ARTIFICIAL NEURAL NETWORK BASED FAULT LOCATION FOR TRANSMISSION LINES

Suhaas Bhargava Ayyagari
University of Kentucky, suhaas.bhargava@gmail.com

Right click to open a feedback form in a new tab to let us know how this document benefits you.

Recommended Citation
https://uknowledge.uky.edu/gradschool_theses/657

This Thesis is brought to you for free and open access by the Graduate School at UKnowledge. It has been accepted for inclusion in University of Kentucky Master's Theses by an authorized administrator of UKnowledge. For more information, please contact UKnowledge@lsv.uky.edu.
STUDENT AGREEMENT:

I represent that my thesis or dissertation and abstract are my original work. Proper attribution has been given to all outside sources. I understand that I am solely responsible for obtaining any needed copyright permissions. I have obtained and attached hereto needed written permission statements(s) from the owner(s) of each third-party copyrighted matter to be included in my work, allowing electronic distribution (if such use is not permitted by the fair use doctrine).

I hereby grant to The University of Kentucky and its agents the non-exclusive license to archive and make accessible my work in whole or in part in all forms of media, now or hereafter known. I agree that the document mentioned above may be made available immediately for worldwide access unless a preapproved embargo applies.

I retain all other ownership rights to the copyright of my work. I also retain the right to use in future works (such as articles or books) all or part of my work. I understand that I am free to register the copyright to my work.

REVIEW, APPROVAL AND ACCEPTANCE

The document mentioned above has been reviewed and accepted by the student’s advisor, on behalf of the advisory committee, and by the Director of Graduate Studies (DGS), on behalf of the program; we verify that this is the final, approved version of the student’s dissertation including all changes required by the advisory committee. The undersigned agree to abide by the statements above.

Suhaas Bhargava Ayyagari, Student

Dr. Yuan Liao, Major Professor

Dr. Zhi Chen, Director of Graduate Studies
ARTIFICIAL NEURAL NETWORK BASED FAULT LOCATION FOR
TRANSMISSION LINES

____________________________________

THESIS

____________________________________

A thesis submitted in partial fulfillment of the
requirements for the degree of Masters of Science in Electrical Engineering
in the College of Engineering at the University of Kentucky

By

Suhaas Bhargava Ayyagari

Lexington, Kentucky

Director: Dr. Yuan Liao, Associate Professor of Electrical & Computer Engineering
ARTIFICIAL NEURAL NETWORK BASED FAULT LOCATION FOR TRANSMISSION LINES

This thesis focuses on detecting, classifying and locating faults on electric power transmission lines. Fault detection, fault classification and fault location have been achieved by using artificial neural networks. Feedforward networks have been employed along with backpropagation algorithm for each of the three phases in the Fault location process. Analysis on neural networks with varying number of hidden layers and neurons per hidden layer has been provided to validate the choice of the neural networks in each step. Simulation results have been provided to demonstrate that artificial neural network based methods are efficient in locating faults on transmission lines and achieve satisfactory performances.

KEY WORDS: Artificial Neural Networks, Feedforward networks, Backpropagation Algorithm, Levenberg-Marquardt algorithm.

Suhaas Bhargava Ayyagari

(12/6/2011)
ARTIFICIAL NEURAL NETWORK BASED FAULT LOCATION FOR TRANSMISSION LINES

By

Suhaas Bhargava Ayyagari

Director of Thesis:
Dr. Yuan Liao

Director of Graduate Studies:
Dr. Zhi Chen

12/6/2011
Dedicated to my Parents
ACKNOWLEDGEMENTS

Firstly, I would like to sincerely express my gratitude to my advisor and my thesis chair Dr. Yuan Liao for all his support and cooperation without which this thesis would never have been possible. I thank him for his constant patience and the financial support he offered to me during the course of my research.

I would also like to thank Dr. Aaron Cramer and Dr. Paul Dolloff for their approval to be a part of my defense committee and for their invaluable suggestions that shaped this thesis to its best.

Finally, I thank my parents, family and friends for their relentless love and their support both financially and emotionally during the course of my research and thereby making this thesis a success. My sincere regards to the Department of Electrical and Computer Engineering and to the Power and energy Institute of Kentucky for providing all the necessary resources and paving way to the successful completion of my Master’s degree.
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS ........................................................................ iii  
LIST OF TABLES ................................................................................. vi  
LIST OF FIGURES ............................................................................... vii  

CHAPTER ONE: INTRODUCTION ............................................................. 1  
1.1 INTRODUCTION ............................................................................ 1  
1.2 MOTIVATION .............................................................................. 3  
1.3 OUTLINE OF THE THESIS ......................................................... 4  

CHAPTER TWO: LITERATURE REVIEW .................................................. 6  
2.1 POWER PROTECTION SYSTEMS .................................................... 6  
2.2 TRANSMISSION LINE FAULT LOCATION TECHNIQUES ..................... 7  
2.2.1 IMPEDANCE BASED METHODS ............................................... 7  
2.2.1.1 SIMPLE REACTANCE METHOD ........................................ 7  
2.2.1.2 TAKAGI METHOD .......................................................... 8  
2.2.1.3 MODIFIED TAKAGI METHOD ........................................... 9  
2.2.2 TRAVELLING WAVE BASED METHODS ..................................... 10  
2.2.3 NEURAL NETWORKS BASED METHODS ................................... 11  

CHAPTER THREE: NEURAL NETWORKS AND THEIR APPLICATION IN  
TRANSMISSION LINE FAULT LOCATION ........................................... 13  
3.1 INTRODUCTION TO NEURAL NETWORKS ..................................... 13  
3.2 MODEL OF A NEURON ................................................................. 15  
3.3 FEEDFORWARD NETWORKS ......................................................... 17  
3.4 LEARNING STRATEGIES ............................................................... 19  

CHAPTER FOUR: FAULT LOCATION IN POWER TRANSMISSION LINES  
USING NEURAL NETWORKS ............................................................... 24  
4.1 INTRODUCTION ........................................................................... 24  
4.2 MODELLING THE POWER TRANSMISSION LINE SYSTEM ............... 24  
4.3 OUTLINE OF THE PROPOSED SCHEME ...................................... 26  
4.4 DATA PRE-PROCESSING .............................................................. 28  
4.5 OVERVIEW OF THE TRAINING PROCESS .................................... 31  
4.6 OVERVIEW OF THE TESTING PROCESS ...................................... 32  

CHAPTER FIVE: EXPERIMENTAL RESULTS ........................................... 34  
5.1 FAULT DETECTION ........................................................................ 34  
5.1.1 TRAINING THE FAULT DETECTION NEURAL NETWORK ............ 34  
5.1.2 TESTING THE FAULT DETECTION NEURAL NETWORK ............ 37  
5.2 FAULT CLASSIFICATION .............................................................. 41  
5.2.1 TRAINING THE FAULT CLASSIFIER NEURAL NETWORK ............ 42  
5.2.2 TESTING THE FAULT CLASSIFIER NEURAL NETWORK ............ 48  
5.3 FAULT LOCATION ......................................................................... 52  
5.3.1 SINGLE LINE – GROUND FAULTS .......................................... 52  
5.3.1.1 TRAINING THE NEURAL NETWORK  
FOR SINGLE LINE – GROUND FAULT LOCATION .............................. 52  
5.3.1.2 TESTING THE NEURAL NETWORK FOR  
SINGLE LINE – GROUND FAULT LOCATION .................................... 60  

iv
5.3.2 LINE–LINE FAULTS...........................................................................................................63
  5.3.2.1 TRAINING THE NEURAL NETWORK FOR LINE–LINE FAULT LOCATION..................63
  5.3.2.2 TESTING THE NEURAL NETWORK FOR LINE–LINE FAULT LOCATION..................69
5.3.3 DOUBLE LINE–GROUND FAULTS ....................................................................................72
  5.3.3.1 TRAINING THE NEURAL NETWORK FOR DOUBLE LINE–GROUND FAULT LOCATION........73
  5.3.3.2 TESTING THE NEURAL NETWORK FOR DOUBLE LINE–GROUND FAULT LOCATION..........81
5.3.4 THREE PHASE FAULTS...................................................................................................85
  5.3.4.1 TRAINING THE NEURAL NETWORK FOR THREE PHASE FAULT LOCATION...............85
  5.3.4.2 TESTING THE NEURAL NETWORK FOR THREE PHASE FAULT LOCATION.................93

CHAPTER SIX: CONCLUSIONS AND SCOPE FOR FUTURE WORK ........................................97
  6.1 CONCLUSIONS..............................................................................................................97

BIBLIOGRAPHY ..................................................................................................................100

VITA ......................................................................................................................................104
LIST OF TABLES

Table 4.1: Sample of inputs to the neural network for various fault cases .............................................30

Table 5.1 Fault classifier ANN outputs for various faults ........................................................................42

Table 5.2 Percentage errors as a function of fault distance and fault resistance for the ANN chosen for single line – ground fault location .................................................................62

Table 5.3 Percentage errors as a function of fault distance and fault resistance for the ANN chosen for line – line fault location ....................................................................................72

Table 5.4 Percentage errors as a function of fault distance and fault resistance for the ANN chosen for double line – ground fault location .................................................................84

Table 5.5 Percentage errors as a function of fault distance and fault resistance for the ANN chosen for three phase fault location .................................................................96
LIST OF FIGURES

Figure 2.1 : Faulted Transmission line illustrating simple-reactance method ................... 8
Figure 2.2 : A single-phase circuit illustrating Takagi method ....................................... 9
Figure 2.3 : Illustration of Travelling wave based fault location ..................................... 10
Figure 3.1 : A basic three-layer architecture of a feedforward ANN ................................. 13
Figure 3.2 : Typical model of a neuron ............................................................................. 15
Figure 3.3 : Step activation function .................................................................................. 16
Figure 3.4 : Piece wise linear activation function ............................................................... 16
Figure 3.5 : Sigmoid unipolar activation function ............................................................... 16
Figure 3.6 : Bipolar activation function ............................................................................. 17
Figure 3.7 : Structure of a two-layered feedforward network ............................................ 18
Figure 3.8 : Scheme of supervised learning ..................................................................... 20
Figure 3.9 : Structure of back-error propagation algorithm ............................................. 22
Figure 3.10: One-line diagram of the studied system ......................................................... 25
Figure 3.11: Snapshot of the studied model in SimPowerSystems .................................... 25
Figure 3.12: Flowchart depicting the outline of the proposed scheme ............................... 27
Figure 3.13: Data pre-processing illustration .................................................................... 28
Figure 4.1 : Mean-square error performance of the network (6-10-1) ............................... 35
Figure 4.2 : Mean-square error performance of the network (6-10-5-1) ........................... 36
Figure 4.3 : Mean-square error performance of the network (6-10-5-3-1) ........................ 36
Figure 4.4 : Regression fit of the outputs vs. targets for the network (6-10-5-3-1) ........... 37
Figure 4.5 : Confusion matrices for Training, Testing and Validation Phases .............. 38
Figure 4.6 : Overview of the ANN (6-10-5-3-1) chosen for fault detection ..................... 39
Figure 4.7 : Chosen ANN for Fault Detection (6 – 10 – 5 – 3 – 1) ................................... 41
Figure 4.8 : Mean-square error performance of the network with configuration (6-5-31-4) ........................................................................................................................................ 44
Figure 4.9 : Mean-square error performance of the network with configuration (6-5-31-4) ........................................................................................................................................ 44
Figure 4.10 : Mean-square error performance of the network with configuration (6-5-4) ........................................................................................................................................ 45
Figure 4.11 : Mean-square error performance of the network with configuration (6-10-4) ........................................................................................................................................ 46
Figure 4.12 : Mean-square error performance of the network with configuration (6-20-4) ........................................................................................................................................ 46
Figure 4.13 : Mean-square error performance of the network with configuration (6-35-4) ........................................................................................................................................ 47
Figure 4.14 : Regression fit of the Outputs vs. Targets of ANN with configuration (6-35-4) ........................................................................................................................................ 48
Figure 4.15 : Gradient and Validation performance of the ANN with configuration (6-35-4) ........................................................................................................................................ 49
Figure 4.16 : Overview of the ANN with configuration (6-35-4), chosen as fault classifier ........................................................................................................................................ 50
Figure 4.17 : Chosen ANN for Fault Classification (6 – 35 – 4) ........................................ 51
Figure 4.18 : Regression fit of the Outputs vs. Targets with configuration (6-5-5-1) ..........................................................
Figure 5.19: Test Phase performance of the Neural Network with configuration (6-5-5-1) ......................................................... 54
Figure 5.20: Regression fit of the outputs versus targets with configuration (6-25-1) ........................................................................ 55
Figure 5.21: Test phase performance of the ANN with configuration (6-25-1) ........................................................................ 55
Figure 5.22: Regression fit of the outputs versus targets with configuration (6-16-1) ........................................................................ 56
Figure 5.23: Test phase performance of the neural network with configuration (6-16-1) ............................................................. 56
Figure 5.24: Regression fit of the outputs versus targets with configuration (6-7-1) ................................................................. 57
Figure 5.25: Test phase performance of the ANN with configuration (6-7-1) ........................................................................ 58
Figure 5.26: Overview of the chosen ANN with configuration (6-7-1) ....................................................................................... 59
Figure 5.27: Mean-square error performance of the network with configuration (6-7-1) ............................................................ 59
Figure 5.28: Gradient and validation performance of the network with configuration (6-7-1) ........................................................ 60
Figure 5.29: Regression plots of various phases of learning of the ANN with configuration (6-7-1) .................................................. 61
Figure 5.30: Structure of the chosen ANN with configuration (6-7-1) .......... 61
Figure 5.31: Mean Square Error performance plot with configuration (6-10-20-5-1) ............................................................... 64
Figure 5.32: Test Phase performance of the ANN with configuration (6-10-20-5-1) ................................................................. 65
Figure 5.33: Mean Square Error performance plot with configuration (6-10-1) ................................................................. 66
Figure 5.34: Test Phase performance of the ANN with configuration (6-10-1) ................................................................. 66
Figure 5.35: Mean Square Error performance of the ANN with configuration (6-10-5-1) ............................................................... 67
Figure 5.36: Test phase performance of the neural network with configuration (6-10-5-1) ............................................................... 67
Figure 5.37: Overview of the chosen ANN for Line-Line Faults (6-10-5-1) .......... 68
Figure 5.38: Regression fit of the outputs versus targets with configuration (6-10-5-1) ............................................................. 69
Figure 5.39: Gradient and validation performance plot of the ANN (6-10-5-1) ................................................................. 70
Figure 5.40 Regression plots of the various phases of learning of the chosen ANN (6-10-5-1) .................................................. 70
Figure 5.41: Structure of the chosen Neural Network (6 – 10 – 5 – 1) ........ 71
Figure 5.42: Mean Square Error performance of the ANN with configuration (6-10-1) ................................................................. 74
Figure 5.43: Test Phase performance of the ANN
Figure 5.44: Mean Square Error performance of the ANN
with configuration (6-10-1) ........................................................................................................... 75
Figure 5.45: Test Phase performance of the ANN
with configuration (6-20-1) ........................................................................................................... 76
Figure 5.46: Mean Square Error performance of the neural network
with configuration (6-10-5-1) ......................................................................................................... 77
Figure 5.47: Test Phase performance of the ANN (6-10-5-1) ................................................................. 78
Figure 5.48: Mean Square Error performance of the neural network
with configuration (6-21-11-1) ......................................................................................................... 79
Figure 5.49: Test phase performance of the ANN (6-21-11-1) ................................................................. 79
Figure 5.50: Overview of the chosen ANN (6-21-11-1) for Double Line-Ground Faults .................. 80
Figure 5.51: Regression fit of the outputs versus targets
with configuration (6-21-11-1) ......................................................................................................... 81
Figure 5.52: Gradient and validation performance plot of ANN
with configuration (6-21-11-1) ......................................................................................................... 82
Figure 5.53: Regression plots of the various stages of learning of ANN (6-21-11-1) ...................... 82
Figure 5.54: Structure of the chosen ANN (6–21–11–1).................................................................... 83
Figure 5.55: Regression fit of the outputs versus targets of ANN
with configuration (6-21-10-1) ......................................................................................................... 86
Figure 5.56: MSE performance of the neural network
with configuration (6-21-10-1) ......................................................................................................... 87
Figure 5.57: Test Phase performance of the ANN
with configuration (6-21-10-1) ......................................................................................................... 87
Figure 5.58: MSE performance of the neural network
with configuration (6-21-1) ................................................................................................................ 88
Figure 5.59: Regression fit for the outputs versus targets of ANN
with configuration (6-21-1) ................................................................................................................ 89
Figure 5.60: Test Phase performance of the ANN
with configuration (6-21-1) ............................................................................................................. 90
Figure 5.61: Regression fit of the outputs versus targets of ANN (6-6-21-16-1) ......................... 91
Figure 5.62: Test Phase performance of the ANN (6-6-21-16-1) ..................................................... 91
Figure 5.63: Overview of the chosen neural network
for three phase fault location ........................................................................................................ 92
Figure 5.64: Mean Square Error performance of the neural network (6-6-21-16-1) ...................... 93
Figure 5.65: Gradient and validation performance plots of the ANN (6-6-21-16-1) ....................... 94
Figure 5.66: Regression plots of the various phases of learning of the ANN (6-6-21-16-1) .......... 94
Figure 5.67: Structure of the chosen ANN (6–6–21–16–1).............................................................. 95
CHAPTER ONE: INTRODUCTION

1.1 INTRODUCTION

In the past several decades, there has been a rapid growth in the power grid all over the world which eventually led to the installation of a huge number of new transmission and distribution lines. Moreover, the introduction of new marketing concepts such as deregulation has increased the need for reliable and uninterrupted supply of electric power to the end users who are very sensitive to power outages [1]. One of the most important factors that hinder the continuous supply of electricity and power is a fault in the power system [2]. Any abnormal flow of current in a power system’s components is called a fault in the power system. These faults cannot be completely avoided since a portion of these faults also occur due to natural reasons which are way beyond the control of mankind. Hence, it is very important to have a well-coordinated protection system that detects any kind of abnormal flow of current in the power system, identifies the type of fault and then accurately locates the position of the fault in the power system. The faults are usually taken care of by devices that detect the occurrence of a fault and eventually isolate the faulted section from the rest of the power system.

Hence some of the important challenges for the incessant supply of power are detection, classification and location of faults [3]. Faults can be of various types namely transient, persistent, symmetric or asymmetric faults and the fault detection process for each of these faults is distinctly unique in the sense, there is no one universal fault location technique for all these kinds of faults. The High Voltage Transmission Lines
(that transmit the power generated at the generating plant to the high voltage substations) are more prone to the occurrence of a fault than the local distribution lines (that transmit the power from the substation to the commercial and residential customers) because there is no insulation around the transmission line cables unlike the distribution lines. The reason for the occurrence of a fault on a transmission line can be due to several reasons such as a momentary tree contact, a bird or an animal contact [1] or due to other natural reasons such as thunderstorms or lightning. Most of the research done in the field of protective relaying of power systems concentrates on transmission line fault protection due to the fact that transmission lines are relatively very long and can run through various geographical terrain and hence it can take anything from a few minutes to several hours to physically check the line for faults [4].

The automatic location of faults can greatly enhance the systems reliability because the faster we restore power, the more money and valuable time we save. Hence, many utilities are implementing fault locating devices in their power quality monitoring systems [2] that are equipped with Global Information Systems for easy location of these faults. Fault location techniques can be broadly classified into the following categories [5]:

- Impedance measurement based methods
- Travelling-wave phenomenon based methods
- High-frequency components of currents and voltages generated by faults based methods
- Intelligence based method
From quite a few years, intelligent based methods are being used in the process of fault detection and location. Three major artificial intelligence based techniques that have been widely used in the power and automation industry are [6]:

- Expert System Techniques
- Artificial Neural Networks
- Fuzzy Logic Systems

Among these available techniques, Artificial Neural Networks (ANN) have been used extensively in this thesis for fault location on electric power transmission lines. These ANN based methods do not require a knowledge base for the location of faults unlike the other artificial intelligence based methods [7].

1.2 MOTIVATION

The prime motive behind this thesis was the significant impact a very accurate fault locator could make if employed in a power transmission and distribution system, in terms of the amount of money and time that can be saved. The main goal of Fault Location is to locate a fault in the power system with the highest practically achievable accuracy. When the physical dimensions and the size of the transmission lines are considered, the accuracy with which the designed fault locator locates faults in the power system becomes very important.

One of the important aspects that this thesis concentrates on is the analysis of the transmission line’s phase voltages and currents during various fault conditions and how they can be effectively utilized in the design of an efficient fault locator. The main objective of this research is to study and successfully design a fault locator that can
detect, classify and locate faults in power transmission lines. This thesis drew its initial motivation from [8] which demonstrates a method that could be used for location of faults in transmission lines using neural fuzzy logic. However, when extensively studied, it can be noted that a fault locator with satisfactorily high accuracy can be easily achieved with the help of artificial neural networks by the use of a large amount of data set for training and the learning process. This eliminates the need for proficiency in power systems which is a necessity when working with expert fuzzy systems [9]. Hence this thesis focuses on the design of a fault locator that can be even used by people who aren’t experts in the field of power systems.

1.3 OUTLINE OF THE THESIS

The second chapter deals with the several problems that hinder the protection of a typical transmission line system. The various kinds of faults and the protection techniques that are currently available and employed are briefly discussed. Some important results from the research on the existing transmission line protection techniques are also provided in this chapter.

The third chapter introduces the concept behind artificial intelligence and neural networks. A few ANN architectures that are usually employed are discussed and the various learning strategies employed in the training process of the neural networks along with the critical factors that affect the size and output of a trained network are discussed in this chapter.
The fourth chapter deals with the actual implementation and development of the neural networks and their architectures proposed for the three different parts of the fault location process namely fault detection, classification and fault location. An overview of the training and testing processes employed with neural networks in this work has been outlined in this chapter.

The fifth chapter presents series of simulation results that have been obtained using MATLAB, SimPowerSystems and the Artificial Neural Networks Toolboxes in Simulink in detail to emphasize the efficiency and accuracy factors of the proposed fault locator. Several neural networks with varying configurations have been trained, tested and their performances have been analyzed in this chapter.

The sixth chapter concludes the entire research work and the thesis. It discusses the results obtained in the previous chapters. Moreover, the scope for future work and possible extensions to this work has been outlined briefly in this chapter.
CHAPTER TWO: LITERATURE REVIEW

This chapter talks about the state of the art research going on in the field of fault location in power transmission lines using artificial neural networks. Sections 2.2.1 – 2.2.2 talk about the different techniques being used for fault location in transmission lines. The section 2.2.3 talks about the various artificial intelligence based methods that are being researched upon in the field of fault location in power transmission lines.

2.1 POWER PROTECTION SYSTEMS

One of the most important components of a power protection system is the relay which is a device that trips the circuit breakers when the input voltage and current signals correspond to the fault conditions designed for the relay operation. Relays in general can be classified into the following categories [10-14]:

- Directional Relays: These relays respond to the phase angle difference between two inputs to the relay.
- Differential Relays: These relays respond to the magnitude of the algebraic sum of two or more of its inputs.
- Magnitude Relays: These relays respond to the magnitude of the input quantity.
- Pilot Relays: These relays respond to the input signals that are communicated to the relay from a remote location.
- Distance Relays: These relays respond to the ratio of two input phasor signals.
Among the various relays that are used for the protection of power lines distance relays are the most relevant to fault locators. Usually a pair of these distance relays is used for the protection of a two-terminal transmission line [13].

2.2 TRANSMISSION LINE FAULT LOCATION TECHNIQUES

The transmission line fault location process, as mentioned before, has been researched for a while and several innovative and efficient techniques have been proposed and analyzed by several authors [15-23]. These techniques can be broadly classified as Impedance based methods, Travelling wave based methods and Artificial Intelligence based methods. Each of these methods is discussed briefly in the following subsections.

2.2.1 IMPEDANCE BASED METHODS

In the case of Impedance based methods, the operation of the distance relay greatly relies on the fault resistance and is not successful in cases with very high fault resistance [16]. Impedance based methods can be classified into single-ended methods and two-ended methods depending upon the number of terminals at which the voltage and current data are collected.

The basic logic behind a single-ended impedance based fault locator is to calculate the location of the fault from the apparent impedance seen looking into the line from one end. The various impedance based methods available in literature are discussed in the upcoming subsections.
2.2.1.1 SIMPLE REACTANCE METHOD

The measured voltage and current values at the terminal are used to calculate the impedance of the line to the fault position as shown in equation (1). Once the line impedance per unit length has been determined, the fault distance can be calculated accordingly as illustrated by equations (2) and (3) [24].

\[ V_A = x \cdot Z_L \cdot I_A + V_f \]  
\[ V_A = x \cdot Z_L \cdot I_A + R_f \cdot I_f \]  

Where \( V_A \) is the voltage at terminal A, 
\( x \) is the distance to the fault from the terminal A, 
\( I_A \) is the current flowing out of the terminal A, \( V_f \) is the fault voltage and 
\( Z_L \) is the line impedance.

\[ x = \frac{(V_A/I_A)}{Z_L} - \frac{R_f}{Z_L(I_A/I_f)} \]  

Figure 2.1 Faulted Transmission Line illustrating simple-reactance method.
2.2.1.2 TAKAGI METHOD

The Takagi method [25] is a very simple yet innovative single-ended impedance-based Fault location technique and is illustrated by Fig 2.2. It requires both the pre-fault and fault data and enhances the simple reactance method by minimizing the effect of fault resistance and reducing the effect of load flow.

![Diagram of the Takagi method](image)

*Figure 2.2 A single-phase circuit illustrating Takagi method.*

The Fault Resistance is given by

\[ R_f = \frac{V_A - Z_C I_A \tanh \gamma x}{(\frac{V_A}{Z_C} \tanh \gamma x - I_A')e^{i\theta}} \]  

(4)

where \( V_A \) is voltage measured at terminal A, \( I_A \) is the flowing out of terminal A, \( \gamma \) is the propagation constant, \( Z_C \) is the characteristic impedance, \( Z_L \) is the line impedance, \( I_A' \) is the superposition current which is the difference between the fault current and the pre-fault current.

And

\[ x = \frac{\text{Im}(V_A I_A')}{\text{Im}(Z_L I_A - I_A)} \]

is the distance to the fault from terminal A. \hspace{1cm} (5)

Where \( Z_L = \gamma Z_C \) \hspace{1cm} (6)
2.2.1.3 MODIFIED TAKAGI METHOD

The modified Takagi method also called the Zero Sequence current method does not require pre-fault data because it uses zero-sequence current instead of the superposition current for ground faults [26]. The location of the fault in this method is given by x in equation (7).

\[
x = \frac{\text{Im}(V_A I_R e^{-j\beta})}{\text{Im}(Z_{1L} I_A I_R e^{-j\beta})}
\]  

(7)

Where \(I_R\) is the zero-sequence current and \(\beta\) is the zero-sequence current angle. The position of the fault ‘x’ is given by equation (7); \(V_A\) is voltage measured at terminal A, \(I_A\) is the flowing out of terminal A and \(Z_{1L}\) is the positive sequence line impedance.

2.2.2 TRAVELLING WAVE BASED METHODS

Travelling wave based methods have been widely used [27-29] for the purpose of fault location and are usually based on the correlation between the forward and backward waves travelling along the transmission line as shown in Fig 2.3. The basic idea is to successively identify the fault initiated by high-frequency travelling waves at the fault locator [5].

![Figure 2.3 Illustration of Travelling wave based Fault Location.](image-url)
The time taken by the high frequency components for propagation is used for the location of fault. In Fig 2.3, a single phase lossless transmission line of length ‘l’ is considered with a travelling wave velocity of $v$, capacitance and inductance per unit length $L'$ and $C'$ and a characteristic impedance of $Z_c$. Assuming the occurrence of a fault at a distance of ‘x’ from the terminal A, the voltage and current values are given by (8) and (9).

$$\frac{\partial e}{\partial x} = -L' \frac{\partial i}{\partial t}$$ \hspace{1cm} (8)

$$\frac{\partial i}{\partial x} = -C' \frac{\partial e}{\partial t}$$ \hspace{1cm} (9)

Whose solutions are given by (10) and (11).

$$e(x,t) = e_f(x - vt) + e_r(x + vt)$$ \hspace{1cm} (10)

$$i(x,t) = \frac{1}{Z_c} e_f(x - vt) - \frac{1}{Z_c} e_r(x + vt)$$ \hspace{1cm} (11)

The times taken for the waves to travel from the fault to the discontinuity $\tau_A$ and $\tau_B$ are to be determined using GPS technology. Once this is done, the fault location ($x$) can be readily determined by the following equation (12)

$$x = \frac{l - c(\tau_A - \tau_B)}{2}$$ \hspace{1cm} (12)

Where: $c$ is the wave propagation speed of 299.79 m/sec.

2.2.3 NEURAL NETWORKS BASED METHODS

Neural networks have been put in use for fault location quite recently [30] and have gained significant importance since Sobajic and Pao used neural networks for the prediction of critical clearing time [31]. Wide usage of neural networks started by late eighties and during early nineties. Neural networks are usually used to achieve greater
efficiency in fault detection, classification and location. A lot of research has been done and abundant literature has been published in the field of fault location using neural networks. Certain significant techniques and results that have been published are briefly discussed here. A majority of the work mentioned here made use of feed-forward multilayer perceptron technique. Kulicke and Dalstein [32] used neural networks for the detection of faults on transmission lines and also differentiated between arcing and non-arcing faults. A new technique for the detection and location of high speed faults using neural networks has been proposed by Rikalo, Sobajic and Kezunovic [33]. Neural network based single ended fault location techniques have been widely researched by Chen and Maun while Song used neural networks for fault location on series compensated lines. Other relevant work in the field of fault location using artificial neural networks can be found in these references [34-38]
CHAPTER THREE: NEURAL NETWORKS AND THEIR APPLICATION IN TRANSMISSION LINE FAULT LOCATION

3.1 INTRODUCTION TO NEURAL NETWORKS

An Artificial Neural Network (ANN) can be described as a set of elementary neurons that are usually connected in biologically inspired architectures and organized in several layers [39]. The structure of a feed-forward ANN, also called as the perceptron is shown in Fig 3.1. There are $N_i$ numbers of neurons in each $i^{th}$ layer and the inputs to these neurons are connected to the previous layer neurons. The input layer is fed with the excitation signals. Simply put, an elementary neuron is like a processor that produces an output by performing a simple non-linear operation on its inputs [40]. A weight is attached to each and every neuron and training an ANN is the process of adjusting different weights tailored to the training set. An Artificial Neural Network learns to produce a response based on the inputs given by adjusting the node weights. Hence we need a set of data referred to as the training data set, which is used to train the neural network.

![Figure 3.1 A basic three-layer architecture of a feedforward ANN.](image)
In Fig 3.1, \( a_1, a_2 \ldots a_{N_0} \) is the set of inputs to the ANN. Due to their outstanding pattern recognition abilities ANNs are used for several purposes in a wide variety of fields including signal processing, computers and decision making. Some important notes on artificial neural networks are [41]:

- Either signal features extracted using certain measuring algorithms or even unprocessed samples of the input signals are fed into the ANN.
- The most recent along with a few older samples of the signals are fed into the ANN.
- The output provided by the neural network corresponds to the concerned decision which might be the type of fault, existence of a fault or the location of a fault.
- The most important factor that affects the functionality of the ANN is the training pattern that is employed for the same.
- Pre-processing and post-processing techniques may be employed as well to enhance the learning process and reduce the training time of the ANN.

One of the biggest drawbacks of applications that make use of artificial neural networks is that no well-defined guide exists to help us choose the ideal number of hidden layers to be used and the number of neurons per each hidden layer. From a different perspective, it is advantageous considering the ability to generalize [39]. A vital feature of ANN is its dedication to parallel computing. Hence it can produce a correct output corresponding to any input even if the concerned input was not fed into the ANN during the training process. Another challenge in the ANN based application development was to synthesize the algorithm for the adaptive learning process. The back-error-propagation algorithm is the basic algorithm in which the neuron weights are
adjusted in consecutive steps to minimize the error between the actual and the desired outputs. This process is known as supervised learning.

### 3.2 MODEL OF A NEURON

Any basic neuron model as shown in Fig 3.2 can be described by a function that calculates the output as a function of $N_0$ inputs to it. The basic idea behind the entire neuron model, including the activation functions illustrated below, has been adopted from [5].

![Figure 3.2 Typical model of a neuron.](image)

The output of the neuron is given by

$$y = f(\varphi) = f\left(\sum_{i=0}^{N_0} w_i a_i\right)$$  \hspace{1cm} (13)

Where: $w_0 a_0$ is the threshold value (polarization), $f(\varphi)$ is the neuron activation function, $\varphi$ is the summation output signal and $y$ is the neuron output.

$$\varphi = W^T A$$  \hspace{1cm} (14)

Where: $W = [w_0 \ w_1 \ \ldots \ w_{N_0}]$, $A = [a_0 \ a_1 \ \ldots \ a_{N_0}]^T$.  

(15)
An activation function decides how powerful the output from the neuron should be, based on the sum of its inputs. Depending upon the application’s requirements, the most appropriate activation function is chosen.

The activation function \( f(\varphi) \) can be in different forms a few of which are described below:

- **Step function**
  \[
  f(\varphi) = \begin{cases} 
  1 & \text{if } \varphi \geq 0 \\
  0 & \text{if } \varphi < 0 
  \end{cases}
  \]

  ![Figure 3.3 Step activation function](image1)

- **Piece wise linear function**
  \[
  f(\varphi) = \begin{cases} 
  1 & \text{if } \varphi > 1 \\
  -1 & \text{if } \varphi < -1 \\
  \varphi & \text{if } |\varphi| < 1 
  \end{cases}
  \]

  ![Figure 3.4 Piece wise linear activation function](image2)

- **Sigmoid unipolar function**
  \[
  f(\varphi) = \frac{1}{1 + e^{-\beta \varphi}}
  \]

  ![Figure 3.5 Sigmoid unipolar activation function](image3)
Based on the way the neurons are interconnected in a model, neural networks can be broadly classified into two types namely feedforward and feedback networks. As the name suggests, feedback networks unlike feedforward networks have a feedback connection fed back into the network along with the inputs. Due to their simplicity and the existence of a well-defined learning algorithm, only feedforward networks have been used in this thesis for the simulation and hence are discussed briefly in the upcoming sections.

3.3 FEEDFORWARD NETWORKS

Feedforward networks are the simplest neural networks where there is no feedback connection involved in the network and hence the information travel is unidirectional [40]. A feedforward network with $N_0$ input and $K_R$ output signals is shown in Fig 3.7. The computation process in the $i^{th}$ layer can be described by the following equation (16)

$$p^{(i)} = f^{(i)}(W^{(i)} g^{(i-1)})$$

(16)

Where $p^{(i)} = \begin{bmatrix} p_1^{(i)} & p_2^{(i)} & \ldots & p_{N_i}^{(i)} \end{bmatrix}^T$ is the signal vector at the output of the $i^{th}$ layer.

- Sigmoid bipolar function

$$f(\varphi) = \tanh(\beta \varphi) = \frac{1 - e^{-2\beta \varphi}}{1 + e^{-2\beta \varphi}}$$

Figure 3.6 Bipolar activation function.
And $W^{(i)} = \begin{pmatrix} w_{10}^{(i)} & w_{11}^{(i)} & \cdots & w_{1N_{i-1}}^{(i)} \\ w_{20}^{(i)} & w_{21}^{(i)} & \cdots & w_{2N_{i-1}}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N_i0}^{(i)} & w_{N_11}^{(i)} & \cdots & w_{N_iN_{i-1}}^{(i)} \end{pmatrix}$ is the weighing matrix between the $(i-1)^{th}$ and the $i^{th}$ layer.

$$g^{(i-1)} = \begin{cases} A & \text{for } i = 1 \\ \begin{bmatrix} 1 \\ p^{(i-1)} \end{bmatrix} & \text{for } i = 2,3,\ldots,R \end{cases}$$ (17)

$A$ is the vector containing the input signals, $f^{(i)}(.)$ is the activation function of the neurons in the $i^{th}$ layer and $R$ is the number of processing layers. All the neurons in a particular layer are assumed to be similar in all aspects and the number of hidden layers can be more than one and is usually determined by the purpose of the neural network. The output of the processed neural network is represented by the output vector:

$$y = p^{(R)} = [y_1 \ y_2 \ \cdots \ y_{N_R}]^T$$ (18)

![Figure 3.7 Structure of a two-layered feedforward network.](image)
3.4 LEARNING STRATEGIES

The basic concept behind the successful application of neural networks in any field is to determine the weights to achieve the desired target and this process is called learning or training. The two different learning mechanisms usually employed are supervised and unsupervised learning. In the case of supervised learning the network weights are modified with the prime objective of minimization of the error between a given set of inputs and their corresponding target values [39]. Hence we know the training data set which is a set of inputs and the corresponding targets the neural network should output ideally. This is called supervised learning because both the inputs and the expected target values are known prior to the training of ANN.

On the other hand, in the case of unsupervised learning, we are unaware of the relationship between the inputs and the target values. We train the neural network with a training data set in which only the input values are known. Hence it is very important to choose the right set of examples for efficient training. These examples are usually chosen using some sort of a similarity principle [40]. The most commonly used unsupervised learning algorithms are the Self-Organizing Map (SOM) and the Adaptive Resonance Theory (ART). The learning strategy employed depends on the structure of the neural network. Feedforward networks are trained using the supervised learning strategy. The supervised learning strategy for a feedforward neural network has been shown in the Fig 3.8.

The set of input-output pairs (shown in Fig 3.8) that are used to train the neural network are obtained prior to the training process either by using physical measurements or by performing some kind of simulations. Fig 3.8 shows that the teacher teaches the
neural network to modify its weights according to the error ‘e’ between the outputs and the targets. The weights of the neural network are then modified iteratively according to equation (19). The general idea behind supervised learning and the mathematics involved has been adopted from [5].

\[ w_{ji}(n + 1) = w_{ji}(n) + \Delta w_{ji}(n) \]  

(19)

Where: \( w_{ji}(n) \) and \( w_{ji}(n+1) \) are the previous and the modified weights connected between the \( i^{th} \) and the \( j^{th} \) adjoining layers. \( \Delta w_{ji}(n) \) stands for the correction or modification factor and \( n \) stands for the number of the iteration. If we consider the \( j^{th} \) neuron in a single layer neural network, the training efficiency is enhanced by minimizing the error between the actual output of the \( j^{th} \) neuron and the output that has been dictated by the teacher. Let \( y_{j}(n) \) and \( p_{j}(n) \) be the actual and the teacher-requested outputs for the \( j^{th} \) neuron in the \( n^{th} \) iteration. Then the error value of that iteration is given by (20).
\[ e_j(n) = p_j(n) - y_j(n) \]  

(20)

The vector \( \mathbf{e}(n) \) that stores the values of all the errors is also a function of the weights \( \mathbf{w}(n) \) for the corresponding layers’ inputs. The value by which the weighing coefficients change (also called the correction factor) is given by the following equation (21).

\[ \Delta w_{ji}(n) = \eta e_j(n)x_i(n) \]  

(21)

Where: \( x_i \) is the \( i^{\text{th}} \) input signal and \( \eta \) is the rate at which the learning process takes place.

As mentioned earlier, learning process aims at the minimization of the error function. The same criterion can also be achieved by the usage of a Least Squares Method (LSM).

Hence, if there are \( L \) neurons in a particular network, the cost function to be ultimately minimized is given by (22).

\[
S_2(\mathbf{w}) = \frac{1}{2} \sum_{j=1}^{L} (p_j - y_j)^2
\]  

(22)

If the number of learning pairs with an input vector \( x(n) \) and an output vector \( d(n) \) of the form \((x(n),d(n))\) are \( P \) in the training set, then during the \( n^{\text{th}} \) iteration of the learning process, we have:

\[
S_2(\mathbf{w}(n)) = \frac{1}{2} \sum_{n=1}^{P} \sum_{j=1}^{L} \left( p_j(n) - y_j(n) \right)^2
\]  

(23)

Since the activation functions that are employed are more than often non-linear, minimization of the above equation (23) is a non-linear problem. Several numerical methods that can handle non-linear functions effectively are available and are based on the steepest-decent method. The steepest-decent method is an extension to the Laplace’s method of integral approximation where the contour integral in a complex plane is deformed to approach a stationary point in the direction of the steepest decent [42]. The back-error-propagation learning technique is based on the steepest-decent method and is usually widely applied in a version known as the Levenberg-Marquardt algorithm [42].
The back-error-propagation algorithm chooses random weights for the neural network nodes, feeds in an input pair and obtains the result. Then we calculate the error for each node starting from the last stage and by propagating the error backwards. Once this is done, we update the weights and repeat the process with the entire set of input output pairs available in the training data set. This process is continued till the network converges with respect to the desired targets. The back-error-propagation technique is widely used for several purposes including its application to error functions (other than the sum of squared errors) and for the evaluation of Jacobian and Hessian matrices. The correction values are calculated as functions of errors estimated from the minimization of equation (23). This process is carried out layer by layer throughout the network in the backward direction. This algorithm is pictorially depicted in Fig 3.9.

![Figure 3.9 Structure of back-error-propagation algorithm [adopted from [5]]](image-url)
The corresponding weighing vectors are shown in blocks $A^{(M)}, A^{(M-1)}, \ldots, A^{(1)}$ and the errors that are propagated to the lower layers are calculated and stored in the blocks $B^{(M-1)}, B^{(M-2)}, \ldots, B^{(2)}$. The back-error-propagation algorithm has been implemented in many ways but the basic idea remains the same. The only thing that changes in each of these implementations is the method used for the calculation of the weights that are iteratively upgraded when passed backward from layer to layer in the neural network. The modifications involved are also used in the training process of recurrent networks. The rate at which the learning process takes place can be estimated by keeping a check on the correction values in successive stages. The total number of iterations required to achieve satisfactory convergence rate depends on the following factors:

- size of the neural network
- structure of the network
- the problem being investigated
- the learning strategy employed
- size of the training/learning set

The efficiency of a chosen ANN and the learning strategy employed can be estimated by using the trained network on some test cases with known output values. This test set is also a part of the learning set. Hence the entire set of data consists of the training data set along with the testing data set. The former is used to train the neural network and the latter is used to evaluate the performance of the trained artificial neural network.
CHAPTER FOUR: FAULT LOCATION IN POWER TRANSMISSION LINES USING NEURAL NETWORKS

4.1 INTRODUCTION

As discussed in the previous chapters, artificial neural networks have been used for the protection of power transmission lines. The excellent pattern recognition and classification abilities of neural networks have been cleverly utilized in this thesis to address the issue of transmission line fault location.

In this chapter, a complete neural-network based approach has been outlined in detail for the location of faults on transmission lines in a power system. To achieve the same, the original problem has been dealt with in three different stages namely fault detection, fault classification and fault location.

4.2 MODELLING THE POWER TRANSMISSION LINE SYSTEM

A 500 kV transmission line system has been used to develop and implement the proposed strategy using ANNs. Fig 4.1 shows a one-line diagram of the system that has been used throughout the research. The system consists of two generators of 500 kV each located on either ends of the transmission line along with a three phase fault simulator used to simulate faults at various positions on the transmission line. The line has been modeled using distributed parameters so that it more accurately describes a very long transmission line.
Figure 4.1 One-line diagram of the studied system.

This power system was simulated using the SimPowerSystems toolbox in Simulink by The MathWorks. A snapshot of the model used for obtaining the training and test data sets is shown in Fig 4.2. In Fig 4.2, ZP and ZQ are the source impedances of the generators on either side. The three phase V-I measurement block is used to measure the voltage and current samples at the terminal A. The transmission line (line 1 and line 2 together) is 300 km long and the three-phase fault simulator is used to simulate various types of faults at varying locations along the transmission line with different fault resistances.

Figure 4.2 Snapshot of the studied model in SimPowerSystems.
The values of the three-phase voltages and currents are measured and modified accordingly and are ultimately fed into the neural network as inputs. The SimPowerSystems toolbox has been used to generate the entire set of training data for the neural network in both fault and non-fault cases.

Faults can be classified broadly into four different categories namely:

- line to ground faults
- line to line faults
- double-line to ground faults
- three-phase faults

There have been 1100 different fault cases simulated for the purpose of fault detection, 1100 different fault cases simulated for fault classification and varying number of fault cases (based on the type of fault) for the purpose of fault location.

4.3 OUTLINE OF THE PROPOSED SCHEME

Although the basic concept behind relays remains the same, the digital technology has had a significant influence on the way relays operate and have offered several improvements over traditional electromechanical relays.

The main goal of this chapter is to design, develop, test and implement a complete strategy for the fault diagnosis as shown in Fig 4.3. Initially, the entire data that is collected is subdivided into two sets namely the training and the testing data sets. The first step in the process is fault detection. Once we know that a fault has occurred on the transmission line, the next step is to classify the fault into the different categories based on the phases that are faulted.
Figure 4.3 Flowchart depicting the outline of the proposed scheme.
Then, the third step is to pin-point the position of the fault on the transmission line. The goal of this thesis is to propose an integrated method to perform each of these tasks using artificial neural networks. A back-propagation based neural network has been used for the purpose of fault detection and another similar one for the purpose of fault classification. For each of the different kinds of faults, separate neural networks have been employed for the purpose of fault location. Each of these steps has been depicted in the flowchart shown in Fig 4.3.

4.4 DATA PRE-PROCESSING

A reduction in the size of the neural network improves the performance of the same and this can be achieved by performing feature extraction. By doing this, all of the important and relevant information present in the waveforms of the voltage and current signals can be used effectively. Voltage and current waveforms have been generated and were sampled at a frequency of 720 Hertz. The voltage and current samples of all the three phases are noted along with the corresponding pre-fault values.

![Data pre-processing illustration](image)
Fig 4.4 shows the current waveform of a Phase B – ground fault at a distance of 60 km from terminal A on a 300 km transmission line. The waveform is the plot of the samples sampled at a frequency of 720 Hz. Hence there are 12 samples per each cycle. Now, the 50th sample (12th sample after the occurrence of the fault) on phase B is noted along with the 26th sample (12th sample before the occurrence of the fault, corresponding to the post-fault sample considered). Once this is done, the inputs to the neural network are the ratios of the voltages and currents in each of the phases before and after the occurrence of fault as shown in Table 4.1. The inputs in matrix format are shown below:

\[
\begin{bmatrix}
\frac{V_a(n+12)}{V_a(n-12)} \\
\frac{V_b(n+12)}{V_b(n-12)} \\
\frac{V_c(n+12)}{V_c(n-12)} \\
\frac{I_a(n+12)}{I_a(n-12)} \\
\frac{I_b(n+12)}{I_b(n-12)} \\
\frac{I_c(n+12)}{I_c(n-12)}
\end{bmatrix}
\]

Where ‘n = 38’ is the sample at which fault occurs.

Hence, there is a set of six inputs each time (3 for the phase voltages and 3 for the phase currents) to all the neural networks discussed in this work [43]. Care has been taken each time to make sure the denominator of each of the inputs is non-zero. If it is zero, the value of n is incremented by 1 and the next sample is taken into consideration for the entire process. The advantage of performing this scaling is to reduce the training computation time. For the sake of illustration, the Table 4.1 shows the voltage and current values that are scaled with respect to their pre-fault values and used as a part of the training set. In Table 4.1, \(V_a\), \(V_b\) and \(V_c\) are the post fault voltage and current sample values and \(V_a(pf)\), \(V_b(pf)\) and \(V_c(pf)\) are the corresponding pre-fault values as illustrated earlier. The given table depicts the values for all the various types of faults and also during the no fault case. The fault has been simulated on a 300 km long transmission line at a distance of 100 km from the terminal A.
Table 4.1 Sample of Inputs to the neural network for various fault cases.

<table>
<thead>
<tr>
<th>Case No:</th>
<th>Input Vector</th>
<th>Fault Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$V_a/V_a$(pf)</td>
<td>$V_b/V_b$(pf)</td>
</tr>
<tr>
<td>1</td>
<td>0.6204</td>
<td>0.9719</td>
</tr>
<tr>
<td>2</td>
<td>0.6573</td>
<td>0.7351</td>
</tr>
<tr>
<td>3</td>
<td>1.2580</td>
<td>0.9141</td>
</tr>
<tr>
<td>4</td>
<td>-0.1882</td>
<td>0.6041</td>
</tr>
<tr>
<td>5</td>
<td>1.0000</td>
<td>0.5516</td>
</tr>
<tr>
<td>6</td>
<td>1.1586</td>
<td>1.0000</td>
</tr>
<tr>
<td>7</td>
<td>-0.1276</td>
<td>0.5841</td>
</tr>
<tr>
<td>8</td>
<td>0.9359</td>
<td>0.5145</td>
</tr>
<tr>
<td>9</td>
<td>0.9864</td>
<td>0.9147</td>
</tr>
<tr>
<td>10</td>
<td>0.3135</td>
<td>0.4373</td>
</tr>
<tr>
<td>11</td>
<td>1.0000</td>
<td>1.0001</td>
</tr>
</tbody>
</table>
4.5 OVERVIEW OF THE TRAINING PROCESS

Two important steps in the application of neural networks for any purpose are training and testing. The first of the two steps namely training the neural network is discussed in this section. Training is the process by which the neural network learns from the inputs and updates its weights accordingly. In order to train the neural network we need a set of data called the training data set which is a set of input output pairs fed into the neural network. Thereby, we teach the neural network what the output should be, when that particular input is fed into it. The ANN slowly learns the training set and slowly develops an ability to generalize upon this data and will eventually be able to produce an output when a new data is provided to it. During the training process, the neural network’s weights are updated with the prime goal of minimizing the performance function. This performance function can be user defined, but usually feedforward networks employ Mean Square Error as the performance function and the same is adopted throughout this work.

As already mentioned in the previous chapter, all the voltages and currents fed into the neural network are scaled with respect to the corresponding voltage and current values before the occurrence of the fault. The outputs, depending upon the purpose of the neural network might be the fault condition, the type of fault or the location of the fault on the transmission line.

For the task of training the neural networks for different stages, sequential feeding of input and output pair has been adopted. In order to obtain a large training set for efficient performance, each of the ten kinds of faults has been simulated at different
locations along the considered transmission line. In view of all these issues, about 100 different fault cases for each of the 10 kinds of faults have been simulated.

Apart from the type of fault, the phases that are faulted and the distance of the fault along the transmission line, the fault resistance also has been varied to include several possible real-time fault scenarios.

- The fault resistance has been varied as follows: 0.25 ohm, 0.5 ohm, 0.75 ohm, 1 ohm, 5 ohm, 10 ohm, 25 ohm, 50 ohm.
- Fault distance has been varied at an incremental factor of every 3 km on a 300 km transmission line.

4.6 OVERVIEW OF THE TESTING PROCESS

As already mentioned in the previous section, the next important step to be performed before the application of neural networks is to test the trained neural network. Testing the artificial neural network is very important in order to make sure the trained network can generalize well and produce desired outputs when new data is presented to it.

There are several techniques used to test the performance of a trained network, a few of which are discussed in this section. One such technique is to plot the best linear regression fit between the actual neural network’s outputs and the desired targets [44]. Analyzing the slope of this line gives us an idea on the training process. Ideally the slope should be 1. Also, the correlation coefficient (r), of the outputs and the targets measures how well the ANN’s outputs track the desired targets. The closer the value of ‘r’ is, to 1, the better the performance of the neural network. Another technique employed to test the
neural network is to plot the confusion matrix and look at the actual number of cases that have been classified positively by the neural network [44]. Ideally this percentage is a 100 which means there has been no confusion in the classification process. Hence if the confusion matrix indicates very low positive classification rates, it indicates that the neural network might not perform well. The last and a very obvious means of testing the neural network is to present it with a whole new set of data with known inputs and targets and calculate the percentage error in the neural networks output. If the average percentage error in the ANN’s output is acceptable, the neural network has passed the test and can be readily applied for future use.

The Neural Network toolbox in Simulink by The MathWorks divides the entire set of data provided to it into three different sets namely the training set, validation set and the testing set. The training data set as indicated above is used to train the network by computing the gradient and updating the network weights. The validation set is provided during to the network during the training process (just the inputs without the outputs) and the error in validation data set is monitored throughout the training process. When the network starts overfitting the data, the validation errors increase and when the number of validation fails increase beyond a particular value, the training process stops to avoid further overfitting the data and the network is returned at the minimum number of validation errors [44]. The test set is not used during the training process but is used to test the performance of the trained network. If the test set reaches the minimum value of MSE at a significantly different iteration than the validation set, then the neural network will not be able to provide satisfactory performance.
CHAPTER FIVE: EXPERIMENTAL RESULTS

5.1 FAULT DETECTION

For the purpose of fault detection, various topologies of Multi-Layer Perceptron have been studied. The various factors that play a role in deciding the ideal topology are the network size, the learning strategy employed and the training data set size.

After an exhaustive study, the back-propagation algorithm has been decided as the ideal topology. Even though the basic back-propagation algorithm is relatively slow due to the small learning rates employed, few techniques can significantly enhance the performance of the algorithm. One such strategy is to use the Levenberg-Marquardt optimization technique. The selection of the apt network size is very vital because this not only reduces the training time but also greatly enhance the ability of the neural network to represent the problem in hand. Unfortunately there is no thumb rule that can dictate the number of hidden layers and the number of neurons per hidden layer in a given problem.

5.1.1 TRAINING THE FAULT DETECTION NEURAL NETWORK

In the first stage which is the fault detection phase, the network takes in six inputs at a time, which are the voltages and currents for all the three phases (scaled with respect to the pre-fault values) for ten different faults and also no-fault case. Hence the training set consisted of about 1100 input output sets (100 for each of the ten faults and 100 for the no fault case) with a set of six inputs and one output in each input-output pair. The output of the neural network is just a yes or a no (1 or 0) depending on whether or not a
fault has been detected. After extensive simulations it has been decided that the desired network has one hidden layer with 10 neurons in the hidden layer. For illustration purposes, several neural networks (with varying number of hidden layers and neurons per hidden layer) that achieved satisfactory performance are shown and the best neural network has been described further in detail. Figures 5.1 – 5.2 show the error performance plots of neural networks with 1 and 2 hidden layers respectively. The chosen network has been depicted in Fig 5.7 and the various error performance plots have been shown in Figures 5.2 – 5.7.

Fig 5.1 shows the training performance plot of the neural network 6-10-1 (6 neurons in the input layer, 1 hidden layer with ten neurons in it and one neuron in the output layer). It can be seen that the network did not achieve the desired Mean Square Error (MSE) goal by the end of the training process.

![Figure 5.1 Mean-square error performance of the network (6-10-1).](image)
Fig 5.2 shows the training performance plot of the neural network with 6-10-5-1 configuration (6 neurons in the input layer, two hidden layers with 10 and 5 neurons respectively and one neuron in the output layer). It is to be noted that the neural network could not achieve the MSE goal of 0.0001 by the end of the training process.

![Figure 5.2 Mean-square error performance of the network (6-10-5-1).](image)

Fig 5.3 shows the training process of the neural network with 6-10-5-3-1 configuration (6 neurons in the input layer, 3 hidden layers with 10, 5 and 3 neurons in them respectively and one neuron in the output layer).

![Figure 5.3 Mean-square error performance of the network (6-10-5-3-1).](image)
From the above training performance plots, it is to be noted that very satisfactory training performance has been achieved by the neural network with the 6-10-5-3-1 configuration (6 neurons in the input layer, 3 hidden layers with 10, 5 and 3 neurons in them respectively and one neuron in the output layer). The overall MSE of the trained neural network is way below the value of 0.0001 and is actually 6.9776 e-5 by the end of the training process. Hence this has been chosen as the ideal ANN for the purpose of fault detection.

5.1.2 TESTING THE FAULT DETECTION NEURAL NETWORK

Once the neural network has been trained, its performance has been tested by three different factors. The first of these is by plotting the best linear regression that relates the targets to the outputs as shown in Fig 5.4.

![Figure 5.4 Regression fit of the outputs vs. targets for the network (6-10-5-3-1).](image)

*Figure 5.4 Regression fit of the outputs vs. targets for the network (6-10-5-3-1).*
The correlation coefficient (r) is a measure of how well the neural network’s targets can track the variations in the outputs (0 being no correlation at all and 1 being complete correlation). The correlation coefficient in this case has been found to be 0.99967 in this case which indicates excellent correlation.

The second means of testing the performance of the neural network is to plot the confusion matrices for the various types of errors that occurred for the trained neural network. Fig 5.5 plots the confusion matrix for the three phases of training, testing and validation. The diagonal cells in green indicate the number of cases that have been classified correctly by the neural network and the offdiagonal cells which are in red indicate the number of cases that have been wrongly classified by the ANN. The last cell in blue in each of the matrices indicates the total percentage of cases that have been classified correctly in green and the vice-verca in red. It can be seen that the chosen neural network has 100 percent accuracy in fault detection.

Figure 5.5 Confusion matrices for Training, Testing and Validation Phases.
The third step in the testing process is to create a separate set of data called the test set to analyze the performance of the trained neural network. A total of 300 different test cases have been simulated with 200 cases corresponding to different types of faults (about 20 cases for each of the ten faults where the fault resistance and the fault location have been varied in each case). The rest of the 100 cases correspond to the no-fault situation.

After the test set has been fed into the neural network and the results obtained, it was noted that the efficiency of the neural network in terms of its ability to detect the occurrence of a fault is a 100 percent. Hence the neural network can, with utmost accuracy, differentiate a normal situation from a fault condition on a transmission line.

Figure 5.6 Overview of the ANN (6-10-5-3-1) chosen for fault detection.
Figure 5.6 presents a snapshot of the trained ANN with the 6 – 10 – 5 – 3 – 1 configuration and it is to be noted that the number of iterations required for the training process were 55. It can be seen that the mean square error in fault detection achieved by the end of the training process was 9.43e-5 and that the number of validation check fails were zero by the end of the training process.

The structure of the chosen neural network for fault detection is shown in Fig 5.7 with the input layer, hidden layers and the output layer labeled. It is to be noted that there are 6 neurons in the input layer, 3 hidden layers with 10, 5 and 3 neurons in them respectively and one neuron in the output layer.

![Diagram of chosen ANN for fault detection](image)

*Figure 5.7 Chosen ANN for Fault Detection (6 – 10 – 5 – 3 – 1)*
5.2 FAULT CLASSIFICATION

Once a fault has been detected on the power line, the next step is to identify the type of fault. This section presents an analysis on the fault classification phase using neural networks. A review of the different neural networks that were analyzed is provided which is followed by the chosen network.

Fault classifiers based on neural networks have been extensively proposed and used in the past and almost all of these classifiers made use of multilayer perceptron neural network and employed the back-propagation learning strategy. Although back-propagation learning strategy is inherently slow in learning and poses difficulty in choosing the optimal size of the network, it is undoubtedly the ideal strategy to be employed when there is a large training set available because back-propagation algorithm can provide a very compact distributed representation of complex data sets.

5.2.1 TRAINING THE FAULT CLASSIFIER NEURAL NETWORK

The same process that was employed in the previous section (section 4.4.1) is also followed in this section in terms of the design and development of the classifier neural network. The designed network takes in sets of six inputs (the three phase voltage and current values scaled with respect to their corresponding pre-fault values). The neural network has four outputs, each of them corresponding to the fault condition of each of the three phases and one output for the ground line. Hence the outputs are either a 0 or 1 denoting the absence or presence of a fault on the corresponding line (A, B, C or G where A, B and C denote the three phases of the transmission line and G denotes the ground). Hence the various possible permutations can represent each of the various faults.
accordingly. The proposed neural network should be able to accurately distinguish between the ten possible categories of faults. The truth table representing the faults and the ideal output for each of the faults is illustrated in Table 4.2.

Table 5.1 Fault classifier ANN outputs for various faults.

<table>
<thead>
<tr>
<th>Type of Fault</th>
<th>Network Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
</tr>
<tr>
<td>A-G fault</td>
<td>1</td>
</tr>
<tr>
<td>B-G fault</td>
<td>0</td>
</tr>
<tr>
<td>C-G Fault</td>
<td>0</td>
</tr>
<tr>
<td>A-B Fault</td>
<td>1</td>
</tr>
<tr>
<td>B-C Fault</td>
<td>0</td>
</tr>
<tr>
<td>C-A Fault</td>
<td>1</td>
</tr>
</tbody>
</table>
Hence the training set consisted of about 1100 input output sets (100 for each of the ten faults and 100 for the no fault case) with a set of six inputs and one output in each input-output pair. Back-propagation networks with a variety of combinations of hidden layers and the number of neurons per hidden layer have been analyzed. Of these, the ones that achieved satisfactory performance are shown followed by the best neural network which has been described further in detail. Figures 5.8 – 5.12 show the error performance plots of neural networks with 1 and 2 hidden layers respectively. The chosen network has been depicted in Fig 5.17 and the various error performance plots have been shown in Figures 5.13 – 5.18.

Fig 5.8 shows the training performance plot of the neural network 6-5-5-31-4 (6 neurons in the input layer, 3 hidden layers with 5, 5 and 31 neurons in them respectively
and four neurons in the output layer). It can be seen that the best validation performance in terms of the Mean Square Error (MSE) by the end of the training process is 0.01289.

Figure 5.8 Mean-square error performance of the network with configuration (6-5-5-31-4).

Figure 5.9 Mean-square error performance of the network with configuration (6-5-31-4).
Fig 5.9 shows the training performance plot of the neural network 6-5-31-4 (6 neurons in the input layer, 2 hidden layers with 5 and 31 neurons in them respectively and four neurons in the output layer). It can be seen that the best validation performance in terms of the Mean Square Error (MSE) by the end of the training process is 0.019773.

Fig 5.10 shows the training performance plot of the neural network 6-5-4 (6 neurons in the input layer, 1 hidden layer with 5 neurons in it and four neurons in the output layer). It can be seen that the best validation performance in terms of the Mean Square Error (MSE) by the end of the training process in this case is 0.029578.

Fig 5.11 shows the training performance plot of the neural network 6-10-4 (6 neurons in the input layer, 1 hidden layer with 10 neurons in it and four neurons in the output layer). It can be seen that the best validation performance in terms of the Mean Square Error (MSE) by the end of the training process in this case is 0.0077.
Figure 5.11 Mean-square error performance of the network with configuration (6-10-4).

Fig 5.12 shows the training performance plot of the neural network 6-20-4 (6 neurons in the input layer, 1 hidden layer with 20 neurons in it and four neurons in the output layer). It can be seen that the best validation performance in terms of the Mean Square Error (MSE) by the end of the training process in this case is 0.0093975.

Figure 5.12 Mean-square error performance of the network with configuration (6-20-4).
Fig 5.13 shows the training performance plot of the neural network 6-35-4 (6 neurons in the input layer, 1 hidden layer with 35 neurons in it and four neurons in the output layer). It can be seen that the best validation performance in terms of the Mean Square Error (MSE) by the end of the training process in this case is 0.00359.

From the above training performance plots, it is to be noted that satisfactory training performance has been achieved by the neural network with the 6-35-4 configuration (6 neurons in the input layer, 35 neurons in the hidden layer and one neuron in the output layer). The overall MSE of the trained neural network is 0.0035986 and it can be seen from Fig 5.13 that the testing and the validation curves have similar characteristics which is an indication of efficient training. Hence this has been chosen as the ideal ANN for the purpose of fault classification.
5.2.2 TESTING THE FAULT CLASSIFIER NEURAL NETWORK

Once the neural network has been trained, its performance has been tested by taking three different factors into consideration. The first of these is by plotting the best linear regression that relates the targets to the outputs as shown in Fig 5.15. The correlation coefficient in this case was found to be 0.98108 which indicates satisfactory correlation between the targets and the outputs. The dotted line in the figure indicates the ideal regression fit and the red solid line indicates the actual fit of the neural network. It can be seen that both these lines track each other very closely which is an indication of very good performance by the neural network.

![Regression fit of the Outputs vs. Targets of ANN with configuration (6-35-4).](image)

*Figure 5.14 Regression fit of the Outputs vs. Targets of ANN with configuration (6-35-4).*

The second factor in the testing process is to plot the Receiver Operating Characteristics curve (ROC). The ROC curves for each of the training, testing and
validation phases have been shown in Fig 5.15 along with the overall ROC curve. The ROC curves are actually plots between the true positive rates (rate of positive classification) and the false positive rates (rate of incorrect classification) of the neural network classifier. Hence, an ideal ROC curve would show points only in the upper-left corner because that is an indication of 100 percent true positivity and 0 percent false positivity in the classification. It is to be noted that the ROC curves plotted in Fig 5.15 are almost perfect since they all have the lines in the upper-left corner.

Figure 5.15 Gradient and Validation performance of the ANN with configuration (6-35-4).

The third step in the testing process is to create a separate set of data called the test set to analyze the performance of the trained neural network. A total of 300 different test cases have been simulated with 550 cases corresponding to different types of faults (about 50 cases for each of the ten faults where the fault resistance and the fault location have been varied in each case). The rest of the 50 cases correspond to the no-fault situation.
After the test set has been fed into the neural network and the results obtained, it was noted that the efficiency of the neural network in terms of its ability to identify the type of the fault is a 100 percent. Hence the neural network can, with utmost accuracy, differentiate between the ten possible types of faults on a transmission line.

Fig 5.16 provides an overview on the neural network and is a screen shot of the training window simulated using the Artificial Neural Network Toolbox in Simulink. Important things to be noted are that the training process converged in about 144 iterations and that the performance in terms of mean square error achieved by the end of the training process was 6.26e-3.

![Neural Network Diagram]

*Figure 5.16 Overview of the ANN with configuration (6-35-4), chosen as fault classifier.*
Fig 5.17 shows the structure of the chosen ANN for the purpose of fault classification and the neural network has 6 neurons in the input layer, 35 neurons in the hidden layer and four neurons in the output layer as shown. Each of the neurons in the output layer would indicate the fault condition on each of the three phases (A, B and C) and the fourth neuron is to identify if the fault is a ground fault. An output of 0 corresponds to no fault while an output of 1 indicates that the phase is faulted.

*Figure 5.17 Chosen ANN for Fault Classification (6 – 35 – 4).*
5.3 FAULT LOCATION

This section talks about the design, development and the implementation of the neural network based fault locators for each of the various types of faults. This forms the third step in the entire process of fault location after the inception of the fault. The following subsections deal with the various kinds of faults and their error performances individually.

5.3.1 SINGLE LINE – GROUND FAULTS

Now that we can detect the occurrence of a fault on a transmission line and also classify the fault into the various fault categories, the next step is to pin-point the location of the fault from either ends of the transmission line. Three possible single line - ground faults exist (A-G, B-G, C-G), corresponding to each of the three phases (A, B or C) being faulted.

5.3.1.1 TRAINING THE NEURAL NETWORK FOR SINGLE LINE – GROUND FAULT LOCATION

Feed forward back – propagation neural networks have been surveyed for the purpose of single line – ground fault location, mainly because of the availability of sufficient relevant data for training. In order to train the neural network, several single phase faults have been simulated on the transmission line model. For each of the three phases, faults have been simulated at every 3 Km on a 300 Km long transmission line. Along with the fault distance, the fault resistance has been varied as mentioned earlier in section 4.4. Hence, a total of 2400 cases have been simulated (100 for each of the three
phases with each of the eight different fault resistances as 0.25, 0.5, 0.75, 1, 5, 10, 25 and 50 ohms respectively). In each of these cases, the voltage and current samples for all three phases (scaled with respect to their pre-fault values) are given as inputs to the neural network. The output of the neural network is the distance to the fault from terminal A. Firstly, a few of the various neural networks (with varying combination of hidden layers and number of neurons per hidden layer) that performed reasonably well are presented along with their respective error performances and then the chosen neural network is shown with all its characteristics depicted in detail. Efficiency of each of the trained networks is analyzed based on their regression performance and their performance in the testing phase. The test performance plots are obtained by simulating various faults on different phases at varying locations and calculating the error in the output produced by the Neural Network. Figures 5.18 – 5.25 show the error performance and regression plots of neural networks with 1 and 2 hidden layers. The chosen network has been depicted in Fig 5.30 and its various error performance plots have been shown in Figures 5.26 – 5.31.

Fig 5.18 plots the best linear regression fit between the outputs and the targets of the neural network with 6 neurons in the input layer, 2 hidden layers with 5 and 5 neuron in them respectively and 1 neuron in the output layer (6-5-5-1). The correlation coefficient (r) as mentioned earlier is a measure of how well the neural network relates the outputs and the targets. The closer the value of r is to 1, the better the performance of the neural network. The value of r in this case is found to be 0.99799. In order to test the performance of this network, 12 different single phase faults have been simulated on different phases with the fault distance being incremented by 25 Km in each case and the
percentage error in calculated output has been calculated. Fig 5.19 shows the results of this test conducted on the neural network (6-5-5-1). It can be seen that the maximum error is almost 4.5 percent.

![Regression fit of the Outputs vs. Targets](image1)

*Figure 5.18 Regression fit of the Outputs vs. Targets with configuration (6-5-5-1).*

![Test Phase performance of the Neural Network](image2)

*Figure 5.19 Test Phase performance of the Neural Network with configuration (6-5-5-1).*

Fig 5.20 plots the best linear regression fit between the outputs and the targets of the neural network with 6 neurons in the input layer, 25 neurons in the hidden layer and 1
neuron in the output layer (6-25-1). The value of the correlation coefficient r in this case is found to be 0.9959. In order to test the performance of this network, 12 different single phase faults have been simulated on different phases with the fault distance being incremented by 25 Km in each case and the percentage error in calculated output has been calculated. Fig 5.21 shows the results of this test conducted on the neural network (6-25-1). It can be seen that the maximum error is around 7 percent which is not very satisfactory.

![Figure 5.20 Regression fit of the outputs versus targets with configuration (6-25-1).](image)

![Figure 5.21 Test phase performance of the ANN with configuration (6-25-1)](image)
Fig 5.22 plots the best linear regression fit between the outputs and the targets of the neural network with 6 neurons in the input layer, 16 neurons in the hidden layer and 1 neuron in the output layer (6-16-1). The value of the correlation coefficient $r$ in this case is found to be 0.99906.

![Regression fit plot](image)

*Figure 5.22 Regression fit of the outputs versus targets with configuration (6-16-1).*

![Performance plot](image)

*Figure 5.23 Test phase performance of the neural network with configuration (6-16-1).*
In order to test the performance of this network, 12 different single phase faults have been simulated on different phases with the fault distance being incremented by 25 Km in each case and the percentage error in calculated output has been calculated. Fig 5.23 shows the results of this test conducted on the neural network (6-16-1). It can be seen that the maximum error is around 4.75 percent.

Fig 5.24 plots the best linear regression fit between the outputs and the targets of the neural network with 6 neurons in the input layer, 7 neurons in the hidden layer and 1 neuron in the output layer (6-7-1). The value of the correlation coefficient r in this case is found to be 0.99924 which is by far the best and the closest to one.

Figure 5.24 Regression fit of the outputs versus targets with configuration (6-7-1).
In order to test the performance of this network, 100 different single phase faults have been simulated on different phases with the fault distance being incremented by 10 Km in each case and the percentage error in calculated output has been calculated. Fig 5.25 shows the results of this test conducted on the neural network (6-7-1). It can be seen that the maximum error is around 1.65 percent which is very satisfactory. It is to be noted that the average error in fault location is just 0.89 percent.

Fig 5.26 shows an overview of the chosen ANN and it can be seen that the training algorithm used is Levenberg - Marquardt algorithm. The performance function chosen for the training process is mean square error. Fig 5.27 plots the mean-square error as a function of time during the learning process and it can be seen that the achieved MSE is about 0.0005056 which is way below the MSE goal of 0.01.
Figure 5.26 Overview of the chosen ANN with configuration (6-7-1).

Figure 5.27 Mean-square error performance of the network with configuration (6-7-1).
5.3.1.2 TESTING THE NEURAL NETWORK FOR SINGLE LINE – GROUND FAULT LOCATION

Several factors have been considered while testing the performance of the neural networks. One prime factor that evaluates the efficiency of the ANN is the test phase performance already illustrated in Fig 5.27. As already mentioned, the average and the maximum error percentages are in tolerable ranges and hence the networks performance is considered satisfactory. Another form of analysis is provided by Fig 5.30, which is the gradient and validation performance plot. It can be seen that there is a steady decrease in the gradient and also that the number of validation fails are 0 during the entire process which indicates smooth and efficient training.

![Gradient and validation performance of the network with configuration (6-7-1).](image)

*Figure 5.28 Gradient and validation performance of the network with configuration (6-7-1).*

The third factor that is considered while evaluating the performance of the network is the correlation coefficient of each of the various phases of training, validation and testing. Fig 5.29 shows the regression plots of the various phases such as training, testing and validation. It can be seen that the best linear fit very closely matches the ideal case with an overall correlation coefficient of 0.99924.
Figure 5.29 Regression plots of various phases of learning of the ANN with configuration (6-7-1).

Fig 5.30 shows the structure of the chosen ANN for single line – ground faults with 6 neurons in the input layer, 7 neurons in the hidden layer and 1 neuron in the output layer (6-7-1).

Figure 5.30 Structure of the chosen ANN with configuration (6-7-1).
Table 5.2 illustrates the percentage errors in Fault location as a function of Fault Distance and Fault Resistance. Two different cases have been considered (shown in adjacent columns), one with a fault resistance of 20 ohms and another with a fault resistance of 60 ohms. It is to be noted that the resistance of 20 ohms was used as a part of training data set and hence the average percentage error in fault location in this case is just 0.1646 %. The second case illustrates the same with a different fault resistance of 60 ohms which is relatively very high and is not a part of the training set. Hence, the performance of the neural network in this case illustrates its ability to generalize and react upon new data. It is to be noted that the average error in this case is just 0.878 % which is very satisfactory. Thus the neural networks performance is considered satisfactory and can be used for the purpose of single line – ground fault location.

Table 5.2 Percentage errors as a function of fault distance and fault resistance for the ANN chosen for single line - ground fault location.

<table>
<thead>
<tr>
<th>Serial No:</th>
<th>% Error vs. Fault Distance (Fault Resistance = 20 Ω)</th>
<th>% Error vs. Fault Distance (Fault Resistance = 60 Ω)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fault Resistance (Ω)</td>
<td>Measured Fault Location</td>
</tr>
<tr>
<td>1</td>
<td>25</td>
<td>25.49</td>
</tr>
<tr>
<td>2</td>
<td>75</td>
<td>75.58</td>
</tr>
<tr>
<td>3</td>
<td>125</td>
<td>125.12</td>
</tr>
<tr>
<td>4</td>
<td>175</td>
<td>175.09</td>
</tr>
<tr>
<td>5</td>
<td>225</td>
<td>225.91</td>
</tr>
</tbody>
</table>
5.3.2 LINE–LINE FAULTS

The design, development and performance of neural networks for the purpose of Line–Line fault location are discussed in this section. Now that we can detect the occurrence of a fault on a transmission line and also classify the fault into the various fault categories, the next step is to pin-point the location of the fault from either ends of the transmission line. Three possible line–line faults exist (A-B, B-C, C-A), corresponding to each of the three phases (A, B or C) being faulted.

5.3.2.1 TRAINING THE NEURAL NETWORK FOR LINE–LINE FAULT LOCATION

Feed forward back–propagation neural networks have been surveyed for the purpose of line–line fault location, mainly because of the availability of sufficient data to train the network. In order to train the neural network, several line–line faults have been simulated on the transmission line model. For each pair formed by the three phases, faults have been simulated at every 3 Km on a 300 Km long transmission line. Along with the fault distance, the fault resistance has been varied as 0.25, 0.5, 0.75, 1, 5, 10, 25 and 50 ohms respectively. Hence, a total of 2400 cases have been simulated (100 for each of the three phases with each of the eight different fault resistances). In each of these cases, the voltage and current samples for all three phases (scaled with respect to their pre-fault values) are given as inputs to the neural network. The output of the neural network is the distance to the fault from terminal A. Hence, each input output pair consists of six inputs and one output. An exhaustive survey on various neural networks has been performed by varying the number of hidden layers and the number of neurons.
per hidden layer. Certain neural networks that achieved satisfactory performance are presented first along with their error performance plots. Of these ANNs, the most appropriate ANN is chosen based on its Mean Square Error performance and the Regression coefficient of the Outputs versus Targets. Figures 5.31 – 5.32 show the MSE and the Test phase performance plots of the neural networks 6 – 10 – 20 – 5 – 1 with 3 hidden layers. Figures 5.33 – 5.34 show the MSE and the Test phase performance plots of the neural network 6 – 10 – 1 with 1 hidden layer.

Fig 5.31 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 3 hidden layers with 10, 20 and 5 neurons in them respectively and 1 neuron in the output layer (6 – 10 – 20 – 5 – 1). It can be seen that the best MSE performance of this neural network is 0.0073438 which is below the MSE goal of 0.01. It was found that the correlation coefficient between the outputs and the targets was 0.98469 in this case.

*Figure 5.31 Mean Square Error performance plot with configuration (6-10-20-5-1).*
In order to test the performance of this network, 12 different line–line faults have been simulated on different phases with the fault distance being incremented by 25 Km in each case and the percentage error in calculated output has been calculated. Fig 5.32 shows the results of this test conducted on the neural network (6-10-20-5-1). It can be seen that the maximum error is around 2.75 percent.

![Graph showing test phase performance of the ANN with configuration (6-10-20-5-1).](image)

*Figure 5.32 Test Phase performance of the ANN with configuration (6-10-20-5-1).*

Fig 5.33 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 10 neurons in the hidden layer and 1 neuron in the output layer (6 – 10 – 1). It can be seen that the best MSE performance of this neural network is 0.0045535 which is below the MSE goal of 0.01. It was found that the correlation coefficient between the outputs and the targets was 0.9825 for this neural network.
Figure 5.33 Mean Square Error performance plot with configuration (6-10-1).

Figure 5.34 Test Phase performance of the ANN with configuration (6-10-1).

In order to test the performance of this network, 12 different line–line faults have been simulated on different phases with the fault distance being incremented by 25 Km in each case and the percentage error in calculated output has been calculated. Fig 5.34 shows the results of this test conducted on the neural network (6-10-1). It can be seen that the maximum error is around 4.65 percent which is unacceptable.
Fig 5.35 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 2 hidden layers with 10 and 5 neurons in them respectively and 1 neuron in the output layer (6 – 10 – 5 – 1). It can be seen that the best MSE performance of this neural network is 0.002089 which is below the MSE goal of 0.01. It was found that the correlation coefficient between the outputs and the targets was 0.98648 for this neural network.

Figure 5.35 Mean Square Error performance of the ANN with configuration (6-10-5-1).

Figure 5.36 Test phase performance of the neural network with configuration (6-10-5-1).
In order to test the performance of this network, 100 different phase to phase faults have been simulated on different phases with the fault distance being incremented by 10 Km in each case and the percentage error in calculated output has been calculated. Fig 5.36 shows the results of this test conducted on the neural network (6-10-5-1). It can be seen that the maximum error is around 1.7 percent which is very satisfactory. It is to be noted that the average error in fault location is just 0.97 percent. Hence, this neural network has been chosen as the ideal network for the purpose of line – line fault location on transmission lines.

Fig 5.37 shows an overview of the chosen ANN and it can be seen that the training algorithm used is Levenberg - Marquardt algorithm. The performance function chosen for the training process is mean square error. Fig 5.38 plots the best linear regression fit between the outputs and the targets and the correlation coefficient for the same has been found to be 0.98648 which is a decently good regression fit.

Figure 5.37 Overview of the chosen ANN for Line-Line Faults (6-10-5-1).
5.3.2.2 TESTING THE NEURAL NETWORK FOR LINE – LINE FAULT LOCATION

Several factors have been considered while testing the performance of the chosen neural network. One prime factor that evaluates the efficiency of the ANN is the test phase performance plot which is already illustrated in Fig 5.38. As already mentioned, the average and the maximum error percentages are in tolerable ranges and hence the network’s performance is considered satisfactory. Another means of evaluating the ANN is provided by Fig 5.41, which is the gradient and validation performance plot. It can be seen that there is a steady decrease in the gradient and also that the number of validation fails did not exceed 1 during the entire process which indicates smooth and efficient training because the validation and the test phases reached the MSE goal at the same time approximately.
The third factor that is considered while evaluating the performance of the network is the correlation coefficient of each of the various phases of training, validation and testing. Fig 5.40 shows the regression plots of the various phases such as training, testing and validation. It can be seen that the best linear fit very closely matches the ideal case with an overall correlation coefficient of 0.98648.
Fig 5.41 shows the structure of the chosen ANN for line – line faults with 6 neurons in the input layer, 2 hidden layers with 10 and 5 neurons in them respectively and 1 neuron in the output layer (6 – 10 – 5 – 1).

Table 5.3 illustrates the percentage errors in Fault location as a function of Fault Distance and Fault Resistance. Two different cases have been considered (shown in adjacent columns), one with a fault resistance of 20 ohms and another with a fault resistance of 60 ohms. It is to be noted that the resistance of 20 ohms was used as a part of training data set and hence the average percentage error in fault location in this case is just 0.1386 %. The second case illustrates the same with a different fault resistance of 60 ohms which is relatively very high and is not a part of the training set. Hence, the performance of the neural network in this case illustrates its ability to generalize and react upon new data. It is to be noted that the average error in this case is just 0.966 % which is
still very satisfactory. Thus the neural networks performance is considered satisfactory and can be used for the purpose of line – line fault location.

Table 5.3 Percentage errors as a function of fault distance and fault resistance for the ANN chosen for line - line fault location.

<table>
<thead>
<tr>
<th>Serial No:</th>
<th>Fault Distance (Km)</th>
<th>Measured Fault Location</th>
<th>Percentage Error</th>
<th>Fault Distance (Km)</th>
<th>Measured Fault Location</th>
<th>Percentage Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>25</td>
<td>25.03</td>
<td>0.01</td>
<td>50</td>
<td>51.17</td>
<td>0.39</td>
</tr>
<tr>
<td>2</td>
<td>75</td>
<td>75.39</td>
<td>0.13</td>
<td>100</td>
<td>102.52</td>
<td>0.84</td>
</tr>
<tr>
<td>3</td>
<td>125</td>
<td>125.67</td>
<td>0.223</td>
<td>150</td>
<td>153.63</td>
<td>1.21</td>
</tr>
<tr>
<td>4</td>
<td>175</td>
<td>175.14</td>
<td>0.047</td>
<td>200</td>
<td>201.98</td>
<td>0.66</td>
</tr>
<tr>
<td>5</td>
<td>225</td>
<td>225.85</td>
<td>0.283</td>
<td>250</td>
<td>255.19</td>
<td>1.73</td>
</tr>
</tbody>
</table>

5.3.3 DOUBLE LINE – GROUND FAULTS

The design, development and performance of neural networks for the purpose of Double Line – Ground fault location are discussed in this section. The third category of faults is the double line – ground faults. Three possible double line – ground faults exist which are denoted as ABG, BCG and ACG (based on which two of the three phases A,B and C are faulted).
5.3.3.1 TRAINING THE NEURAL NETWORK FOR DOUBLE LINE – GROUND FAULT LOCATION

Feed forward back – propagation algorithm was once again used for the purpose of double line – ground fault location on transmission lines. The reason for doing so, as already mentioned is that these networks perform very efficiently when there is availability of a sufficiently large training data set. For the purpose of training the neural network, several double line – ground faults have been simulated on the modeled transmission line on each of the three phases. The various factors that were varied were the fault distance (incremented by 3 km each time), the fault resistance (one of the chosen eight different fault resistances) and the phases that were faulted. About 100 fault cases were simulated for each phase with each of the eight different resistances as 0.25, 0.5, 0.75, 1, 5, 10, 25 and 50 ohms respectively. Hence a total of 2400 fault cases were simulated on the transmission line. In each of these cases, the voltage and current samples on all three phases (scaled with respect to their pre-fault values) are fed as inputs to the neural network. The neural network’s output is the distance to the fault from terminal A. Thus each input output pair fed into the neural network has a set of six inputs and one output.

An exhaustive survey on various neural networks has been performed by varying the number of hidden layers and the number of neurons per hidden layer. A few neural networks that achieved satisfactory performance are presented first along with their error performance plots. Of these ANNs, the most appropriate ANN is chosen based on its Mean Square Error performance and the Regression coefficient of the Outputs vs. Targets. Figures 5.42 – 5.45 show the MSE and the Test phase performance plots of the
neural networks 6 – 10 – 1 and 6 – 20 – 1 with 1 hidden layer. Figures 5.46 – 5.49 show the MSE and the Test phase performance plots of the neural network 6 – 10 – 5 – 1 and 6 – 21 – 11 – 1 with 2 hidden layers.

Fig 5.42 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 10 neurons in the hidden layer and 1 neuron in the output layer (6 – 10 – 1). It can be seen that the best MSE performance of this neural network is 0.0047967 which is below the MSE goal of 0.01 (denoted by the black dotted line). It was found that the correlation coefficient between the outputs and the targets was 0.98193 in this case.

![Figure 5.42 Mean Square Error performance of the ANN with configuration (6-10-1).](image)

In order to test the performance of this network, 12 different double line – ground faults have been simulated on different phases with the fault distance being incremented
by 25 Km in each case and the percentage error in ANN’s output has been calculated. Fig 5.43 shows the results of this test conducted on the neural network (6-10-1). It can be seen that the maximum error is higher than 5 percent which is exorbitantly high.

![Graph showing error vs. fault location](image)

*Figure 5.43 Test Phase performance of the ANN with configuration (6-10-1).*

Fig 5.44 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 20 neurons in the hidden layer and 1 neuron in the output layer (6 – 20 – 1). It can be seen that the best MSE performance of this neural network is 0.0013561 which is below the MSE goal of 0.01 (denoted by the black dotted line in the figure). It was found that the correlation coefficient between the outputs and the targets was 0.98804 for this neural network.
Figure 5.44 Mean Square Error performance of the ANN with configuration (6-20-1).

Figure 5.45 Test Phase performance of the ANN with configuration (6-20-1).
In order to test the performance of this network the same method adopted for the earlier case is followed. 12 different double line – ground faults have been simulated on different phases with the fault distance being incremented by 25 Km in each case and the percentage error in ANN’s output has been calculated. Fig 5.45 shows the results of this test conducted on the neural network (6-20-1). It is to be noted that the maximum error is higher than 4.75 percent which is too high for this purpose.

Fig 5.46 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 2 hidden layers with 10 and 5 neurons in them respectively and 1 neuron in the output layer (6 – 10 – 5 – 1). It can be seen that the best MSE performance of this neural network is 0.00338785 which is below the MSE goal of 0.01 (denoted by the black dotted line in the figure). It was found that the correlation coefficient between the outputs and the targets was 0.98913 for this neural network.

![Figure 5.46 Mean Square Error performance of the neural network with configuration (6-10-5-1).](image)
In order to test the performance of this network the same method adopted for the earlier case is followed. 12 different double line – ground faults have been simulated on different phases with the fault distance being incremented by 25 Km in each case and the percentage error in ANN’s output has been calculated. Fig 5.47 shows the results of this test conducted on the neural network (6-10-5-1). It is to be noted that the maximum error is higher than 3.5 percent which is still not satisfactory for this purpose.

![Graph](image)

*Figure 5.47 Test Phase performance of the ANN (6-10-5-1).*

Fig 5.48 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 2 hidden layers with 21 and 11 neurons in them respectively and 1 neuron in the output layer (6 – 21 – 11 – 1). It can be seen that the best MSE performance of this neural network is 0.00159395 which is below the MSE goal of 0.01 (denoted by the black dotted line in the figure). It was found that
the correlation coefficient between the outputs and the targets was 0.99329 for this neural network which indicates very good regression fit.

**Figure 5.48 Mean Square Error performance of the neural network with configuration (6-21-11-1).**

**Figure 5.49 Test phase performance of the ANN (6-21-11-1).**
In order to test the performance of this network, 100 different double line–ground faults have been simulated on different phases with the fault distance being incremented by 10 Km in each case and the percentage error in calculated output has been calculated. Fig 5.49 shows the results of this test conducted on the neural network (6-21-11-1). It can be seen that the maximum error is around 1.71 percent which is very satisfactory. It is to be noted that the average error in fault location is just 0.863 percent. Hence, this neural network has been chosen as the ideal network for the purpose of double line–ground fault location on transmission lines.

Fig 5.50 shows an overview of the chosen ANN and it can be seen that the training algorithm used is Levenberg - Marquardt algorithm. The performance function chosen for the training process is mean square error. Fig 5.51 plots the best linear regression fit between the outputs and the targets. As already mentioned, the correlation coefficient in this case is found to be 0.99329 which is very good.

![Figure 5.50 Overview of the chosen ANN (6-21-11-1) for Double Line-Ground Faults.](image-url)
5.3.3.2 TESTING THE NEURAL NETWORK FOR DOUBLE LINE – GROUND FAULT LOCATION

Now that the neural network has been trained, the next important step is to analyze the performance of this network which is called testing. The methods and means by which this neural network has been tested are discussed here under. One important factor that helps test the network is the test phase performance plot as shown in Fig 5.51. It is to be noted that both the average as well as the maximum error percentages are in acceptable levels and hence the networks performance is satisfactory. Another means of determining the efficiency of a trained neural network is to check the gradient and validation performance plot as shown in Fig 5.54. It can be seen that there is a steady decrease in the gradient and also that the maximum number of validation fails is 3 during

*Figure 5.51 Regression fit of the outputs versus targets with configuration (6-21-11-1).*
the training process. This indicates efficient training because the validation phase follows
the test phase closely if the number of validation fails is low. This further implies that the
neural network can generalize new data fed into it more effectively.

Figure 5.52 Gradient and validation performance plot of ANN with configuration (6-21-11-1).

Figure 5.53 Regression plots of the various stages of learning of ANN (6-21-11-1).
The third factor that is considered while evaluating the performance of the network is the correlation coefficient of each of the various phases of training, validation and testing. Fig 5.53 shows the regression plots of the various phases such as training, testing and validation. It can be seen that the best linear fit very closely matches the ideal case with an overall correlation coefficient of 0.99329.

Fig 5.54 shows the structure of the chosen ANN for double line - ground fault location with 6 neurons in the input layer, 2 hidden layers with 21 and 11 neurons in them respectively and 1 neuron in the output layer (6 – 21 – 11 – 1).

![Structure of the chosen ANN (6 – 21 – 11 – 1).](image)

*Figure 5.54 Structure of the chosen ANN (6 – 21 – 11 – 1).*
Table 5.4 illustrates the percentage errors in Fault location as a function of Fault Distance and Fault Resistance. Two different cases have been considered (shown in adjacent columns), one with a fault resistance of 20 ohms and another with a fault resistance of 60 ohms. It is to be noted that the resistance of 20 ohms was used as a part of training data set and hence the average percentage error in fault location in this case is just 0.091 %. The second case illustrates the same with a different fault resistance of 60 ohms which is relatively very high and is not a part of the training set. Hence, the performance of the neural network in this case illustrates its ability to generalize and react upon new data. It is to be noted that the average error in this case is just 1.122 % which is still acceptable. Thus the neural networks performance is considered satisfactory and can be used for the purpose of double line – ground fault location.

<table>
<thead>
<tr>
<th>Serial No:</th>
<th>% Error vs. Fault Distance (Fault Resistance = 20 Ω)</th>
<th>% Error vs. Fault Distance (Fault Resistance = 60 Ω)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fault Distance (Km)</td>
<td>Measured Fault Location</td>
</tr>
<tr>
<td>1</td>
<td>25</td>
<td>25.53</td>
</tr>
<tr>
<td>2</td>
<td>75</td>
<td>75.18</td>
</tr>
<tr>
<td>3</td>
<td>125</td>
<td>125.11</td>
</tr>
<tr>
<td>4</td>
<td>175</td>
<td>175.16</td>
</tr>
<tr>
<td>5</td>
<td>225</td>
<td>225.39</td>
</tr>
</tbody>
</table>
5.3.4 THREE PHASE FAULTS

The design, development and performance of neural networks for the purpose of three-phase fault location are discussed in this section. The fourth and the final category of faults are the three phase faults. There exists only one kind of three phase faults which is denoted as ABC fault where in all the three phases A, B and C are faulted.

5.3.4.1 TRAINING THE NEURAL NETWORK FOR THREE PHASE FAULT LOCATION

Feed forward back–propagation algorithm was once again used for the purpose of three phase fault location on transmission lines. The reason for doing so, as already mentioned is that these networks perform very efficiently when there is availability of a sufficiently large training data set. For the purpose of training the neural network, several three phase faults have been simulated on the modeled transmission line. The various factors that were varied were the fault distance (incremented by 3 km each time) and the fault resistance (one of the chosen eight different fault resistances). About 100 fault cases were simulated with each of the eight different resistances as 0.25, 0.5, 0.75, 1, 5, 10, 25 and 50 ohms respectively. Hence a total of 800 fault cases were simulated on the transmission line. In each of these cases, the voltage and current samples on all three phases (scaled with respect to their pre-fault values) are fed as inputs to the neural network. The neural network’s output is the distance to the fault from terminal A. Thus each input output pair fed into the neural network has a set of six inputs and one output. An exhaustive survey on various neural networks has been performed by varying the number of hidden layers and the number of neurons per hidden layer. A few neural
networks that achieved satisfactory performance are presented first along with their error performance plots. Of these ANNs, the most appropriate ANN is chosen based on its Mean Square Error performance and the Regression coefficient of the Outputs vs. Targets. Figures 5.55 – 5.57 show the MSE and the Test phase performance plots of the neural network 6 – 21 – 10 – 1 with 2 hidden layers. Figures 5.58 – 5.60 show the MSE and the Test phase performance plots of the neural network 6 – 21 – 1 with 1 hidden layer.

Fig 5.55 plots the best linear regression fit between the outputs and the targets of the neural network with 6 neurons in the input layer, 2 hidden layers with 21 and 10 neurons in them respectively and 1 neuron in the output layer (6 – 21 – 10 – 1). The correlation coefficient (r) as mentioned earlier is a measure of how well the neural network relates the outputs and the targets. The closer the value of r is to 1, the better the performance of the neural network. The value of r in this case is found to be 0.99706.
Fig 5.56 MSE performance of the neural network with configuration (6-21-10-1).

Fig 5.56 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 2 hidden layers with 21 and 10 neurons in them respectively and 1 neuron in the output layer (6 – 21 – 10 – 1). It can be seen that the best MSE performance of this neural network is 0.00067433 (denoted by the dotted green line) which is below the MSE goal of 0.01 (denoted by the black line).

Fig 5.57 Test Phase performance of the ANN with configuration (6-21-10-1).
In order to test the performance of this network, 12 different three phase faults have been simulated on the transmission line with the fault distance being incremented by 25 Km in each case and the percentage error in ANN’s output has been calculated. Fig 5.57 shows the results of this test conducted on the neural network (6-21-10-1). It can be seen that the maximum error is higher than 3 percent which is fairly satisfactory. However neural networks that can perform better are more desirable.

Fig 5.58 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 1 hidden layer with 21 neurons in it and 1 neuron in the output layer (6 – 21 – 1). It can be seen that the best MSE performance of this neural network is 0.00076875 (denoted by the dotted green line) which is below the MSE goal of 0.01 (denoted by the black dotted line).

![Figure 5.58 MSE performance of the neural network with configuration (6-21-1).](image)

*Figure 5.58 MSE performance of the neural network with configuration (6-21-1).*
Fig 5.59 plots the best linear regression fit between the outputs and the targets of the neural network with 6 neurons in the input layer, 1 hidden layer with 21 neurons in it and 1 neuron in the output layer (6 – 21 – 1). The correlation coefficient (r) as mentioned earlier is a measure of how well the neural network relates the outputs and the targets. The closer the value of r is, to 1, the better the performance of the neural network. The value of r in this case is found to be 0.99804 which is an improvement from the previous case (6-21-10-1).

![Regression fit for the outputs versus targets of ANN with configuration (6-21-1).](image)

*Figure 5.59 Regression fit for the outputs versus targets of ANN with configuration (6-21-1).*

In order to test the performance of this network, 12 different three phase faults have been simulated on the transmission line with the fault distance being incremented by 25 Km in each case and the percentage error in ANN’s output has been calculated. Fig 5.60 shows the results of this test conducted on the neural network (6-21-1). It can be
seen that the maximum error is just lower than 3 percent which is a significant improvement from the previous case.

Figure 5.60 Test Phase performance of the ANN with configuration (6-21-1).

Fig 5.61 plots the best linear regression fit between the outputs and the targets of the neural network with 6 neurons in the input layer, 3 hidden layers with 6, 21 and 16 neurons in them respectively and 1 neuron in the output layer (6 – 6 – 21 – 16 – 1). The correlation coefficient (r) as mentioned earlier is a measure of how well the neural network relates the outputs and the targets. The closer the value of r is, to 1, the better the performance of the neural network. The value of r in this case is found to be 0.99897 which is very close to 1.
In order to test the performance of this network, 100 different three phase faults have been simulated on the transmission line with the fault distance being incremented by 10 Km in each case and the percentage error in ANN’s output has been calculated. Fig 5.62 shows the results of this test conducted on the neural network (6-6-21-16-1). It can
be seen that the maximum error is around 1.62 percent which is very satisfactory. It is to be noted that the average error in fault location is just 0.677 percent. Hence, this neural network has been chosen as the ideal network for the purpose of three phase fault location on transmission lines.

Fig 5.63 shows an overview of the chosen ANN and it can be seen that the training algorithm used is Levenberg - Marquardt algorithm. The performance function chosen for the training process is mean square error.

![Neural Network Diagram](image)

**Figure 5.63 Overview of the chosen neural network for three phase fault location.**

Fig 5.64 shows the performance of the neural network (in terms of training, testing and validation) with 6 neurons in the input layer, 1 hidden layer with 21 neurons in it and 1 neuron in the output layer (6 – 6 – 21 – 16 – 1). It can be seen that the best MSE performance of this neural network is 0.00060607 (denoted by the dotted green line) which is below the MSE goal of 0.01 (denoted by the black dotted line).
5.3.4.2 TESTING THE NEURAL NETWORK FOR THREE PHASE FAULT LOCATION

Now that the neural network has been trained, the next step is to analyze the performance of this network which is called testing. The methods and means by which this neural network has been tested are discussed here in this section. One important factor that helps test the network is the test phase performance plot as shown in Fig 5.64. It is to be noted that both the average as well as the maximum error percentages in accurately determining the location of the fault are in acceptable levels and hence the network’s performance is satisfactory.

Figure 5.64 Mean Square Error performance of the neural network (6-6-21-16-1).
Another important means of determining the efficiency of a trained neural network is to check the gradient and validation performance plot as shown in Fig 5.67. It
can be seen that there is a steady and smooth decrease in the gradient and also that the maximum number of validation fails is 0 during the training process. This indicates efficient training because the validation phase follows the test phase closely if the number of validation fails is low. This is further indicated by the test and validation curves on Fig 5.66. This further implies that the neural network can generalize new data fed into it more effectively.

The third factor that is considered while evaluating the performance of the network is the correlation coefficient of each of the various phases of training, validation and testing. Fig 5.66 shows the regression plots of the various phases such as training, testing and validation. It can be seen that the best linear fit very closely matches the ideal case with an overall correlation coefficient of 0.99329.

Fig 5.67 shows the structure of the chosen ANN for three-phase faults with 6 neurons in the input layer, 1 hidden layer with 21 neurons in it and 1 neuron in the output layer (6 – 6 – 21 – 16 – 1).

![Figure 5.67 Structure of the chosen ANN (6 – 6 – 21 – 16 – 1).](image-url)
Table 5.5 illustrates the percentage errors in Fault location as a function of Fault Distance and Fault Resistance. Two different cases have been considered (shown in adjacent columns), one with a fault resistance of 20 ohms and another with a fault resistance of 60 ohms. It is to be noted that the resistance of 20 ohms was used as a part of training data set and hence the average percentage error in fault location in this case is just 0.178 %. The second case illustrates the same with a different fault resistance of 60 ohms which is relatively very high and is not a part of the training set. Hence, the performance of the neural network in this case illustrates its ability to generalize and react upon new data. It is to be noted that the average error in this case is just 0.836 % which is still acceptable. Thus the neural networks performance is considered satisfactory and can be used for the purpose of three phase fault location.

Table 5.5 Percentage errors as a function of fault distance and fault resistance for the ANN chosen for three phase fault location.

<table>
<thead>
<tr>
<th>Serial No:</th>
<th>% Error vs. Fault Distance (Fault Resistance = 20 Ω)</th>
<th>% Error vs. Fault Distance (Fault Resistance = 60 Ω)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fault Distance (Km)</td>
<td>Measured Fault Location</td>
</tr>
<tr>
<td>1</td>
<td>25</td>
<td>25.51</td>
</tr>
<tr>
<td>2</td>
<td>75</td>
<td>75.17</td>
</tr>
<tr>
<td>3</td>
<td>125</td>
<td>125.52</td>
</tr>
<tr>
<td>4</td>
<td>175</td>
<td>175.69</td>
</tr>
<tr>
<td>5</td>
<td>225</td>
<td>225.46</td>
</tr>
</tbody>
</table>
CHAPTER SIX: CONCLUSIONS AND SCOPE FOR FUTURE WORK

6.1 CONCLUSIONS

This thesis has studied the usage of neural networks as an alternative method for the detection, classification and location of faults on transmission lines. The methods employed make use of the phase voltages and phase currents (scaled with respect to their pre-fault values) as inputs to the neural networks. Various possible kinds of faults namely single line-ground, line-line, double line-ground and three phase faults have been taken into consideration into this work and separate ANNs have been proposed for each of these faults.

All the neural networks investigated in this thesis belong to the back-propagation neural network architecture. A fault location scheme for the transmission line system, right from the detection of faults on the line to the fault location stage has been devised successfully by using artificial neural networks.

The simulation results obtained prove that satisfactory performance has been achieved by all of the proposed neural networks in general. As further illustrated, depending on the application of the neural network and the size of the training data set, the size of the ANN (the number of hidden layers and number of neurons per hidden layer) keeps varying. The importance of choosing the most appropriate ANN configuration, in order to get the best performance from the network, has been stressed upon in this work. The sampling frequency adopted for sampling the voltage and current waveforms in this thesis is just 720 Hz which is very low compared to what has been used in the literature (a major portion of the works in literature utilized 2 kHz – 5 kHz).
This is of significant importance because, the lower the sampling frequency, the lesser the computational burden on the industrial PC that uses the neural networks. This means a lot of energy savings because a continuous online detection scheme of this kind consumes a large amount of energy, a major portion of which is due to the continuous sampling of waveforms. The above mentioned are some significant improvements that this thesis offers over existing neural network based techniques for transmission line fault location.

To simulate the entire power transmission line model and to obtain the training data set, MATLAB R2010a has been used along with the SimPowerSystems toolbox in Simulink. In order to train and analyze the performance of the neural networks, the Artificial Neural Networks Toolbox has been used extensively.

Some important conclusions that can be drawn from this thesis are:

- Neural Networks are indeed a reliable and attractive scheme for an ideal transmission line fault location scheme especially in view of the increasing complexity of the modern power transmission systems.
- It is very essential to investigate and analyze the advantages of a particular neural network structure and learning algorithm before choosing it for an application because there should be a trade-off between the training characteristics and the performance factors of any neural network.
- Back Propagation neural networks are very efficient when a sufficiently large training data set is available and hence Back Propagation networks have been
chosen for all the three steps in the fault location process namely fault detection, classification and fault location.

As a possible extension to this work, it would be quite useful to analyze all the possible neural network architectures and to provide a comparative analysis on each of the architectures and their performance characteristics. The possible neural network architectures that can be analyzed apart from back propagation neural networks are radial basis neural network (RBF) and support vector machines (SVM) networks.
BIBLIOGRAPHY


44. Howard Demuth, Mark Beale, Martin Hagan, The MathWorks user’s guide for MATLAB and Simulink, Neural Networks Toolbox 6.
VITA

Suhaas Bhargava Ayyagari was born on July 2nd, 1989 in Andhra Pradesh, India. He received his undergraduate degree in the field of Electronics and Communication Engineering from Amrita Vishwa Vidyapeetham, Coimbatore, India in June 2006. In Fall 2010, he enrolled as a MSEE student in the Department of Electrical and Computer Engineering at the University of Kentucky. He is currently with the Physical Modeling Team at The MathWorks in Boston, MA.