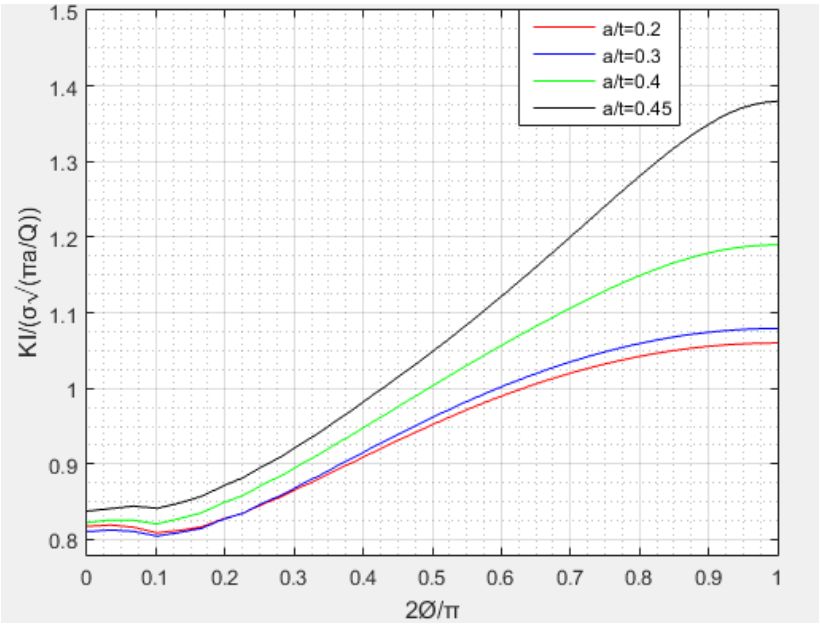


Figure 3.3. Variation of stress intensity factor for different values of a/t (a/c=0.5)

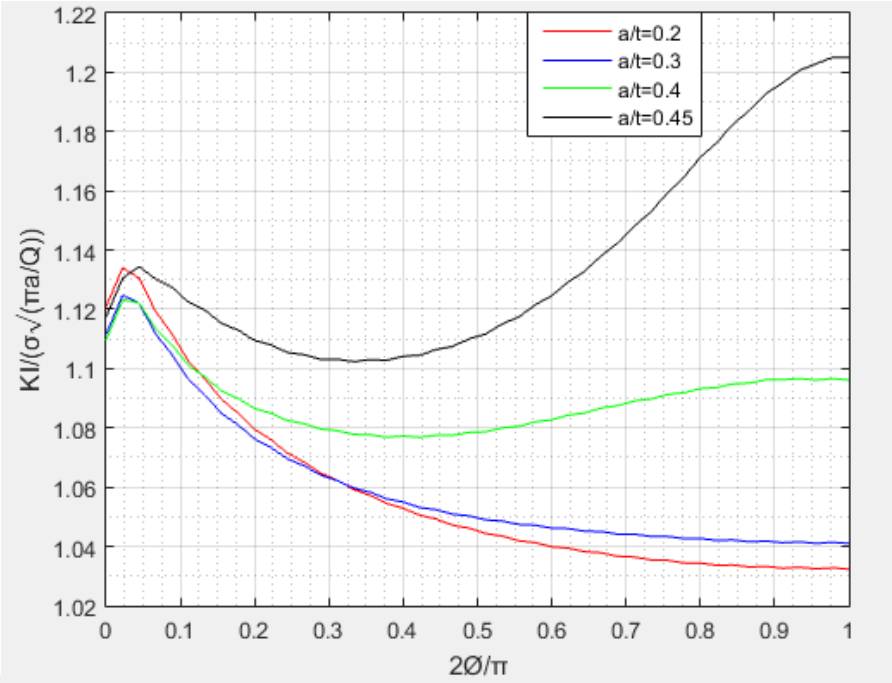


Figure 3.4. Variation of stress intensity factor for different values of a/t (a/c=1)

In ANN simulations, for case 1 (two semi elliptical surface cracked body – one crack at each side), number of input neurons was 4 and number of output neuron was 1. Since the number of hidden layers and corresponding neurons were unknown, different network structures were formed and accuracy of these network models were compared with each other using 760 test data in order to get the most appropriate structure. Results of aforementioned study are shown in table 3.1.

Table 3.1. Deviation values of 760 test data for different types of network structures (case 1)

Number of hidden layers	Number of hidden layer neurons	Deviation %
1	5	2.519
1	8	1.966
1	10	1.701
1	15	1.415
1	20	1.151
2	5	1.218
2	10	0.407
2	15	0.32
2	16	0.852
2	18	0.969

As seen in table 3.1 above, minimum deviation value was 0.32% and so number of hidden layers and hidden layer neurons were selected as 2 and 15 respectively. Schematic representation of this network is given in figure 3.5.

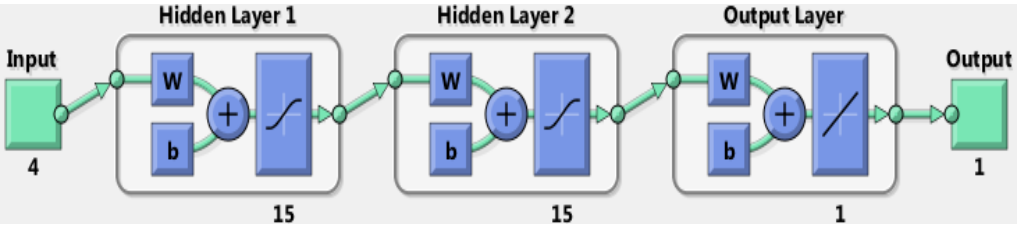


Figure 3.5. Schematic representation of the trained model for case 1

Simulation / training results of the ANN model in figure 3.5 are shown in figure 3.6, figure 3.7, figure 3.8 and figure 3.9 below.

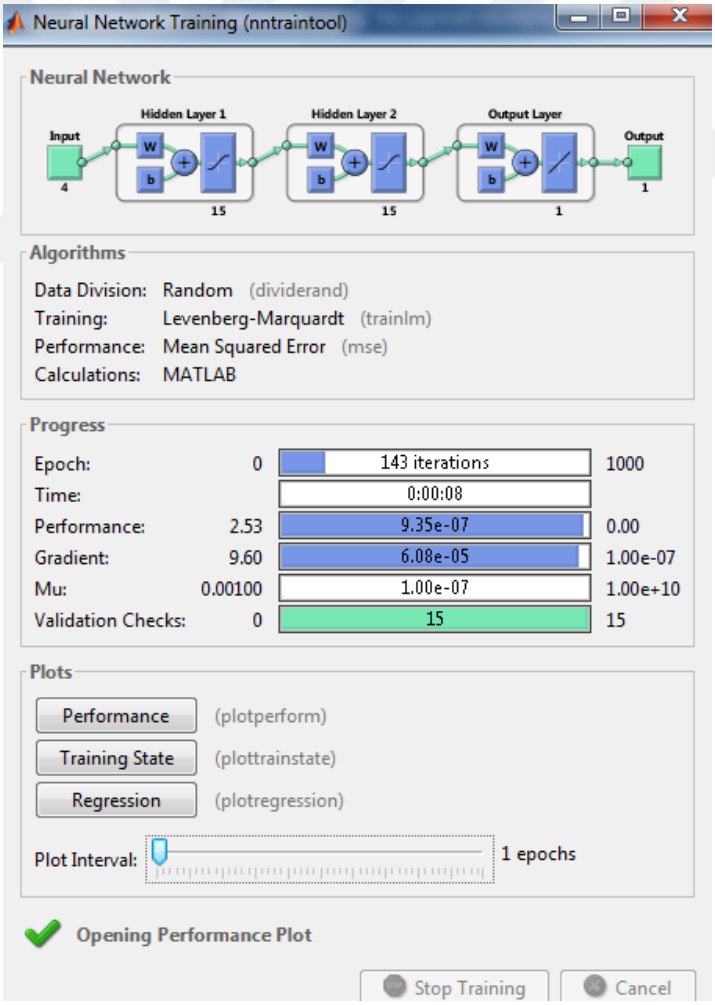


Figure 3.6. Output screen at the end of training process for case 1 (4 input neurons, 2 hidden layers, 15 neurons for each hidden layers and 1 output neuron)

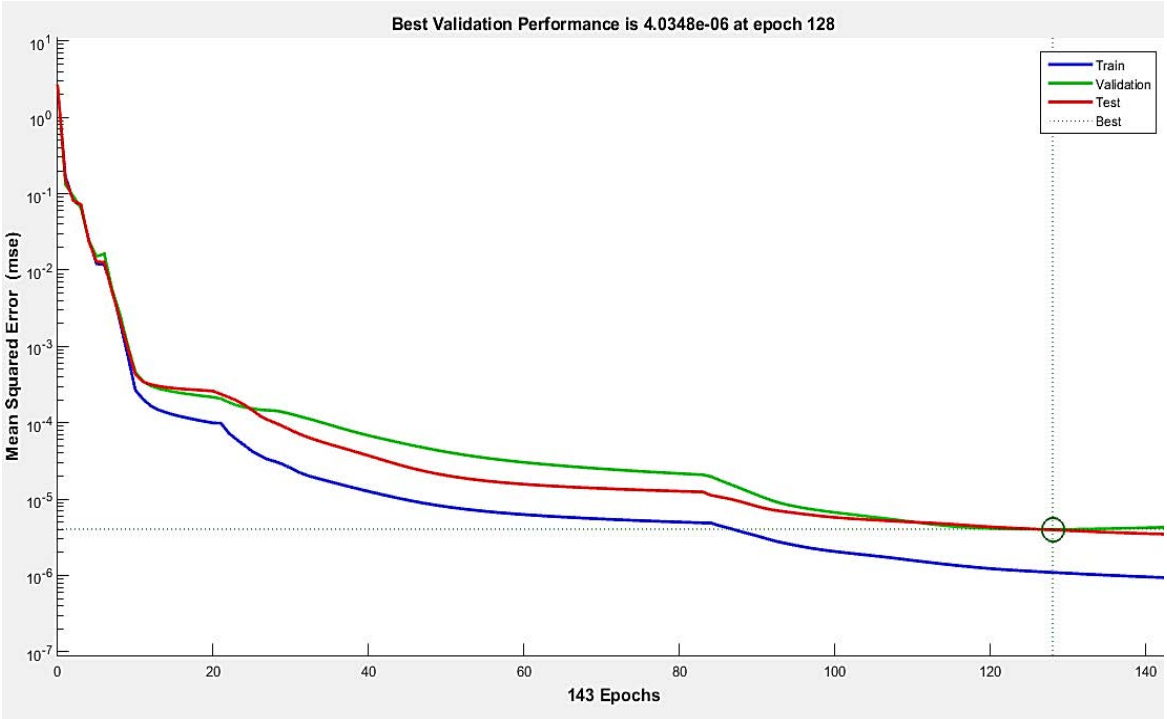


Figure 3.7. Performance graph of the training process (case 1)

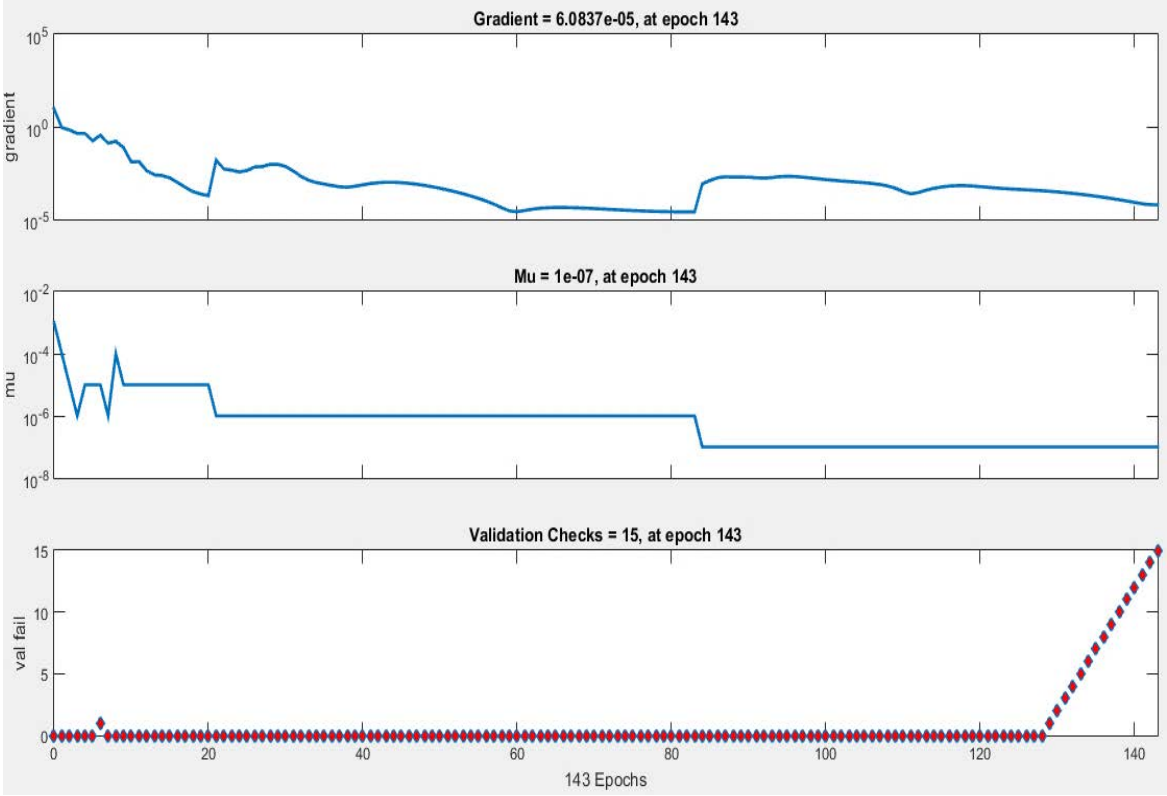


Figure 3.8. Change in the value of gradient, damping factor and number of validation check during training (case 1)

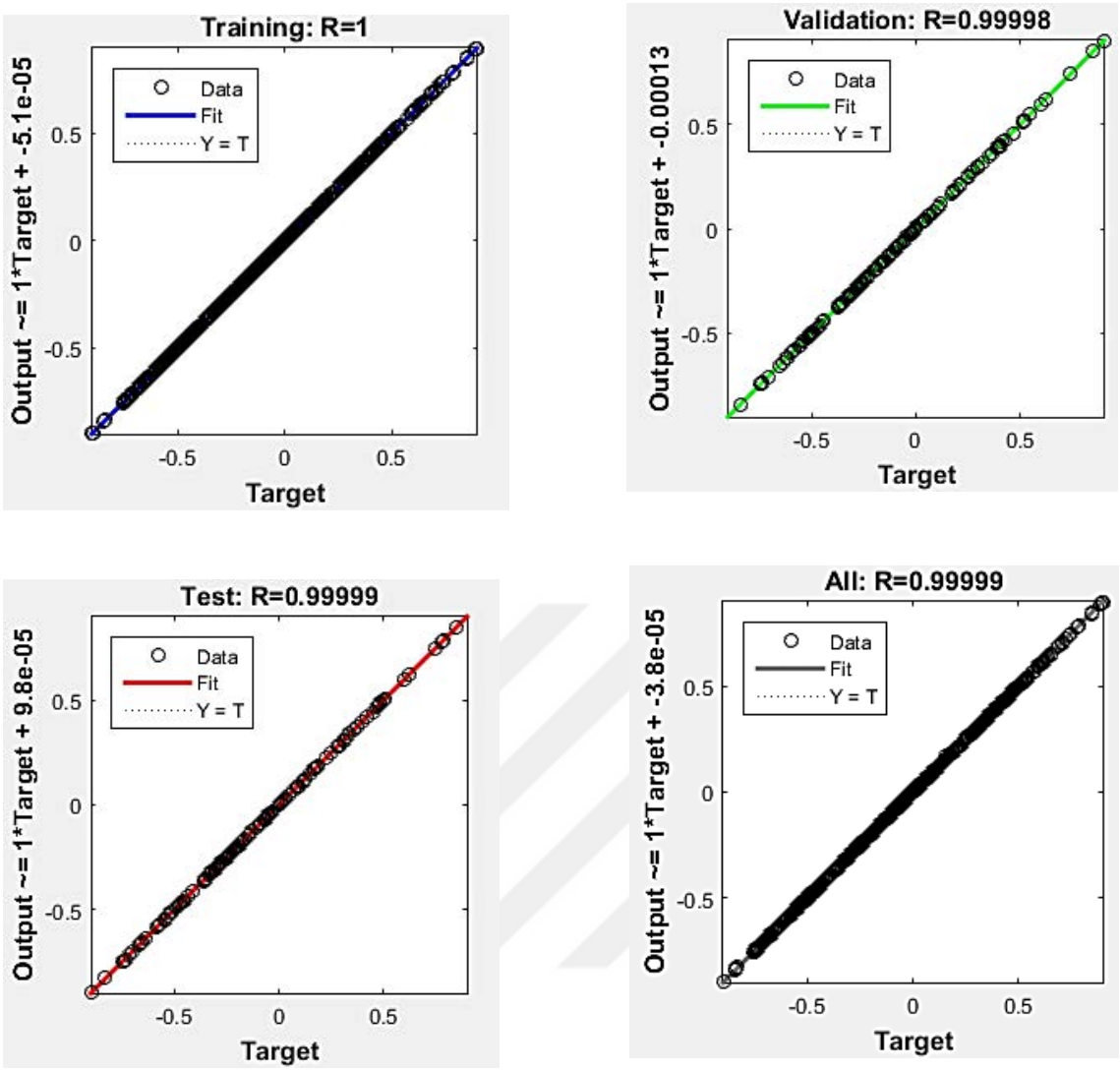


Figure 3.9. Correlation coefficient for training, validation and test data (case 1)

As seen in figure 3.9, the correlation coefficient for training, validation and test data was very close to 1 in case 1. In other words, the statistical relationship between output data of the trained model and target data was great.

This trained ANN model can be used for estimating SIF value of plates / bodies which have two semi elliptical surface cracks, one crack at the front side and at the back side. This model can be directly used with Matlab nntool module or an easy program-code which is generated using trained network weights. Of course, normalization (between -0.9 and 0.9) is needed to estimate SIF for different values of a, a/t, a/c, parametric angle and KI. So minimum and maximum values of these variables which are needed to do normalization are given in table 3.2.

Table 3.2. Minimum and maximum values needed for normalization (case 1)

	a(m) First Input Neuron	a/c Second Input Neuron	Angle (º) Third Input Neuron	a/t Fourth Input Neuron	KI (Pa√m) Output Neuron
Minimum	0,0015	0,3	0,00	0,1	41282
Maximum	0,045	1,8	90,00	0,45	252180

The weights of the trained neural network model which can be utilized to calculate new SIF values are presented as follows. There are 331 weights in total.

Table 3.3. Weights of the trained neural network for case 1 – between input neurons (4) and first hidden layer neurons (15)

Hidden Layer 1	Input Neuron 1 a	Input Neuron 2 a/c	Input Neuron 3 Angle	Input Neuron 4 a/t
Neuron 1	-0.42701	0.94449	0.85053	0.65408
Neuron 2	1.2402	-2.1679	0.19388	0.062815
Neuron 3	0.34417	-0.41592	0.54089	1.5928
Neuron 4	0.1643	-0.083261	1.6154	0.05083
Neuron 5	-0.36543	0.33353	-0.40976	-1.2158
Neuron 6	-0.62662	0.16397	1.0986	0.010568
Neuron 7	0.41838	-0.36187	-0.54367	-0.024711
Neuron 8	-0.65558	-0.064784	-0.40357	-0.011902
Neuron 9	-0.97312	0.48175	-0.10424	-0.7897
Neuron 10	0.37438	-0.67761	1.2339	0.028733
Neuron 11	-0.20426	0.15174	1.1676	0.010823
Neuron 12	0.21878	0.95215	0.97847	-0.065358
Neuron 13	-0.08453	0.043741	-2.761	-0.0030389
Neuron 14	2.1669	-0.044939	0.085944	-0.0021232
Neuron 15	0.056003	-0.88445	0.88815	0.037334

Table 3.4. Weights of the trained neural network for case 1 – between five of first hidden layer neurons (1-5) and second hidden layer neurons (15)

	Hidden 1 Neuron 1	Hidden 1 Neuron 2	Hidden 1 Neuron 3	Hidden 1 Neuron 4	Hidden 1 Neuron 5
Hidden 2 Neuron 1	-0.3215	-0.8446	-0.20217	-0.20144	-0.6414
Hidden 2 Neuron 2	0.39491	0.30376	-0.49667	-0.84706	-1.1407
Hidden 2 Neuron 3	0.82872	0.80843	0.3291	-1.1579	-1.1665
Hidden 2 Neuron 4	-0.51058	0.25985	-0.83835	1.423	0.29119
Hidden 2 Neuron 5	0.46518	0.001291	1.3892	-0.14459	0.086084
Hidden 2 Neuron 6	0.10942	-0.10132	-0.019605	1.0256	0.3452
Hidden 2 Neuron 7	-0.072161	-0.13344	0.21014	0.22298	0.55573
Hidden 2 Neuron 8	-0.69952	0.60918	0.33515	1.3807	0.28218
Hidden 2 Neuron 9	0.047818	-0.55185	-0.48336	0.034563	-1.2371
Hidden 2 Neuron 10	0.74063	0.69689	-0.68432	-0.60947	0.5943
Hidden 2 Neuron 11	0.71763	1.1707	-0.75717	-0.1526	0.96512
Hidden 2 Neuron 12	-0.088566	0.8537	0.23353	0.20498	0.20898
Hidden 2 Neuron 13	0.85399	0.052766	0.062017	-0.1275	0.56747
Hidden 2 Neuron 14	-0.3615	-0.79864	0.41507	0.14989	0.16949
Hidden 2 Neuron 15	0.12788	0.76244	-0.37496	-0.6977	-0.12389

Table 3.5. Weights of the trained neural network for case 1 – between five of first hidden layer neurons (6-10) and second hidden layer neurons (15)

	Hidden 1 Neuron 6	Hidden 1 Neuron 7	Hidden 1 Neuron 8	Hidden 1 Neuron 9	Hidden 1 Neuron 10
Hidden 2 Neuron 1	0.42393	1.1186	-0.83278	-0.087296	-0.64169
Hidden 2 Neuron 2	0.19131	0.14367	-0.17467	-0.021887	-0.18483
Hidden 2 Neuron 3	-0.31162	-0.4474	0.16819	0.092968	-0.16538
Hidden 2 Neuron 4	0.5384	0.62739	-0.42028	0.065541	0.069728
Hidden 2 Neuron 5	0.27536	-0.30876	-0.58348	0.025174	-0.13438
Hidden 2 Neuron 6	-0.80907	-0.080145	-1.0571	0.70178	0.055557
Hidden 2 Neuron 7	-0.33107	-0.85041	0.4723	-0.0080075	0.054043
Hidden 2 Neuron 8	-0.058243	1.1824	-0.40283	0.23221	-0.37391
Hidden 2 Neuron 9	-0.6812	-0.77293	-0.47245	0.27638	-0.25189
Hidden 2 Neuron 10	-0.9932	1.2919	0.35618	1.354	0.38969
Hidden 2 Neuron 11	0.032561	-0.14252	-0.54883	0.53154	0.616
Hidden 2 Neuron 12	1.0365	2.0561	0.51968	-0.019701	0.38611
Hidden 2 Neuron 13	-0.71646	-0.52628	-1.0127	0.013605	-0.34072
Hidden 2 Neuron 14	-0.22589	-0.063222	-0.97837	-0.046975	-0.39371
Hidden 2 Neuron 15	-0.61784	-0.56204	0.9297	-0.0032049	-0.13688

Table 3.6. Weights of the trained neural network for case 1 – between five of first hidden layer neurons (11-15) and second hidden layer neurons (15)

	Hidden 1 Neuron 11	Hidden 1 Neuron 12	Hidden 1 Neuron 13	Hidden 1 Neuron 14	Hidden 1 Neuron 15
Hidden 2 Neuron 1	0.50652	-0.65793	-0.3648	0.5921	1.2845
Hidden 2 Neuron 2	0.0035707	-0.18061	0.78762	-0.12277	-0.5683
Hidden 2 Neuron 3	-0.35708	1.2729	-0.15221	0.80107	1.009
Hidden 2 Neuron 4	0.37078	-0.51	-0.2307	-0.77743	-0.20083
Hidden 2 Neuron 5	0.30357	0.31036	-0.71597	0.096744	-0.52224
Hidden 2 Neuron 6	-0.43593	0.41164	0.011278	0.25884	-0.046408
Hidden 2 Neuron 7	-0.15922	0.37388	0.3937	0.059262	0.51525
Hidden 2 Neuron 8	0.16937	0.61789	-0.053082	-0.3212	-0.71607
Hidden 2 Neuron 9	0.34156	0.4547	0.093368	0.54569	-0.48429
Hidden 2 Neuron 10	-1.4508	-0.70737	0.060086	1.489	-0.98288
Hidden 2 Neuron 11	-1.3411	0.42265	1.1978	0.55676	-0.10703
Hidden 2 Neuron 12	0.63658	0.11929	-0.036442	1.044	-0.26534
Hidden 2 Neuron 13	-0.17472	-0.26463	-0.23347	3.111	-0.29454
Hidden 2 Neuron 14	0.26741	0.55687	0.72461	-0.91425	0.1019
Hidden 2 Neuron 15	0.55861	0.031693	-2.1144	0.16156	-0.82013

Table 3.7. Weights of the trained neural network for case 1 – between second hidden layer neurons (15) and output neuron (1)

	Hidden 2 Neuron 1	Hidden 2 Neuron 2	Hidden 2 Neuron 3	Hidden 2 Neuron 4	Hidden 2 Neuron 5
Output neuron	1.2937	-1.7401	1.3352	-0.5922	-0.60146
	Hidden 2 Neuron 6	Hidden 2 Neuron 7	Hidden 2 Neuron 8	Hidden 2 Neuron 9	Hidden 2 Neuron 10
Output neuron	-0.19176	-1.6159	-0.41521	0.60459	-0.004914
	Hidden 2 Neuron 11	Hidden 2 Neuron 12	Hidden 2 Neuron 13	Hidden 2 Neuron 14	Hidden 2 Neuron 15
Output neuron	0.13383	1.251	3.0383	0.56691	-0.64608

Table 3.8. Bias weights of the trained neural network for case 1

Neuron	Corresponding Bias	Neuron	Corresponding Bias
Hidden 1 Neuron 1	-3.0685	Hidden 2 Neuron 1	-1.4028
Hidden 1 Neuron 2	-2.5259	Hidden 2 Neuron 2	0.13168
Hidden 1 Neuron 3	-2.7273	Hidden 2 Neuron 3	-0.92564
Hidden 1 Neuron 4	-2.0641	Hidden 2 Neuron 4	0.73684
Hidden 1 Neuron 5	2.2634	Hidden 2 Neuron 5	-0.28852
Hidden 1 Neuron 6	0.59668	Hidden 2 Neuron 6	0.39252
Hidden 1 Neuron 7	-0.30205	Hidden 2 Neuron 7	-0.54189
Hidden 1 Neuron 8	-0.44699	Hidden 2 Neuron 8	0.071495
Hidden 1 Neuron 9	0.17927	Hidden 2 Neuron 9	-0.67006
Hidden 1 Neuron 10	0.031039	Hidden 2 Neuron 10	-0.12947
Hidden 1 Neuron 11	-0.75374	Hidden 2 Neuron 11	-0.20851
Hidden 1 Neuron 12	1.9494	Hidden 2 Neuron 12	-1.099
Hidden 1 Neuron 13	-3.1826	Hidden 2 Neuron 13	0.62752
Hidden 1 Neuron 14	2.7321	Hidden 2 Neuron 14	0.9354
Hidden 1 Neuron 15	-1.7077	Hidden 2 Neuron 15	-2.0187
Output Neuron	-1.9611		

Graphs of Ansys and trained ANN results (case 1 – 2 hidden layer – 15 hidden layer neurons for each hidden layer) for some new values (test values) of input variables are presented in the following figures.

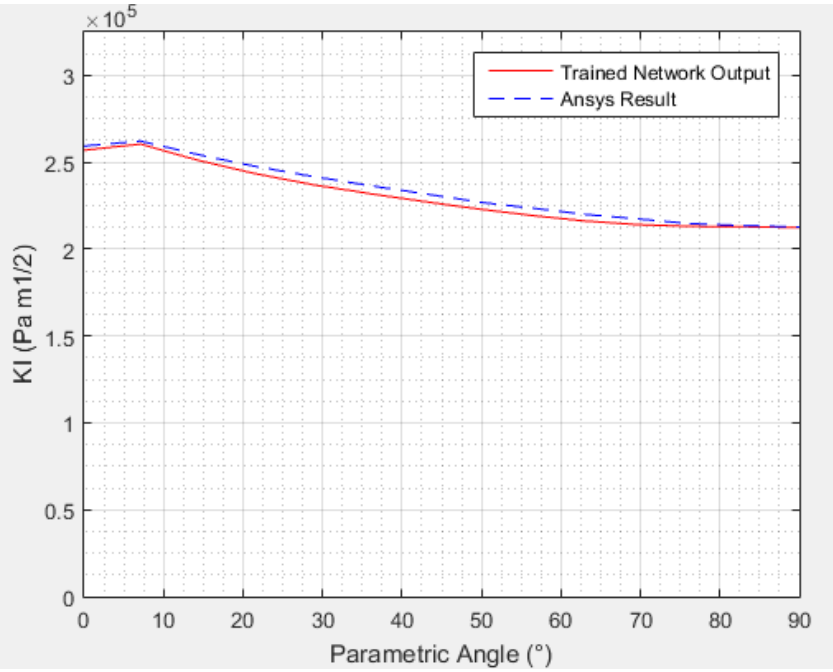
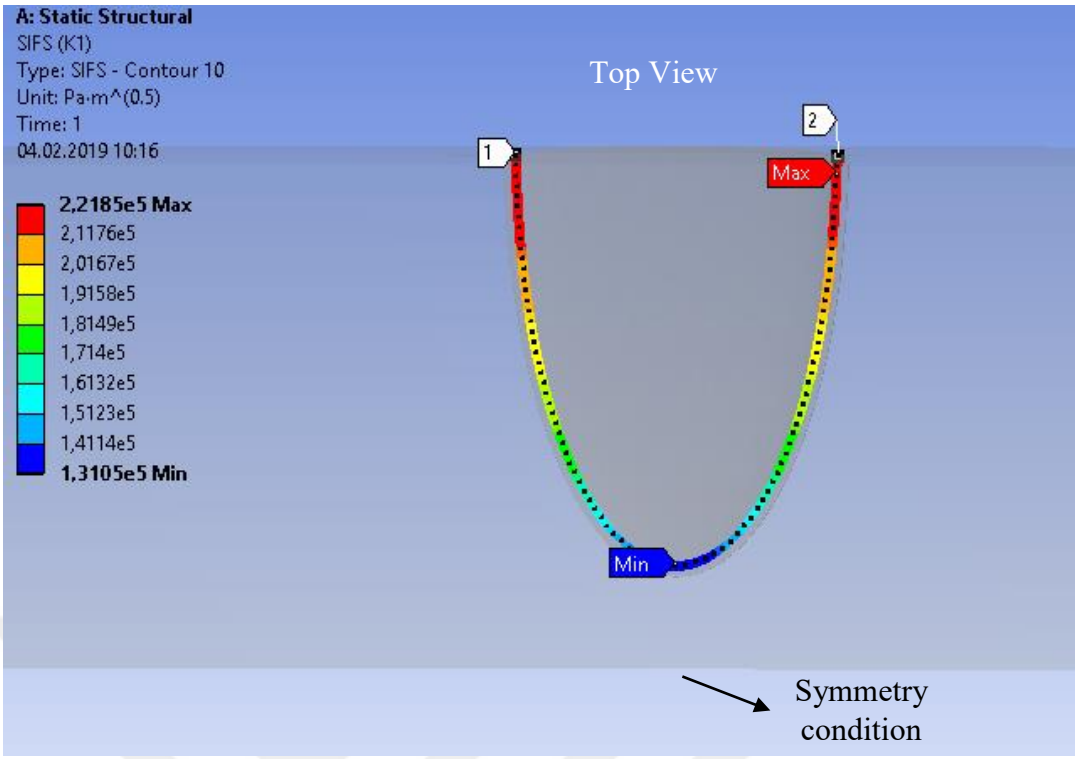
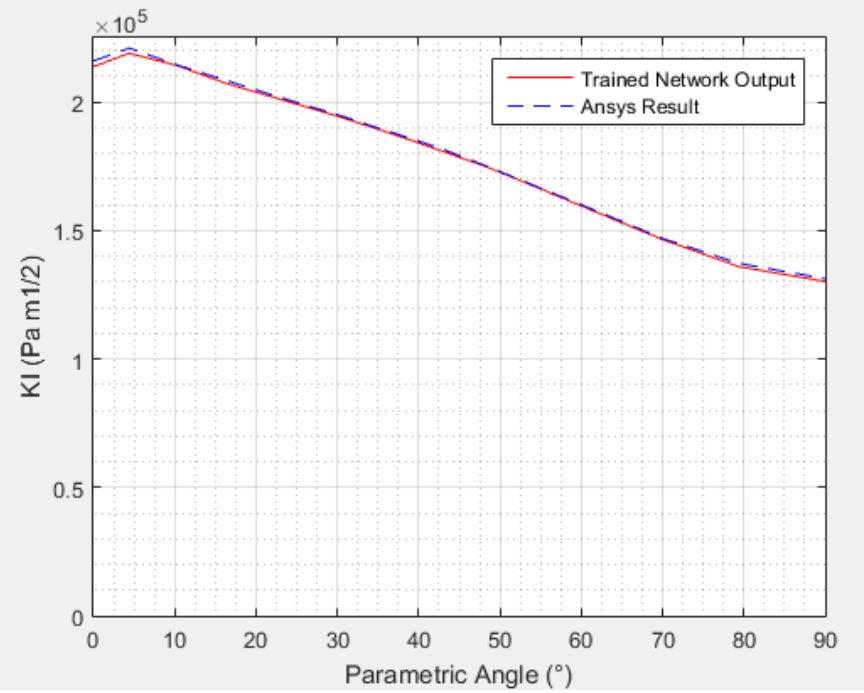


Figure 3.10. Variation of SIF along the crack front for case 1 (a=0.0442 m, a/c=1.3, a/t=0.33, deviation=1.32%)

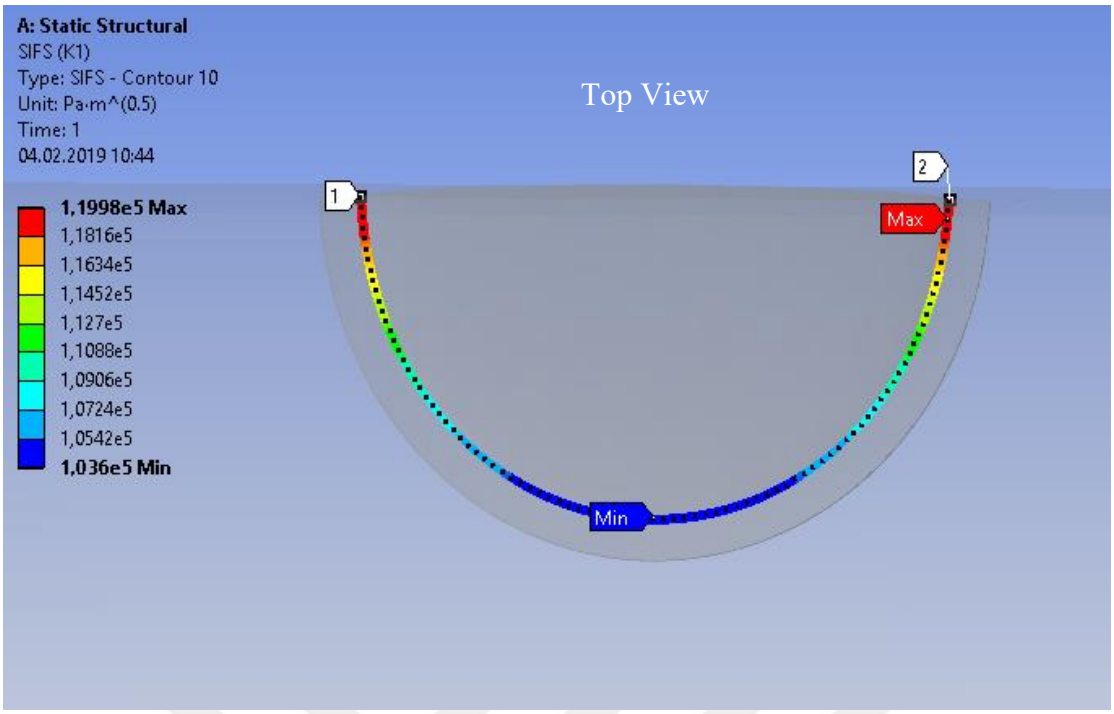


(a)

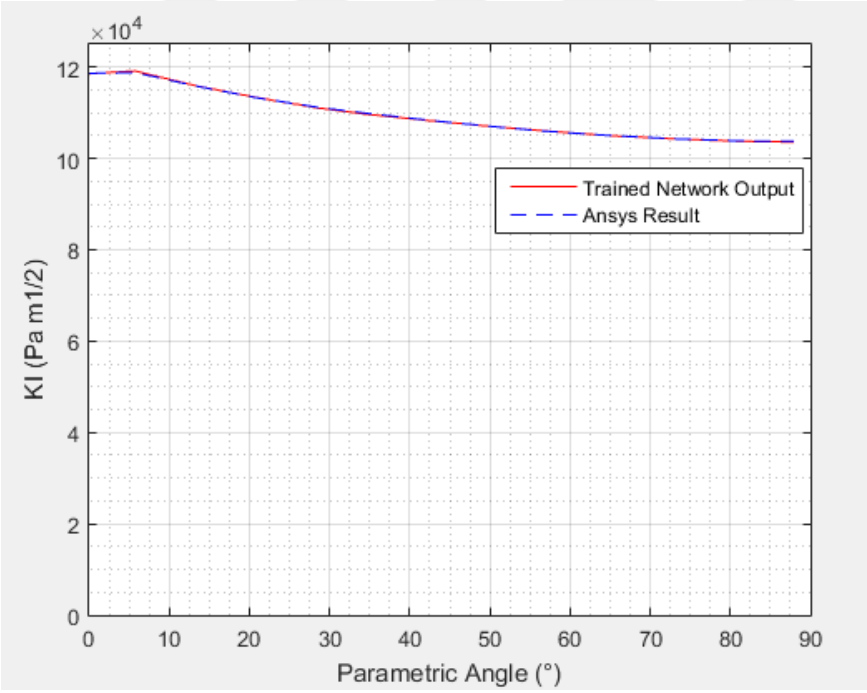


(b)

Figure 3.11. Variation of SIF along the crack front for case 1 – (a) Ansys result and (b) Comparative graph (a=0.0425 m, a/c=2.5, a/t=0.41, deviation=0.49%)

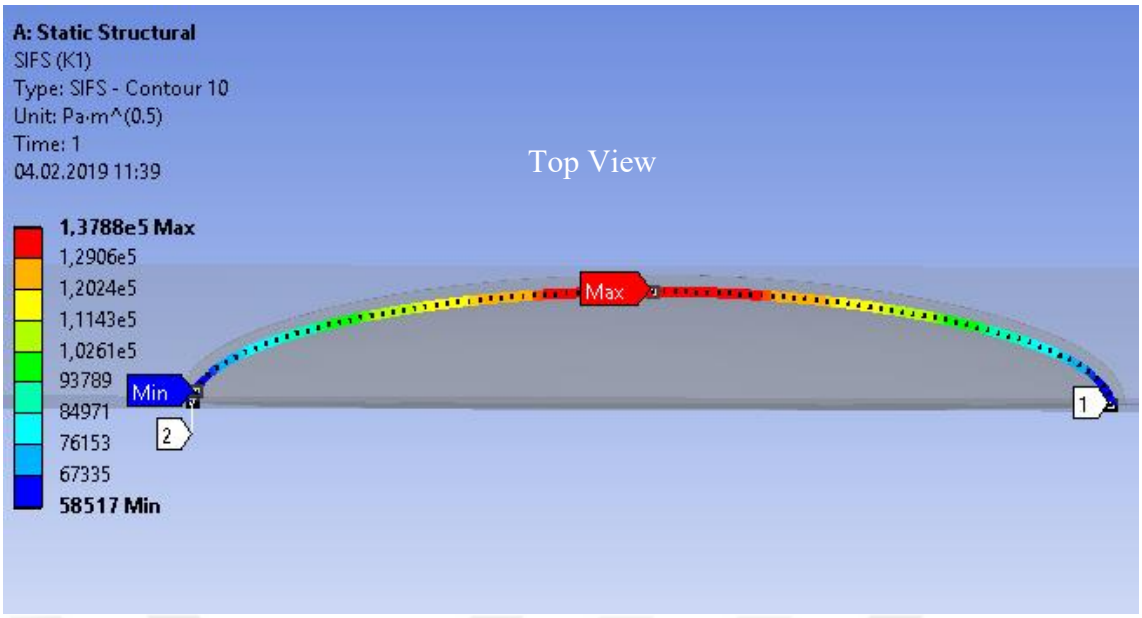


(a)

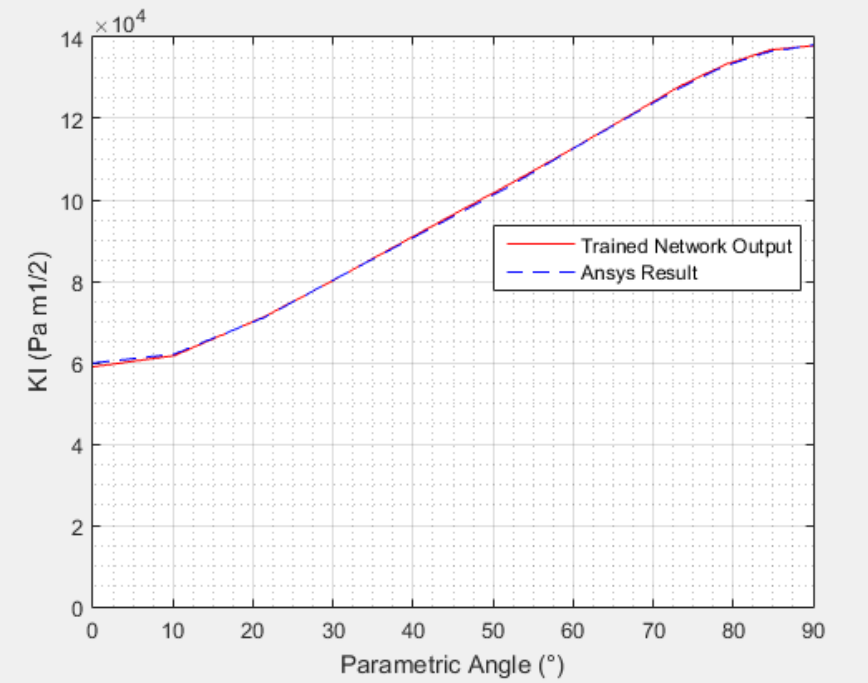


(b)

Figure 3.12. Variation of SIF along the crack front for case 1 – (a) Ansys result and (b) Comparative graph (a= 0.0088 m, a/c= 1.1, a/t= 0.15, deviation= 0.08%)



(a)



(b)

Figure 3.13. Variation of SIF along the crack front for case 1 – (a) Ansys result and (b) Comparative graph (a= 0.003125 m, a/c= 0.25, a/t= 0.44, deviation= 0.38%)

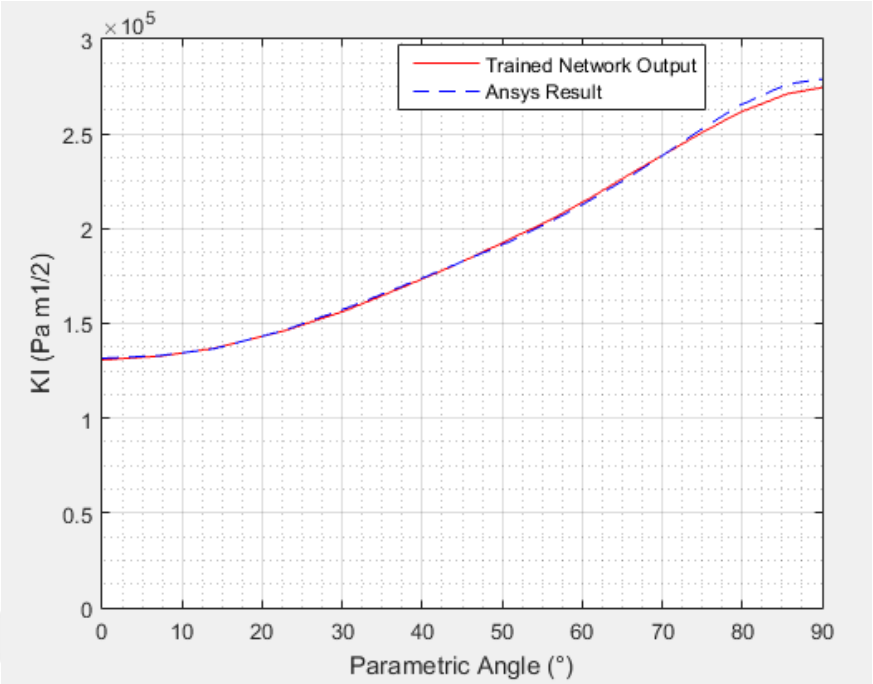


Figure 3.14. Variation of SIF along the crack front for case 1 (a=0.012 m, a/c=0.4, a/t=0.47, deviation=0.65%)

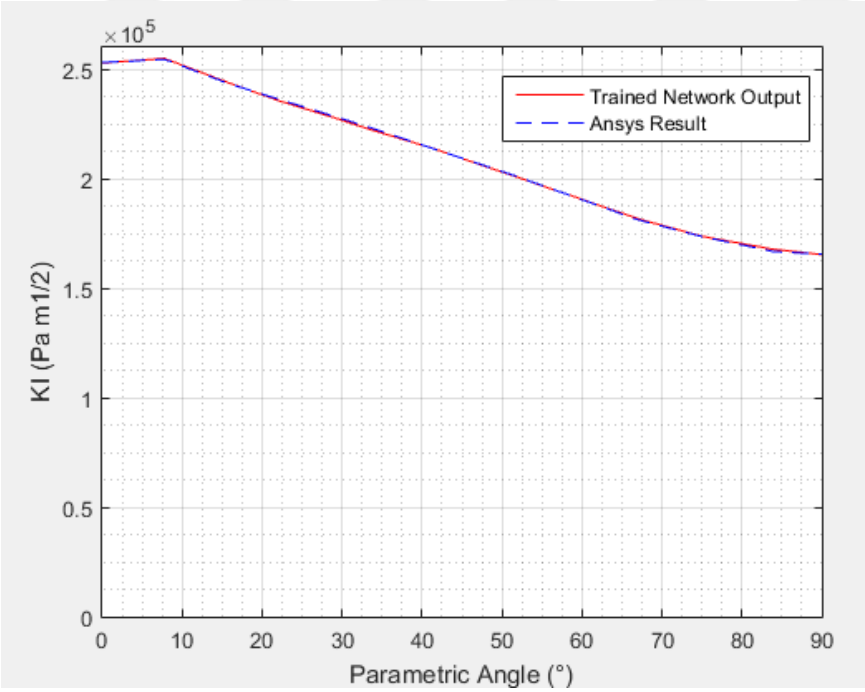
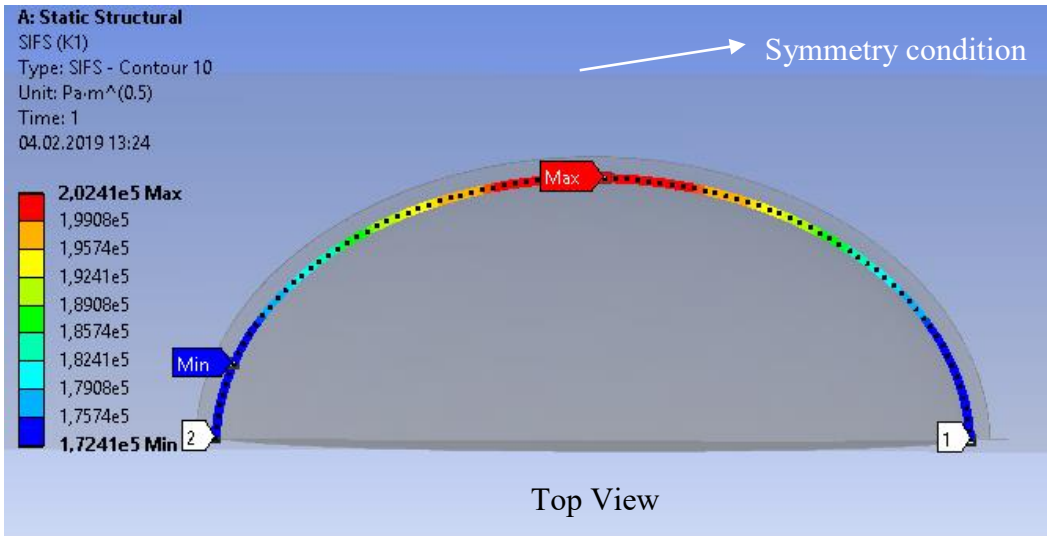
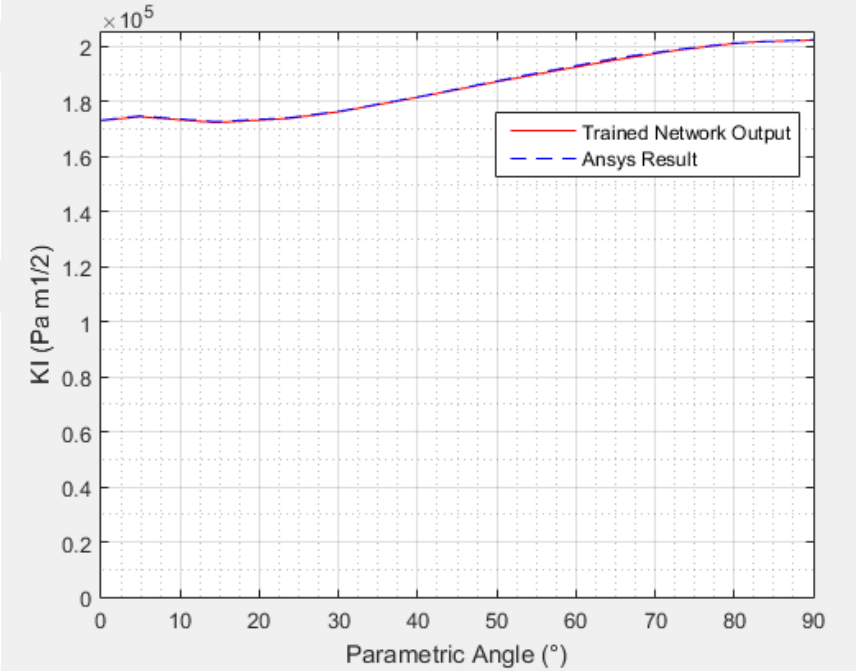


Figure 3.15. Variation of SIF along the crack front for case 1 (a=0.05 m, a/c=2, a/t=0.25, deviation=0.2%)



(a)



(b)

Figure 3.16. Variation of SIF along the crack front for case 1 – (a) Ansys result and (b) Comparative graph (a=0.0189 m, a/c=0.7, a/t=0.38, deviation=0.13%)

3.2. Analysis and Results for Case 2

Figure 3.17 and figure 3.18 below shows the SIF (normalized) distribution for different values of a/h.

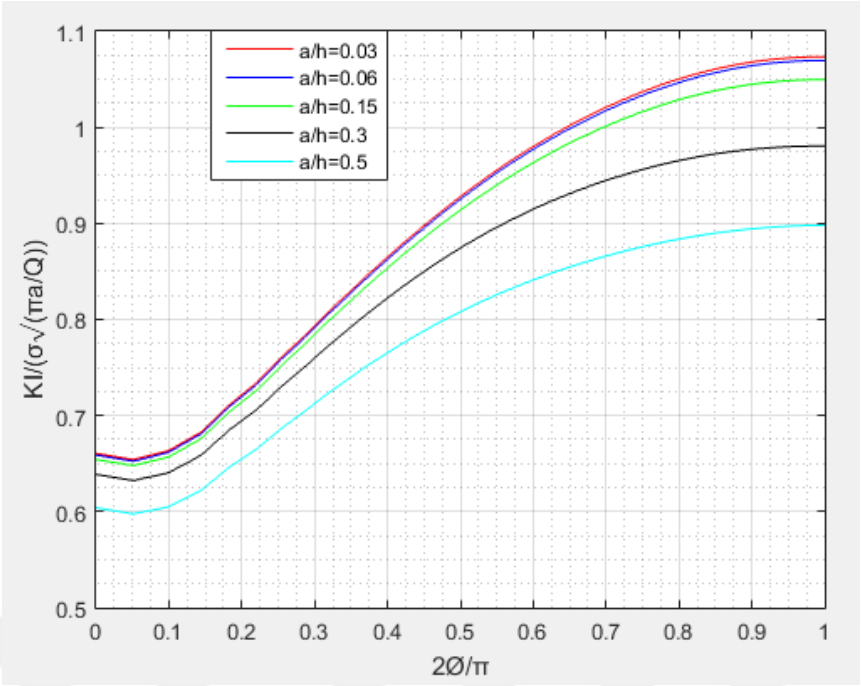


Figure 3.17. Variation of stress intensity factor for different values of a/h (a/c=0.3, a/t=0.2)

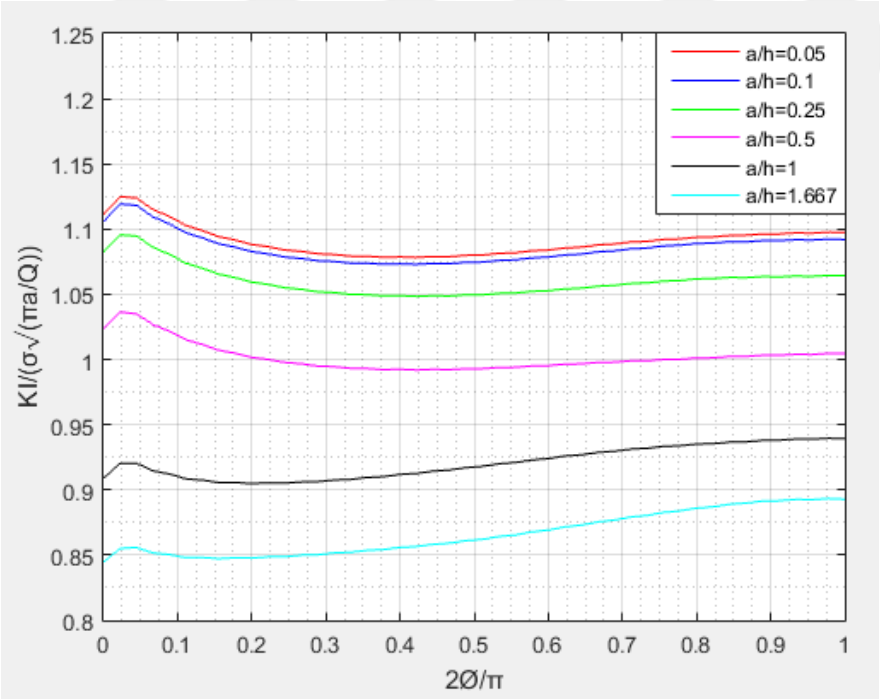


Figure 3.18. Variation of stress intensity factor for different values of a/h (a/c= 1, a/t= 0.4)

Same procedure in case 1 was followed for case 2 (four semi elliptical surface cracked body – two cracks at each side). In case 2, there were 5 input neurons and 1 output neuron. As stated before, extra input neuron was h for the second case. Actually, a/h could be used as an input neuron instead of h in ANN, but h was also a good choice, so h was selected as an extra input neuron. The results of the study, which was done to determine optimum number of hidden layer and hidden layer neurons are given in table 3.9.

Table 3.9. Deviation values of 1139 test data for different types of ANN in case 2

Number of hidden layer	Number of hidden layer neurons	Deviation %
1	5	3.45
1	10	1.53
1	20	1.29
2	10	0.66
2	14	0.58
2	15	0.54
2	16	0.63
2	20	0.82
3	10	0.83
3	14	0.49
3	15	0.68
3	20	1.38

The best result was obtained as 0.49%. Number of hidden layers and corresponding hidden layer neurons was 3 and 14 respectively for the best case. Schematic representation of this neural network structure is shown in figure 3.19 below.

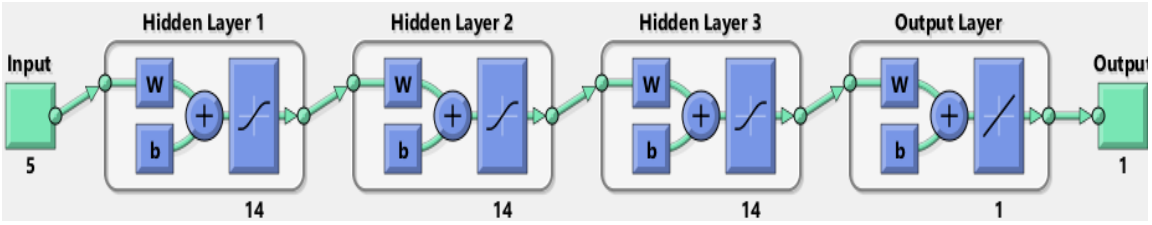


Figure 3.19. Schematic representation of the trained model for case 2

Training results of the model for case 2 are shown in figure 3.20, figure 3.21, figure 3.22 and figure 3.23 as follows.

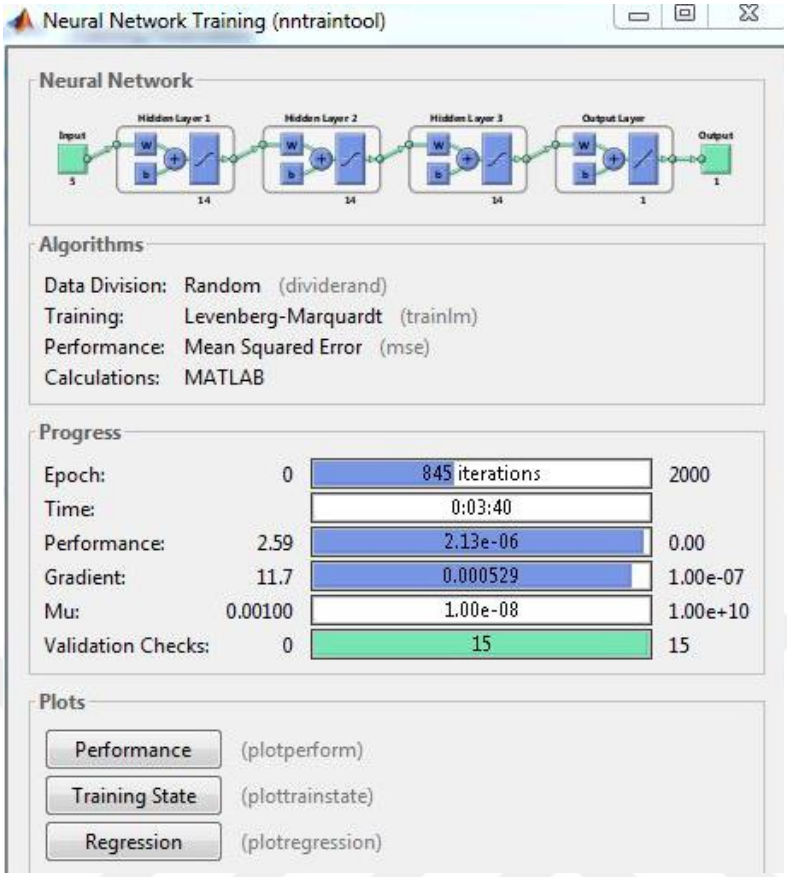


Figure 3.20. Output screen at the end of training process for case 2 (5 input neurons, 3 hidden layers, 14 neurons for each hidden layers and 1 output neuron)

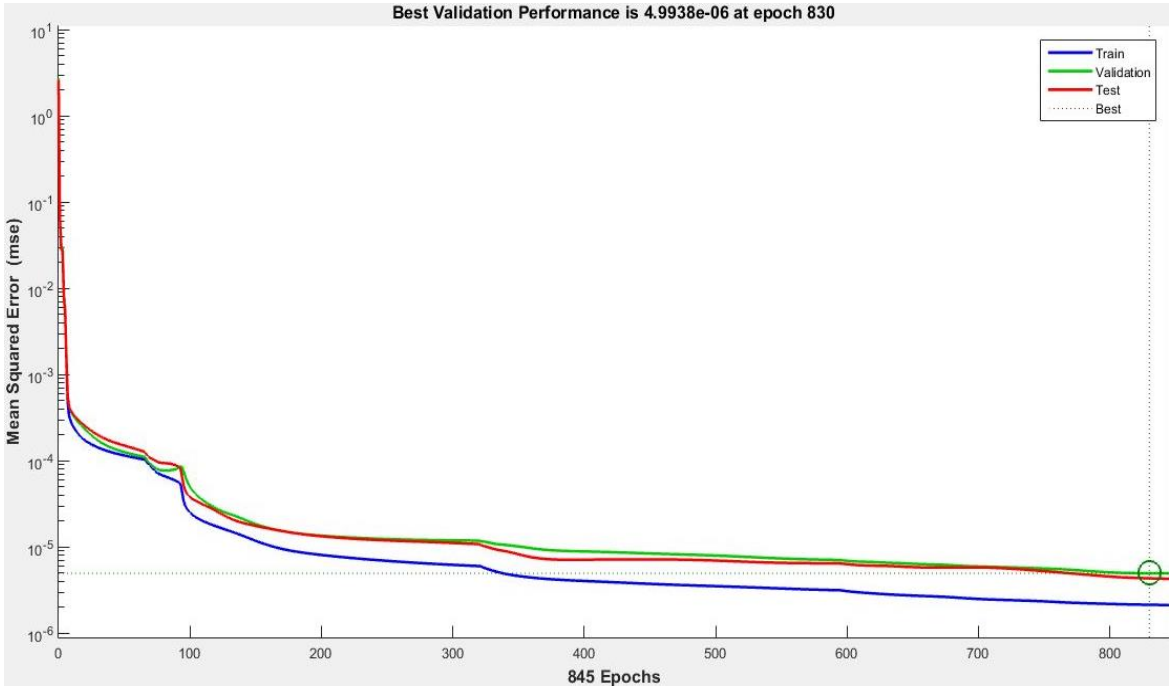


Figure 3.21. Performance graph of the training process (case 2)

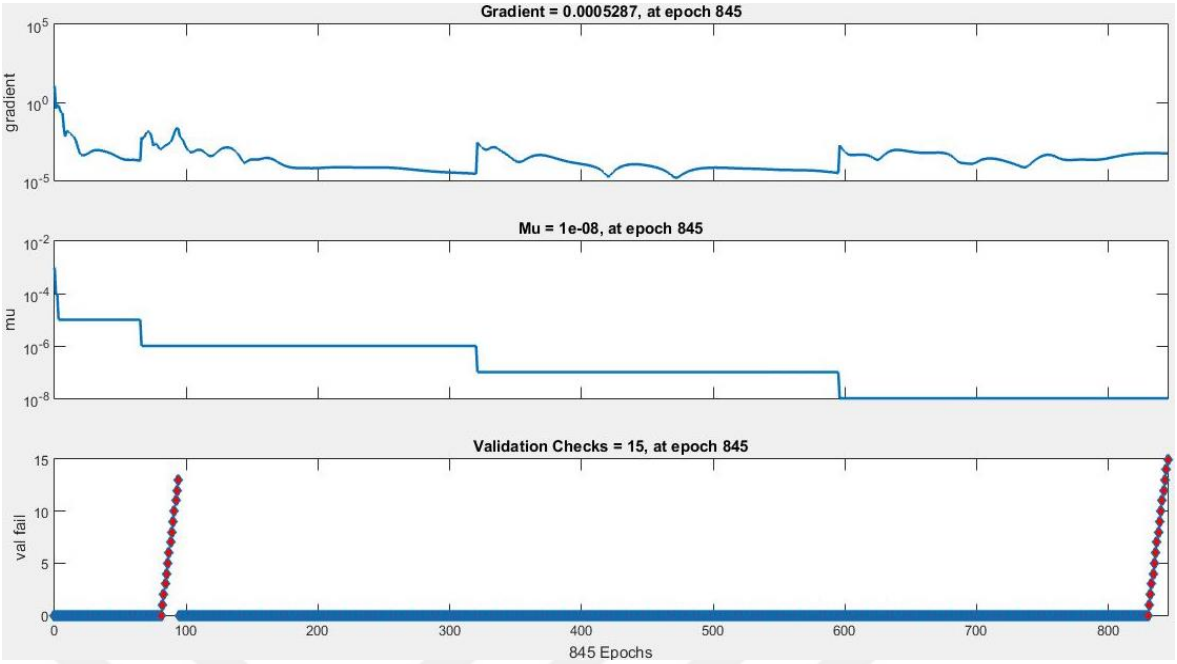


Figure 3.22. Change in the value of gradient, damping factor and number of validation check during training (case 2)

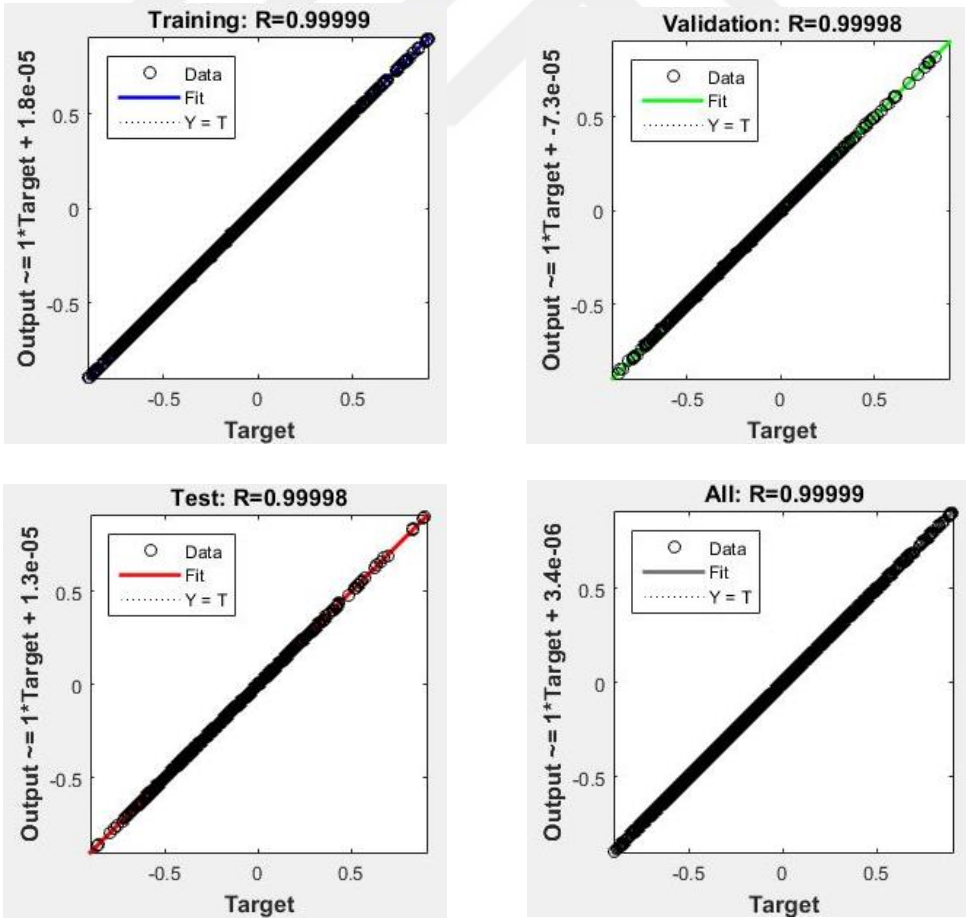


Figure 3.23. Correlation coefficient for training, validation and test data (case 2)

As shown in figure 3.23, the correlation between output and target data was quite well. So weights of this network model can be used to estimate SIF values of plates which contain two semi elliptical surface cracks in both sides, front and back. As mentioned in the previous study for case 1, normalization is needed to estimate new stress intensity factor values. Table 3.10 shows the minimum and maximum values of a, a/c, h, parametric angle and a/t.

Table 3.10. Minimum and maximum values needed for normalization (case 2)

	a(m) First Input Neuron	a/c Second Input Neuron	h (m) Third Input Neuron	Angle (º) Fourth Input Neuron	a/t Fifth Input Neuron	KI (Pa√m) Output Neuron
Minimum	0,0015	0,3	0,0025	0,00	0,1	37332
Maximum	0,045	1,8	0,15	90,00	0,45	224030

The weights of the trained network which can be used to calculate new SIF values for case 2 are given as follows. There are 519 weights in total.

Table 3.11. Weights of the trained neural network for case 2 – between input neurons (5) and first hidden layer neurons (14)

Hidden Layer 1	Input Neuron 1 a	Input Neuron 2 a/c	Input Neuron 3 h	Input Neuron 4 Angle	Input Neuron 5 a/t
Neuron 1	1.0915	-0.15327	-0.15692	0.70624	1.2628
Neuron 2	0.01693	-0.43757	0.02355	0.87277	0.0052101
Neuron 3	-0.0021194	-0.25133	-0.011249	-0.8989	0.0042421
Neuron 4	0.8491	0.087192	-3.0433	-0.34526	-1.0325
Neuron 5	0.23869	-0.10184	-1.995	0.022941	0.00089902
Neuron 6	-0.070998	-0.030719	-0.05481	-0.53242	-0.011168
Neuron 7	-0.1898	-0.15371	-0.026677	0.088734	0.040566
Neuron 8	-0.033088	0.035834	-0.088377	3.5867	-0.0035772
Neuron 9	0.56783	0.073921	-0.024985	-0.1346	0.019894
Neuron 10	0.015561	0.25292	0.0056634	0.50251	-0.0091427
Neuron 11	-0.29586	0.35175	0.085431	-0.10228	-0.049459
Neuron 12	0.40334	0.255	-0.21209	0.040556	0.049014
Neuron 13	-0.1333	0.046731	-0.019336	-0.7502	-0.90116
Neuron 14	-1.1516	-0.38473	-0.048309	-0.16799	-0.056272

Table 3.12. Weights of the trained neural network for case 2 – between five of first hidden layer neurons (1-5) and second hidden layer neurons (14)

	Hidden 1 Neuron 1	Hidden 1 Neuron 2	Hidden 1 Neuron 3	Hidden 1 Neuron 4	Hidden 1 Neuron 5
Hidden 2 Neuron 1	-0.30156	-0.44793	-0.20202	-0.10219	0.019826
Hidden 2 Neuron 2	-0.050305	-1.085	-2.2141	0.44215	-0.40958
Hidden 2 Neuron 3	-0.1071	0.6549	0.23295	-0.0090748	-1.1705
Hidden 2 Neuron 4	0.038803	0.34249	-0.24311	0.18069	5.4888
Hidden 2 Neuron 5	0.049521	-0.18286	0.047912	0.020069	-1.0049
Hidden 2 Neuron 6	0.059901	0.51512	1.379	0.025512	-1.201
Hidden 2 Neuron 7	-0.70981	-0.30653	-0.42189	0.020777	-0.40223
Hidden 2 Neuron 8	-0.015308	0.38711	0.13803	-0.026693	0.8129
Hidden 2 Neuron 9	0.014094	-0.357	-0.6547	0.18446	-5.2823
Hidden 2 Neuron 10	0.074961	-2.2339	-0.37341	0.061475	-0.77314
Hidden 2 Neuron 11	0.46266	-0.68548	1.7319	0.15273	-1.6127
Hidden 2 Neuron 12	-0.20277	-0.44257	0.23871	-0.088718	-4.5798
Hidden 2 Neuron 13	0.17865	-0.87569	-2.0584	0.4676	-0.97385
Hidden 2 Neuron 14	0.67561	3.8097	1.1527	-0.020641	1.2123

Table 3.13. Weights of the trained neural network for case 2 – between five of first hidden layer neurons (6-10) and second hidden layer neurons (14)

	Hidden 1 Neuron 6	Hidden 1 Neuron 7	Hidden 1 Neuron 8	Hidden 1 Neuron 9	Hidden 1 Neuron 10
Hidden 2 Neuron 1	0.98466	-1.2891	-2.4112	-2.6756	1.7245
Hidden 2 Neuron 2	0.53351	-0.97961	0.19449	-0.27808	1.0762
Hidden 2 Neuron 3	-0.27834	-0.80785	-0.026329	-0.076969	-0.13879
Hidden 2 Neuron 4	1.3717	0.070298	0.18441	5.3644	2.0447
Hidden 2 Neuron 5	-0.67142	-0.28954	0.42736	-1.3489	-0.5978
Hidden 2 Neuron 6	0.74131	-0.46677	-1.2823	0.16468	0.46524
Hidden 2 Neuron 7	0.82491	-1.5965	-1.1629	1.7945	1.6
Hidden 2 Neuron 8	-0.11638	-0.16074	0.13978	1.4467	-0.13568
Hidden 2 Neuron 9	3.9418	-0.7314	0.18654	-1.8478	3.0989
Hidden 2 Neuron 10	0.84101	0.67924	-1.8988	1.9545	1.2396
Hidden 2 Neuron 11	-0.92988	2.2383	-0.71915	-0.51694	-1.0571
Hidden 2 Neuron 12	1.4416	-0.10142	-0.4952	-2.995	1.8547
Hidden 2 Neuron 13	1.1515	1.1962	-0.1326	0.75842	1.6161
Hidden 2 Neuron 14	-3.3145	0.27305	-1.1508	-1.5574	-2.7703

Table 3.14. Weights of the trained neural network for case 2 – between four of first hidden layer neurons (11-14) and second hidden layer neurons (14)

	Hidden 1 Neuron 11	Hidden 1 Neuron 12	Hidden 1 Neuron 13	Hidden 1 Neuron 14
Hidden 2 Neuron 1	-2.1994	0.14952	0.38555	1.3627
Hidden 2 Neuron 2	-0.18189	-0.47321	0.6818	-0.95061
Hidden 2 Neuron 3	-0.30386	-0.091085	-0.31293	-0.38075
Hidden 2 Neuron 4	-1.1809	-1.4401	0.245	-1.5755
Hidden 2 Neuron 5	0.10023	0.60188	-0.27159	0.41592
Hidden 2 Neuron 6	-0.30166	0.16904	-0.2453	-0.21051
Hidden 2 Neuron 7	-1.283	-0.35916	-0.17708	-0.86378
Hidden 2 Neuron 8	-0.11652	-0.45793	0.27111	0.35243
Hidden 2 Neuron 9	-3.6305	1.1681	0.42808	0.44174
Hidden 2 Neuron 10	-0.57565	-0.71235	0.030133	0.60822
Hidden 2 Neuron 11	0.78198	-0.027381	-2.5173	-0.023424
Hidden 2 Neuron 12	-0.25266	0.04734	-0.34934	2.1482
Hidden 2 Neuron 13	0.60337	-0.6942	1.4654	-1.2087
Hidden 2 Neuron 14	4.8783	2.476	0.39282	-1.0294

Table 3.15. Weights of the trained neural network for case 2 – between five of second hidden layer neurons (1-5) and third hidden layer neurons (14)

	Hidden 2 Neuron 1	Hidden 2 Neuron 2	Hidden 2 Neuron 3	Hidden 2 Neuron 4	Hidden 2 Neuron 5
Hidden 3 Neuron 1	0.48792	1.5911	0.70391	-1.5683	1.2768
Hidden 3 Neuron 2	-1.1829	1.7719	-0.83676	0.55071	-0.63037
Hidden 3 Neuron 3	-1.409	0.34674	-1.467	0.36564	-0.60663
Hidden 3 Neuron 4	0.22773	-0.13991	0.97268	0.94437	-1.4344
Hidden 3 Neuron 5	-0.77096	0.86034	-1.1853	1.4272	0.11966
Hidden 3 Neuron 6	2.5949	3.0229	0.52493	-0.1708	-2.1661
Hidden 3 Neuron 7	0.72154	0.60897	0.014369	-0.22702	0.18548
Hidden 3 Neuron 8	0.39692	0.40231	1.4766	-0.48711	1.3766
Hidden 3 Neuron 9	0.56237	-1.9749	-0.74059	0.040125	0.62056
Hidden 3 Neuron 10	-0.35604	2.6759	-0.53756	1.2463	-0.45025
Hidden 3 Neuron 11	1.204	0.5165	-0.29581	-0.30227	-0.80392
Hidden 3 Neuron 12	0.56269	-0.96393	1.4105	0.49196	0.41359
Hidden 3 Neuron 13	-0.68732	-0.85675	0.33881	0.23779	-0.35665
Hidden 3 Neuron 14	-0.61021	-0.51551	2.3447	0.34212	0.071467

Table 3.16. Weights of the trained neural network for case 2 – between five of second hidden layer neurons (6-10) and third hidden layer neurons (14)

	Hidden 2 Neuron 6	Hidden 2 Neuron 7	Hidden 2 Neuron 8	Hidden 2 Neuron 9	Hidden 2 Neuron 10
Hidden 3 Neuron 1	-0.54047	0.13995	2.4254	-3.2544	0.85881
Hidden 3 Neuron 2	-0.15006	-0.27567	-2.4397	-0.41347	0.22406
Hidden 3 Neuron 3	-0.6843	-0.55949	-0.57878	-2.1559	0.33745
Hidden 3 Neuron 4	0.68274	-0.98037	0.75695	-3.8808	-0.65833
Hidden 3 Neuron 5	0.082187	1.629	0.95974	-4.5351	-0.46093
Hidden 3 Neuron 6	1.4622	-0.92506	0.11408	0.70707	-0.47452
Hidden 3 Neuron 7	0.048872	-0.922	0.21467	0.71919	-0.22168
Hidden 3 Neuron 8	0.4965	-0.68202	1.5046	1.3107	-0.68548
Hidden 3 Neuron 9	-1.1255	0.84017	-1.0205	1.9879	0.15822
Hidden 3 Neuron 10	-0.045041	-0.79547	-1.6868	0.040975	0.12995
Hidden 3 Neuron 11	-0.059603	-0.99046	-0.8073	-0.54721	-0.21513
Hidden 3 Neuron 12	0.86512	-0.7559	-0.83564	-1.094	-0.28217
Hidden 3 Neuron 13	0.043486	0.17555	-0.46567	0.34901	0.31948
Hidden 3 Neuron 14	1.5109	2.1869	-1.4442	2.6571	-0.2287

Table 3.17. Weights of the trained neural network for case 2 – between five of second hidden layer neurons (11-14) and third hidden layer neurons (14)

	Hidden 2 Neuron 11	Hidden 2 Neuron 12	Hidden 2 Neuron 13	Hidden 2 Neuron 14
Hidden 3 Neuron 1	-0.33401	-0.28126	-2.0985	-0.020685
Hidden 3 Neuron 2	0.19553	0.77562	-0.67455	0.41775
Hidden 3 Neuron 3	0.31976	0.24627	-0.47507	0.61992
Hidden 3 Neuron 4	-3.5536	-0.31732	1.2575	2.3676
Hidden 3 Neuron 5	-0.20153	0.054166	0.19574	0.91705
Hidden 3 Neuron 6	-1.0355	0.045199	-0.67906	-1.1842
Hidden 3 Neuron 7	-0.0112	-0.081335	-0.079661	-0.50929
Hidden 3 Neuron 8	0.21992	-0.34582	0.31342	1.7256
Hidden 3 Neuron 9	0.29966	-0.73143	2.2809	-1.6553
Hidden 3 Neuron 10	3.3501	1.2815	-1.6182	-0.076039
Hidden 3 Neuron 11	0.38135	0.1119	-0.27676	0.32178
Hidden 3 Neuron 12	0.14668	1.7087	1.1373	0.16558
Hidden 3 Neuron 13	-0.33798	-0.0051902	0.4454	-0.41927
Hidden 3 Neuron 14	0.89925	0.42521	1.6836	-2.7714

Table 3.18. Weights of the trained neural network for case 2 – between third hidden layer neurons (14) and output neuron (1)

	Hidden 3 Neuron 1	Hidden 3 Neuron 2	Hidden 3 Neuron 3	Hidden 3 Neuron 4	Hidden 3 Neuron 5
Output neuron	-3.0675	-1.1443	-0.99036	-0.18037	-0.41857
	Hidden 3 Neuron 6	Hidden 3 Neuron 7	Hidden 3 Neuron 8	Hidden 3 Neuron 9	Hidden 3 Neuron 10
Output neuron	0.24138	-5.8344	1.5425	-1.8328	2.5842
	Hidden 3 Neuron 11	Hidden 3 Neuron 12	Hidden 3 Neuron 13	Hidden 3 Neuron 14	
Output neuron	-4.2545	-1.7598	-4.9477	0.96746	

Table 3.19. Bias weights of the trained neural network for case 2

Neuron	Bias	Neuron	Bias	Neuron	Bias
Hidden 1 Neuron 1	-1.9736	Hidden 2 Neuron 1	1.3712	Hidden 3 Neuron 1	1.2874
Hidden 1 Neuron 2	-1.2323	Hidden 2 Neuron 2	0.83569	Hidden 3 Neuron 2	-0.8938
Hidden 1 Neuron 3	1.223	Hidden 2 Neuron 3	-2.3163	Hidden 3 Neuron 3	0.92272
Hidden 1 Neuron 4	-2.0328	Hidden 2 Neuron 4	-0.33238	Hidden 3 Neuron 4	-2.8767
Hidden 1 Neuron 5	-2.4289	Hidden 2 Neuron 5	0.056568	Hidden 3 Neuron 5	1.272
Hidden 1 Neuron 6	-0.50732	Hidden 2 Neuron 6	-0.83125	Hidden 3 Neuron 6	0.074583
Hidden 1 Neuron 7	-0.13996	Hidden 2 Neuron 7	-0.78137	Hidden 3 Neuron 7	1.796
Hidden 1 Neuron 8	4.2614	Hidden 2 Neuron 8	0.041732	Hidden 3 Neuron 8	-0.60849
Hidden 1 Neuron 9	1.0754	Hidden 2 Neuron 9	-1.5447	Hidden 3 Neuron 9	1.5064
Hidden 1 Neuron 10	0.55214	Hidden 2 Neuron 10	-2.0486	Hidden 3 Neuron 10	4.245
Hidden 1 Neuron 11	-0.37748	Hidden 2 Neuron 11	-1.4753	Hidden 3 Neuron 11	0.487
Hidden 1 Neuron 12	0.50327	Hidden 2 Neuron 12	-0.15825	Hidden 3 Neuron 12	0.65162
Hidden 1 Neuron 13	1.4872	Hidden 2 Neuron 13	-0.57347	Hidden 3 Neuron 13	1.1572
Hidden 1 Neuron 14	-2.2588	Hidden 2 Neuron 14	7.7324	Hidden 3 Neuron 14	-2.9705
Output N.	1.2917				

Comparative graphs of Ansys and trained ANN results (case 2 – 3 hidden layers – 14 hidden layer neurons for each hidden layer) for some new values of input variables are shown in the following figures.

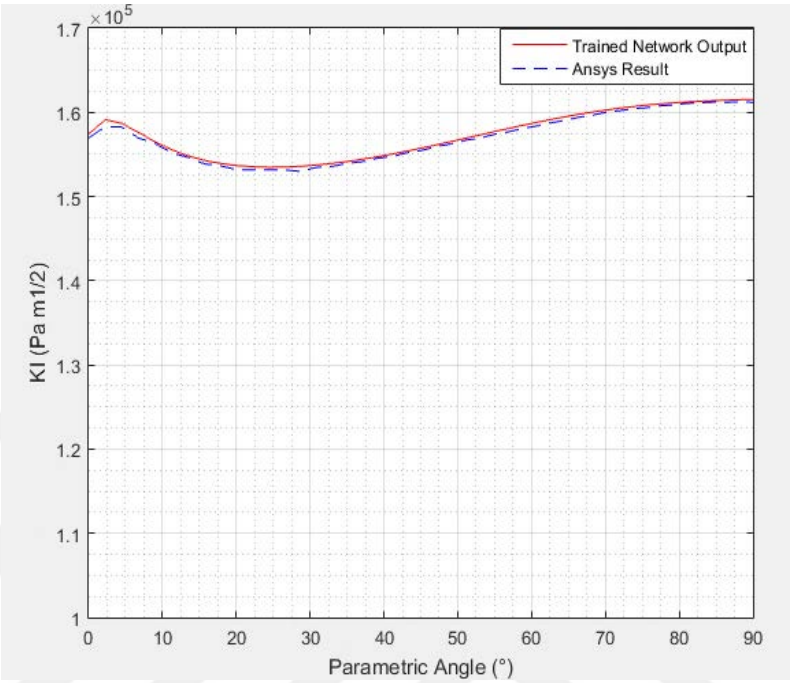


Figure 3.24. Variation of SIF along the crack front for case 2 (a=0.0168 m, a/c=0.8, h=0.05 m / a/h=0.336, a/t=0.27, deviation=0.21%)

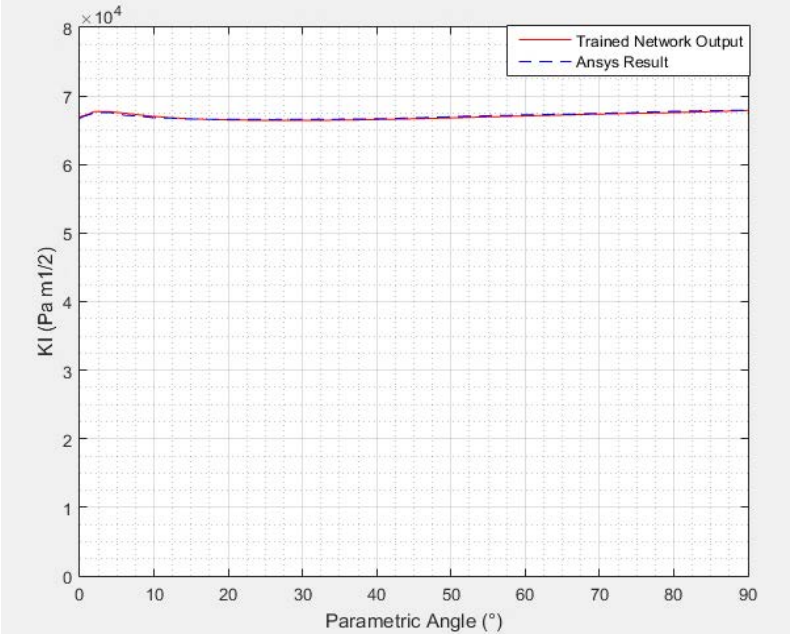
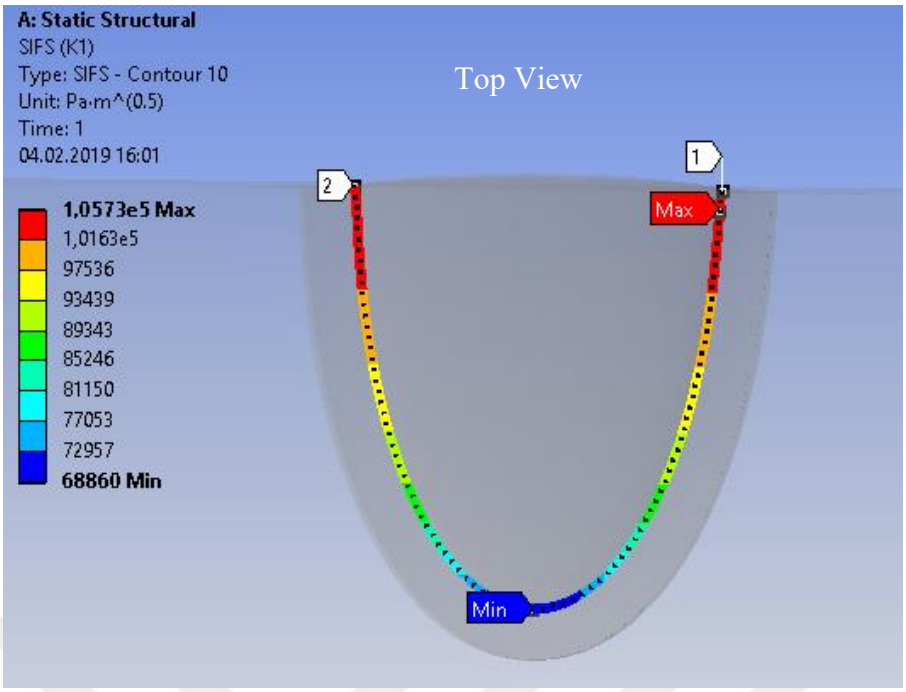
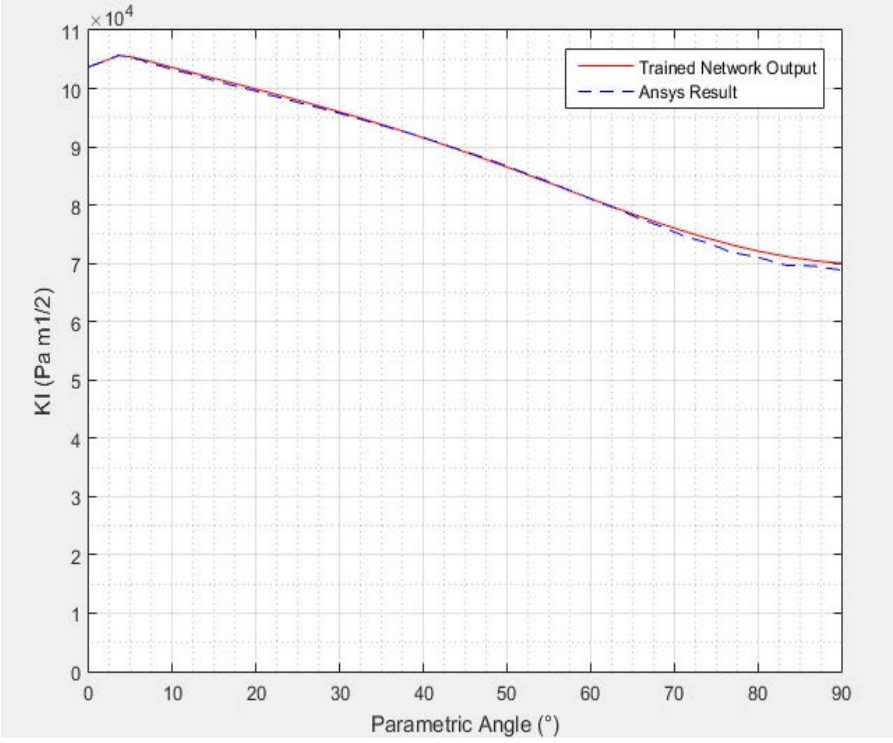


Figure 3.25. Variation of SIF along the crack front for case 2 (a=0.005 m, a/c=1, h=0.003 m / a/h=1.67, a/t=0.35, deviation=0.19%)

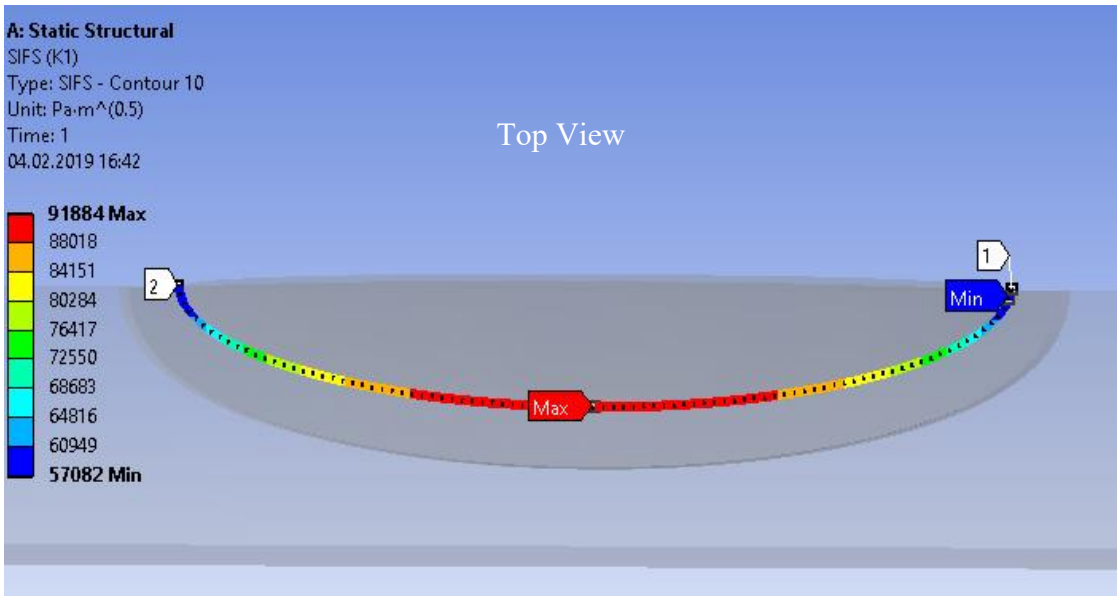


(a)

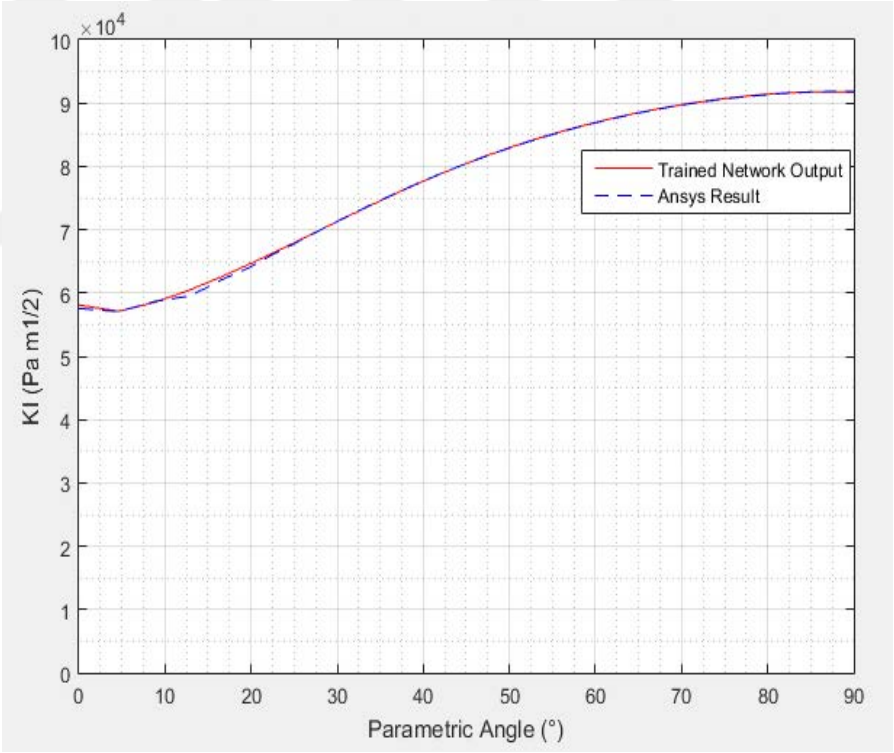


(b)

Figure 3.26. Variation of SIF along the crack front for case 2 – (a) Ansys result and (b) Comparative graph (a=0.01125 m, a/c=2.25, h=0.015 m / a/h=0.75, a/t=0.25, deviation=0.43%)

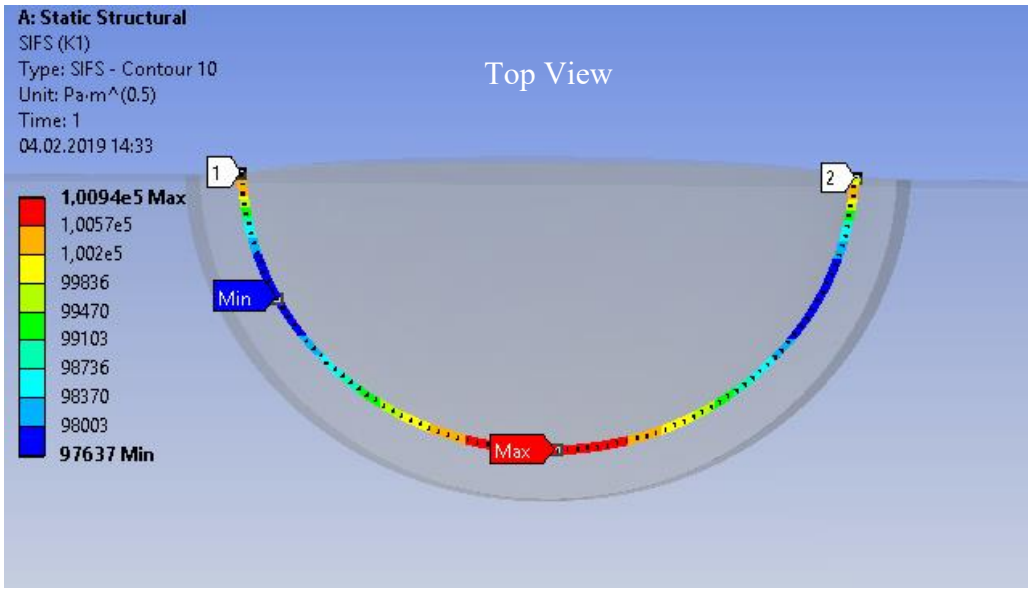


(a)

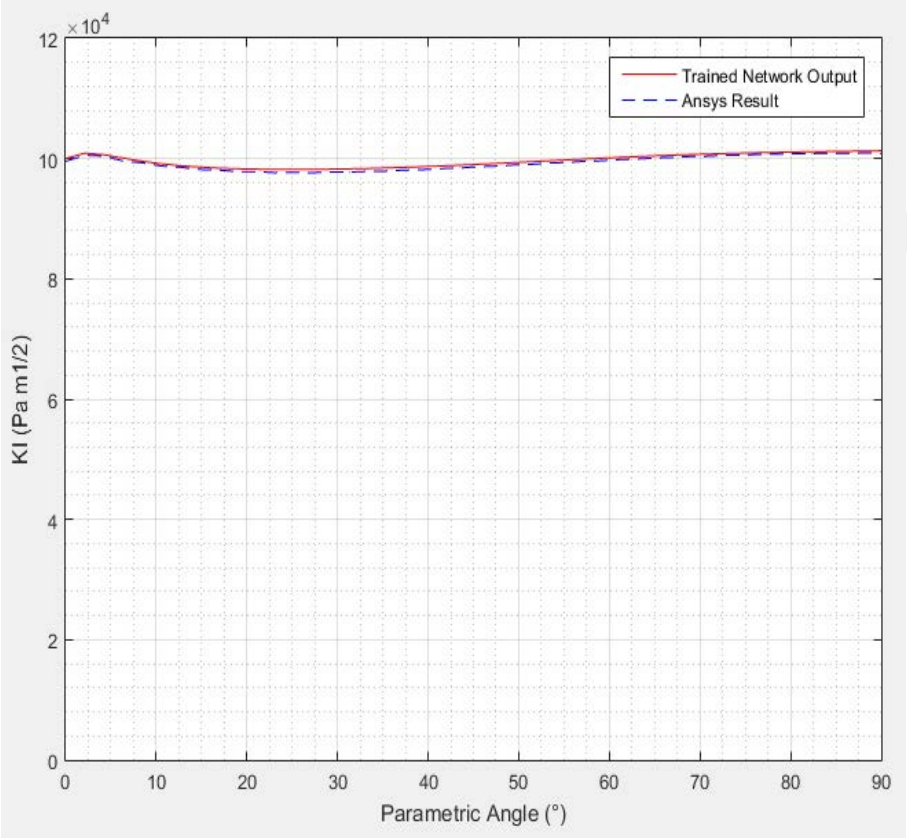


(b)

Figure 3.27. Variation of SIF along the crack front for case 2 – (a) Ansys result and (b) Comparative graph (a=0.003 m, a/c=0.3, h=0.015 m / a/h=0.2, a/t=0.25, deviation=0.15%)

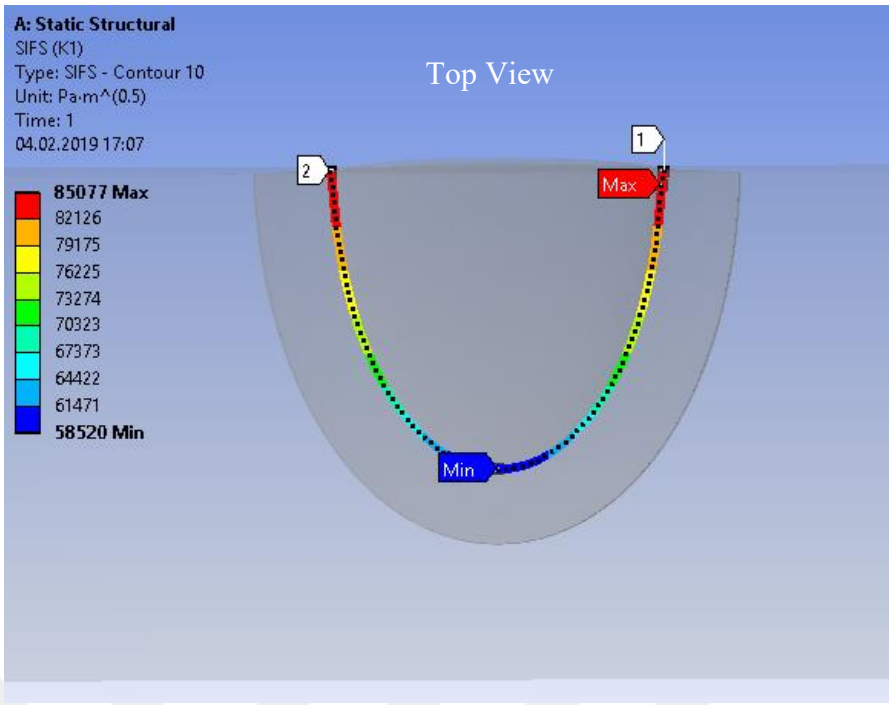


(a)

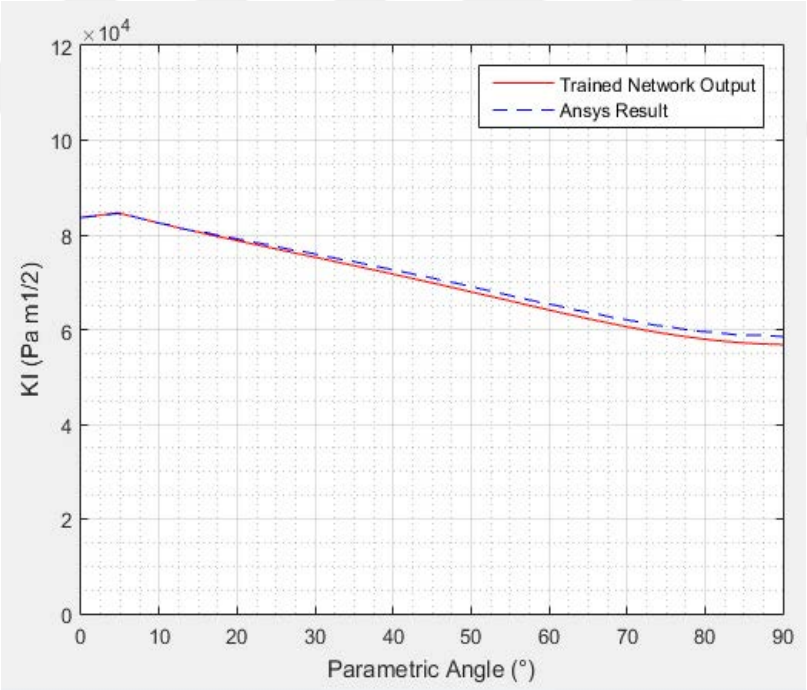


(b)

Figure 3.28. Variation of SIF along the crack front for case 2 – (a) Ansys result and (b) Comparative graph (a=0.0081 m, a/c=0.9, h=0.013 m / a/h=0.623, a/t=0.33, deviation=0.39%)



(a)



(b)

Figure 3.29. Variation of SIF along the crack front for case 2 – (a) Ansys result and (b) Comparative result (a=0.0054 m, a/c=1.8, h=0.025 m / a/h=0.216, a/t=0.3, deviation=1.25%)

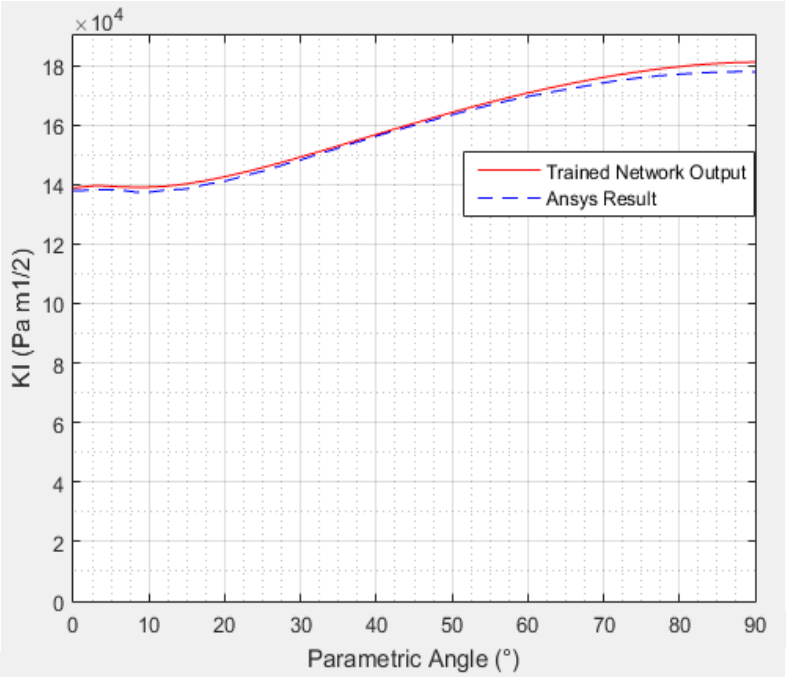


Figure 3.30. Variation of SIF along the crack front for case 2 (a=0.014 m, a/c=0.5, h=0.05 m / a/h=0.28, a/t=0.32, deviation=0.93%)

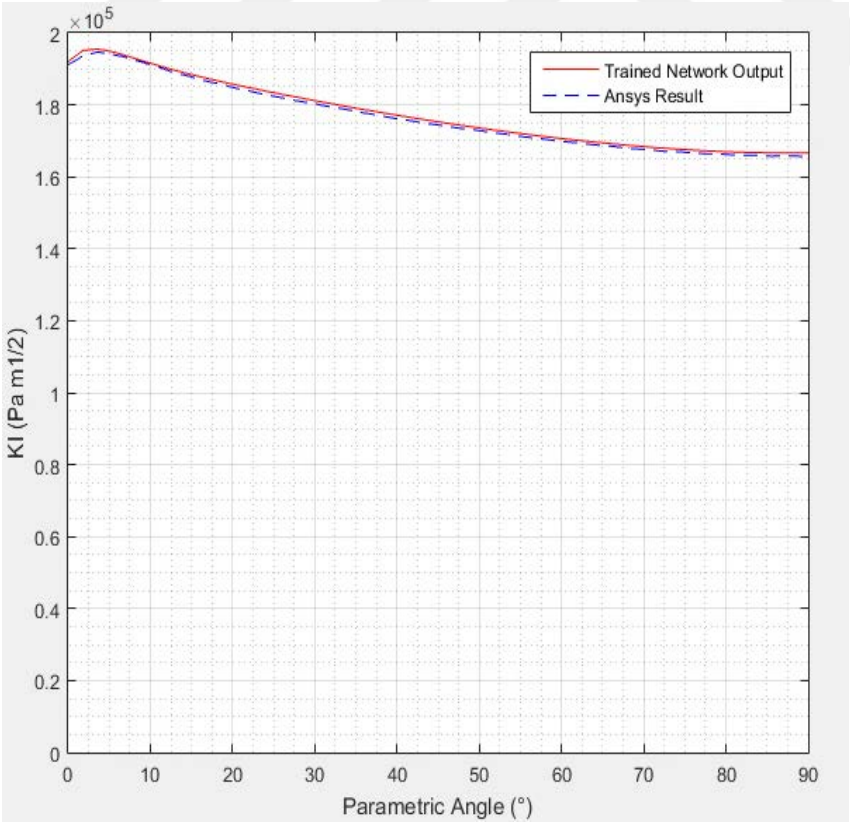
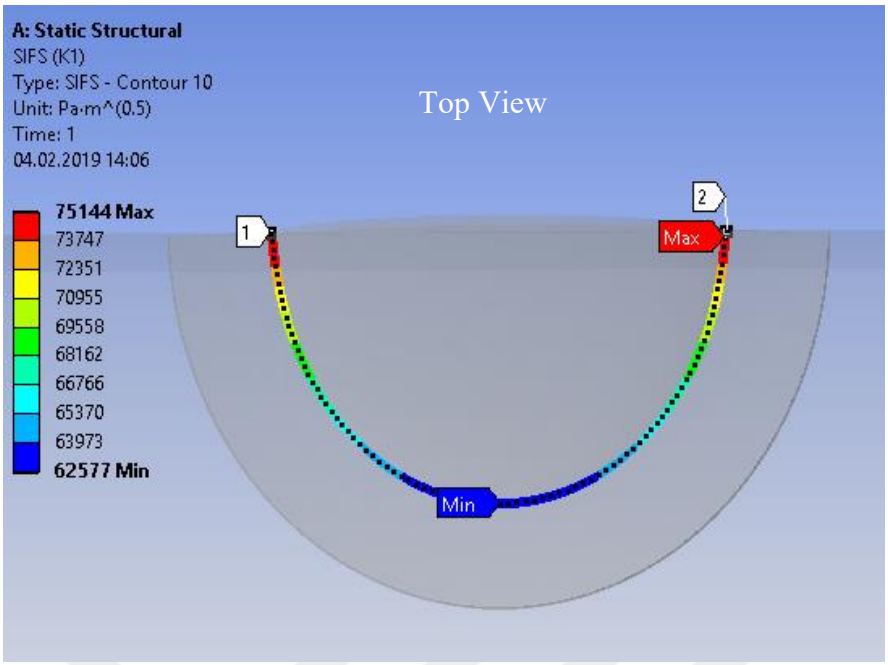
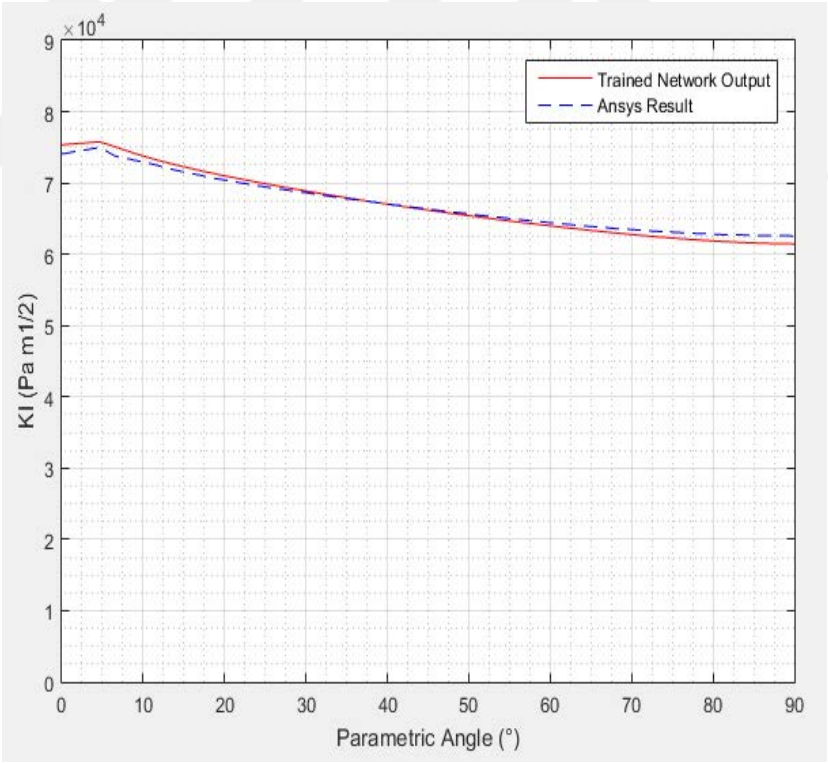


Figure 3.31. Variation of SIF along the crack front for case 2 (a=0.024 m, a/c=1.2, h=0.15 m / a/h=0.16, a/t=0.36, deviation=0.46%)



(a)



(b)

Figure 3.32. Variation of SIF along the crack front for case 2 (a) Ansys result (b) Comparative graph (a=0.0036 m, a/c=1.2, h=0.02 m / a/h=0.18, a/t=0.2, deviation=0.85%)

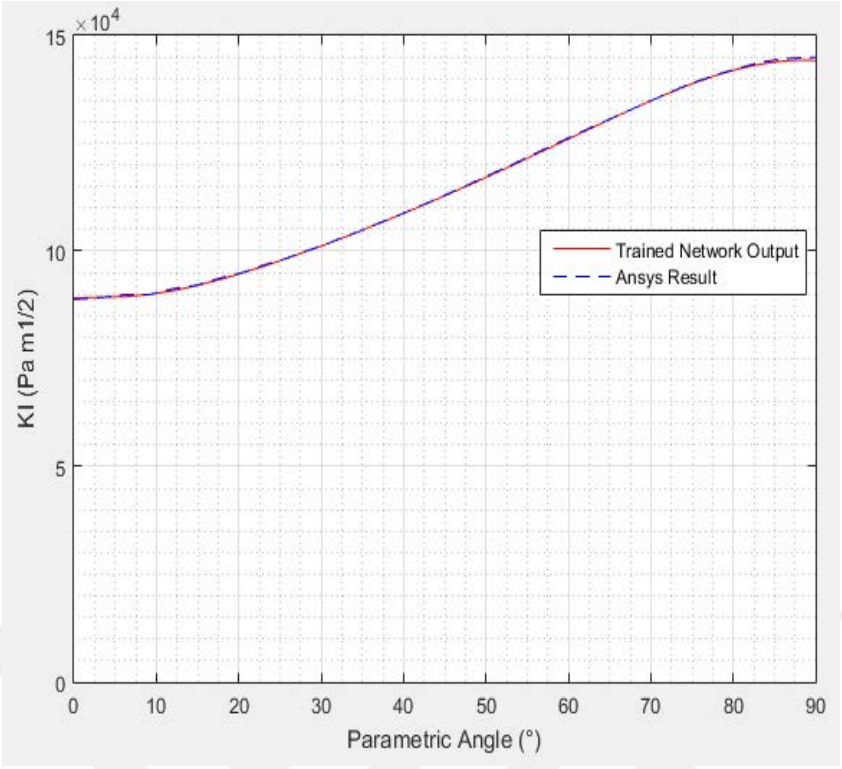


Figure 3.33. Variation of SIF along the crack front for case 2 (a=0.008 m, a/c=0.5, h=0.005 m / a/h=1.6, a/t=0.45, deviation=0.17%)

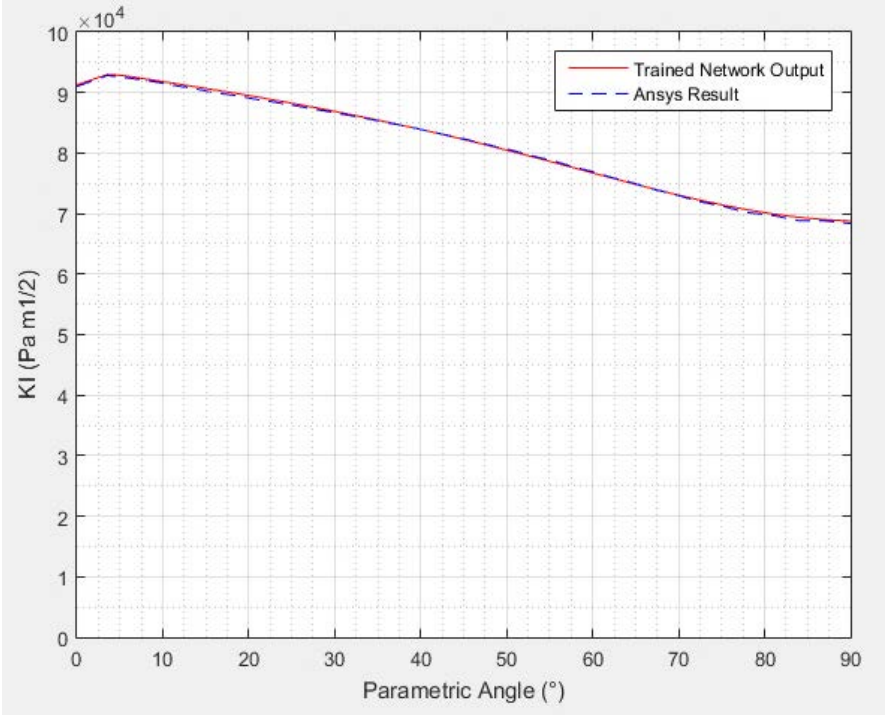


Figure 3.34. Variation of SIF along the crack front for case 2 (a=0.01 m, a/c=2, h=0.008 m a/h=1.25, a/t=0.35, deviation=0.28%)

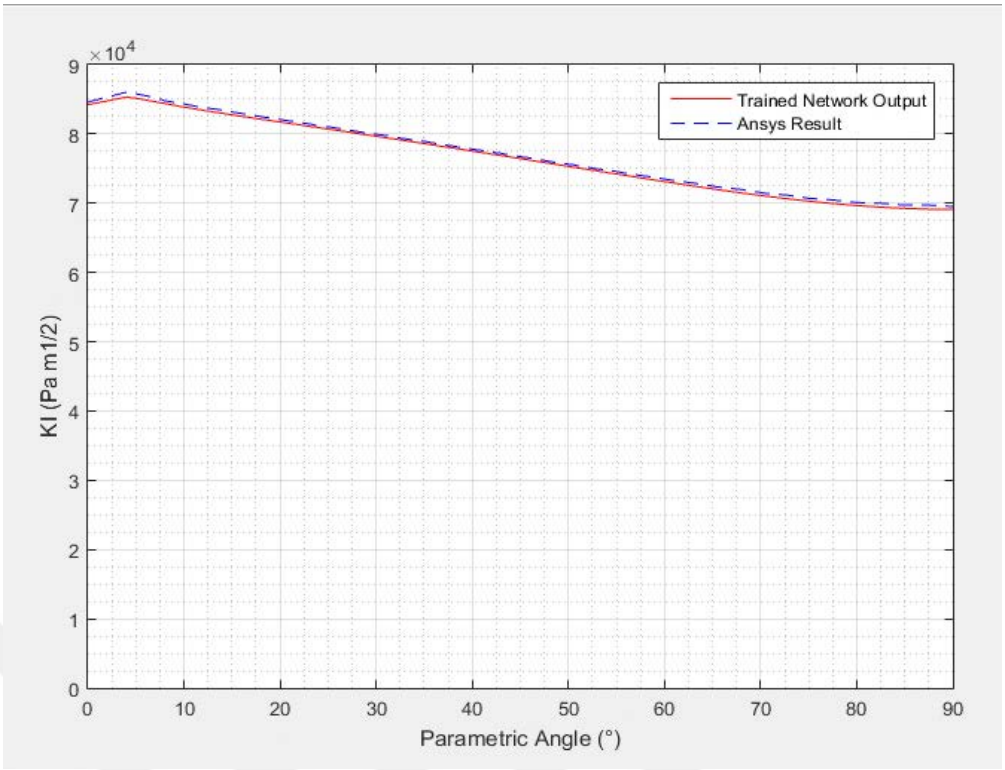


Figure 3.35. Variation of SIF along the crack front for case 2 (a=0.006 m, a/c=1.5, h=0.01 m / a/h=0.6, a/t=0.42, deviation=0.5%)

In the Ansys simulations in case 1 and case 2, applied stress value was taken as 1 MPa (unit value). So some of the simulations were repeated, taking the applied stress as 100 MPa to check the correlation between the applied stress and corresponding SIF. As it was expected, as a result of this control study, deviation value was calculated as 0.02% more or less. Since the deviation value was too small, it can be considered that there is a linear relationship between the applied stress and SIF. So these trained ANN models can be used for different values of applied stresses (MPa). For instance, if the applied stress value is 100 MPa, it is enough to multiply the trained network output by 100 so as to get the result.

3.3. Conclusions and Recommendations

As a consequence of this thesis, two neural network models which have the potential to estimate SIF values, were developed for two different types of semi elliptical surface cracked plates / bodies.

For the first case (two semi elliptical surface cracked plate-one of them is at the front side and the other one is at the back side), trained model contains 2 hidden layer and 15 neuron

for each hidden layer. For the second case (four semi elliptical surface cracked plate-two of them is at the front side and the others are at the back side), final neural network model consists of 3 hidden layer and 14 neuron for each hidden layer. Since the second case / problem is more complex than case 1, ANN model for the second case is more complicated (there is an extra input node h-vertical distance between two parallel semi elliptical cracks) than case 1 in order to get the best relationship between input, output and target data.

As presented in the previous section, deviation values of these neural network models which can be used for case 1 and case 2 are 0.32% and 0.49% (deviation between the Ansys result and the trained ANN model result) respectively. 760 new data were used so as to check the accuracy of the model and calculate the deviation percentage for case 1. Also 1139 test data were used in order to determine the deviation percentage for case 2. These deviation values are well enough for SIF estimation.

There are 331 and 519 weights for the trained model of case 1 and case 2 respectively. Weights of the trained models were given in the analysis and results section. These weights are very important for ANN modeling. Because if these weights are known, there is no need to use the trained ANN Matlab nntool module file. Only a simple code which is created using the weights of the trained models and the transfer functions, is enough to estimate the values of SIFs of case 1 and case 2.

By means of these ANN models, there is no need to do any time consuming simulations and any numerical calculations. These trained ANN models can be used like an explicit SIF formula. Moreover, these trained network models can calculate / estimate SIF values of too many cases (500, 1000, 10000 etc.) simultaneously within seconds with good margin of error and therefore it is a time saving process. Because, some of the simulations which are executed to calculate SIF value of the cracked plate in Ansys Workbench, take one hour more or less.

If the variety of training data is increased, better deviation values may be calculated. Moreover, if more test data are used to check the accuracy of the trained model, better ANN structures may be obtained.

Neural network modeling process can also be used to estimate SIF values of other complicated bodies (have no analytical formula to calculate SIF) which contain different types of cracks like semi elliptical surface crack, edge crack, embedded elliptical crack, center crack, etc., following the procedure in this thesis.

REFERENCES

[1] S. Kumar, S.V. Barai, Concrete Fracture Models and Applications, Introduction to Fracture Mechanics of Concrete, Chapter 1, Springer, **2011**

[2] C. H. Wang, Introduction to Fracture Mechanics, Technical Report, 1, **1996**

[3] S. Rusia, K. K. Pathak, Application of Artificial Neural Network to Analyze Hexagonal Plate with Hole Considering Different Geometrical and Loading Parameters, **2016**

[4] S. Rusia, K. K. Pathak, Application of Artificial Neural Network for Analysis of Triangular Plate with Hole Considering Different Geometrical and Loading Parameters, Open Journal of Civil Engineering, **2016**

[5] P. E. Nicholas, K. P. Padmanaban, D. Vasudevan, I. J. Selvaraj, Neural Network Based Buckling Strength Prediction of Laminated Composite Plate with Central Cutout, Applied Mechanics and Materials, **2014**

[6] Z. Ali, K.E.S. Meysam, A. Iman, B. Aydin, B. Yashar, Finite Element Method Analysis of Stress Intensity Factor in Different Edge Crack Positions and Predicting Their Correlation Using Neural Network Method, **2013**

[7] M.A. Kutuk, N. Atmaca, I. H. Guzelbey, Explicit Formulation of SIF Using Neural Networks for Opening Mode of Fracture, **2006**

[8] L. S. Jabur, N. R. Mohsin, Stress Intensity Factor for Double Edge Cracked Finite Plate Subjected to Tensile Stress, **2015**

[9] P. Rubio, B. M. Abella, L. Rubio, Neural Approach to Estimate the Stress Intensity Factor of Semi Elliptical Cracks in Rotating Cracked Shafts in Bending, **2017**

[10] N. Kilic, B. Ekici, S. Hartomacioglu, Determination of Penetration Depth at High Velocity Impact Using Finite Element Method and Artificial Neural Network Tools, **2015**

[11] P. Jonsen, H. A. Haggblad, Fracture Energy Based Constitutive Models for Tensile Fracture of Metal Powder Compacts, 1, **2007**

[12] W. F. Hosford, Mechanical Behaviour of Materials, Second Edition, Cambridge University Press, 228, **2010**

[13] W. F. Hosford, Mechanical Behaviour of Materials, Second Edition, Cambridge University Press, 229, **2010**

[14] R. G. Budynas, Advanced Strength and Applied Stress Analysis, Second Edition, McGraw Hill, 520, **1999**

[15] W. F. Hosford, Mechanical Behaviour of Materials, Second Edition, Cambridge University Press, 230-231, **2010**

[16] R. G. Budynas, Advanced Strength and Applied Stress Analysis, Second Edition, McGraw Hill, 523-524, **1999**

[17] B. Yıldırım, Ansys Yapısal Analiz Uygulamaları, Bölüm 7, Kırılma Mekaniği Analizleri, Anova, 223, **2017**

[18] M. Sharobeam, J. D. Landes, Numerical Solutions for Ductile Fracture Behaviour of Semi Elliptical Surface Crack, **1999**

[19] S. De, Mane 4240 / CIVL 4240 - Introduction to Finite Elements, https://slideplayer.com/slide/4033899/ (**19/12/2018**)

[20] S. Balkissoon, S. R. Gunakala, D. Comissiong, V. Job, The Comparative Analysis Of The Two Dimensional Laplace Equation Using The Galerkin Finite Element Method With The Exact Solution For Various Domains With Triangular Elemental Meshing, 3, **2015**

[21] https://www.quora.com/What-is-the-major-difference-between-a-neural-network - and-an-artificial-neural-network, (**19/12/2018**)

[22] M. R. Veronez, S. F. Souza, M. T. Matsuoka, A. Reinhardt, R. M. Silva, Regional Mapping of the Geoid Using GNSS (GPS) Measurements and an Artificial Neural Network, **2011**

[23] https://www.analyticsvidhya.com/blog/2016/03/introduction-deep-learning- funda mentalsneural-networks/, (**19/12/2018**)

[24] P. Samui, Handbook of Research on Advanced Computational Techniques for Simulation Based Engineering, 142, **2015**

[25] Defining a semi elliptical crack, https://www.sharcnet.ca/Software/Ansys/17.0/en-us/help/wb_sim/ds_frac_mesh_define_crack.html, (**30/12/2018**)

[26] F. Bre, J. M. Gimenez, V. D. Fachinotti, Prediction of Wind Pressure Coefficients on Building Surfaces Using Artificial Neural Networks, 4, **2017**

[27] M. H. Beale, M. T. Hagan, H. B. Demuth, Matlab R2017a Neural Network Toolbox User's Guide, Mathworks, Chapter 9, Advanced Topics, 9-29, **2017**

[28] H. Yu, B. M. Wilamowski, Industrial Electronics Handbook, Vol. 5 Intelligent Systems, Chapter 12, Levenberg Marquardt Training, Crc Press, 12-13, **2011**

[29] D. Kim, S. S. Roy, T. Lansivaara, R. Deo, P. Samui, Handbook of Research on Predictive Modelling and Optimization Methods in Science and Engineering, IGI Global, 332, **2018**

[30] D. Kriesel, A Brief Introduction to Neural Networks, 100, **2005**

[31] Multilayer Shallow Neural Network Architecture, https://de.mathworks.com /help/deeplearning/ug/multilayer-neural-network-architecture.html (**07/01/2019**)

[32] M. H. Beale, M. T. Hagan, H. B. Demuth, Matlab R2017a Neural Network Toolbox User's Guide, Mathworks, Chapter 9, Advanced Topics, 9-31, **2017**

[33] What is Underfitting and Overfitting in Machine Learning and How to Deal with It, https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning -and-how-to-deal-with-it-6803a989c76 (**13/01/2019**)

[34] Overfitting / Underfitting How Well Does How Well Does Your Model Fit, https://meditationsonbianddatascience.com/2017/05/11/overfitting-underfitting-how-well-does-your-model-fit/ (**13/01/2019**)

[35] M. H. Beale, M. T. Hagan, H. B. Demuth, Matlab R2017a Neural Network Toolbox User's Guide, Mathworks, Chapter 9, Advanced Topics, Early Stopping, 9-35, **2017**

HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
THESIS/DISSERTATION ORIGINALITY REPORT

HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
TO THE DEPARTMENT OF MECHANICAL ENGINEERING

Date: 27/05/2019

Thesis Title / Topic: Estimation of Stress Intensity Factors for Cracked Finite Plates Using a Hybrid Model of Artificial Neural Network and Finite Element Method (ANSYS)

According to the originality report obtained by myself/my thesis advisor by using the *Turnitin* plagiarism detection software and by applying the filtering options stated below on 22/04/2019 for the total of 98 pages including the a) Title Page, b) Introduction, c) Main Chapters, d) Conclusion sections of my thesis entitled as above, the similarity index of my thesis is 6 %.

Filtering options applied:

1. Bibliography/Works Cited excluded
2. Quotes excluded / included
3. Match size up to 5 words excluded

I declare that I have carefully read Hacettepe University Graduate School of Sciene and Engineering Guidelines for Obtaining and Using Thesis Originality Reports; that according to the maximum similarity index values specified in the Guidelines, my thesis does not include any form of plagiarism; that in any future detection of possible infringement of the regulations I accept all legal responsibility; and that all the information I have provided is correct to the best of my knowledge.

I respectfully submit this for approval.

27/05/2019

Name Surname: Yusuf YABİR

Student No: N15237213

Department: Mechanical Engineering

Program: Mechanical Engineering

Status: ☒ Masters ☐ Ph.D. ☐ Integrated Ph.D.

ADVISOR APPROVAL

APPROVED.

Prof. Dr. Bora YILDIRIM

CURRICULUM VITAE

Credentials

Name, Surname	: Yusuf, YABİR
Place of Birth	: Ankara
Marital Status	: Married
E-mail	: yyabir@hotmail.com
Address	: Beşikkaya Mh. 1960. Sk. 6/12 Altındağ/ANKARA

Education

B.Sc.	: Gazi University, Department of Mechanical Engineering (2007-2011)
M.Sc.	: Hacettepe University, Department of Mechanical Engineering (2016-2019)

Foreign Languages

English

Work Experiences

2016-2018	: Research Assistant, Hacettepe University Department of Mechanical Engineering, ANKARA
2018-	: Mechanical Engineer, Gazi University, ANKARA