TO STUDY INDUCTION MOTOR EXTERNAL FAULTS DETECTION AND CLASSIFICATION USING ANN AND FUZZY SOFT COMPUTING TECHNIQUES

A Thesis submitted to Gujarat Technological University

for the Award of

Doctor of Philosophy

in

Electrical Engineering

by

Kalpesh Jayantilal Chudasama

Enrollment No. 119997109001

GUJARAT TECHNOLOGICAL UNIVERSITY
AHMEDABAD

October - 2016

TO STUDY INDUCTION MOTOR EXTERNAL FAULTS DETECTION AND CLASSIFICATION USING ANN AND FUZZY SOFT COMPUTING TECHNIQUES

A Thesis submitted to Gujarat Technological University

for the Award of

Doctor of Philosophy

in

Electrical Engineering

by

Kalpesh Jayantilal Chudasama

119997109001

under supervision of

Dr. Vipul A. Shah

GUJARAT TECHNOLOGICAL UNIVERSITY
AHMEDABAD

October – 2016

DECLARATION

I declare that the thesis entitled To Study Induction Motor External Faults Detection and Classification Using ANN and Fuzzy Soft Computing Techniques submitted by me for the degree of Doctor of Philosophy is the record of research work carried out by me during the period from September 2011 to October 2016 under the supervision of Dr. Vipul Shah and this has not formed the basis for the award of any degree, diploma, associateship, fellowship, titles in this or any other University or other institution of higher learning.

I further declare that the material obtained from other sources has been duly acknowledged in the thesis. I shall be solely responsible for any plagiarism or other irregularities, if noticed in the thesis.

Signature of the Research Scholar: …………………………… Date:…………………

Name of Research Scholar: Kalpesh Jayantilal Chudasama

Place: Ahmedabad

CERTIFICATE

I certify that the work incorporated in the thesis To Study Induction Motor External Faults Detection and Classification Using ANN and Fuzzy Soft Computing Techniques submitted by Shri Kalpesh Jayantilal Chudasama was carried out by the candidate under my supervision/guidance. To the best of my knowledge: (i) the candidate has not submitted the same research work to any other institution for any degree/diploma, Associateship, Fellowship or other similar titles (ii) the thesis submitted is a record of original research work done by the Research Scholar during the period of study under my supervision, and (iii) the thesis represents independent research work on the part of the Research Scholar.

Signature of Supervisor: ……………………………… Date: ………………

Name of Supervisor: Dr. Vipul A Shah

Place: Ahmedabad

Originality Report Certificate

It is certified that PhD Thesis titled To Study Induction Motor External Faults Detection and Classification Using ANN and Fuzzy Soft Computing Techniques by Shri Kalpesh Jayantilal Chudasama has been examined by us. We undertake the following:

a. Thesis has significant new work / knowledge as compared already published or are under consideration to be published elsewhere. No sentence, equation, diagram, table, paragraph or section has been copied verbatim from previous work unless it is placed under quotation marks and duly referenced.
b. The work presented is original and own work of the author (i.e. there is no plagiarism). No ideas, processes, results or words of others have been presented as Author own work.
c. There is no fabrication of data or results which have been compiled / analysed.
d. There is no falsification by manipulating research materials, equipment or processes, or changing or omitting data or results such that the research is not accurately represented in the research record.
e. The thesis has been checked using **Turnitin** (copy of originality report attached) and found within limits as per GTU Plagiarism Policy and instructions issued from time to time (i.e. permitted similarity index <=25%).

Signature of the Research Scholar: …………………………… Date: …………

Name of Research Scholar: Kalpesh Jayantilal Chudasama

Place: Ahmedabad

Signature of Supervisor: ……………………………… Date: ………………

Name of Supervisor: Dr. Vipul A. Shah

Place: Ahmedabad

Kalpesh Thesis

ORIGINALITY REPORT

12%	10%	1%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	inpressco.com Internet Source	4%
2	www.ijert.org Internet Source	3%
3	Submitted to Institute of Technology, Nirma University Student Paper	2%
4	chius.homestead.com Internet Source	1%
5	lib.tkk.fi Internet Source	1%
6	waset.org Internet Source	1%
7	www.ro.feri.uni-mb.si Internet Source	1%

EXCLUDE QUOTES ON EXCLUDE MATCHES < 1%

EXCLUDE BIBLIOGRAPHY	ON

PhD THESIS Non-Exclusive License to GUJARAT TECHNOLOGICAL UNIVERSITY

In consideration of being a PhD Research Scholar at GTU and in the interests of the facilitation of research at GTU and elsewhere, I, Kalpesh Jayantilal Chudasama having Enrollment No. 119997109001 hereby grant a non-exclusive, royalty free and perpetual license to GTU on the following terms:

a) GTU is permitted to archive, reproduce and distribute my thesis, in whole or in part, and/or my abstract, in whole or in part (referred to collectively as the “Work”) anywhere in the world, for non-commercial purposes, in all forms of media;

b) GTU is permitted to authorize, sub-lease, sub-contract or procure any of the acts mentioned in paragraph (a);

c) GTU is authorized to submit the Work at any National / International Library, under the authority of their “Thesis Non-Exclusive License”;

d) The Universal Copyright Notice (©) shall appear on all copies made under the authority of this license;

e) I undertake to submit my thesis, through my University, to any Library and Archives. Any abstract submitted with the thesis will be considered to form part of the thesis.

f) I represent that my thesis is my original work, does not infringe any rights of others, including privacy rights, and that I have the right to make the grant conferred by this non-exclusive license.

g) If third party copyrighted material was included in my thesis for which, under the terms of the Copyright Act, written permission from the copyright owners is required, I have obtained such permission from the copyright owners to do the acts mentioned in paragraph (a) above for the full term of copyright protection.

h) I retain copyright ownership and moral rights in my thesis, and may deal with the copyright in my thesis, in any way consistent with rights granted by me to my University in this non-exclusive license.

i) I further promise to inform any person to whom I may hereafter assign or license my copyright in my thesis of the rights granted by me to my University in this non-exclusive license.

j) I am aware of and agree to accept the conditions and regulations of PhD including all policy matters related to authorship and plagiarism.

Signature of the Research Scholar:____________________________

Name of Research Scholar: Kalpesh Jayantilal Chudasama

Date: Place: Ahmedabad

Signature of Supervisor: _______________________________

Name of Supervisor: Dr. Vipul A. Shah

Date: Place: Ahmedabad

Seal:

Thesis Approval Form

The viva-voce of the PhD Thesis submitted by Shri Kalpesh Jayantilal Chudasama (Enrollment No. 119997109001) entitled To Study Induction Motor External Faults Detection and Classification Using ANN and Fuzzy Soft Computing Techniques was conducted on ………………….…………. (day and date) at Gujarat Technological University.

(Please tick any one of the following option)

- ☐ We recommend that he/she be awarded the PhD degree.
- ☐ We recommend that the viva-voce be re-conducted after incorporating the following suggestions.

(briefly specify the modifications suggested by the panel)

- ☐ The performance of the candidate was unsatisfactory. We recommend that he/she should not be awarded the PhD degree.

(The panel must give justifications for rejecting the research work)

--	--
Name and Signature of Supervisor with Seal	1) (External Examiner 1) Name and Signature
--	--
2) (External Examiner 2) Name and Signature	3) (External Examiner 3) Name and Signature

Abstract

Induction motors are widely used as electrical load in all kind of industries. Induction motors are appears to various abnormalities/faults during their operation. Accurate fault identification is the prime industrial need to reduce breakdown maintenance and revenue losses. Induction motors appears to different kind of faults or abnormalities which can majorly categorized in two parts external (e.g., phase failure, unbalance supply, stalling, overvoltage, undervoltage overload and reverse phase sequence) and internal (stator interterm, rotor bar, eccentricity and bearing failure) faults. There are invasive and noninvasive methods for fault detection. The noninvasive methods are more preferable because they are based on easily accessible and economical to diagnose the machine conditions without disintegrating the machine.

Low cost thermal protection devices like eutectic alloy or bi-metal type overload relays, Electromagnetic relays and static relays are not suitable for multiclass faults identification. In conventional protection, relays applied for one hazard may operate for others as some overlap found particularly in overload versus faults like unbalance voltages/currents, single phasing etc. It is more difficult to estimate negative sequence current if loss of phase occurs while running. The fault data sets scatter plot (shown in Fig. 3.9 and Fig. 5.7) found complex and linearly nonseparable. The relay logic used to identify these faults requires sophisticated techniques for accurate, generalized and reliable operation.

The current interest of academia for the multiple fault diagnosis in induction machines is using soft computing techniques mainly artificial neural network (ANN) and fuzzy logic. The guiding principle of soft computing techniques is exploiting the tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, and low cost solution. ANN and clustering based fuzzy logic are suitable and well proven for complex and linearly non separable fault identification task. ANN or fuzzy can be used as relaying logic for multiple fault identification. Neural network provide a natural framework for fault identification and it can approximate abnormal behaviour of dynamial systems through learning approach. Fuzzy logic can be used to provide a general heuristic solution to a particular problem. It can provide a heuristic output as a result of some complex computations by quantifying the actual numerical data into heuristic and linguistic terms.

The main objective of thesis is to identify external faults using ANN and fuzzy logic approaches.

In first phase, external faults like overload (OL), overvoltage (OV), undervoltage (UV), single phasing of any phase (SP) and voltage unbalances (VUB) were created alongwith normal output (N) conditions for different operating voltage and load conditions in MATLAB/Simulink environment. Data sets of three phase RMS voltages and RMS currents were obtained as input training and testing feature vector. The classification error obtained with well-known statistical linear discriminant analysis (LDA) classifier for independent test inputs is high (26.1%). Multilayer perceptron neural network (MLPNN) with Levenberg-Marquardt (LM) algorithm is used for external faults identification on simulation data sets. Different single hidden layer MLPNN configurations were offline train and tested with trial and error method. They are tested with increasing hidden neurons for finding best generalized ANN configuration to detect external faults with highest stastical parameter like validation subset classification accuracy. Classification accuracy (or classification error) is natural performance measure for classification problems. Train subset classification accuracy and validation error is also considered to find best generalized well trained MLPNN configuration. It is observed through simulation that ANN can detect all kind of faults occurs 1 cycle before its refreshing rate.

In second phase, multilayer perceptron neural network (MLPNN), subtractive clustering based Sugeno fuzzy inference system (SC_FIS), probabilistic neural network (PNN) and also hybrid adaptive neurofuzzy inference approach were used for the external faults identification in MATLAB environment for experimentally obtained data sets. Experiments were performed to obtain input vector and evaluate the performance of MLPNN, SC_FIS, PNN, and adaptive neurofuzzy inference system (ANFIS) on real time data sets. Subtractive clustering based fuzzy inference systems (SC_FIS & ANFIS) are used to obtain rules besides good classification accuracy alongwith neural networks in this work. To simulate the real field operation of induction motor data sets were logged with different loading and at various supply voltage conditions. Induction motor coupled DC generator laboratory type setup was used for the practical data sets. Normal conditions and five external fault conditions OL, OV, UV, SP and VUB were experimentally simulated to obtain the input feature vector. Three phase voltages and currents were sampled and logged. Representative samples are used for training and testing inputs. Best generalized

MLPNN configuration is obtained for practical data sets same as discussed for simulation data sets. SC_FIS is used for external faults identification with practical data sets. FISs are generated using different cluster radius (0.1 to 0.9). FISs are train and tested with 10-times random subsampling train and test data sets. Total average test classification accuracy and also test RMSE error are used to find best generalized FIS configuration.

PNN is used with training and validation (subset of total training data sets) data sets and classification accuracy against different radial basis function spread is obtained to find spread for which PNN gives most generalized results. The best generalized ANFIS configuration is obtained by comparing test classification accuracy of ANFISs obtained with different cluster radius. Independent test data sets used as checking data for ANFIS.

Conventional statistical LDA and simple probabilistic Naïve Bayes classifier (NBC) are also used for fault identification performance comparison in terms of statistical measurers with soft computing classifiers using total train and independent test practical data sets. Soft computing classifiers performance found excellent from classifiers performance comparison with respect to classification accuracies (Table 7.1) and with respect to other statistical measures sensitivity, specificity, precision and F-measure for train and test data sets (Table 7.14 and 7.15 respectively). MLPNN, PNN, SC_FIS and ANFIS show impressive results for train data sets classification accuracy and other statistical measurers. MLPNN and SC_FIS generalization performance found better based on independent test data sets classification accuracies (98.61% and 97.2% respectively) and other statistical measures for all six output conditions. MLPNN outperforms for external faults identification in all statistical measures. SC_FIS and neurofuzzy have advantage of obtaining rules for external faults besides good statistical parameters results.

Acknowledgment

I would like to express my sincere gratitude to my supervisor Dr. Vipul Shah. He has been wise and trusted throughout the entire process. His trust of my abilities has given me invaluable opportunities to improve my research and communication skills, which helped me in my study and future career. I am extremely thankful to my DPC members Dr. Ramji Makwana and Dr. Devang Bhat for their valuable suggestions and all help.

I would like to specially thanks Dr. Shishir Shah, Associate Professor, University of Houston for his valuable inputs and suggestions.

I am very thankful to my Organization, Institute and Department for their kind support. I am thankful to GTU V.C., Registrar, Controller of Examination and Ph.D. section for their kind support. I extend my sincere gratitude to all those people who helped me in all their capacity to complete this work.

I am very much thankful to my young son Jalpan and my wife Hiral for their consistent support, motivation and patience throughout out my research work. I am very much thankful to my newly born daughter Peehu for her smiles. I am thankful to my elder brothers and their family members for their support and help as and when needed.

I am thankful to my parents, to whom this dissertation is dedicated to, have been always constant source of inspiration and strength for me.

Above all, I am very much thankful to almighty GOD for giving me this beautiful life and make me able to reach this stage of life.

K. J. Chudasama

Table of Contents

List of Abbreviations

Abbreviations	Original Phrase
ANN	Artificial neural network
FIS	Fuzzy inference system
MLPNN	Multilayer perceptron neural network
SC_FIS	Subtractive clustering based Sugeno fuzzy inference system
PNN	Probabilistic neural network
ANFIS	Adaptive neuro-fuzzy inference system
LDA	Linear discriminant analysis
NBC	Naïve Bayes classifier
RMS	Root means square
RMSE	Root means square error
GD	Gradient decent
TP	True positives
TN	True negatives
FP	False positives
FN	False negatives
HP	Horse power
AI	Artificial intelligence
SVM	Support vector machines
NEMA	National electrical motor association
LT	Low-Tension
HT	High-Tension
LV	Low-Voltage
MV	Medium-Voltage
HV	High-Voltage

NN	Neural network
FFNN	Feedforward neural network
OL	Overload
OV	Overvoltage
UV	Undervoltage
SP	Single phasing
VUB	Voltage unbalances
N	Normal output
LM	Levenberg-Marquardt
BP	Backpropagation
CT	Current transformer
VT	Voltage transformer

List of Figures

List of Tables

List of Appendices

CHAPTER - 1

Introduction

1.1 Overview

Induction motors are most widely used industrial load and consumes a major part of overall electrical consumption. Fault identification in electrical machines and power systems is increasing interest research area for academicians as well as for industry. The wide variety of environments and conditions motor exposed to, misoperations and manufacturing defects can make it subject to incipient faults or gradual deterioration and can lead to motor failure if left undetected. Most electric motor failures interrupt process, reduce production and may damage related machinery. Sometimes a small HP motor failure can also create hours of plant stoppage in continuous processing industries. Reliable and healthy operation of induction motors is the major need of industries. There many ways used by the industry to tackle the problem like preventive and corrective maintenance, keeping spare motors, protective system etc. In some industries very expensive scheduled maintenance performed in order to prevent sudden motor failures. Therefore there is considerable demand to reduce maintenance cost and prevent unscheduled downtime for electric motors and drive systems. Early fault detection or correct fault detection and classification allows scheduling maintenance which reduces the maintenance efforts by reducing failure and downtime and improves the overall availability of motor driven system. It increases the revenue by reducing failures.

Induction motors appears to different kind of faults or abnormalities which can majorly divided in two parts, external and internal faults. Overload (OL), undervoltage (UV), overvoltage (OV), Single phasing (SP), voltage unbalances (VUB), locked rotor, earthfault between supply feeder and motor terminals and three phase fault at the terminals are categorized as external faults and short circuit, Stator interturn failure, bearing, rotor faults, eccentricity as internal faults. Internal as well as external faults accurate diagnosis is equally necessary and which can be lead to development of comprehensive protective scheme for all faults. There are invasive and noninvasive methods for fault detection. The noninvasive methods are more preferable because they are based on easily accessible and inexpensive measurements to diagnose the machine conditions without disintegrating the machine [1] [2]. These schemes are suitable for on-line monitoring and fault detection purposes [2]. We have worked on most probable five external faults OL, UV, OV, SP and VUB identification.

Fuses and bi-metallic overload relays used for Low-Voltage (LV) induction motors protection are roughly emulate the induction motor thermal limit curve. Overload relays for Medium-Voltage (MV) induction motors utilize simple thermal models and embedded temperature sensors to monitor the winding temperature. These techniques first calculate the losses in a motor, and then estimate the stator winding temperature based on motor's thermal model. However, the main drawback of these thermal model-based approaches is that the thermal parameters are not constant and measurements must be made for each motor under different operating conditions. Embedded temperature sensors, on other hand, may result in false alarms or trips because of disintegration of the connections, noise interference and their large time constant. [3] [4].

Electromechanical relays and static relays are not suitable for multiclass faults identification. Induction motor fault diagnosis is likely a highly complex nonlinear mapping problem as both the inputs and outputs are multiple variables without clear linear relationships. The scatter plot matrix for output classes (five considered external faults and normal condition) with respect to input (3 phase RMS voltages and currents) variable is linearly non separable and complex. In conventional protective schemes relays applied for one hazard may operate for others as some overlap found particularly in overload versus faults, unbalance voltages/currents and single phasing etc. In conventional microprocessor based relays negative sequence protection used for cases like voltage unbalance or single phasing it is difficult to estimate negative sequence current and even more difficult if loss of phase occurs while running [5].

The history of fault monitoring and fault isolation started with the use of electromechanical relays to protect the motor against faults [6]. However these relays are slow in operation, consume significant power and require periodic maintenance due to mechanical parts involved. The traditional relays based on electromechanical and solid state relays are mostly replaced by microprocessor based relays. Modern numerical relays run on background software. Nowadays fast and sophisticated microprocessors, microcontrollers, and digital signals processors availability to compact, faster, more accurate, flexible and reliable protective relays, make them cost effective as compared to the conventional ones. In numeric relays, the analog current and voltage signals monitored through current transformers (CTs) and/or voltage transformers (VTs) are conditioned, sampled at specified instants of time and converted to digital form for numerical manipulation, display and recording. The outputs of the analog pre-processor digitalized using A/D converters. The analog preprocessing and analog interface constitute the data acquisition system. Numerical relays process the data numerically using a relaying algorithm to calculate the fault discriminate and make trip decisions [7].

The current development of computer software based on intelligent systems components leads attention of relay engineers to use them in the diagnosis of faults in power system components such as induction motors. Recently soft computing techniques are used in this relay logic block [8].

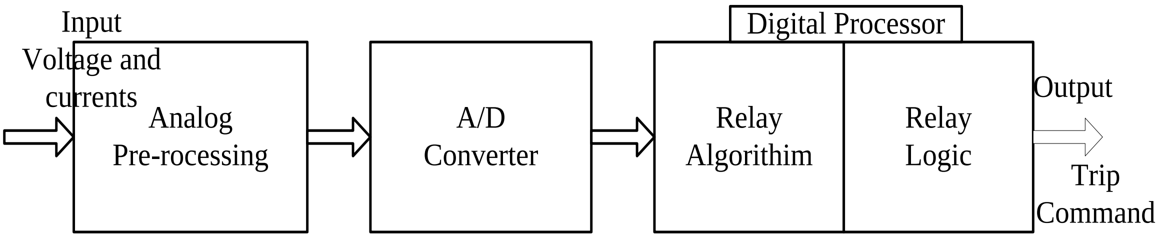


FIGURE 1.1

Block Diagram of Digital Relay Scheme [8]

Artificial Neural networks (ANNs) and fuzzy logic systems are parameterised computational nonlinear algorithms for numerical processing of data. The acquired knowledge is stored in internal parameters. Neural and fuzzy though different technologies can be used to accomplish the specification of mathematical relationships among numerous variables in complex problem, performing with some degree of impression [9]. Others include recently emerged different technologies like thermal measurements, chemical analysis, and axial flux. Intelligent numerical relays using artificial intelligence soft computing techniques such as ANNs and fuzzy logic systems are presently under active research and development stage. This work presents mainly ANN and Fuzzy Logic based three phase induction motor external faults identification.

Automated fault diagnostics and condition monitoring are important parts of most of the world's industrial processes. It is difficult to develop an analytical model that adequately describes induction motor performance in its all operation points with any power source in case of induction motor fault identification. If the expert knowledge of process is available a simple signal-based diagnostics can be adopted with knowledge-based models. It is difficult for a human expert to distinguish fault from the normal operation. Multiple information sources may need for accurate decision. Thus, the data-based models are the most interesting approach for the induction motor diagnostics [10]. In this presented work, the fault identification system is built using RMS features retrieved from the voltage and current signals and decision making part relies on data-based (pattern) classification model.

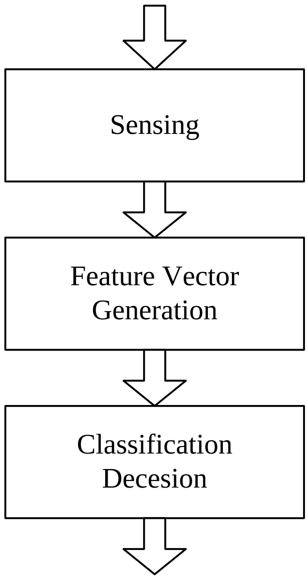


FIGURE 1.2

Basic Pattern Recognition System

Classification establishes (or tries to find) the structure in data, whereas pattern recognition attempts to take new data and assign them to one of the classes defined in the classification process [11]. Pattern recognition can be defined as a process of identifying structure in data by comparisons to known structure; the known structure is developed through methods of classification [12]. Classification and pattern recognition are general names for data-based algorithms that classify things based on multiple numerical measurements i.e. features. The classifiers can be trained to represent direct relationship between measurement data of the system and certain fault conditions. During the last years ANN based models like Multilayer Perceptron Neural Network (MLPNN) and Radial basis function neural networks have been a popular research subject and also their application in data based model widely studied. With ANN models it is possible to estimate a nonlinear function without requiring a mathematical description of how the output functionally depends on the input, neural networks (NNs) learn from examples. The most commonly mentioned advantages of ANN are their ability to model any nonlinear system, the ability to learn, highly parallel structure and the ability to deal with inconsistent data. Application of a NN in the decision making part of the fault diagnostics system is also called NN based fault classification or pattern recognition [13]. Developments in diagnosis systems have led to the consideration of radically different diagnosis strategies by making extensive use of artificial intelligence (AI) techniques. They have numerous advantages over conventional diagnosis techniques. They could give improved performance of fault identification if properly tuned.

ANN found good potential in different applications like fault diagnosis, pattern recognition, forecasting, systems dynamic modeling, robotic control etc. Neural Networks are robust to input and system noise, have learning capabilities and can perform in real time. A large number of input variables can be simultaneously fed to a multi-input neural network. An ANN with its excellent pattern recognition capabilities can be effectively used for faults identification of an induction motor. ANN design does not require a complete mathematical model of the induction motor. Moreover once designed the internal structure of ANN can be easily changed, if modifications or additions need to be made.

In presented work, three-layer MLPNNs and Probabilistic neural networks (PNNs) are evaluated for external fault diagnosis of induction motor. ANN with single hidden layer is capable of approximate any function regardless of its complexity. Presently there is no satisfactory method to define how many neurons should be used in hidden layers and usually found by trial and error method. Large number of neurons and layers may cause overfitting and may cause decrease the generalization ability. Many researchers suggested ANN in classification but less have discussed about generalized ANN. There must be trade-off between learning and generalization in classification. Standard Gradient Decent (GD) algorithm is too slow for practical problems. Levenberg-Marquardt (LM) is algorithm is the blend of GD and Newton technique. It is much faster and efficient than gradient descent [14]. Present study has attempted to find best generalized and well trained

MLPNN configuration using LM backpropagation (BP) optimization for the induction motor external faults detection and classification. PNN belongs to family of radial basis function NN which due to their robustness widely used in pattern classification problems. This study has also used PNN for external faults identification and evaluates best generalized PNN configuration.

The interpretation of the fault conditions is a fuzzy concept using rigorous mathematical formulations in parameter estimation approach is generally impractical and inaccurate. Engineers prefer the accurate fault detection as well as the heuristic knowledge behind the faults diagnosis. We have used subtractive clustering based Sugeno fuzzy inference system (SC_FIS) for the external fault identification. It is not feasible to make direct rules using expert knowledge or by observing data sets for induction motor external faults identification for varying load and supply voltage. Clustering is a very effective technique to identify natural groupings in data and in this case allows to group fault patterns into broad categories. Subtractive clustering which does the clustering without prior information about the number of clusters and initial guess of the cluster centers. In subtractive clustering based FIS, combination of subtractive cluster estimation method and a linear least-squares estimation procedure provides a fast and robust algorithm for identifying fuzzy models from numerical data without involving any nonlinear optimization [15]. Performance of adaptive neuro-fuzzy system (ANFIS) is also evaluated for external faults identification. In literature survey related with application of clustering based Sugeno fuzzy systems in fault diagnosis or classification tasks, it is observed that the cluster radius of subtractive clustering is assumed and the FIS is used for diagnosis or classification. Present study attempted to find best generalized cluster radius Sugeno fuzzy and adaptive neurofuzzy configuration for external faults identification of induction motor for practical data sets.

In addition, the external faults identification performance of neural nets and subtractive clustering based FIS approach is also compared with widely used probabilistic Naïve Bayes Classifier (NBC) and well-known Linear Discriminant Analysis (LDA) classifier with respect to statistical parameters like total classification accuracy, sensitivity, specificity, precession and F-measure to find its effectiveness for the problem.

1.2 Definition of the Problem

This thesis uses soft commuting techniques ANN and fuzzy logic for the identification of external faults generally experienced by induction motors alongwith normal conditions. This thesis evaluates the potential of ANN and fuzzy logic as relaying logic in induction motor external faults diagnosis. The ANN and fuzzy based methods used are tested with real time data sets obtained using 3 HP three phase induction motor. This study also compares the performance of ANN based and fuzzy based classifiers, alognwith probabilistic NBC and well-known statistical LDA classifier using statistical measures like classification accuracy, sensitivity, specificity, precision and overall F-measure.

1.3 Objectives and Scope of the Study

- To study induction motor external faults (OL, OV, UV, SP and VUB) and soft computing techniques like MLPNN, Subtractive clustering based Sugeno fuzzy inference systems (SC_FIS and ANFIS) and PNN.
- Statistical measures study for multiclass classification performance comparison that includes classification accuracy, sensitivity, specificity, precision and F-measure.
- To develop induction motor external faults simulation in Matlab/Simulink environment. Plotting and visualization of data sets obtained through simulation. External faults classification using LDA and MLPNN based on simulation data sets.
- To obtain real time data sets of three phase RMS current and RMS voltage values using experimentation and external faults identification using MLPNN, PNN and Sugeno FISs.
- To obtain best generalized configuration for MLPNN, PNN and Sugeno FISs.
- To compare the performance of ANN and fuzzy classification techniques including conventional well known stastical LDA and Naive bayes with respect to different stastical measures using practically obtained data sets.

1.4 Significance of the Study

Induction motors are the most common electrical machines, because of relatively low manufacturing cost and the ease of control. As indicated before, accurate identification of faults and protecting them is an important aspect to reduce financial losses. Currently AI based fault diagnosis is widely studied and proposed by researchers for power systems and its components. This study evaluates the potential of ANN and fuzzy based techniques as relaying logic for induction motor external faults generalized and accurate identification.

This study applied MLPNN for the accurate fault identification with fast LM BP algorithm using early stopping for generalization. Plotting of complex, overlapping, linearly non separable data sets is done class wise. Subtractive clustering based Sugeno fuzzy inference system is used for external faults identification and rules responsible for the five external faults and normal conditions are obtained. This study attempted to find best generalized well trained neural network (MLPNN and PNN) and fuzzy logic (Sugeno based FISs) configuration for external faults detection and classification. This study also attempted conventional LDA and probabilistic NBC for external faults identification alongwith neural network and fuzzy classifiers for statistical performance comparison.

1.5 Outline of Thesis

Chapter 2 presents literature survey mainly for induction motor faults identification studies using AI and some other approaches. Chapter 3 summarizes the different external faults appears to induction motor and discusses external faults simulation. OL, OV, UV, SP and VUB external faults are simulated alongwith normal conditions at different operating voltages and load torque in MATLAB/SIMULINK. Scatter plots are used to visualize the linearly non separable and complex data sets. Chapter 4 discusses the neural network structure, advantages, generalization and LM algorithm. Different single layer MLPNN configurations are tested using growing hidden neuron phenomena and best generalized well trained configuration is found for external faults identification. Experimental setup details used for obtaining practical data sets for external faults are presented in chapter 5. MLPNN, SC_FIS, PNN, ANFIS, NBC and LDA are used for induction motor external faults identification using experimentally obtained real-time data sets and result are discussed in chapter 6. The classifiers are compared with total train and independent test classification accuracy for practical data sets in chapter 7. They are also compared using other stastical parameters like sensitivity, specificity, precession, F-measure alongwith for independent test and total train data sets in chapter 7. Finally, chapter 8 concludes the thesis.

References

1. Aroui T, Koubaa Y, Toumi A (2008) 'Magnetic Coupled Circuit Modelling of Induction machines oriented to Diagnostics', Leonardo Journal of sciences, Vol. 7, issue 13, pp.103-121, ISSN: 1583-0233.

2. Chow MY, Sharpe RN, Hung JC (1993) 'On the Application and Design of Artificial Neural Networks for motor fault detection – Part I', IEEE Transactions on Industrial Electronics, Vol.40, No2, pp.181-188, ISSN: 0278-0046.

3. Zhang P, Du Y, Habetler TG, Lu B (2009) 'A survey of condition monitoring and protection methods for medium voltage induction motors', *Proceedings of the Energy conversion congress and exposition, IEEE*, pp. 3165-3174.

4. Du Y, Habetler TG, Harley RG (2007) 'Methods for thermal protection of medium voltage induction motors – A review', *Proceedings of the 2008 international conference on condition monitoring and diagnosis*, China, April 21-24.

5. Bamber M et. al. (2011) AC motor protection. In: (May 2011) Network Protection and Automation Guide, pp. 337-351. Available: www.fecime.org/referencias/npag/chap19-336-351.pdf. [Accessed 2 March 2015]

6. Downs CL (2004) Motor protection. In: Elmore WA (2nd Edition) Protective relaying theory and applications. Marcel Dekker, Inc., New York, pp. 153-155.

7. Venkateshmurthy BS, Venkatesh V (2013) 'Advanced numerical relay incorporating the latest features which can compute the interfacing with automation using DSP', Journal of Engineering Computers and Applied sciences (JEC & AS) Vol. 2, No. 1, pp. 4-7, ISSN: 2319-5606.

8. Hammo R (2014) Faults identification in three phase induction motors using support vector machines. Master of Technology Management Plan II Graduate Projects, Paper1, Bowling Green State University.
Available: http://scholarworks.bgsu.edu/ms_tech_mngmt/1. [Accessed 31 October 2014]

9. Rajasekaran S, Vijayalakshmi Pai GA (2007) Neural Networks, Fuzzy logic, and Genetic Algorithms Synthesis and Applications. Prentice Hall of India, New Delhi.

10. Sanna P (2004) Support Vector Machine Based Classification in Conditioning Monitoring of induction motors. Abstract Doctoral Dissertation. Helsinki University of Technology. Available: http://lib.tkk.fi/Diss/2004/isbn9512271559/isbn9512271559.pdf

11. Ross TJ (2009) *Fuzzy logic with engineering applications.* 2nd edition, Wiley, India.
12. Bezdek J (1981) *Pattern Recognition with fuzzy objective function algorithms*. Springer Science and Business Media, New York.
13. Nejjari H , Benbouzid MEH (2000) 'Monitoring and Diagnosis of Induction Motors Electrical Faults using a current park's vector pattern learning approach', IEEE Transactions on Power Electronics, Vol. 36, No. 3, pp. 730-735, ISSN: 0278-0046.
14.Hagan MT, Menhaj MB (1994) 'Training feedforward networks with Marquardt algorithm, IEEE Transactions on neural networks, Vol. 5, No. 6, ISSN: 1045-9227.
15. Chiu SL (1994) 'Fuzzy model identification based on cluster estimation', Journal of intelligent Fuzzy Systems, Vol. 2, pp. 267-278, ISSN: 1064-1246.

CHAPTER - 2

Literature Review

2.1 Overview

This review mainly covers some topics like induction motor external and internal faults identification, AI based fault identification, ANN generalization, NBC and LDA based fault identification, induction motor modelling and comparison of classifiers performance.

All above topics are broadly classified and considered in three categories (1) ANN based approaches (2) Fuzzy logic based approaches (3) Hybrid and other approaches.

2.2 ANN Based Approaches

Accurate models of faulty machine and model based techniques are essentially required for achieving a good fault diagnosis, while soft computing approaches such as neural networks, fuzzy logic technique provide good analysis of a system even of in absence of accurate fault model [1].

As a likely result of on-going computer technology development massive parallel processing and soft computing will significantly enhance traditional computation methods and natural consequence of this growth is the emergence of field of intelligent systems. It is a practical alternative for solving mathematically intractable and complex problems. The main subdivisions of the area are ANN and Fuzzy systems. The soft computing systems have very distinct features while operated with specialised hardware. The mathematical power of machine intelligence is commonly attributed to the neural like system architecture used and the fault tolerance arising from the massively interconnected structure. Another aspect of soft computing systems is that they use fuzzy/continuous levels instead of zero and one digital levels and in this way much more information is passed through the system. The third feature is survivability in the presence of faults, means they work correctly if they are partially damaged [2].

The Noninvasive parameter estimation technique require mathematical model and elaborate understanding of system dynamics based on system parameters. The parameters are usually chosen to reflect the motor conditions and usually difficult to obtain [3]. On the other hand, ANN is also a noninvasive technique but unlike parameter estimation technique ANN can performs fault detection based on measurements and training without need of complex and rigorous mathematical models. ANN is proposed for fault identification and other power system applications [4] & is emerging technologies promising for future widespread industrial usage [5].

Power system and electrical machines problem regarding classification or the encoding of an unspecified non-linear function are well-suited for ANN. ANNs can be especially useful for problems that need quick results, such as those in real time applications. Induction motor detection and classification are essential for relaying decision for alarming or tripping. The implementation of pattern recognizer for power system diagnosis has provided great advances in the protection field. An artificial neural network can be used as a pattern classifier for induction motor protection relay operation. The RMS magnitudes of three phase voltages and current signals of the transmission line are measured using current and voltage transformers and filters. The waveforms are sampled and digitized using analog to digital converters. After data acquisition, the signal is fed to pattern recognizer. The ANN (or fuzzy) recognizer module then verifies for the fault and generate a signal for alarm or tripping if it exists. The ANN (or fuzzy) module can be trained offline or online and trip decision is depend upon how ANN (or fuzzy) module is trained [6].

As in [7] maintenance as well as down time expenses can be reduced by appropriate fault detection schemes and proper monitoring of the incipient faults. Many of the conventional methods used to determine these faults are either very expensive, off-line, require the need of an expert, or impractical for small machines. ANN have been proposed and have demonstrated the capability of solving the motor monitoring and fault detection problem using an inexpensive, reliable, and noninvasive procedure.

The neural network for induction motor fault diagnosis was first proposed by [8] for bearing and stator turn fault of single phase induction motor. Neural network was trained using controllable data sources (developed using a computer programme) for initial design and training of ANN. An overview of feedforward nets and the BP training algorithm and a general methodology for the design of feedforward artificial neural networks to perform motor fault detection was discussed in [3]. Feedfoward layered artificial neural network (ANN) structure and standard gradient decent BP algorithm was used for identification of external faults of induction motor by [4] [9] on simulation measurements. Simple statistical parameters such as the overall RMS value of a signal can give useful information; for example, the RMS value of the vibration velocity is a convenient measure of the overall vibration severity. The RMS value of the stator current provides a rough indication of the motor loading in similar way [10] [11] [4]. Three phase RMS currents and voltages of the induction motor were used as input feature vector in [4] and angular speed was also considered in [9]. PC based monitoring and external fault detection scheme for 3 phase induction motor using ANN was presented in [12]. Generalization of ANN, finding optimal configuration and statistical parameters for classification issues are not addressed in [4] [9]. We have evaluated the best generalized and well trained configuration for MLPNN for induction motor external faults identification using statistical parameters comparison. Early stopping is widely used generalization technique because it is simple to understand and implement and reported superior to regularization in many cases [13]

An algorithm was proposed and experimentally validated in [14] for discrimination of unbalance supply voltage and phase losses fault using neural network. Authors used ratio of third harmonic to fundamental fast Fourier transform magnitude components of line currents and voltages at different load levels in their approach. In [15], ANN was used for detecting voltage unbalance faults for traction motor irrespective of the load and fault percentages. Stator current Park’s vector approach was used by [16] as input to train and test ANN for classifying stator open phase and voltage unbalance case.

In [17] stator interturn, bearing and rotor bar fault protection scheme was presented using principle components extracted from three phase RMS currents as input error vector for ANN. MLPNN based relaying scheme was presented in [18] for classifying and locating faults in TSCS transmission lines using three phase voltage and currents as input to MLPNN using EMTP-ATP simulation. In [19], two kinds of neural networks MLPNN with BP algorithm and self-organizing map (SOM) were compared for internal faults diagnosis using current and vibration signals. They found ANN diagnoses faults accurately with variable load and speeds.

Different kernel functions with different scaling range were applied to train SVM in [20] for identifying external faults in induction motor. The choice of suitable kernel for the given application, speed, size, and optimal design for multiclass classification are the challenges that limits the use of SVM for multiclass classification.

Dr. D.F. Specht in [21] proposed PNN which is feedforward neural network developed from radial basis function model. PNN is widely used in classification and pattern recognition problems. PNN follows an approach of Bayesian classifiers and use Parzen estimators which were developed to construct probability density functions [22] and has the distinct features from other networks in learning like no learning process are required, no need of initial weight setting is required, and no relationship between learning process and recalling processes. It has advantages like training of PNN is much simpler and instantaneous that of MLPNN, can be used in real time, network begin to generalize once one pattern representing each category has been observed and improve with additional patterns, decision surface can be made simple or complex by choosing appropriate smoothing parameter, and erroneous samples are tolerated [21].

Radial basis function was utilized to train ANN for the detection of bearing and stator interturn fault in [23]. Instantaneous current and angular velocity depending upon rotor speed were utilized in their approach. In [24], PNN was used to recognize power disturbance produced with Labview software using discrete wavelet transform (DWT). A performance comparative study is presented of three types of neural networks MLPNN, RBF and PNN for induction motor bearing fault detection using GA for section of features [25]. Induction machine drive speed sensor and current sensor faults diagnosis is presented in [26] using PNN and KNN algorithm alongwith PCA for feature extraction using MATLAB simulation model.

Matlab/Simulink simulated faults and radial basis function neural network was used in [27] for detection and classification of mainly different voltage unbalance external fault conditions. Instantaneous values of three phase voltages and currents were used and authors demonstrated ANN classifies the faults correctly. The radial basis function spread is selected 1 for generalization. However the spread is not selected with any comparison of statistical parameters like classification accuracy for different spread of radial basis function neural network. We have evaluated the best spread for best generalized PNN configuration for external faults identification using test classification accuracy and three phase RMS voltages and currents as input patterns.

2.3 Fuzzy Logic Based Approaches

ANNs do not provide heuristic interpretation of the solution due to its numerically oriented structures. Engineers prefer the accurate fault detection as well as the heuristic knowledge behind the fault detection process. Fuzzy logic is a technology that can easily provide heuristic reasoning while being difficult to provide exact solutions [7].

Ref. [28] discussed about practical utilisation of the fuzzy logic based protection device scheme. It contains mainly three blocks. First is fuzzification where real input values are fuzzified. Second is fuzzy reasoning block where the fuzzy signals are processed and after comparison with fuzzy settings some fuzzy decision signal are generated. The third is defuzzification where the fuzzy outputs are converted to crisp numbers real output or decision signal. It is common that most of the relay decisions in protective device are of discrete type (0-1).The relay output generated is either for tripping/alarm of showing healthy state. There is no intermediate case. FL is multi-valued. It deals with degree of membership and degrees of truth.

Fuzzy logic allow for a gradual transition between true and false. Decision making with FL can be compared to classification of the object to one of two or more classes where the border between them is not sharp, but is specified with some fuzziness. An element in such cases is classified as belonging with some degree between zero and unity to given set. When the degree of membership is high the fuzzy final decision can be made.

Following important features make fuzzy a promising alternative for protection and control application: ability of processing uncertain, inaccurate or corrupt data, possibility of expressing non sharp relationships and rules in close to natural language way, easy interpretation of internal signal of fuzzy system, improvement of efficiency and selectivity in decision making because of fuzzy settings and decision characteristics [28].

Fuzzy logic (FL) Mamdani approach was used to diagnose stator open condition and stator voltage unbalance of induction motor by [29] using compositional rule of fuzzy inference and stator current amplitude as input linguistic variables. FL approach was used to diagnose stator turn fault and open phase condition by [30] and to diagnose overload, voltage unbalance, undervoltage and single phase to ground by [31] using induction motor mathematical modelling. Three phase RMS stator currents were used as input variables in [30] and in addition speed and leakage current were also considered as input variables in [31]. They constructed rules for faults detection based on the observation of data obtained through simulation.

FL was utilized in [32] for extracting heuristics underlying stator fault diagnosis using stator current concordia pattern based fuzzy decision system. FISs are widely used for processes simulation and control which can be designed from expert knowledge or data sets. But, FIS based on only expert knowledge may suffer from inaccuracy whereas the fuzzy system inferred from data sets for complex systems is more accurate [33].

The main role of the relaying principle is detecting and classifying the faults based on input samples. It is not feasible to make direct rules using expert knowledge or by observing data sets for induction motor external faults identification for varying load and supply voltage. Clustering is a very effective technique to identify natural groupings in data and in this case allows to group fault patterns into broad categories. In [34], k means, fuzzy k means and subtractive clustering techniques were used alongwith Mamdani and Sugeno FIS for standard IRIS flower identification and Mackey-Glass time series case. Authors concluded that subtractive clustering technique in conjunction with FIS methods in all sample tests showed a better performance than any other technique. Subtractive clustering which does the clustering without prior information about the number of clusters and initial guess of the cluster centers. Combination of subtractive cluster estimation method and a linear least-squares estimation procedure provides a fast and robust algorithm for identifying fuzzy models from numerical data without involving any nonlinear optimization [35].

2.4 Hybrid and Other Approaches

Methodology behind a novel hybrid neurofuzzy system which merges the neural network and fuzzy logic technologies to solve fault detection problems and training procedure for neurofuzzy fault detection system is discussed for single phase induction motor bearing and stator turn fault detection in [7]. In [36] trained neurofuzzy fault detector provide accurate fault detector performance; also provide the heuristic reasoning behind the fault detection process and the actual motor fault conditions through the fuzzy rules. Expert knowledge was extracted using neural fuzzy fault detection in [36] for bearing and stator turn fault. Stator winding faults and voltage unbalance faults are addressed in [37] and authors used adaptive neurofuzzy inference system (ANFIS) to accurately detect both conditions. MCSA widely used for internal faults diagnosis in literature. This technique depends upon locating faults according to the position of specific harmonic components in stator current spectrum like broken rotor bars, air-gap eccentricity, bearing fault, faults in stator windings, etc. Signal processing techniques are applied to the measured sensor signals in order to generate features or parameters (e.g. amplitudes of frequency components associated with faults) which are sensitive to the presence or absence of specific faults. In [38], two approaches based on discrete wavelet transform were utilized for the induction motor fault detection wherein first fault detection criteria is the comparison between threshold determined experientially during healthy condition of motor and DWT coefficients of fault currents using selected mother wavelet 'db3' at the sixth level of resolution were utilized and second based on comparison of modulus maxima of the DWT coefficients. Wavelet Packet Transform based protection system was developed in [39] coefficients of the Wavelet Packet Transform line currents compared experimentally decided threshold for detecting and diagnosing various disturbance single phasing, phase to earth and short circuit faults occurring in induction motor. Using high resolution spectral analysis of stator current spectrum through experiment the voltage unbalance and open phase external fault condition was identified in [40].

In [41], detection and diagnosis of rolling element bearing defects with different severity levels were discussed using vibration signals based on classification techniques. Authors used two stastical classifier LDA & QDA and two types of neural networks RBF and MLP for classification accuracy comparison. In [42], performance of Bayes net and Naïve Bayes classifier (NBC) were compared for helical gear box fault and Naïve Bayes based practical system was demonstrated more efficient with lesser features. NBC detection system is proposed in [43] to identify temporary short circuit occurrence in induction motor stator winding using wavelet decomposition of current signal and NBC performance also compared with LDA using confusion matrix. Protection scheme for incipient faults using microcontroller was demonstrated in [44].

Authors in [45] presented MATLAB/Simulink model technique for induction motor tests which also helpful for evaluate steady state characteristic of motors. Induction machine model have been simulated in [46] [47] [48] for its behaviour analysis in Matlab/Simulink using symmetrical I.M d-q model. As in [46] dynamic simulation of small induction motor based on mathematical modelling is one of the key steps in the validation of the design process of motor drive system. D-Q model provide guidelines dynamic for simulation of induction motor, which can also applied for some faults data generation. “Ref. [48]” shows the results of studies under acceleration, variable load and open phase using this model.

References

1. Dash R (2010) Fault diagnosis in induction motor using soft computing techniques. M Tech by Research thesis. NIT Rourkela. Available: http://ethesis.nitrkl.ac.in/2809/ . [Accessed 20 February 2014]

2. Wilamowski, BM (2003) ‘Neural network architectures and learning’, Proceedings of *2003 IEEE International Conference on Industrial Technology*, Maribor, Slovenia, Vol.1, pp.TU1-T12.

3. Chow MY, Sharpe RN, Hung JC (1993) ‘On the Application and Design of Artificial Neural Networks for motor fault detection – Part I’, IEEE Transactions on Industrial Electronics, Vol. 40, No 2 , pp 181-188, ISSN: 0278-0046.

4. Kola S, Varatharasa L (2000) ‘Identifying three phase I.M faults using neural networks’, ISA Transactions, Elsevier, Science Direct, Vol. 39, issue 4, pp. 433-439. ISSN: 0019-0578.

5. Chow MY, Sharpe RN, Hung JC (1993) ‘On the Application and Design of Artificial Neural Networks for motor fault detection – Part II’, IEEE Transactions on Industrial Electronics, Vol. 40, No 2 , pp. 189-196, April 1993, ISSN: 0278-0046.

6. Badriram (2011) Power System Protection and Switchgear. Tata McGraw-Hill Education, pp. 520-527.

7. Goode PV, Chow MY (1995) ‘Using a Neural/Fuzzy system to extract Heuristic knowledge of incipient faults in I.M: part I-Methodology’, IEEE Transactions on Industrial Electronics, Vol. 42, No.2, pp. 131-138, ISSN: 0278-0046.

8. Chow MY, Lee S (1991) ‘Methodology for on line incipient fault detection in single phase Squirrel Cage I.M. using ANN’, IEEE Transactions on Industrial Electronics, Vol. 06, No. 3, pp. 536-545, ISSN: 0278-0046.

9. Tag Eldin EM, Emara HR, Aboul-Zahab EM, Refaat S (2007) ‘Monitoring and Diagnosis of External faults in Three Phase Induction Motor using Artificial Neural Network’, *Proceedings of the IEEE Power Engineering Society General Meeting*, Tempa, FL.

10. Riley CM, Lin BK, Habetler TG, Kliman GB (1999) 'Stator Current Harmonics and their Causal Vibrations: A Preliminary Investigation of Sensorless Vibration Monitoring Applications', IEEE Transactions on Industrial Applications, Vol. 35, no. 1, pp. 94-99, ISSN: 0093-9994.

11. Sin ML, Soong WL, Ertugrul (2003) 'Induction machine on –line condition monitoring and fault diagnosis – A survey', *Proceedings of the* Australian *Universities Power Engineering Conference, Christchurch*, New Zealand, 2003, pp. 1-6.

12. Kola S, Shwan D (2007) 'ANN based fault identification scheme implementation for 3 phase I.M.', ISA Transactions, Elsevier, Science Direct, Vol. 46, issue 2, pp.433-439, ISSN: 0019-0578.

13. Prechelt L (1996) 'Early stopping –But When?, Proceeding Neural Networks: Ticks of the Trade, Springer-Verlag, London, UK, pp.55-69.

14. Reffat SS, Abu-Rub H, Iqbal A (2014) 'ANN-based system for a discrimination between unbalanced supply voltage and phase loss in induction motors', Proceedings of *7th IET international conference on Power Electronics,* Machines and Drives (PEMD 2014), Manchester, pp. 1-6.

15. Mossavi SS, Djerdir A, Ait-Amirat Y, Khaburi DA (2012) 'Fault detection in 3 phase traction motor using Artificial Neural Networks', *Proceedings of the Transportation Electrification Conference and Exposure (ITEC), IEEE,* Dearborn, MI, pp. 1-6.

16. Nejjari H , Benbouzid MEH (2000) 'Monitoring and Diagnosis of Induction Motors Electrical Faults using a current park's vector pattern learning approach', IEEE Transactions on Power Electronics, Vol. 36, No. 3, pp. 730-735, ISSN: 0278-0046.

17. Özgönenel O, Yalçin T (2011) 'A complete motor protection algorithm based on PCA and Ann: A real time study', Turkish Journal of Electrical and Computer Science, Vol.19, No. 3, pp. 317-334, ISSN: 1300-0632.

18. Hosny A, Safiuddin M (2009) 'ANN-based protection system for Controllable Series-Compensated transmission lines', *Proceedings of the Power Systems Conference and Exposition (PSCE '09),* IEEE/PES, Seattle, WA, pp.1-6.

19. Lingxin L, Mechefske CK, Weidong L (2004) 'Electric motor faults diagnosis using artificial neural networks', Insignt-Non Destructive Testing and Condition Monitoring, Vol. 46, No.10, pp. 616-621.

20. Hammo R (2014) Faults identification in three phase induction motors using support vector machines. Master of Technology Management Plan II Graduate Projects, Paper1, Bowling Green State University.

21. Specht DF (1990) 'Probabilistic Neural Networks', Neural Networks, Vol. 3, pp. 109-118, ISSN: 0893-6080.

22. Temurtas F (2009) 'A comparative study on thyroid disease diagnosis using neural networks', Expert system with applications, Vol. 36, pp. 944-949, ISSN: 0917-4174.

23. Abd Alla AN (2006) 'Three phase I.M motor faults detection by using Radial Basis function neural Network', Journal of Applied Science, Vol. 6, No.13, Asian Network for scientific information, ISSN 1812-5654.

24. Wang C, Yang W, Chen J, Liao G (2009) 'Power disturbance recognition using probabilistic neural networks', *Proceedings of the International Multi Conference of Engineers and Computer Scientists (IMECS),* Hong Kong, pp 1573-1577.

25. Samanta B, Al-Balushi KR, Al-Araimi, SA (2005) 'Artificial neural networks and genetic algorithm for bearing fault detection', Soft Computing, Vol. 10, pp. 264-271, ISSN: 1433-7479.

26. Silva AA, Bazzzi AM, Gupta Shalabh (2012) 'fault diagnosis in electric drives using machine learning approach', *Proceedings of the 2013 IEEE International Electrical Machines and Drive Conference,* Chicago, IL , pp. 722-726.

27. Siddeque A, Yadav G, Singh B (2004) 'Identification of 3 phase induction motor faults using ANN', *Proceedings of the 2004 IEEE International Symposium on Electrical Insulation,* Indianapolis, IN USA, 2004, Vol. 39, (4), pp. 433-439.

28. Rebizant W, Wiszniewski A (2011) Digital Signal Processing in Power System Protection and Control. Springer Science & Business Media, pp. 228-237.

29. Zeraoulia M, Mamoune A, Mangel H, Benbouzid MEH (2005) 'A simple FL approach for induction motors stator condition monitoring', Journal of Electrical Systems, Vol. 1, No.1, pp. 15-25.

30. Mini VP, Sivakortaiah S, Ushakumari S (2010) 'Fault detection and diagnosis of an induction motor using FL', Proceedings of the IEEE Region 8 SIBIRCON-2010, Irkutsk Listvyanka, Russia, July 11-15, pp. 459-464.

31. Mini VP, Ushakumari S (2011) 'Incipient fault detection and diagnosis of induction motor using fuzzy logic', *Proceedings of the Recent Advances in Intelligent Computational Systems, IEEE,* Trivandrum, pp. 675-681, 2011.

32. Zidani F, Benbouid MEH, Diallo D, NaïSaïd MM (2003) 'Induction Motor stator faults diagnosis by a current concordia pattern based fuzzy decision system', IEEE Transaction on Energy Conversion, Vol. 18, No. 4, pp. 469-475, ISSN: 0885-8969.

33. Guillaume S (2001) 'Designing fuzzy inference system from data sets: An interpretability oriented review', IEEE Transactions on Fuzzy Systems, Vol. 9, No. 3, pp. 426-442, ISSN: 1063-6706.

34. Moreno JE, Castillo O, Castro JR, Martínez LG, Melin P (2007) 'Data Mining for extraction of fuzzy IF-THEN rules using Mamdani and Takagi-Sugeno-Kang FIS', Engineering Letters, Vol. 15, No. 1, pp. 82-88, ISSN: 1816-0948.

35. Chiu SL (1994) 'Fuzzy model identification based on cluster estimation', Journal of Intelligent Fuzzy systems, Vol. 2, pp. 267-278, ISSN: 1064-1246.

36. Goode PV, Chow MY (1995) 'Using a Neural/Fuzzy system to extract Heuristic knowledge of incipient faults in I.M: part II-Application', IEEE transactions on industrial Electronics, Vol. 42, No.2, pp. 139-146, April 1995, ISSN: 0278-0046.

37. Ahmed SM, Abu-Rub H, Refaat SS, Iqbal A (2012) 'Diagnosis of stator turns -to-turn fault and stator voltage unbalance using ANFIS' International Journal of Electrical and Computer Engineering, Vol. 3, No. 1, pp. 129-135, ISSN: 2088-8708.

38. Khan MAS, Rahman MA (2006) 'Discrete Wavelet transform based detection of disturbances in induction motors', *Proceedings of the International Conference on the Electrical and Computer Engineering (ICECE'06), IEEE*, Dhaka, Bangladesh, pp. 462-465.

39. Khan MAS, Radwan TS, Rahman MA, 'Real-Time Implementation of Wavelet Packet Transform-Based Diagnosis and Protection of Three Phase Induction motors', IEEE Transactions on Energy Conversion, Vol. 22, No.3, pp. 647-654, ISSN: 0885-8969.

40. Benbouzid MEH,Vieira M, Theys MC (1999) 'Induction Motors faults detection and localization stator current advanced signal processing techniques', IEEE Transactions on Power Electronics, Vol. 14, No. 1, pp. 14-22, ISSN: 0278-0046.

41. Cococcioni M, Lazerni B, Volpi S (2013) 'Robust Diagnosis of Rolling Element Bearings Based on Classification Techniques', IEEE Transaction on industrial informatics, Vol. 9,pp. 2274-2283, ISSN: 1551-3203.

42. Vedant, Sugumaran V, Amarnath M, Kumar H (2013) 'Fault diagnosis of helical gear box using sound signal using Naïve Bayes and Bayes net', International Journal Engineering Technology Research, Vol. 1, pp. 98-105, ISSN: 2347-4904.

43. Asfani D, Purnomo M, Sawitri D (2013) 'Naïve Bayes classifier for temporary short circuit fault detection in stator winding', Proceeding of 9th IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives (SDEMPED), Valencia,Spain,pp.288-294.

44. Sudha M, Anbalagan P (2009) ‘A protection Scheme for three phase I.M form Incipient faults using Embedded Controller’, Asian Journal of Scientific Research, Vol. 2, pp.28-50, ISSN: 1992-1454.

45. Ayasun S, Nwankpa CO (2005) ‘Induction Motor Tests Using MATLAB/SIMULNK and Their Integration Into Undergraduate Electric Machinery Courses’, IEEE Transactions on Education, Vol. 48. No. 1, pp. 37-46.

46. Shi KL, Chan TF,Wong YK, Ho SL (1999) ‘Modelling and simulation of 3 phase induction motor using Simulink’, International journal of Electrical Engineering Education, Vol. 36, Manchester, pp. 163-172, ISSN: 0020-7209.

47. Ansari AA, Deshpande DM (2010) ‘Mathematical Model of Asynchronous Machine in Matlab/ Simulink’, International Journal of Engineering Science and Technology, pp. 1260-1267, ISSN: 2141-2839.

48. Orille AI, Sowilam GMA, Valencia JA (1999) ‘A New simulation of symmetrical 3 phase I.M under transformations of park’, Computers and Industrial Engineering, Vol. 37, issue 1-2, pp. 359–362, ISSN:0360-8352.

CHAPTER - 3

Induction Motor External Faults and its Simulation

3.1 Induction Motor

Induction motors are complex electromechanical devices widely used for conversion of power from electrical to mechanical form in various industrial applications because they are robust, controlled and most suitable for many applications like pumps, fans, compressors, machine tools etc. The focus of this study is mainly related with LV/MV small and medium size squirrel cage induction motors faults identification.

3.2 Induction Motor External Faults

3.2.1 Overload

As the mechanical load on induction motor increases the motor begins to draw high current and speed decrease. Below the normal rated current heat dissipation is more than the heat produced and vice versa above normal rated current. After certain amount of load heat generation rate is higher than heat dissipation rate than the insulation is threatened.

Overload protection is always applied to motors to protect them against overheating. The National Electric Code requires that an overload protective device be used in each phase of induction motor unless protected by other means as because single phasing in the primary of a delta-wye transformer that supplies motor will produce three phase motor currents in a 2:1:1 relationship. If the two units of current appeared in the phase with no overlaod device the motor would be unprotected [1].

The overload protection can also divide in two stages, alarming and tripping. In case of pre-warning alarm (for example 90 % of full load) operator get some time to find out possible source of overload and to resolve the cause. If the overload becomes higher (for example greater than 10-15 %) than tripping is required [2].

The limitation of this scheme that ambient temperature and cooling effect will not be considered on current base fault identification so soft computing based overload protection can be used for prewarning.

3.2.2 Overvoltage

Induction motor is designed to withstand overvoltage upto +10% as general voltage design motor manufacture specification. When voltage increases beyond it motor overheat because of increase in core losses. Current draw is only controlled by the load and at rated current and 10% overvoltage the motor will be overloaded by approximately 10%. The core loss is 20 to 30% greater than normal and could cause the machine to overheat.

3.2.3. Undervoltage

As the voltage across motor reduces slip increases, motor speed drops and current increases. This is because the power to be delivered remains constant and voltage is reduced from normal rated voltage. The increase of current can harm the insulation of the motor windings.

When the voltage is reduced of normal the developed torque moves to lower and in order to develop the torque to drive the load motor slow down (slip increase) and draws more current from supply. The current changes drastically as voltage reduces below 75 to 80% of rated voltage. In some cases, a large drop in voltage may cause the motor to stall also [3].

3.2.4 Single Phasing

Single phasing is the worst case of voltage unbalance and can be happened because of open winding in motor or any open circuit in any phase anywhere between the secondary of transformer and the motor or one pole of circuit breaker open or opening of fuse [4] [5]. The single phasing causes unbalanced currents to flow and the negative sequence component of these unbalanced current causes the rotor to overheat. The negative sequence current increases the rotor copper losses also. It is the worst case of voltage unbalance. Negative sequence currents generated will be approximately six times the negative sequence voltage. Thus effect of increase negative sequence current is 6 times the effect of similar increase in positive sequence current due to thermal overload [6].

3.2.5 Voltage Unbalance

Unbalanced supply voltage causes negative sequence currents to circulate in the motor, which increases the stator and rotor heating. The main causes of voltage unbalance condition are open delta transformers, lack of adequate transpositions in supply lines, single phase fuse failure, pole discrepancy of a circuit breaker, unbalanced loading, unequal tap settings, high resistance connections, Shunted single phase load, unbalanced primary voltage and defective transformer [7] [4]. Voltage unbalance can also be causes by unsymmetrical fault within induction motor or such a fault on the feeder feeding the induction motor from supply side. Presence of small voltage unbalance results in large current unbalance by a factor of six times and negative sequence phase components cause increased stator and rotor copper loss, eddy current loss, overheating , reduction in output torque and efficiency. Unbalance also causes mechanical problem like vibration. So Induction motor voltage unbalance monitoring is required to prevent or protect motor from failure.

The negative-sequence current usually produces very little torque, especially if the unbalance is small, which implies a small negative-sequence current. Its major effect is to increase the losses, primarily the stator I^2R losses. The winding carrying the largest current will overheat, but in time the excess heat is distributed throughout the machine more or less uniformly. This may cause the machine to be derated, with the derating being highly dependent on the ratio of sequence impedances given by equation of ratio of starting to running current [3].

NEMA standard suggest no derating required up to 1% unbalance, from 1 to 5% motor derating require and above 5% operation is not recommended [8] [9] [10]. Standard motor are capable of operating under condition of supply voltage unbalance of 1% for long period. Voltage unbalance more than 1% is considered voltage unbalance condition in simulation and experimental study in this thesis and less than 1% voltage unbalance is considered as normal condition. All types of voltage unbalance like single phase and two phase undervoltage and overvoltage unbalance, three phase undervoltage and overvoltage unbalance and one phase, two phase angle displacement are considered in the simulation data sets.

TABLE 3.1

Relative Insulation Life for Different % Voltage Unbalances for Induction Motor (for 100% Motor Loading and 1 Service Factor) [11]

Voltage Unbalance (%)	**Derating Required**
0	1
1	0.9
2	0.64
3	0.37
4	0.17

Percentage line unbalance considered based on NEMA definition

% Line Unbalance Voltage Ratio = (Maximum Voltage from average line voltage magnitude /Average Voltage) x 100% ……………………………………………… (3.1)

The magnitude of the NEMA unbalanced voltage in percentage and negative sequence voltage in percentage is almost equivalent for all practical purpose [8].

3.3 Induction Motor External Faults Simulation

A three phase induction motor external faults simulation is prepared in Matlab/Simulink environment [12] with varying operating voltages and load. OL, OV, UV, SP (for each phase), VUB and normal conditions are simulated to obtain three phase RMS line voltages and RMS line current values. The fault simulation is prepared using 3 phase, 50Hz, 4 kW/5.4 HP, 400 V, 1430 rpm, star connected induction motor. Induction motor block in Matlab/Simulink is based on arbitrary reference frame theory and contains highly nonlinear modelling equations. Induction motor is used in stationary reference frame. Data sets are obtained using ode 23tb stiff solver in simulation. The three phase steady state RMS voltages and currents values are obtained as data sets (patterns) and used as input feature vectors for training, for example in MLPNN based fault identification algorithm for MLPNN training. We have prepared 174 data sets for training and 46 data sets for independent testing. The training data sets and testing data sets are shown in Appendix A and Appendix B respectively. The number of train and test data sets patterns used for six output conditions are shown in Table 3.2. Subsection 3.3.1 to 3.3.6 discusses details of how different output conditions data sets are obtained and also shows example of how the independent test data sets (patterns) are obtained for output conditions. Three phase RMS voltages and current values obtained using simulation at 1.2s and used as test vector. Table 3.3 shows some example of independent test patterns.

TABLE 3.2

Number of Patterns for simulation Train and Independent Test Data Sets

Sr. No.	Condition	Train Data	Independent Test Data
1	Normal Output (N)	31	6
2	Overload (OL)	19	6
3	Overvoltage(OV)	30	6
4	Undervoltage(UV)	20	5
5	Single Phasing (SP)	25	15
6	Voltage Unbalance (VUB)	49	8

Some of the test (unseen) patterns used for MLPNN testing (in chapter 4) obtained for different external fault conditions alongwith normal conditions are shown in Table 3.3.

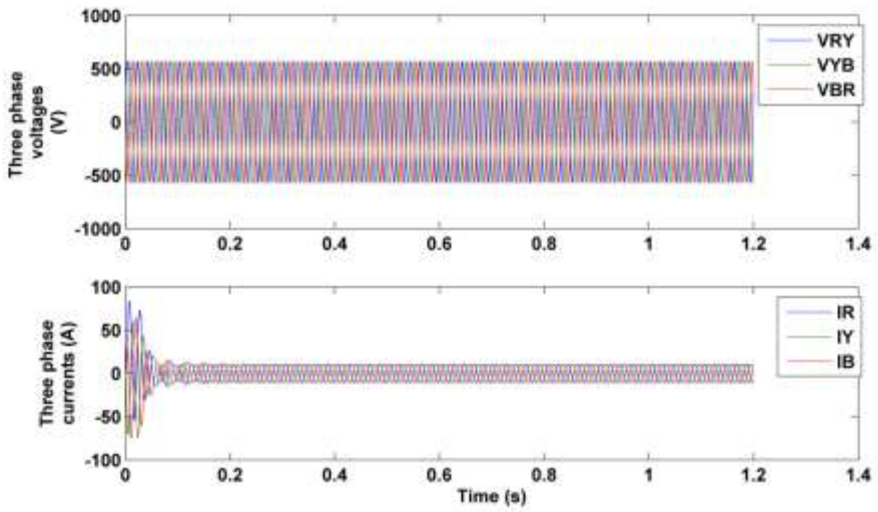
TABLE 3.3

Examples of Test Inputs for Simulation Data Sets

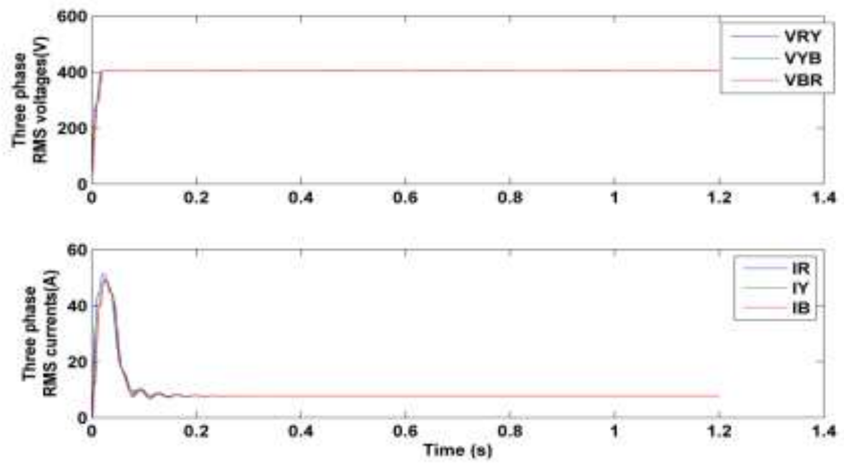
Sr. No.	Output	Inputs					
		VRY (In1)	VYB (In2)	VBR (In 3)	IR (In 4)	IY (In 5)	IB (In 6)
1	N	405.1	405.4	405.6	7.74	7.72	7.73
2	N(92.5% UV within normal limit)	369.7	369.9	370	8.16	8.16	8.16
3	N (VUB 0.52%)	397.7	394.4	396.2	8.08	7.68	7.57
4	OL	399.7	400	400	10.33	10.33	10.33
5	OV	443.2	443.4	443.5	7.44	7.45	7.44
6	UV	347.6	347.8	348	8.55	8.54	8.54
7	SP(R phase)	300.2	412.7	374.6	0	16.26	16.26
8	SP(y phase)	370.6	265.7	400.4	20.1	0	20.1
9	SP(B phase)	384.3	352.4	263.6	17.6	17.6	0
10	VUB (2 phase UV)	365.7	390.0	355.8	6.819	11.62	8.294
11	VUB (3 phase OV)	427.1	438.4	432.5	6.51	7.68	8.09

3.3.1 Normal Condition

Induction motor normal operation data sets are obtained with rated load torque and also with some other normal variant loading (60-105% of full load) condition and different normal balanced voltage of the range ±10% of rated voltage, with which motor mostly operates in industry. Fig. 3.1 (a) shows the three phase voltage and currents and Fig. 3.1(b) shows three phase RMS voltages and currents for normal condition Sr. No. 1 of Table 3.3



(a)

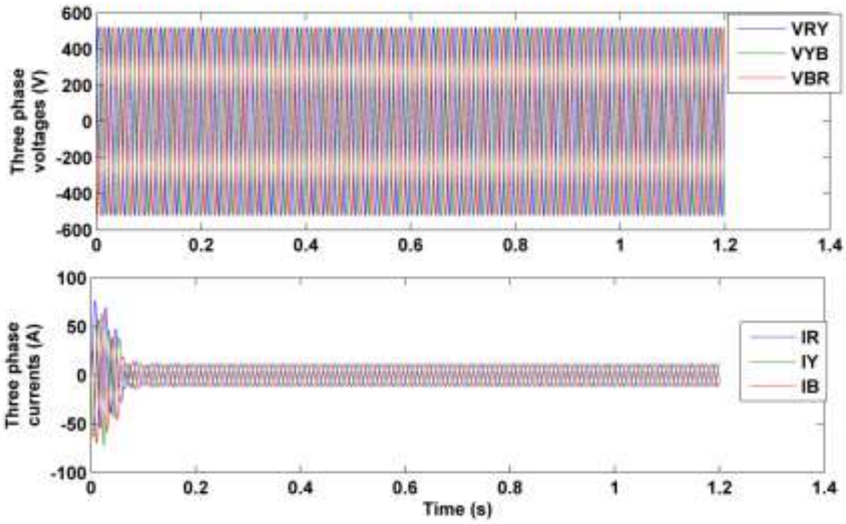


(b)

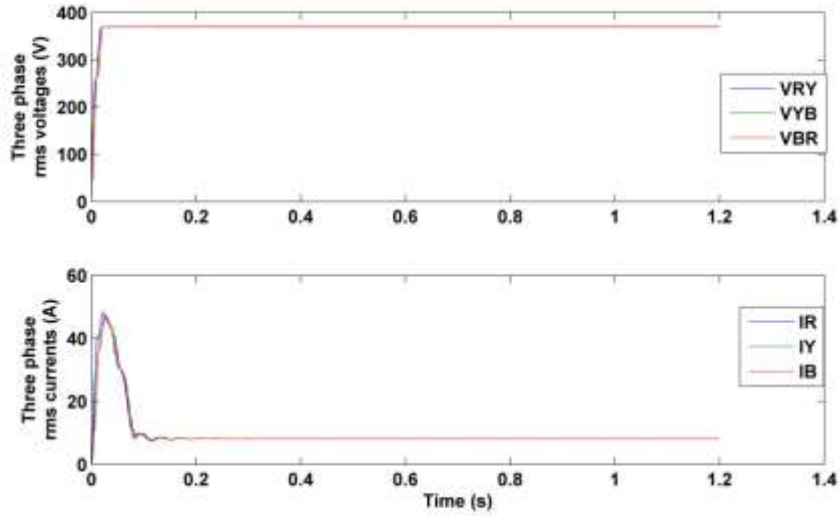
FIGURE 3.1

Normal Condition (Sr. No.1 of Table 3.3)
(a) Three Phase Voltages and Currents (b) Three Phase RMS Voltages and Currents

Fig. 3.2 (a) shows the three phase voltage and currents and Fig. 3.2 (b) shows three phase RMS voltages and currents for normal condition (with 92.5% rated normal voltages) Sr. No. 2 of Table 3.3.



(a)

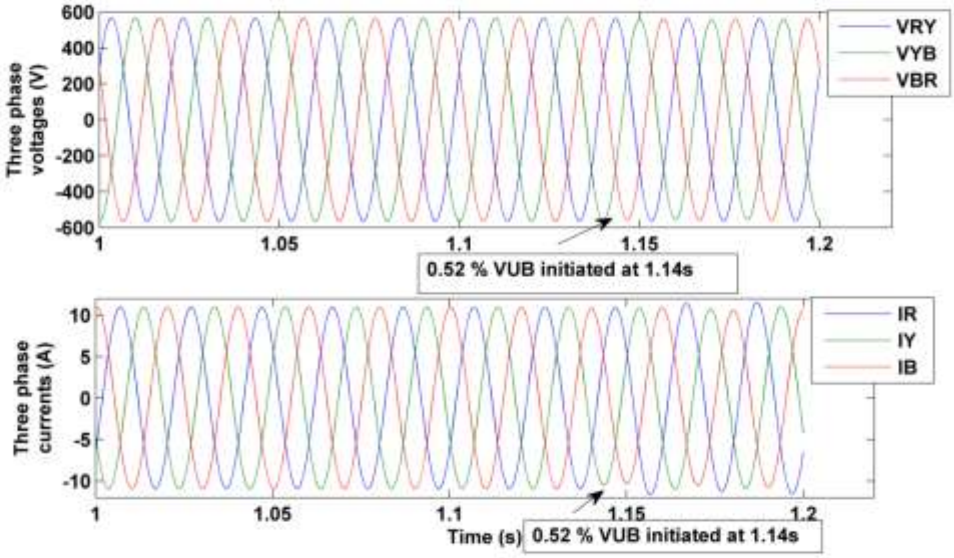


(b)

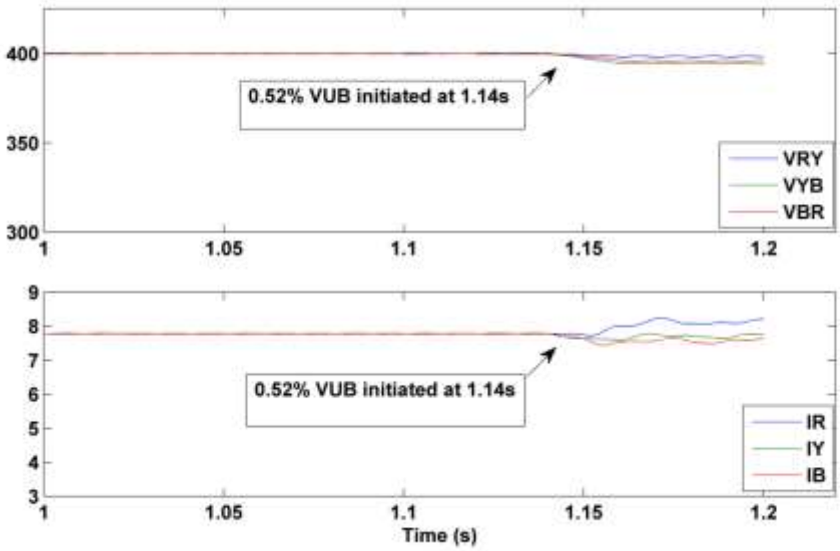
FIGURE 3.2

Normal Condition (Sr. No. 2 of Table 3.3) (a) Three Phase Voltages and Currents for (b) Three Phase RMS Voltages and Currents

Fig. 3 (a) shows the three phase voltages and currents and Fig. 3(b) shows three phase RMS voltages and currents for normal condition with 0.52% VUB in supply voltages. 0.52% VUB initiated at 1.14s.



(a)



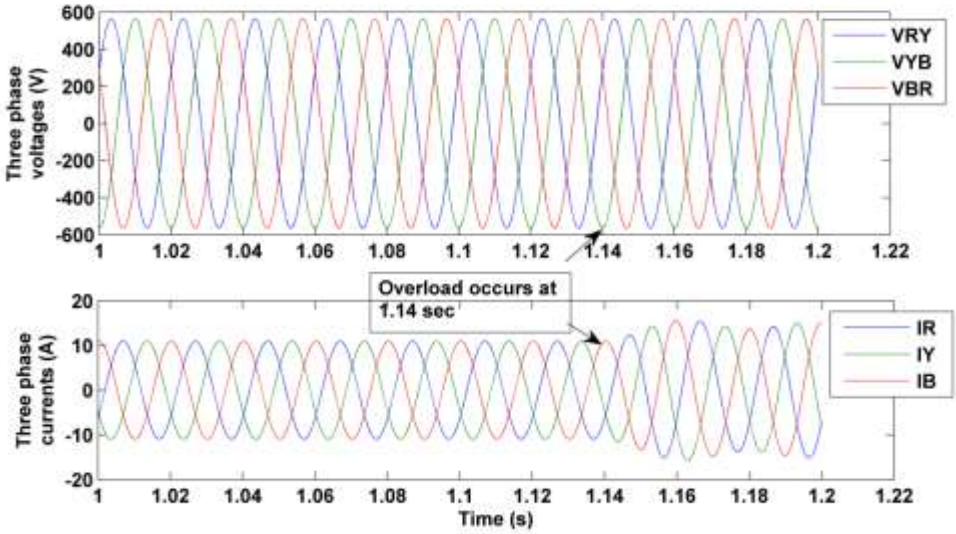
(b)

FIGURE 3.3

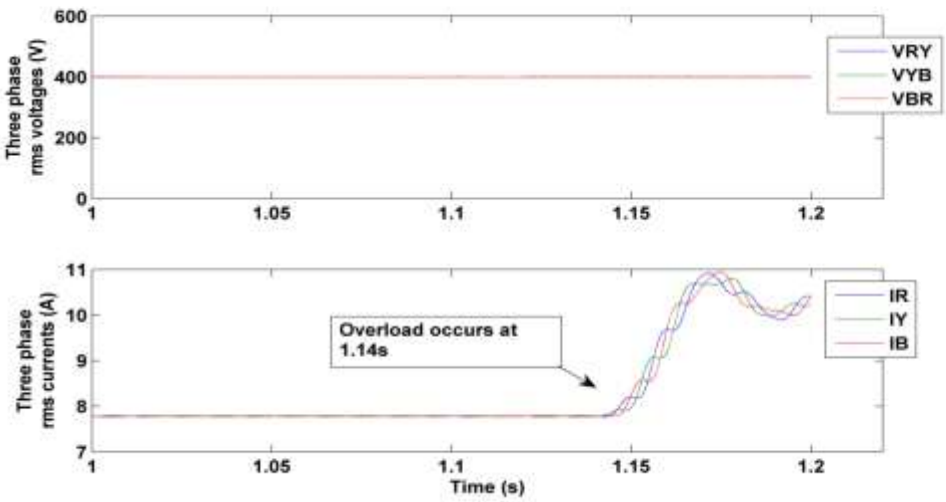
Normal Condition (Sr. No. 3 of Table 3.3) (a) Three Phase Voltages and Currents for (b) Three Phase RMS Voltages and Currents

3.3.2 Overload Condition

Loading condition above 105% to 150% of normal load is considered as motor overload condition. Fig. 3.4 (a) shows the three phase voltages and currents and Fig. 3.4 (b) shows three phase RMS voltages and currents for OL condition. 124% OL of rated current initiated at 1.14s.



(a)



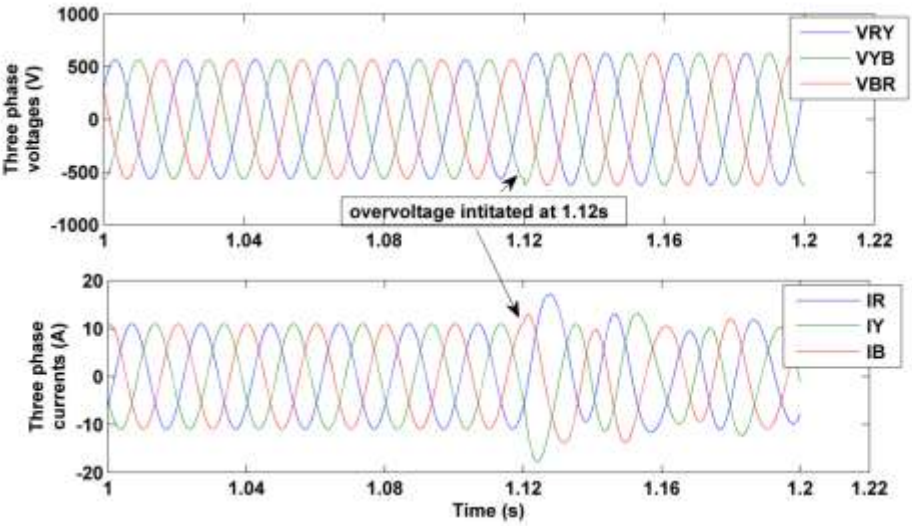
(b)

FIGURE 3.4

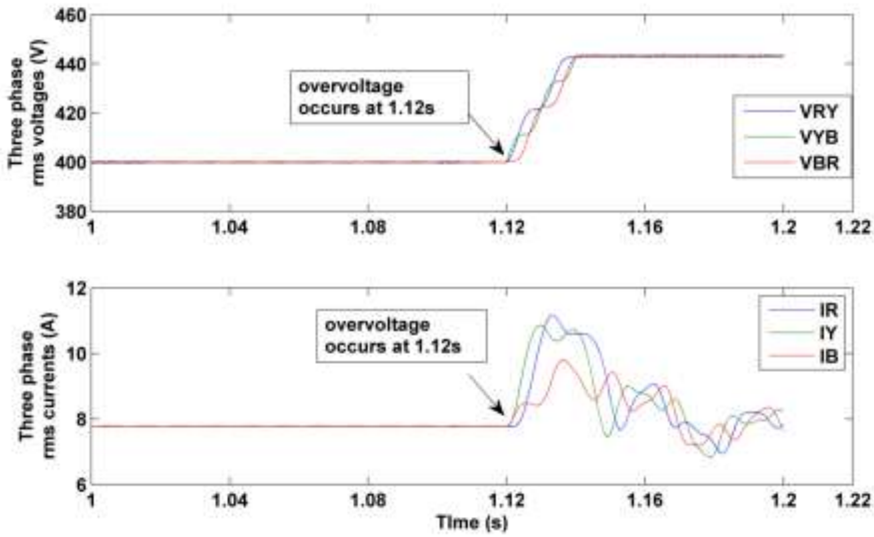
OL Condition (Sr. No 4 of Table 3.3) (a) Three Phase Voltages and Currents for (b) Three Phase RMS Voltages and Currents

3.3.3 Overvoltage Condition

The operating voltages more than 10% rated operating voltages are considered as overvoltage condition in simulation. Fig. 3.5 (a) shows the three phase voltages and currents and Fig. 3.5 (b) shows three phase RMS voltages and currents for OV condition. 110.8% OV of rated voltage initiated at 1.12s.



(a)



(b)

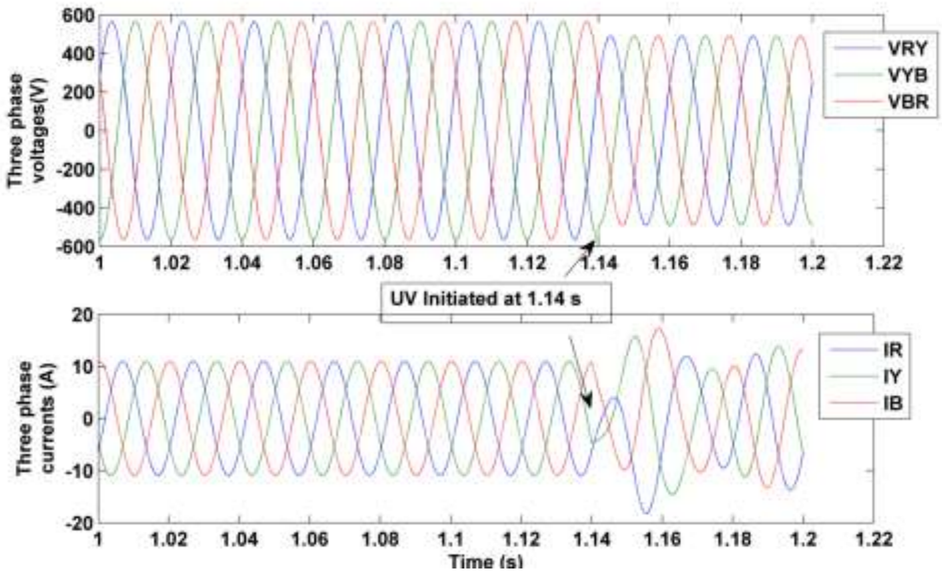
FIGURE 3.5

OV Condition (Sr. No.5 of Table 3.3) (a) Three Phase Voltages and Currents for (b) Three Phase RMS Voltages and Currents

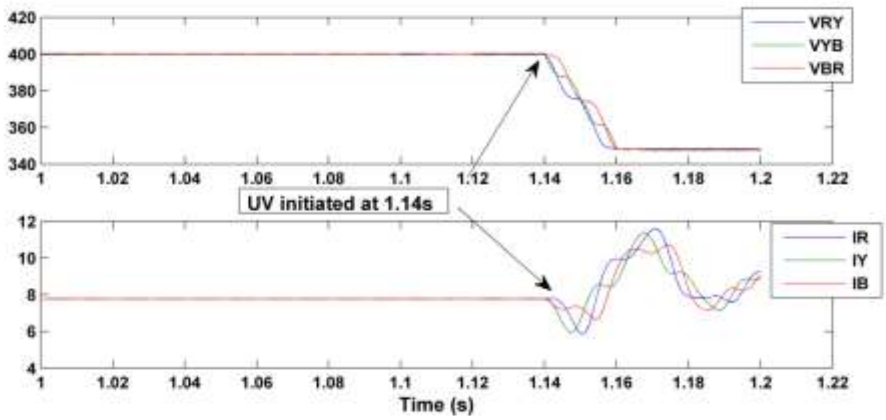
3.3.4 Undervoltage Condition

The operating voltages less than 10% rated operating voltages are considered undervoltage condition in simulation.

Fig. 3.6 (a) shows the three phase voltages and currents and Fig. 3.6 (b) shows three phase RMS voltages and currents for UV condition. 87% UV of rated voltage initiated at 1.14s.



(a)



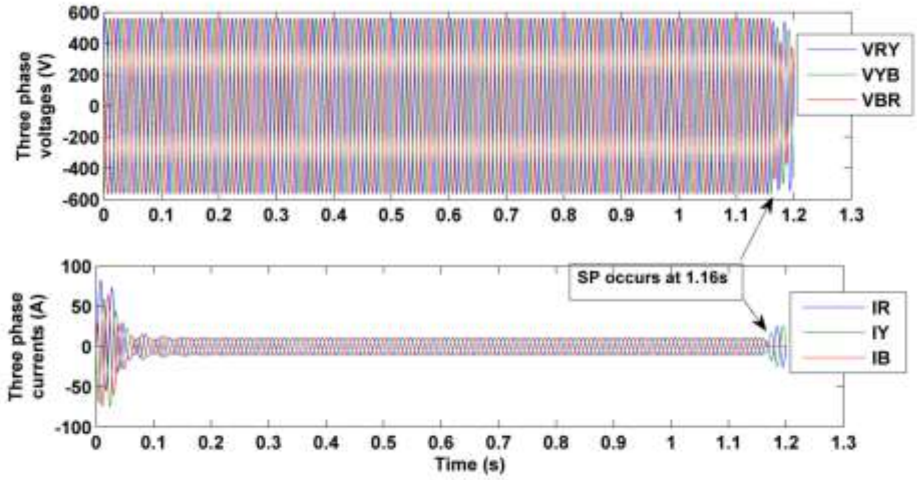
(b)

FIGURE 3.6

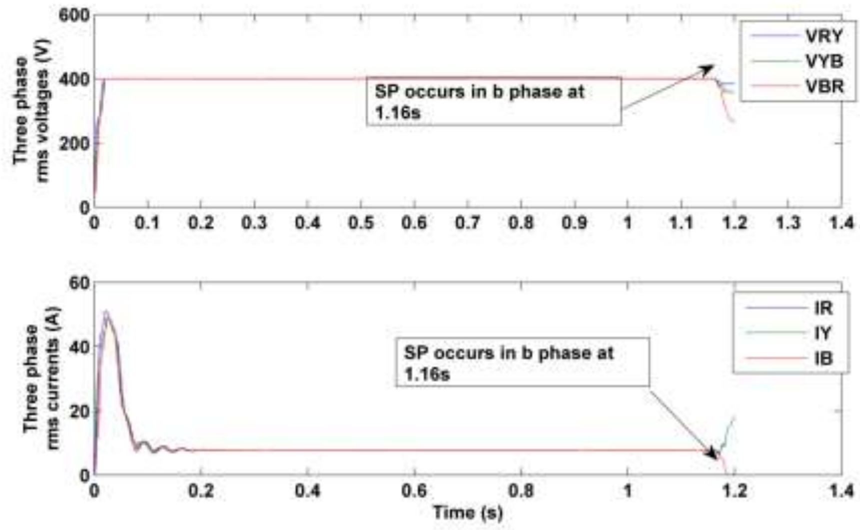
UV Condition (Sr. No. 6 of Table 3.3) (a) Three Phase Voltages and Currents (b) Three Phase RMS Voltages and Currents

3.3.5 Single Phasing Condition

Opening of any of three phases is considered in single phasing condition. Fig. 3.7 (a) shows the three phase voltages and currents and Fig. 3.7 (b) shows three phase RMS voltages and currents for SP condition. SP in B phase initiated at 1.16s.



(a)



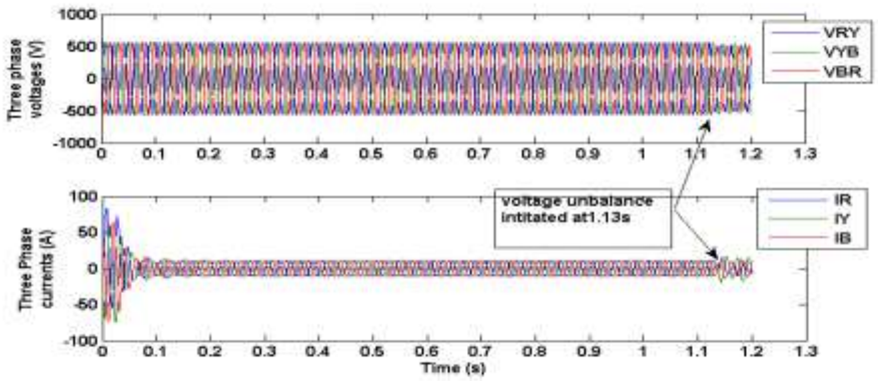
(b)

FIGURE 3.7

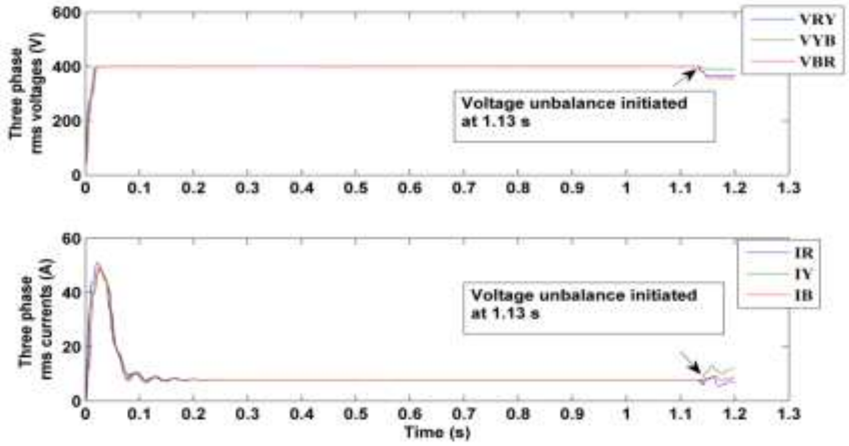
SP Condition in B phase (Sr. No. 9 of Table 3.3 at 1.16s) (a) Three Phase Voltages and Currents (b) Three Phase RMS Voltages and Currents

3.3.6 Voltage Unbalance Condition

Standard motor are capable of operating under condition of supply voltage unbalance of 1% for long period. Derating is requiring for voltage unbalance between 1 to 5% for safe operation which is generally not taken care in field. We have considered voltage unbalance more than 1% as fault which. All types of voltage unbalance like single phase and two phase undervoltage and overvoltage unbalance, three phase undervoltage and overvoltage unbalance and one phase, two phase angle displacement considered in the case. Fig. 3.8 (a) shows the three phase voltages and currents and Fig. 3.8 (b) shows three phase RMS voltages and currents for VUB condition. Two phase undervoltage VUB initiated at 1.13s.



(a)



(b)

FIGURE 3.8

VUB Condition (Sr. No. 10 of Table 3.3) (a) Three Phase Voltages and Currents for (b) Three Phase RMS voltages and Currents

3.3.7 Scatter Plot Visualization of Simulation Data Sets

Fig. 3.9 shows the scatter plot visualization for fault data sets patterns obtained using simulation. Plot displays input variable relations with respect to different fault class and found linearly nonseparable and overly complex.

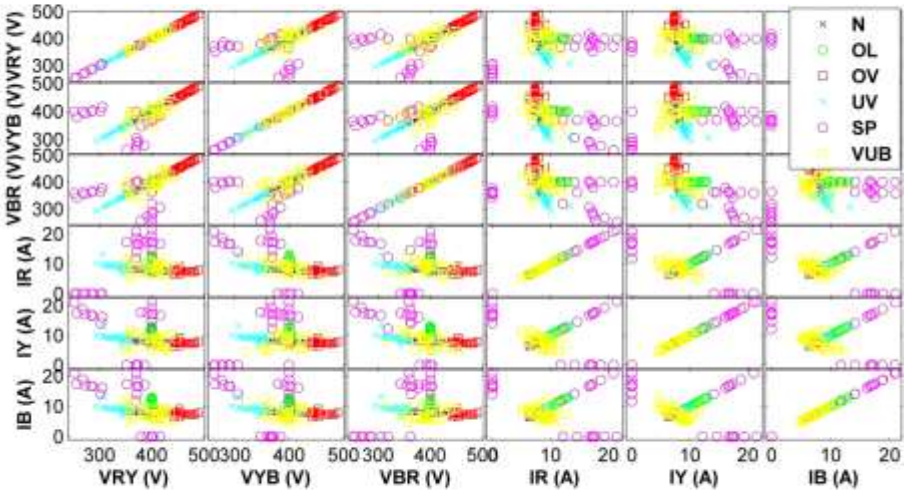


FIGURE 3.9

Scatter Plot of Simulation Data Sets

3.3.8 External Faults Identification With Linear Discriminant Analysis (LDA) and Discussions

3.3.8.1 LDA

Besides visualization of data sets complexity using scatter plot we have also tested with conventional and widely used LDA for classification results. Discriminant analysis approaches are well known statistical approaches and widely used in pattern recognition tasks. It can be easily extended to multiclass cases Via multiple discriminant analysis [13] [14]. LDA analysis can be used to study the difference between groups of objects (two or more) with respect to several variables simultaneously; for determining wheather meaningful differences exist between the groups [13].

The basic idea of LDA is to find a linear transformation that best discriminate among classes and the classification is then performed in the transformed space based on some metric such as Euclidean distance.

- **Two-Class LDA:**

Fisher first introduced LDA for two classes and his idea was to transform the multivariate observations X to univariate observations Y such that the Y's derived from two classes were separated as much as possible. For example, suppose a set of n numbers q-dimensional samples $X_1, X_2, \ldots, X_n$ (where $X_i = (X_{i1},\ldots,X_{iq})$) belonging to two different classes, namely c_1 and c_2. For these two classes, the scatter matrices are given as

$$S_i=\sum_{x\in c_i}(X-\overline{X}_i)(X-\overline{X}_i)' \quad \ldots\ldots (3.2)$$

Where in (3.2), $\overline{X}_i=\frac{1}{n_i}\sum_{x\in c_i} x$. n_i is the number of samples in c_i. Hence the total intra-class scatter matrix is given by

$$\widehat{\sum}_w=S_1+S_2=\sum_i\sum_{x\in c_i}(X-\overline{X}_i)(X-\overline{X}_i)' \quad \ldots\ldots (3.3)$$

The inter-class scatter matrix is given by

$$\widehat{\sum}_b=(\overline{X}_1-\overline{X}_2)(\overline{X}_1-\overline{X}_2)' \quad \ldots\ldots (3.4)$$

Fisher's criterion suggested the linear transformation Φ to maximize the ratio of the determinant of the inter-class scatter matrix of the projected samples to the intra-class scatter matrix of the projected samples:

$$J(\Phi)=\frac{\Phi^T\widehat{\sum}_b\Phi}{\Phi^T\widehat{\sum}_W\Phi} \quad \ldots\ldots (3.5)$$

If $\widehat{\sum}_w$ is non-singular, above (3.5) can be solved as a conventional eigenvalue problem and Φ is given by the eigenvectors of matrix $\widehat{\sum}_w^{-1}\widehat{\sum}_b$ [13].

- **Multi-Class LDA:**

If the numbers of classes are more than two, then a natural extension of Fisher linear discriminant possible using multiple discriminant analysis. As in two-class case, the projection is from high dimensional space to a low dimensional space and the transformation suggested still maximizes the ratio of intra-class scatter to the inter-class scatter. The maximization should be done among several competing classes unlike the two-class case. Suppose that now there are p classes. The intra-class matrix is calculated similar to (3.3):

$$\hat{\sum}_w = S_1 + \ldots + S_p = \sum_{i=1}^{p} \sum_{x \in c_i} (X-\bar{X}_i)(X-\bar{X}_i)' \ldots\ldots (3.6)$$

Inter-class scatter matrix slightly differs in computation and is given by

$$\hat{\sum}_b = \sum_{i=1}^{p} m_i (\bar{X}_i - \bar{X})(\bar{X}_i - \bar{X})' \ldots\ldots (3.7)$$

Where in (3.7), m_i is the number of training samples for each class, $\bar{X}_i$ is the mean for each class and $\bar{X}$ is the total mean vector given by $\bar{X} = \frac{1}{m} \sum_{i=1}^{p} m_i \bar{X}_i$, Transformation Φ can be obtained by solving generalized eigenvalue problem

$$\hat{\sum}_b \Phi = \lambda \hat{\sum}_w \Phi \ldots\ldots (3.8)$$

λ is known as eigenvalue. Once the transformation Φ is given, the classification is then performed in the transformed space based on some distance metric such as Euclidean distance

$d(X,Y)=\sqrt{\sum_i (X_i - Y_i)^2}$ and cosine measure $d(X,Y) = 1 - \frac{\sum_i X_i Y_i}{\sqrt{\sum_i (X_i)^2} \sqrt{\sum_i (Y_i)^2}}$

. Then upon arrival of the new instance Z, it is classified to $\arg\min_k d(Z\Phi, \bar{X}_k \Phi)$, where $\bar{X}_k$ is the centroid of k-th class [13].

3.3.8.2 Classification Results Obtained Using LDA and Discussions

We have used MATLAB classify function for linear discriminant analysis based fault diagnosis with own written codes.

The classification accuracy results obtained for total train (174) and (46) independent test data sets are obtained as follows. Total classification Accuracy is defined as the total number of correct decisions to total number of cases.

Total train classification accuracy (with 174 total train data sets) = 70.11%

Independent test classification error=0.261 (with independent 46 test data sets) =26.1%

Independent test classification accuracy (1- classification error) = 73.9%.

The programme is also tested with 10-fold cross validation by splitting total train data sets in 10-fold train and test data sets.

CVMCR (misclassification test error with 10-fold cross validation) = 0.3448 = 34.48%,

It is observed the classification error obtained with widely LDA is high for this complex and multi-class fault identification problem.

We have used ANN in next chapter and shown results obtained for ANN. The three phase steady state RMS values of voltages and currents are obtained for normal and external faults condition which used as input patterns for MLPNN training. We have tested MLPNN with separate test patterns and also discussed results obtained in next chapter. The test pattern variables are mostly within ±7.5% of train pattern variables values for simulation and practical data sets. It is observed MLPNN can identify any unseen external faults with high classification accuracy.

As single phasing is more severe fault among all these faults and requires early tripping we have taken RMS values after 2 cycles for simulation train data sets. Steadystate RMS values can also be taken for SP case. Present numerical protection for single phasing provides delay about 5 sec. External faults not demand instantaneous tripping and can protected with proper time delayed protection. The same phenomena can be possible with the use of ANN and Fuzzy with suitable and less time delay than present numerical protection.

.

The limitation of this fault identification scheme is that it should be blocked during starting period. However as future scope this problem can be rectified by taking starting period input values for training for each condition.

References

1. Horowitz SH, Phadke AG (2008) Power system Relaying. In: Horowitz SH, Phadke AG (3^{rd} edition) *rotating machinery protection*, John Wiley and Sons, England, pp. 159-178.

2. Distribution Automation Handbook, section 8.11, Motor Protection, pp. 6-13.

3. Anderdson PM (1999) *Power System Protection*, IEEE Press Power Engineering, NJ, Willy interscience, New York, USA, pp. 783-787.

4. Kersting, WH (2004) 'Caused and effects of single phasing induction motors', *Proceedings of Rural Electrical Power Conference, IEEE,* Vol. 4, pp. 1-6.

5. Downs CL (2004) Motor protection. In: Elmore WA (2^{nd} Edition) Protective relaying theory and applications. Marcel Dekker, Inc., New York, pp. 153-155.

6. Oza, BA, Nair N., Metha R, Makwana, V (2010) Power system protection and switchgear. Tata Mc-Graw Hill, India, pp. 306-311.

7. Sudha M, Anbalagan P (2009) 'A Protection scheme for three-phase induction motor from incipient faults using embedded controller', Asian Journal of Scientific Research, Vol. 2, pp. 28-50, ISSN:1992-1454.

8. Cummings P, Dunki-Jacobs J, Kerr R (1985) 'Protection of induction motors against unbalanced voltage operation', IEEE Transactions on Industrial Applications, Vol. IA-21, No. 4, pp. 778-792, ISSN: 0093-9994.

9. NEMA Standard MGI-14.34, 1980

10. NEMA Standard MGI-20.55, 1980

11. Paoletti GJ, Rose A (1989) improving existing motor protection for M.V. motors, IEEE transactions on industry applications, Vol. 25, No.3, 1989, ISSN: 0093-9994.

12. MATLAB 2010b, Mathworks Inc.

13. Tao L, Shenghuo Z, Mitsunori O (2006) 'Using discriminant analysis for multi-class classification: an experimental investigation', Knowledge and information systems, 2006, pp. 453-472, ISSN: 0219-1377.

14. Johnson RA, Wichern DW (1988) *Applied multivariate statistical analysis*. Prentice Hall,NJ,USA.

CHAPTER - 4

External Faults Detection and Classification Using MLPNN

4.1 Artificial Neural Networks

4.1.1 Introduction

ANN functionally motivated by human brain represents a simplified model of biological nervous system. ANN is highly interconnected network and connected through number of processing elements. These processing elements are known as neurons. These neurons are connected though weighted links. ANNs have ability to acquire knowledge through learning and make it available for use. ANN can acquire knowledge through various existing learning mechanisms. ANN architectures are classified into various types based on their learning mechanisms and other features. Some classes of NN refer to this learning process as training and the ability to solve problem using the acquired knowledge as inference [1]. ANN is trained with known examples of problem. Trained ANN can identify unknown instances of the problem.

The NN are robust systems and fault tolerant. The NN possess the capabilities to generalize thus they can predict new outcomes from learnt patterns i.e. past trends. They can recall full patterns from incomplete, noisy or partial patterns. The NN can process information in parallel at high speed and distributed manner. ANNs are widely used for the applications like classification, clustering, pattern recognition, vector quantization, function approximation, forecasting, control, optimization, pattern completion and search [2] [3].

Data classification is a basic issue in pattern recognition, data mining, and forecasting. The goal classification is to assign a new object to a class from a given set of predefined classes based on the attribute values or features of the object. Classification is based on some discovered model, which forms an important piece of knowledge about the application domain. There has been wide range of methods for classification task and ANN is one of the popular and widely used techniques among them. In a classification task, the pattern which is to classify is typically fed into the networks as activation of a set of input units. This activation is then spread through the network via the connections, finally resulting in activation of the output units, which is interpreted as the classification result. There are a large number of different neural network architectures. One of the most popular neural network architectures used for classification is the MLP. The units are organized into different layers. The neural network is said to be feed-forward because the activation values propagate in one direction only, from the units in the input layer through a number of hidden layers, to end up in the output layer [4].

In the field of fault diagnosis, neural networks are frequently employed and about major publications utilizing a classification procedure for fault diagnosis, relied on neural networks. Since efficient tools for network training and implementation have become easily available, it is likely that neural networks are used in more than half of the applications today. They provide a means to achieve decent classification results with relatively moderate design effort.

NN based fault diagnosis is basically a general purpose solution. Prior knowledge of motor diagnosis models is not required. Only the training data should be obtained in advance. The knowledge about the problem is distributed among artificial neurons and between the connection weights and biases and set them to a value such that ANN performs better at the applied input. Thus, the entire training process is a means of evaluating right combination of weights and biases for which ANN performs at its best. A good ANN architecture gives the best performance in the least number of layers and least number of neurons. At training stage, the feature vector is applied as input to neural network and network adjusts its variable parameter, the weights and biases, to capture the relationship between input pattern and outputs [3].

The inherent drawbacks of neural network learning such as over training and under training can be resolved by generalization and proper selection of hidden neurons. However, the short comings of neural networks based motor fault diagnosis is that qualitative and linguistic information from the operator cannot directly utilized or embedded in neural networks structures and it is difficult to interpret the input and output mapping of a trained ANN into meaningful fault diagnosis rules [5].

4.1.2 Learning Methods

Learning methods in NN can be broadly classified into three basic types: supervised, unsupervised and reinforced [2] [1]. Here supervised and unsupervised learning are discussed.

4.1.2.1 Supervised Learning

In this every input pattern that is used to train the network is associated with an output pattern, which is the target or the desired pattern. A teacher is assumed to be present during the learning process, when a comparison is made between the network's computed output and the correct expected output, to determine the error. The error can then be used to change network parameters, which result in an improvement in performance. Perceptron learning rule, BP (generalized Widrow-Hoff learning or continuous Perceptron learning) rule, Widrow-Hoff learning rule, correlation and outstar learning rules are the examples of supervised learning of neural network.

4.1.2.2 Unsupervised Learning

In this learning, the target output is not presented to the network. It is as if there is no teacher to present the desired patterns hence the system learns of its own by discovering and adapting to structural features in input patterns. Hebbian learning rule, Winner-Take all learning rule are example of unsupervised learning.

4.1.3 MLPNN

The feedforward neural networks (FFNNs) allow only for one directional signal flow. Furthermore, most of FFNNs are organized in layers. Multilayer perceptron neural networks (MLPNNs) are widely used FFNNs in different kind of applications due to their fast operation, ease of implementation, and smaller training set requirements [6]. Based on universal approximation theorem one hidden layer is sufficient for a NN to approximate any continuous mapping from the patterns to the output patterns to an arbitrary degree of accuracy. Standard feedforward networks are universal function approximator and with only a single hidden layer can approximate any continuous function on any compact set and any measurable function to any desired degree of accuracy [7] [8]. The MLPNN can be used in cases where the shapes of the class boundary are complex and linearly not separable. Minimum amount of neurons and number of instances are necessary to program given task into MLPNN [9] [10]. There is no analytical method for determining the number of neurons in the hidden layer. Therefore it only found by trial and error [6] [11] [12].

There is no clear and exact rule due to complexity of the network mapping. Neurons depend on the function to be approximated and its degree of nonlinearity affects the size of network. Large number of neurons and layers may cause overfitting and may cause decrease the generalization ability.

The data sets obtained from the induction motor fault simulation of RMS three phase voltages and line currents are used as concurrent input training vector to train the neural network. The input data matrix should be preprocessed for the efficient and better form of NN training. The goal of normalization is to ensure that the statistical distribution of values for each net input and output is roughly uniform. We have used MATLAB function MAPMINMAX to processes matrices by normalizing the minimum and maximum values of each row to y_{min}(-1) and y_{max}(+1) and given as,

$$y = (y_{max} - y_{min}) * \frac{x - x_{min}}{y_{max} - y_{min}} + y_{min} \qquad (4.1)$$

An example of single hidden layer neural network is shown in Fig. 1. This network consist input layer, hidden layer and output layer. The FFNN is also used for nonlinear transformation of a multidimensional input variable into another multidimensional variable in the output. In theory, any input-output mapping should be possible if neural network has enough neurons in hidden layers (size of output layer is set by the number of outputs required). Presently there is no satisfactory method to define how many neurons should be used in hidden layers and usually found by trial and error method. On the other side, networks with larger number of neurons lose their ability for generalization, and it is more likely that such network will try to map noise supplied to the input also.

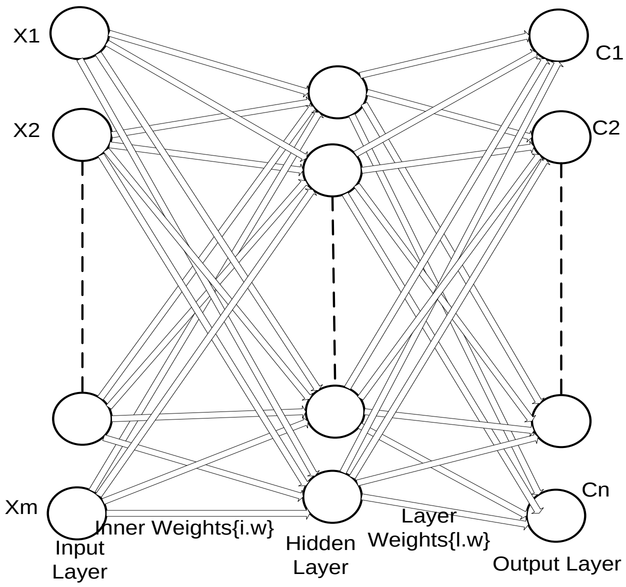


FIGURE 4.1

Single Hidden Layer MLPNN

Each input of MLPNN is weighted with randomly initialized weights and bias. The sum of weighted inputs and the bias forms the input to transfer function. Tansig transfer function is utilized in hidden layer. The output a^i of i^{th} neuron of hidden layer can be given as

$a^i =$ tansig (I.W. * p + b^i) ……………………………………………………………………………….(4.2)

Wherein (4.2) I.W is weight vector of different element of input vector p to i the hidden layer neuron and b^i is weight assigned to neuron. Tansig activation function is used in hidden and output layer. The output o^i of i^{th} neuron of output layer can be given as

$o^i =$ tansig (L.W. * n + b^i)………………………………………………………………………………….(4.3)

Wherein (4.3) L.W. is weight vector of different element of hidden layer to i^{th} output layer neuron, n is output vector of hidden layer and b^i is weight assigned to the i^{th} output layer neuron. o^i is output of the i^{th} output neuron. In our case, number of input layer neurons are 6 same as number of input variables and output layer neurons are equal to 6 that of number of classes.

4.1.4 Levenberg-Marquardt Backpropagation Optimization

Standard BP is a gradient descent algorithm, as is the Widrow-Hoff learning rule, in which the network weights are moved along the negative of the gradient of the performance function. It works on

$$x_{k+1} = x_k - \alpha g_k \quad \text{......(4.4)}$$

Wherein (4.4) x_k is vector of current weights, g_k is current gradient and α is learning rate. In batch mode training all the inputs are applied to the network and the gradients calculated at each training example are added together to determine the change in the weights and biases before the weights are updated in the direction of the negative gradient of mean square error. Standard Gradient Decent (GD) algorithm is too slow for practical problems. As in [13] Levenberg proposed algorithm is the blend of GD and Newton technique and the weight update rule is

$$x_{k+1} = x_k - (H+\mu I)^{-1} g \quad \text{......(4.5)}$$

Wherein (4.5) $H = J^T J$ and $g = J^T e$

Where J is the Jacobian matrix that contains first derivatives of the network errors with respect to weight and biases, g is the average error gradient, I is the identity matrix, μ is weight damping factor, and H is approximation to the hessian (matrix of mixed partials) which is obtained by averaging outer products of the first order derivative (gradient).

Steepest Decent type method used until the approach a minimum and gradual switch over to the quadratic approximation. The algorithm adjusts μ according to wheather error increasing or decreasing. If error increases as a result of the update then retracts the step means reset the weights to previous values and increase μ by set increasing factor and try for an update again. The step accepted if error decreases as result of update. Marquardt improved this method by replacing the identity matrix (I) by diagonal of the Hessian with a view insight to use benefit of hessian when μ is high, by scaling each component of the gradient according to the curvature, results in larger movement along the directions where the gradient is smaller so the classic error valley does not occur. The resulting update rule is

$$x_{k+1} = x_k - (H+\mu \mathrm{diag}[H])^{-1} g \quad \text{......(4.6)}$$

For moderately sized models (few hundred parameters) LM is much faster than gradient descent [14]. We have used 'trainlm' Matlab function for LM optimization [15].

4.1.5 Early Stopping Generalization

NN learns during learning to approximate behaviour adaptively from training data while generalization is the ability to predict well beyond the training data. Overfitting in complex model such as NN can occurs when a model begins to memorize training data rather than learning to generalize from trend [16]. We have used early stopping to improve generalization of NN and to make it robust.

The usual approach for evaluating the generalization performance of an ANN is to divide the available data into three subsets training, validation and testing [17]. The second subset is the validation set. In early stopping error computed for a validation set at the same time that the network is being trained with the training set. Early stopping is performed to avoid the case when the MSE might decrease in the training set but increases in the validation set. This happens when the network starts memorizing the training patterns. Thus network must be instructed to stop the training when the above situation occurs. When the validation error increases for a specified number of iterations the training is stopped, and the weights and biases at the minimum of the validation error are returned. Test subset is independent to check network generalization [15]

4.2 Classification Results Obtained Using MLPNN for Simulation Data Sets

The neural networks are designed to match an arbitrary function by reducing an appropriate error measure usually defines as sum of squares of the errors [18]. Neural networks of such types have been shown to be universal approximators and they can fit any function to an arbitrary accuracy if their structure is sufficiently large.

As diagnosis tools they are trained with exactly the same target values as for instance the polynomial classifier which means that they also approximate the class conditional posterior probability. The classifier needs to map the relationship between computed or measured symptoms and the fault indicators with binary desired value. The classification methods usually compute a value between 0 and 1, an additional maximum operation is needed and that used to determine fault diagnosis result in induction motor external faults classification methods. A MLPNN model with insufficient or excessive number of neurons in the hidden layer most likely cause the problems of bad generalization and overfitting. The determination of appropriate number of hidden layers is one of the most critical tasks in neural network design. As referred above, there is no analytical method for determining the number of neurons in the hidden layer and therefore it only found by trial and error. Several single hidden layer neural network configurations were tested with growing neurons to find the best generalized neural network configuration using trial and error method. For that, the training data sets are divided in 2 parts training and validation subsets. Early stopping is used to stop the neural network training. ANN training is stopped when validation error is found increasing while training error decreasing for consecutive 6 epochs. The fault identification target output is shown in Table 4.1 for MLPNN training. Validation subset classification accuracy, train subset classification accuracy, and validation error are considered and compared to find best generalized well trained ANN configuration. Each ANN configuration is tested atleast 20 times with reinitialized weights and biases. All configurations are also tested with independent test data sets. The results of validation accuracy, validation error, train accuracy and also train error for different configuration are shown in Table 4.2. It is observed from Table 4.2 that the MLPNN structure 6-10-6 have highest validation subset, train subset, independent test classification accuracy and least validation subset error. The MLPNN structure 6-10-6 is considered as best generalized configuration. In single hidden layer MLPNN configurations the number of input layer neurons are selected 6 same as number of input variables and output neurons equal to six same as six output classes.

TABLE 4.1

Target Output

Output Condition	**Target output**					
VUB	1	0	0	0	0	0
SP	0	1	0	0	0	0
UV	0	0	1	0	0	0
OV	0	0	0	1	0	0
OL	0	0	0	0	1	0
N	0	0	0	0	0	1

TABLE 4.2

MLPNN Configurations Validation Accuracy, Validation Error, Train Accuracy and Train Error With Different Hidden Neurons

Hidden Neurons	**Validation Error**	**Validation Subset Classification Accuracy**	**Training Error**	**Train Subset Classification Accuracy**	**Independent Test Classification Accuracy**
5	0.0354	92.3	0.0115	95.4	97.8
6	0.0288	92.3	0.00083	97.7	97.8
7	0.0213	94.2	0.0089	97.1	95.7
8	0.0202	94.2	0.0014	97.7	97.8
9	0.0233	92.3	0.00029	97.7	95.7
10	**0.0087**	**96.2**	**0.0027**	**98.9**	**97.8**
11	0.0177	94.2	0.00597	98.3	97.8
12	0.0238	92.3	0.00192	97.7	97.8

4.2.1 Results Obtained for Test Patterns of Table 3.3 Using Best Generalized MLPNN Configuration

Some of the test patterns for different output conditions are shown in Table 3.3 of previous chapter. The best ANN configuration output is evaluated using programme in MATLAB environment and shown in Table 4.3 for the respective test input.

Fig. 4.2 to Fig. 4.9 are the MLPNN outputs obtained through simulation after connecting best generalized MLPNN network configuration with simulation diagram for the test cases of Table 3.3. The MLPNN layer details used in simulation are shown in APPENDIX E. The MLPNN refresh rate is set at 0.1 second. In case of external faults identification using MLPNN, It is observed through simulation that ANN can detect all kind of faults occurs 1 cycle before its refreshing rate. This fault identification can further convey to call for alarm or tripping.

TABLE 4.3

Output Results Obtained for Test Patterns of Table 3.3 using Best MLPNN Configuration

Sr. No. of Test Pattern Mention in Table 3.3	VUB	SP	UV	OV	OL	N
1	0.02	0	0	0	0	0.93
2	0	0	0.03	0	0	0.91
3	0.02	0	0	0	0	0.96
4	0	0	0	0	1	0
5	0	0	0	1	0	0.21
6	0.01	0	0.89	0	0.02	0
7	0	0.99	0	0	0	0
8	0	1	0.08	0	0	0
9	0	1	0.01	0	0	0
10	1	0	0	0	0.08	0
11	1	0	0	0	0	0

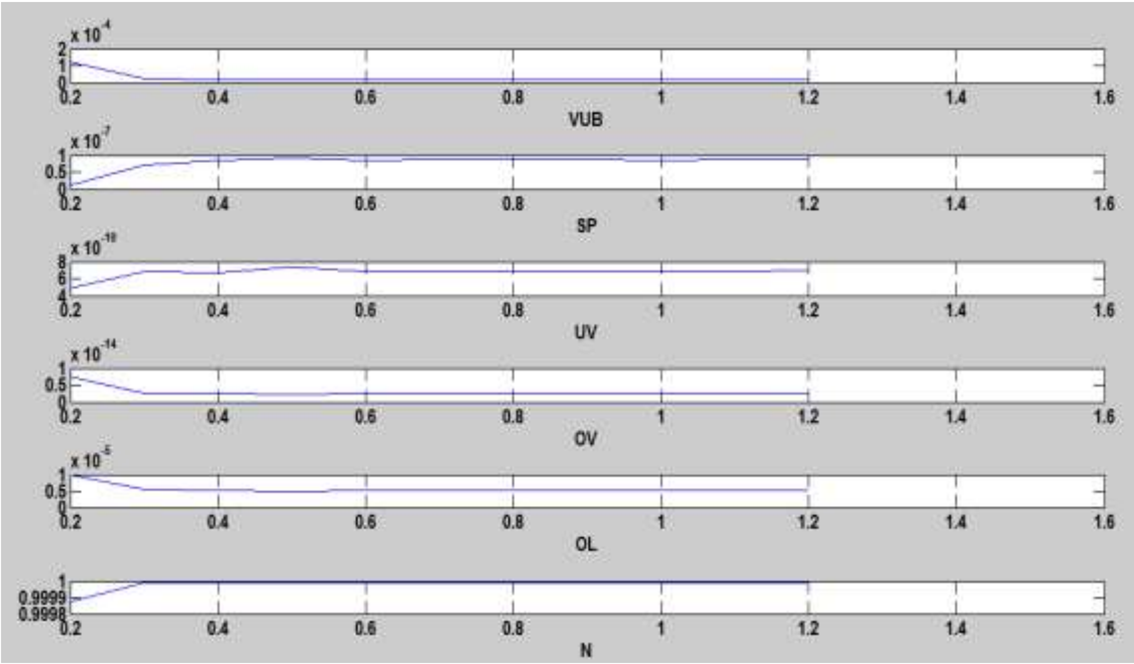


FIGURE 4.2

ANN Output Status for Normal Condition (Sr. No. 1 of Table 3.3)

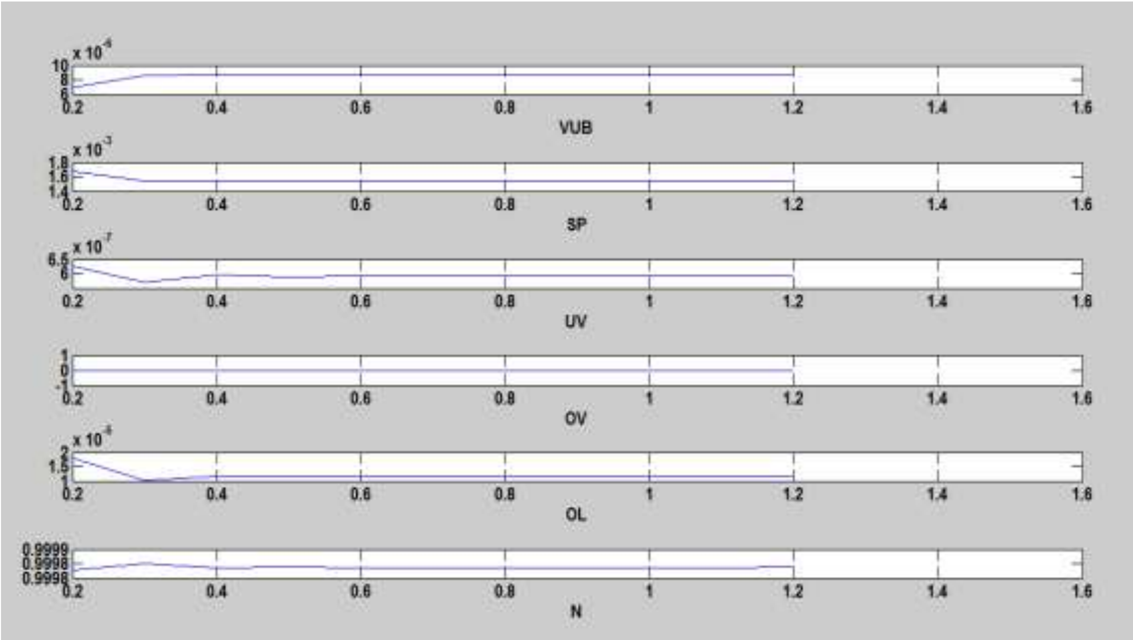


FIGURE 4.3

ANN Output Status for Normal Condition With 92% UV (Sr. No. 2 of Table 3.3)

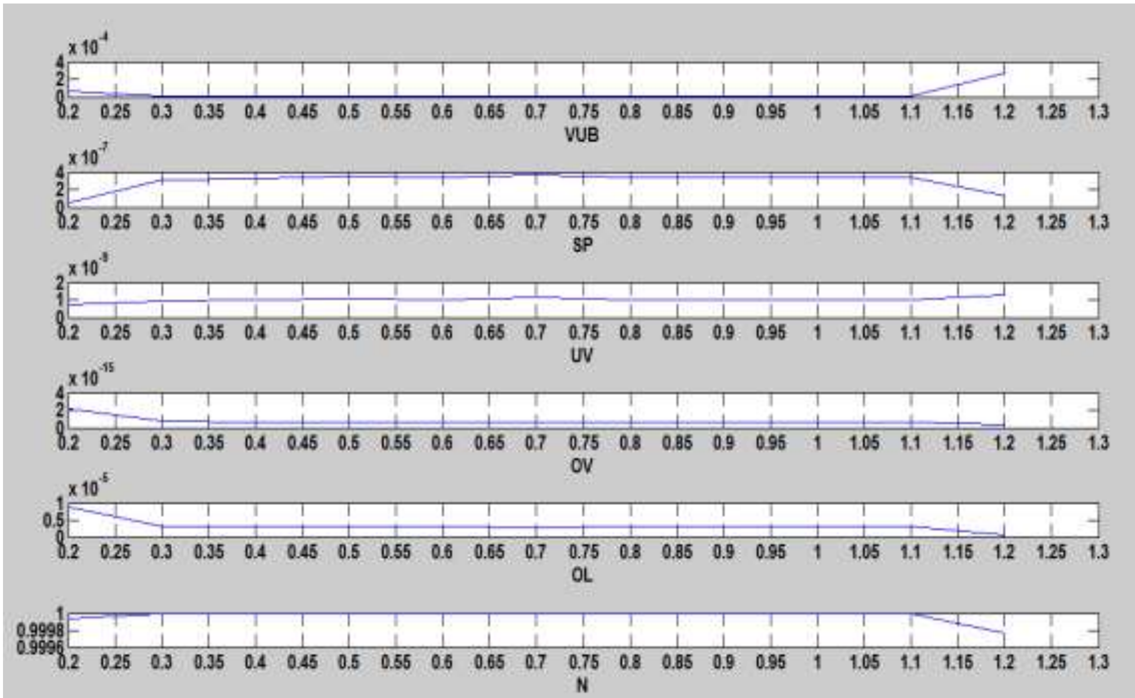


FIGURE 4.4

ANN Output Status for Normal Condition With 0.52% VUB (Sr. No. 3 of Table 3.3)

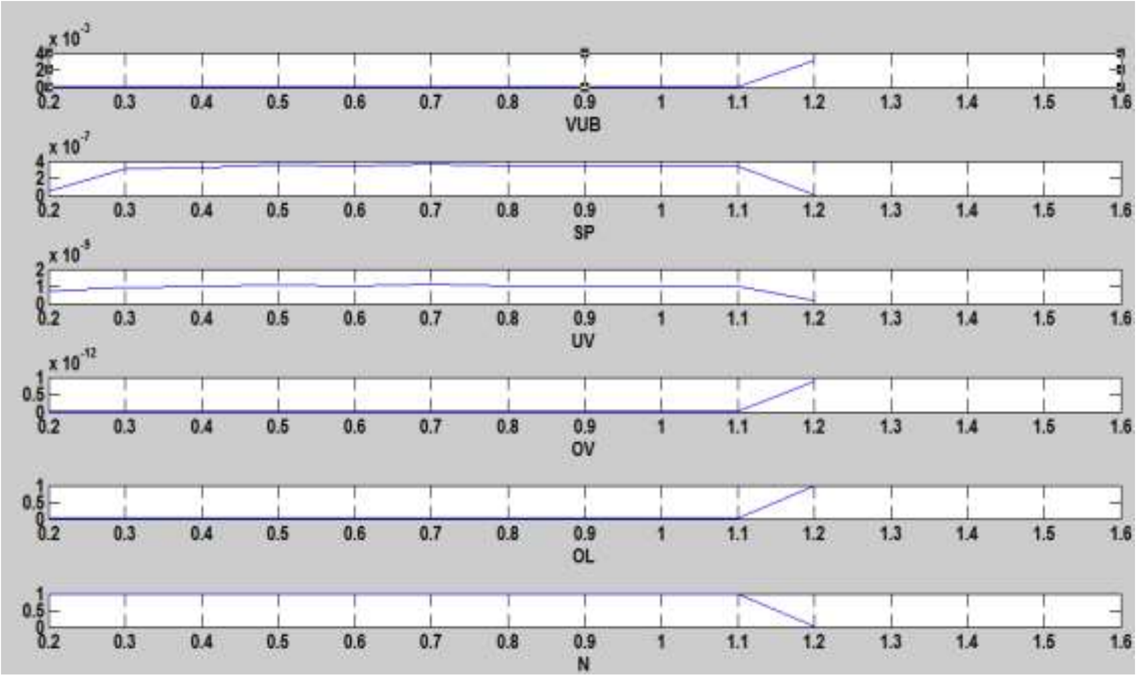


FIGURE 4.5

ANN Output Status for OL condition (Sr. No. 4 of Table 3.3)

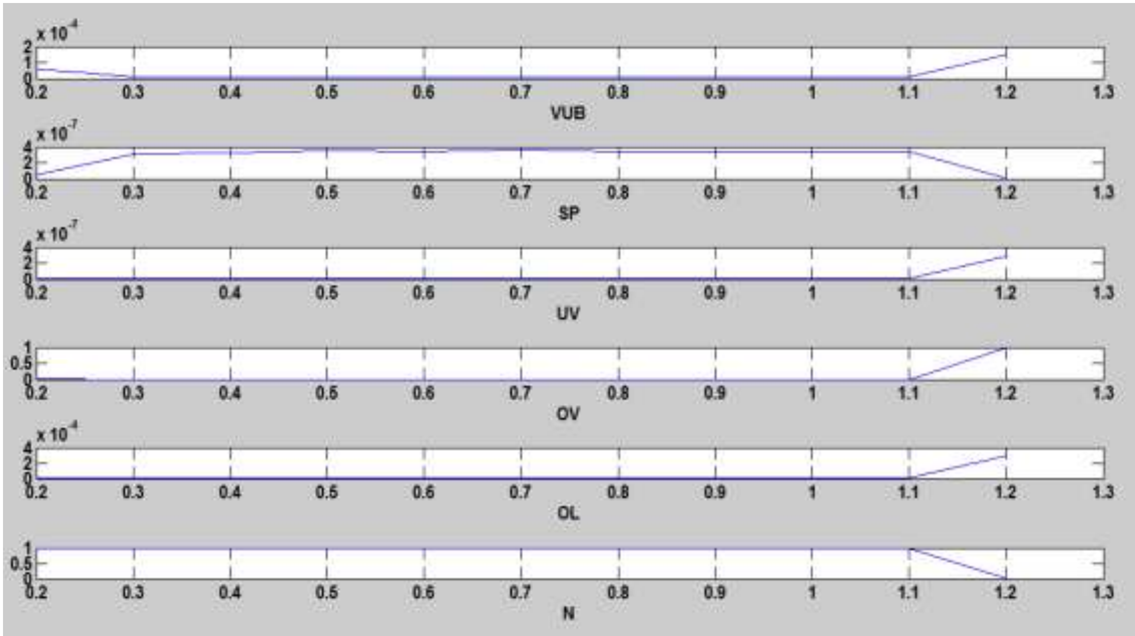


Figure 4.6

ANN Output Status for (OV condition) Sr. No. 5 of Table 3.3

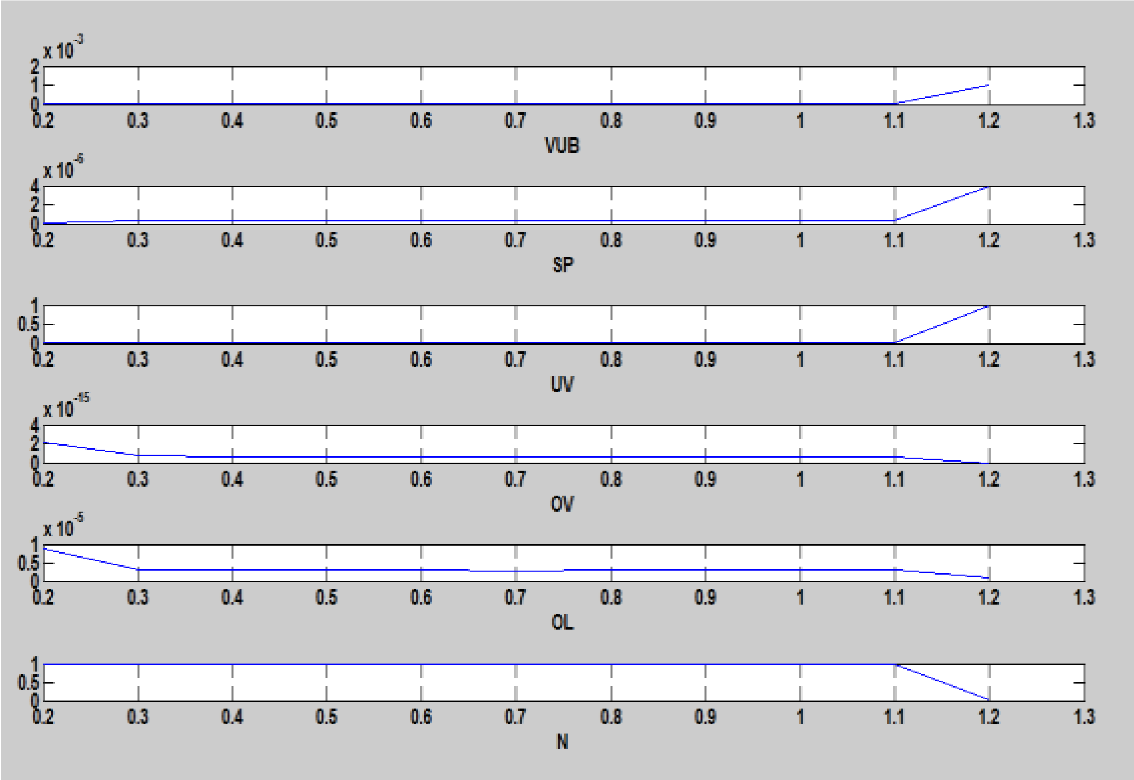


FIGURE 4.7

ANN status for UV condition (Sr. No. 6 of Table 3.3)

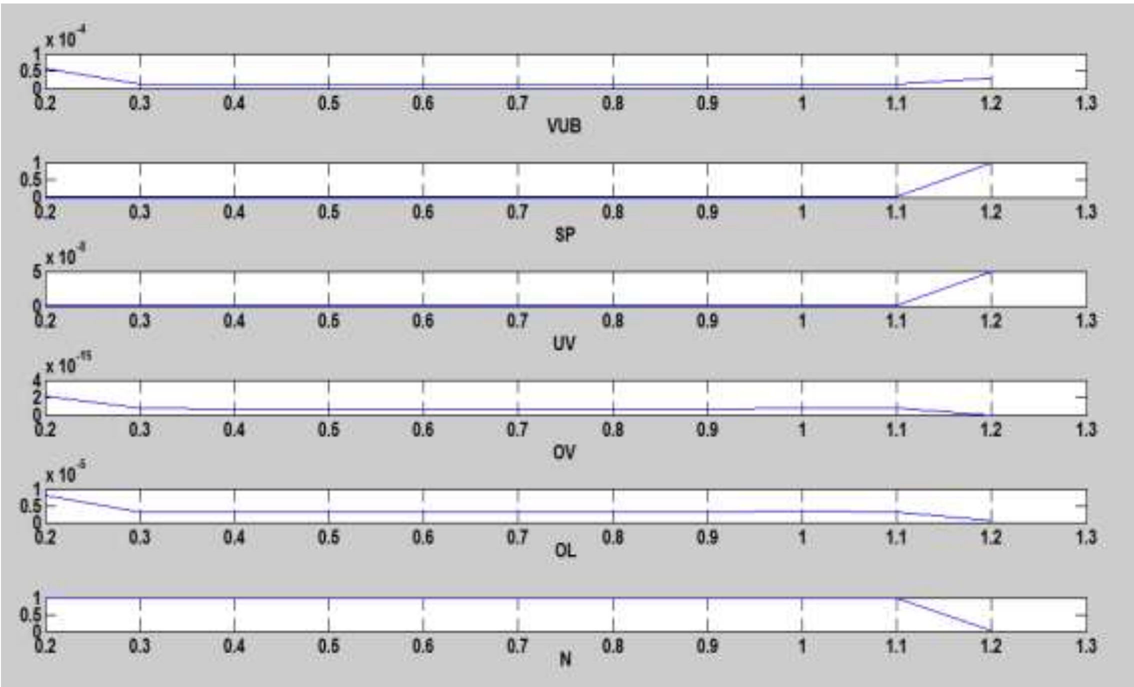


FIGURE: 4.8

ANN Output Status for SP condition (Sr. No. 9 of Table 3.3)

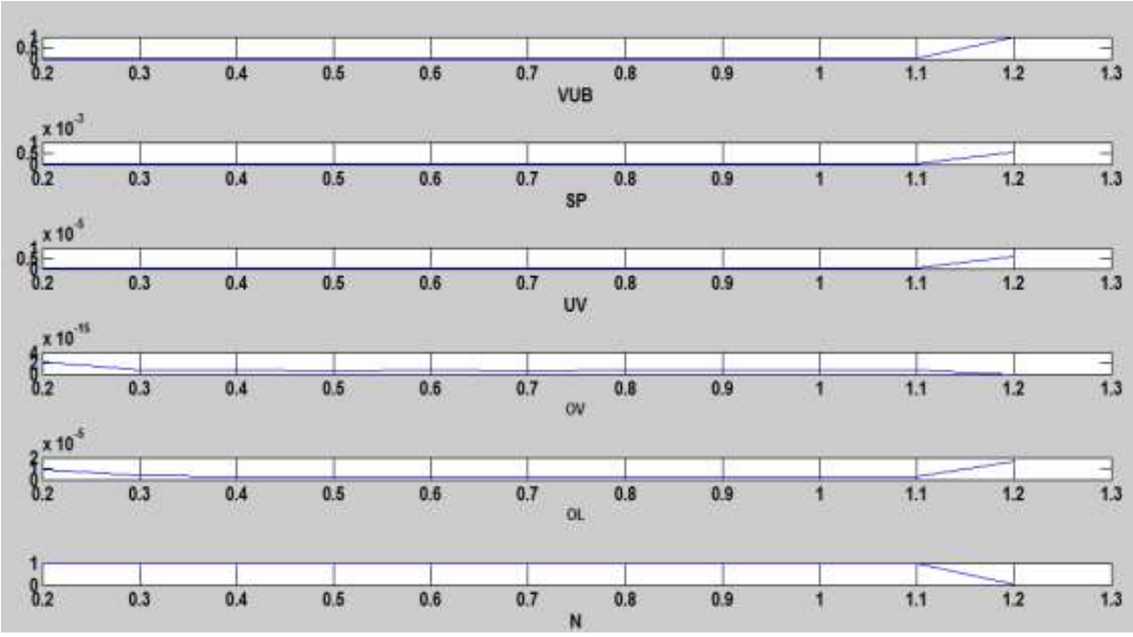


FIGURE: 4.9

ANN Output Status for VUB condition (Sr. No. 10 of Table 3.3)

References

1. Rajasekaran S, Vijayalakshmi Pai GA (2007) Neural Networks, Fuzzy logic, and Genetic Algorithms Synthesis and Applications. Prentice Hall of India, New Delhi.

2. Mehrotra K, Mohan CK, Ranka S (1997) *Elements of Artificial Neural Networks*. Penram International Publishing Pvt. Ltd, India.

3. Kharat PA, Dudul SV (2012) 'Daubechies Wavelet neural network classifier for the diagnosis of epilepsy', WSEAS transactions on biology and biomedicine, Vol. 9, issue 4, pp. 103-113, ISSN: 1109-9518.

4. Dubey A (2013) A study of classification techniques using soft computing. PhD Thesis, Guru Ghasidas Vishwavidyalaya, Available: http://shodhganga.inflibnet.ac.in/bitstream/10603/11748/10/10_chapter_04.pdf ,[Accessed 7 July 2014]

5. Qiang S, Gao XZ, Zhuang X (2003) 'State-of-the-art in soft computing-based motor fault diagnosis', *Proceedings of 2003 IEEE Conference on Control Applications (CCA 2003)*, Vol.2, pp.1381-1386.

6. Umut O, Mahmut H, Mahumut O (2011) EEG signals classification using the k-means clustering and a perceptron neural network model, Expert system with applications Vol. 38, pp. 13475-13481.

7. Hornik K (1989) ‘Multilayer feedforward networks are universal approximates’, Neural Networks, Vol. 2, pp. 359-366, ISSN: 0893-6080.

8. Jawadelar A, Paraskar S, Jadhav S, Dhole G (2014) ‘Artificial neural network-based induction motor fault classifier using continuous wavelet transform’, System Science and Control Engineering, Vol. 2, pp. 684-690, ISSN: 2164-2583.

9. Kotsiantis SB (2007) ‘Supervised machine learning: A review of classification techniques’, informatica, Vol. 31, pp. 249-268, ISSN: 0350-5596.

10. Kon. MA, Plaskota L. (2000) ‘Information complexity of neural networks’, Neural Networks. Vol. 13, pp. 365-375, ISSN: 0893-6080.

11. Haykin S (2008) *Neural Networks and learning machines (3rd ed.)* Pearson Education, New Jersy, USA.

12. Oğulata SN, Şahin C, Erol R (2009) ‘Neural Network-Based computer aided diagnosis in classification of primary generalized epilepsy by EEG signals’, Journal of Medical Systems, Vol. 33, pp. 107-112, ISSN: 0148-5598

13. Rowies S. (1999) Levenberg _Marqardt Optimization. Available: http:// WWW. Cs. Nyu. edu/~rowies/note/lm. Pdf [Accessed 12 December 2012]

14. Ranganathan A (2004) The Levenberg-Marquardt Algorithm, Technical Report, Honda Research Institute, Available: http:// www. ananth.in /docs/ lmtut.pdf.

15. Matlab 7.10 Mathworks Inc. (2010).

16. Peter Zhang G. (2000) ‘Neural Networks for classification: A Survey’, IEEE Transactions on Systems, Man and Cybernetics – Part c: Applications and Reviews, Vol. 30, No. 4, pp. 451-461.

17. Hessami M, Francois A, Viau A. (2004) ‘Selection of Artificial Neural Network Model for the post – calibration of wheather Radar Rainfall estimation’, Journal of Data Science, Vol. 2, 2004, pp.107-124, ISSN: 1683-8602.

18. Isermann R (2006) Fault diagnosis with classification methods. In: Isermann R Fault *diagnosis systems,* Springer Berlin Heaidelberg, pp. 295-310.

CHAPTER - 5

Experimental Setup for Induction Motor External Faults Real-Time Data Sets

5.1 Experimental Setup

Five external faults and normal conditions are experimentally created on operational three phase induction motor. Real time RMS three phase voltages and currents data sets are obtained for the external faults detection and classification using Logger. 3 Phase, 2.2 kW/3 HP, 415V, 4.7A, 50 Hz, 1430 rpm induction motor coupled with 3 HP 220V, 1500 rpm separately excited dc generator set is used for the experiment. The induction motor is supplied through three phase autotransformer. DC generator is loaded with rheostat setup. Average RMS values of three phase voltage and currents are logged at the time interval of 0.5s using logger of sample rate 10.4 kHz. A representative set is prepared for the training and testing of classifiers. Fig. 5.1shows the block diagram of experimental setup and Fig. 5.2 and Fig. 5.3 shows the experimental setup and its details.

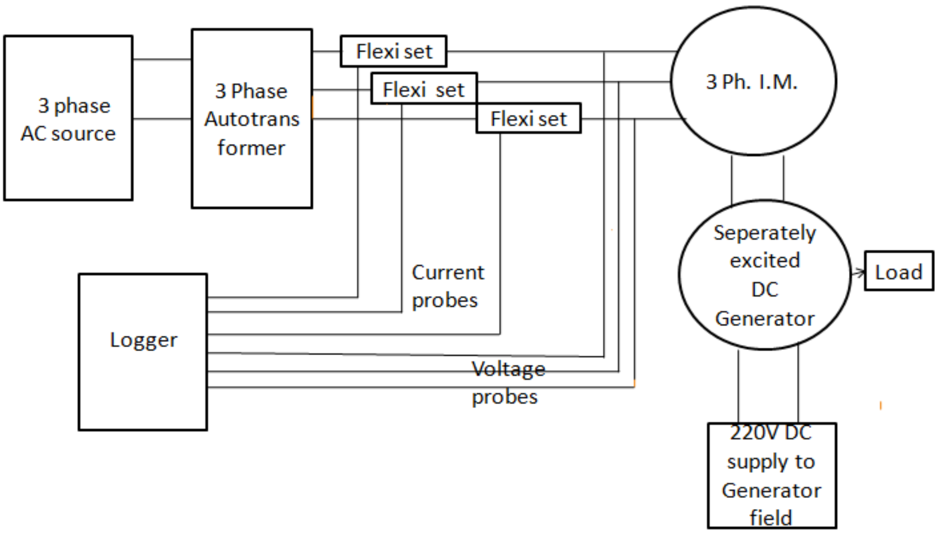


FIGURE 5.1

Experimental Block Diagram

FIGURE 5.2

Experimental Setup

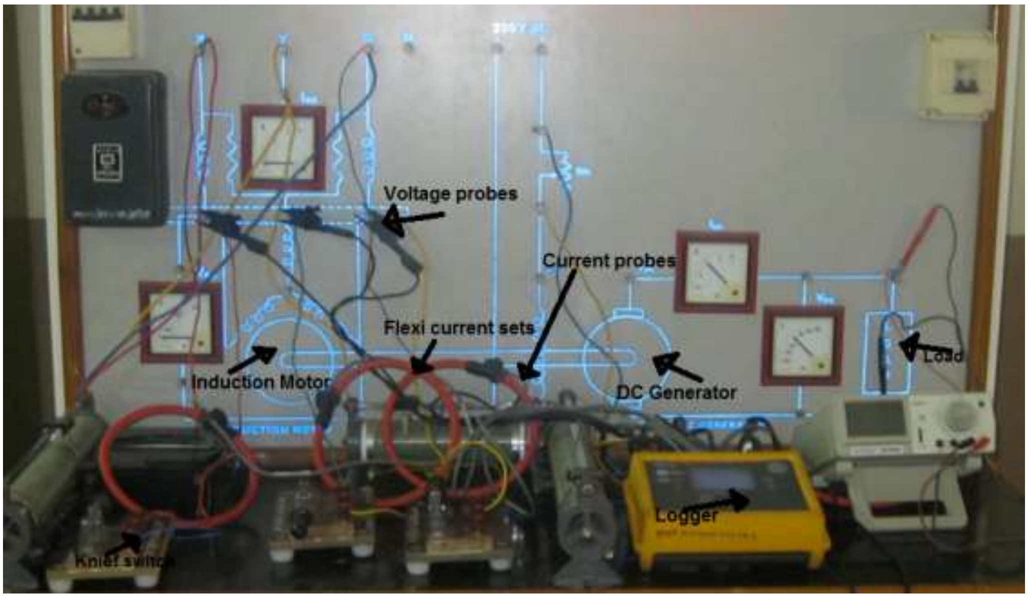


FIGURE 5.3

Experimental Setup Details

5.2 Power Log PC Software and Fluke 1735

Power Log is PC software for fluke 1735 and it's some other versions. The software accepts data downloaded from fluke logger. After transferring logged data to a PC for graphical and tabular evaluation, data also can be exported to spread sheet. With the use of fluke 1735 logger voltage, current and power studies can be conducted. By selecting logging menu in parameter configuration averaging time can be adjusted. When a flexi set or current probe is connected to the instrument it is automatically recognized, but only at power up. In current probes or flexi set submenu in instrument setup current measuring range can be selected with current transformer or without current transformer. In Power network submenu of instrument setup menu power type like single phase, three phase star or delta can be selected with nominal phase voltage and frequency. With the use of logging function Record/Measure button logging function can be started. Power Log PC software provides data downloaded, analysis and reporting in one package.

5.3 Experimental Results for Normal and Different External Faults Condition

Fig. 5.4 shows the real time logged three phase voltage and current waveform for normal, OL, OV and UV conditions and Fig. 5.5 shows for SP conditions respectively. Similarly the real time data sets are logged for VUB conditions and shown in Fig 5.6. Matlab scatter plot for practical data sets is shown in Fig. 5.7. L12, L23, L31 are three phase RMS line voltages and L1, L2 and L3 are three phase RMS line currents in Fig. 5.4, Fig. 5.5 and Fig. 5.6. OV and UV conditions are created using 3 phase autotransformer arrangement. OL is created using increasing load on dc generator through rheostatic load arrangement. VUB conditions are created using rheostats in two phases and single phasing by use of knife switch in each phase. Data sets for normal conditions are obtained with load varying from 50% to 105% and voltage within ±10% variations. While above 109.5% and less than 90.5% rated operating voltages are taken for overvoltage and undervoltage condition. Overvoltage condition data sets are taken upto nearly 113 % of rated operating voltage and undervoltage upto 86 % of rated operating voltages is considered. Similarly overload is considered from 105% to 120% of full load in experimentation.

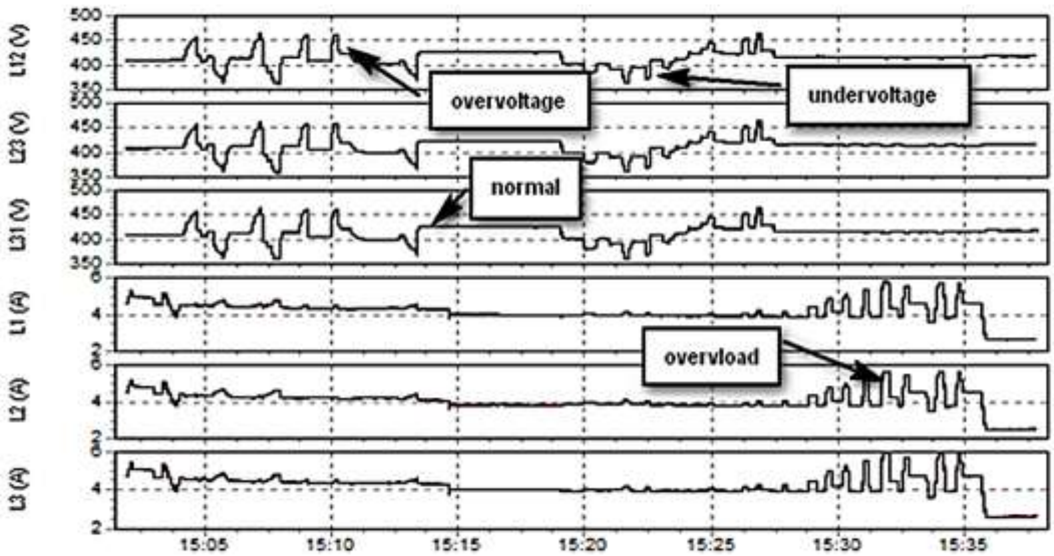


FIGURE 5.4

Three Phase RMS Voltages and RMS Currents for Normal, OL, OV and UV Condition

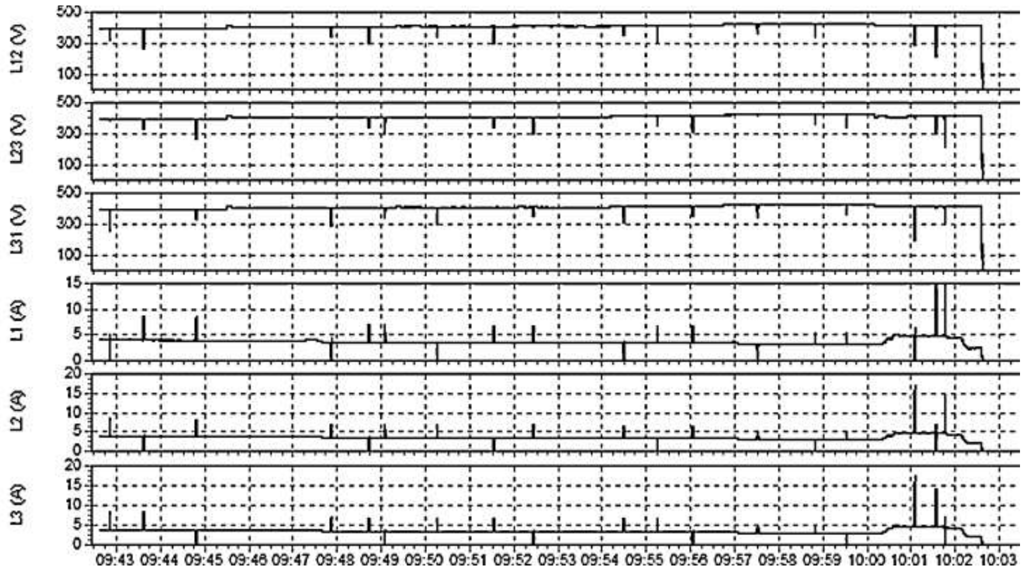


FIGURE 5.5

Three Phase RMS Voltages and RMS Currents for SP condition

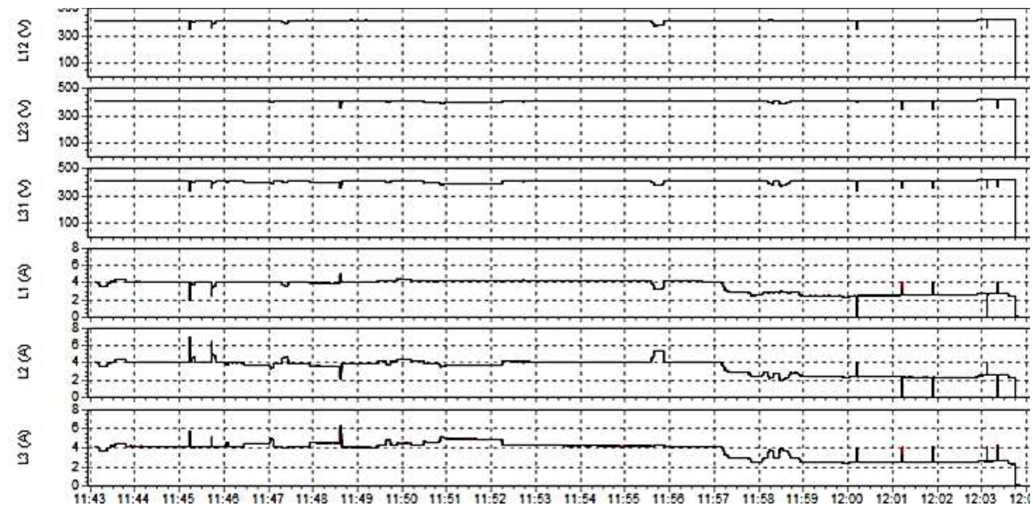


FIGURE: 5.6

Three Phase RMS Voltages and RMS Currents for VUB Condition

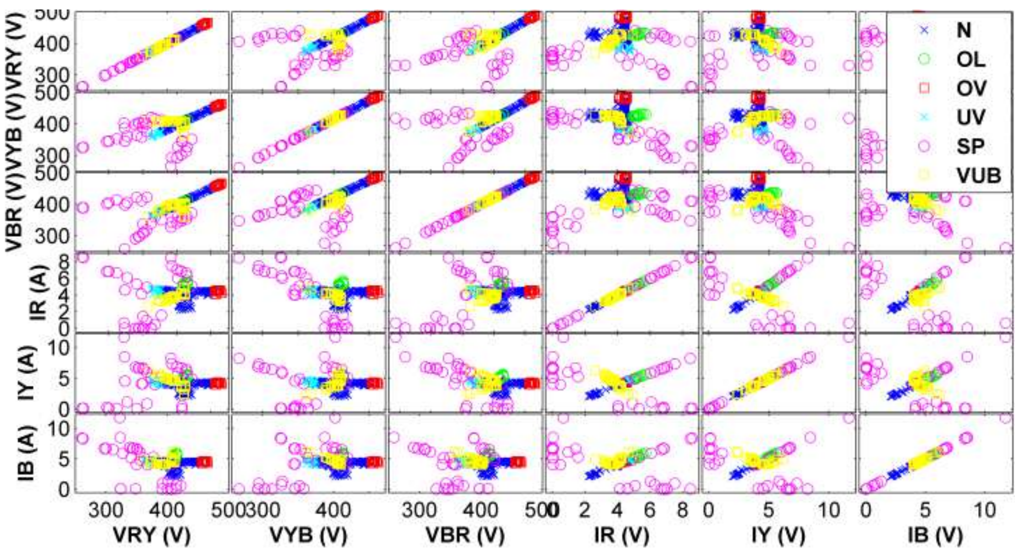


FIGURE 5.7

Scatter Plot of Practical Data Sets

We have prepared 321 representative data sets (patterns) for finding best generalized MLPNN, SC_FIS, PNN and ANFIS configurations in Chapter 6. The details and results for each technique are discussed in Chapter 6. All classifier performance including LDA and NBC is compared in Chapter 8 using statistical measures like total classification accuracy, sensitivity, specificity, precision and overall F-measure using 72 independent test data sets and 321 input train patterns.

The 321 input train and 72 independent test patterns are shown in Appendix-C and Appendix-D respectively. The number of train and test data sets patterns used for different output (normal and external faults) conditions are shown in Table 5.1. Table 5.2 shows the some example patterns of practically obtained independent test (unseen) data sets.

TABLE 5.1

Number of Patterns for Practical Train and Independent Test Data Sets

Sr. No.	Condition	Train Data	Independent Test Data
1	Normal (N)	140	22
2	Overload (OL)	30	10
3	Overvoltage(OV)	30	8
4	Undervoltage(UV)	30	10
5	Single Phasing (SOP)	41	9
6	Voltage Unbalance Condition (VUB)	50	13

TABLE 5.2

Example of Independent Test Inputs for Practical Data Sets

Sr. No.	**Output Condition**	**VRY**	**VYB**	**VBR**	**IR**	**IY**	**IB**
1	N	410.571	408.268	409.445	4.568	4.459	4.691
2	N (VUB within normal limit 0.83 %)	400.055	405.94	401.769	4.036	4.391	4.173
3	OL	414.895	413.257	414.102	5.755	5.659	5.986
4	OV	462.103	459.365	461.412	4.473	4.255	4.527
5	UV	360.702	357.452	359.448	4.759	4.609	4.841
6	SP(R Phase)	351.106	411.287	309.22	0.014	6.436	6.491
7	SP(B Phase)	421.497	341.179	363.798	5.345	5.195	0.014
8	VUB (1 Phase)	388.336	410.161	395.398	3.518	4.909	4.091
9	VUB (2 Phase)	376.975	405.018	384.651	3.45	5.25	4.255

CHAPTER - 6

Induction Motor External Faults Identification Using MLPNN, SC_FIS, PNN, ANFIS, NBC and LDA Classifier for Practical Data Sets

This Chapter deals with external faults identification of induction motor using soft computing (MLPNN, SC_FIS, PNN and ANFIS) and conventional (LDA and NBC) classification methods. The data sets patterns obtained through experimentation are utilized as input feature vector to MLPNN, SS based FIS, PNN, NBC and LDA classifier for evaluating external faults identification performance.

6.1 External Faults Identification Using MLPNN

A MLPNN model with excessive or insufficient number of neurons in the hidden layer most likely cause the problems of bad generalization and overfitting. The determination of appropriate number of hidden layers is one of the most critical tasks in neural network design. There is no analytical method for determining the number of neurons in the hidden layer. Therefore it only found by trial and error [1] [2]. As cited in Chapter 4 one hidden layer is sufficient for feedforward networks to approximate any continuous mapping from the patterns to the output patterns to an arbitrary degree of accuracy. Several single hidden layer neural network configurations were tested with growing neurons to find the optimal neural network configuration using trial and error method. For that, the training data sets are divided in 2 parts training and validation subsets. Early stopping is used to stop the neural network training. ANN training is stopped when validation error is found increasing while training error decreasing for consecutive 6 epochs. The fault diagnostic target output assignment is shown in Table 6.1. The six input variables (three phase RMS voltages and currents) constitute input and six output conditions constitute the output of MLPNN.

Validation subset accuracy alongwith validation error and train subset accuracy are considered and compared to find optimal MLPNN configuration. This comparison is shown in Table 6.2. Each ANN configuration is tested atleast 20 times with reinitialized weights and biases. MLPNN configuration 6-12-6 with 12 hidden neurons is considered as the best generalized and well trained configuration as it has high validation subset classification accuracy, least validation error and also high train subset accuracy. The best generalized MLPNN configuration is used for classifiers comparison with 72 independent test data sets in chapter 7**.**

TABLE 6.1

Target Output

Output Condition	**Target output**					
VUB	1	0	0	0	0	0
SP	0	1	0	0	0	0
UV	0	0	1	0	0	0
OV	0	0	0	1	0	0
OL	0	0	0	0	1	0
N	0	0	0	0	0	1

TABLE 6.2

MLPNN Configurations Validation Accuracy, Validation Error, Train Accuracy and Train Error With Different Hidden Neurons for Practical Data Sets

Hidden Neurons	**Validation Error**	**Validation Subset Classification Accuracy**	**Training Error**	**Train Subset Classification Accuracy**
5	0.0063	99	0.0022	99.1
6	0.0061	99	0.0017	99.7
7	0.0081	99	0.0024	99.7
8	0.0054	97.9	0.0013	99.4
9	0.0046	99	0.00029	99.7
10	0.0068	97.9	0.00035	99.4
11	0.0039	99	0.0047	99.1
12	**0.0029**	**100**	**0.00086**	**100**

6.2 External Faults Identification Using Subtractive Clustering Based Sugeno Fuzzy Inference System (SC_FIS)

Fuzzy logic based system is able to approximate the complex relationship related to diagnosis task [3]. The principle of this theory is to quantify the uncertainties in a given system using MFs. Fuzzy clustering methods are one of the strategies implemented to identify these MFs by organizing data samples into clusters so that the data samples within clusters are more similar to each other [4]. A fuzzy logic approach helps in accurate diagnoses induction motor faults and is also able to extract the heuristics related to diagnosis of faults.

6.2.1 Fuzzy logic and systems

6.2.1.1 Fuzzy Logic

In crisp logic, the truth values acquired by propositions or predictors are two valued namely true or false which may be treated numerically equivalent to (0 1). However in fuzzy logic truth values are multivalued and are numerically equivalent to (0-1) [5].

6.2.1.2 Fuzzy Set

If X is universe of discourse and x is a particular element of X, then fuzzy set 'A' defined on X may be written as a collection of ordered pairs

$$A = \{(x, \mu_A (x), x \in X\} \qquad (6.1)$$

Wherein (6.1) each pair (x, μ_A (x)) is called singleton. μ_A (x) is the membership function and associated with a fuzzy set A such that the function maps every element of the universe discourse X to the interval [0 1] [5].

6.2.1.3 Fuzzy Logic Proposition

A fuzzy logic proposition 'P' is a statement that involves some fuzzy concepts. Linguistic statements that tend to express subjective ideas typically involve fuzzy propositions. The truth value assigned to P can be any value on the interval [0 1]. Suppose proposition P is assigned to fuzzy set A; then the truth value of a proposition denoted T(P) is given by

$$T(P) = \mu_A (x) \text{ where } 0 \le \mu_A \le 1 \qquad (6.2)$$

Equation (6.2) indicates that degree of truth of proposition P: x ∊ A is equal to the membership grade of x in the fuzzy set A [6].

6.2.1.4 Fuzzy Inference System

The fuzzy inference system is a popular computing framework based on the concepts of fuzzy set theory, fuzzy if-then rules, and fuzzy reasoning. It has found successful applications in wide variety of fields such as automatic control, data classification, decision analysis, expert systems, time series prediction, and robotics and pattern recognition. Because of multidisciplinary nature the fuzzy inference system is known by numerous other names such as fuzzy rule based system, fuzzy model, fuzzy expert system, fuzzy associative memory, fuzzy logic controller and simply fuzzy system.

Fuzzy inference system can take either fuzzy inputs or crisp inputs and output it produce almost always fuzzy. Sometime it is necessary to have crisp outputs according to need of applications. Therefore a method of defuzzification is needed to extract crisp value that best represents a fuzzy set. A fuzzy inference system with a crisp output is shown in Fig. 6.1 where the border line indicates a basic fuzzy inference system with fuzzy output and defuzzification block transforms an output fuzzy set into a crisp single value [7].

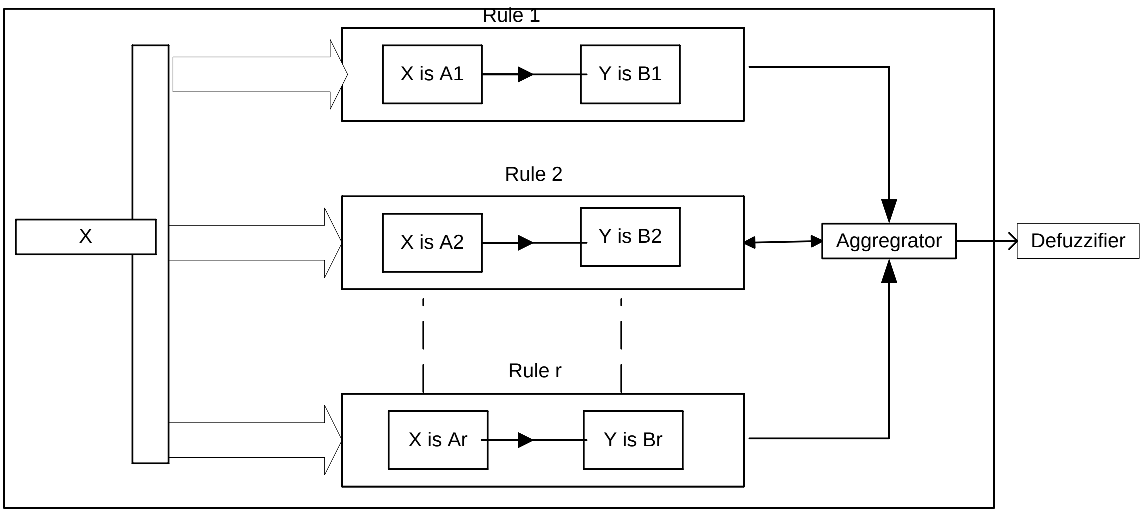


FIGURE 6.1

Block Diagram of Fuzzy Inference System [7]

With the crisp inputs and outputs, a fuzzy inference system implements a nonlinear mapping from its input space to output space. This mapping is accomplished by a number of fuzzy if then rules each of which describes the total behaviour of the mapping. In particular, the antecedent of a rule defines a fuzzy region in the input space, while the consequence specifies output in the fuzzy region. Mamdani and Sugeno type of FISs widely used in various applications. The difference between these two FISs lies in the consequents of their fuzzy rules, and thus their aggregation and defuzzification procedures differ accordingly. Mamdani fuzzy models are based on expert knowledge means well suited to human input. Sugeno FIS is computationally efficient and works well with optimization and adaptive techniques. It has guaranteed continuity of the output space and well suited to mathematical analysis. It is similar to the Mamdani method in many respects. The first two parts of the fuzzy inference process, fuzzifying the inputs and applying the fuzzy operator, are exactly the same. The main difference between Mamdani and Sugeno is that the Sugeno output membership functions are either linear or constant

6.2.1.5 Sugeno Fuzzy Inference System

Sugeno or also known as Takagi-Sugeno-Kang method of fuzzy inference introduced in 1985 [8] in an effort to develop a systematic approach to generating fuzzy rules from a given input-output data set. A typical fuzzy rule in a Sugeno fuzzy model has the form

If x is A and y is B then z = f(x,y)……………………………………………………(6.3)

Wherein (6.3) A and B are fuzzy sets in the antecedent, while z = f(x,y) is a crisp function in the consequent. Usually f(x,y) is a polynomial in the input variables x and y, but it can be any function as long as it can approximately describe the output of the model within the fuzzy region specified by the antecedent of the rule. When f(x,y) is a first order polynomial the resulting FIS is called first order Sugeno fuzzy model. For a first order Sugeno fuzzy model since each rule has a crisp output the overall output is obtained via weighted average, thus avoiding the time consuming process of defuzzification required in mamdani model. Without time consuming and mathematically intractable defuzzification operation, the Sugeno fuzzy model is most popular candidate for sample-data-based fuzzy modelling [7].

6.2.2 Clustering

Natural groupings in data from data sets can be very effectively identify using clustering and so it allows concise representation of relationships embedded in the data. In this case of fault identification clustering allows us to group fault patterns into broad categories and hence provides easier interpretability. The most representative offline fuzzy clustering techniques include mountain clustering, FCM, and subtractive clustering. Mountain clustering relies on dividing the data space into grid points and calculating a mountain function at every grid point which is a representation of density of data at that point. The disadvantage of this algorithm is computation increases exponentially with increased in input data dimension as the mountain function has to be calculated at each grid data point so not suitable for problems of dimension higher than two or three [9].

Several parameters need to specify like number of clusters c, fuzziness component m, termination tolerance, fuzzy partition matrix U in case of FCM. Performance of FCM depends on the initial membership matrix values. So several runs each starting with different values of membership grades of data points probably give good performance in FCM.

6.2.2.1 Subtractive Clustering

Subtractive clustering is a fast one pass algorithm for estimating number of clusters and cluster centers in a set of data when it is difficult to decide how many clusters should be for data set. “Ref [10]” presented substractive clustering algorithm, modified form of mountain clustering method, as the basis of fuzzy identification algorithm and instead of grid point each data point considered as potential cluster centre. In Yeager’s [11] mountain clustering computation grows exponentially with the dimension of the problem because mountain function has to be evaluated at each grid point. As data points used for cluster centers computation is proportional to the problem size instead of problem dimension. It also eliminates the need of specifying grid resolution and does not involve any iterative nonlinear optimization. However the actual centers are not necessarily located at one of the data points, but in most case it is a good approximation.

Since each data point is a candidate for cluster centers, a potential measure at data point x_i is defined as

$$P_i = \sum_{j=1}^{n} \exp\left(-\alpha\left\|x_i - x_k\right\|^2\right) \quad \text{……(6.4)}$$

Wherein (6.4) $\alpha = (r_a/2)^2$ and r_a is positive constant represents a neighbourhood radius. A data point will have a high potential value if it has many neighbouring data points. The first cluster center x_{k1}^* is chosen as the point having largest potential value p_{k1}^*. Next the potential measure of each data point is revised as follows:

$P_i \Leftarrow P_i - P^*_{k1} \exp\left(-\beta \left\| x_{i} - x^*_{k1} \right\|^2\right)$ …………………………………………………………(6.5)

Where $\beta = (r_b/2)^2$.

r_b is a positive constant which defines a neighbourhood that has measurable reductions in potential measure. Therefore the data points near the first cluster center x^*_{k1} will have reduced potential measure and therefore are unlikely to select as the next cluster centre. To avoid obtaining closely spaced cluster centers a good choice is $r_b = 1.5\ r_a$. After revising the potential function according (5), the next cluster center is selected as the point having the greatest potential value and the process continues until a sufficient number of clusters are attained given as

$P_i \Leftarrow P_i - P^*_k \exp\left(-\beta \left\| x_{i} - x^*_k \right\|^2\right)$ ……………………………………………………..……. (6.6)

In Yager and Filev's mountain clustering procedure [11] the process of revising potential repeats until $p^*_k < \varepsilon p^*_1$ where ε is an important factor affecting the results; if ε is too large, too few data points will be accepted as cluster centers and viceversa. Chiu developed additional criteria for accepting and rejecting cluster centers as it was found difficult to establish a single value of ε that works well for data patterns in which two thresholds are utilized one for the potential above which data point accepted and other for threshold below which rejection of data point and for in between two threshold (6.7) is utilized.

$\frac{D_{min}}{r_a} + \frac{P^*_k}{P^*_1} \geq 1$ ……………………………………………………………………….. (6.7)

Wherein (6.7) d_{min} is the shortest distances and p^*_k is location of k^{th} cluster centre between all previously found cluster centers [10].

6.2.3 Subtractive Clustering Based Sugeno Fuzzy Inference System (SC_FIS)

Subtractive clustering was applied to extract the rules for identifying each class of data after the input and output were assigned. The clusters found in the data sets of a given group identify regions in the input space that map into associated class so it can translate each cluster centre into a fuzzy rule for identifying the class [12]. The fundamental feature of clustering based rule extraction method of rules generation helps avoid combinatorial explosion of rules with increasing dimension of input space [13].

As discussed in [10], cluster center x_i^* was considered as fuzzy rule that described the system behaviour. A set of c cluster centers $\{x_1^*, x_2^*, \ldots, x_c^*\}$ were considered in M dimension space and among them first N dimensions corresponded to input variables and last M-N corresponded to output variables. Each x_i^* vector was decomposed into two component vectors y_i^* and z_i^* , where y_i^* contained the first N elements of x_i^* (i.e the coordinates of cluster center in input space) and z_i^* contained M-N elements (i.e., the coordinates of the cluster centers in outspace).

The computational model was viewed as fuzzy inference system which employed each rule in following form.

If Y_1 is A_{i1} & Y_2 is A_{i2} &. . . then Z_1 is B_{i1} & Z_2 is B_{i2}. ……………………………………(6.8)

Where in (6.8) Y_j is the j^{th} input variable, Z_j is the j^{th} output variable and A_{ij} is the exponential membership function in the i^{th} rule associated with j^{th} input. B_{ij} is a singleton membership function in the i^{th} rule associated with j^{th} output centered around Z_{ij}^*. The membership function A_{ij} is given by (6.9).

$$A_{ij}\,(Y_j) = \exp\left\{-\frac{1}{2}\left(\frac{Y_j-Y_{ij}^*}{\sigma_{ij}}\right)^2\right\} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(6.9)}$$

Where in (6.9) Y_{ij}^* is the j^{th} element of y_i^* and $\sigma_{ij}^2=2/(r_a)^2$. r_a is positive constant represents a neighbourhood radius in subtractive clustering.

This computational scheme is equivalent to an inference method that uses multiplication as the AND operator, weights the consequence of each rule by the rule's degree of fulfilment, and computes the final output value as weighted average of all the consequences. For the optimization of rules, Z_{ij}^* (Z_{ij}^* is the j^{th} element of Z_i^*) was considered as a linear function of the input variables, instead of constant, as

$$Z_{ij}^* = G_{ij}\, y + h_{ij} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(6.10)}$$

Where in (6.10) G_{ij} is N element vector of coefficients and h_{ij} is a scalar constant. The if-then rule then becomes the Takagi-Sugeno type.

As in [8], given a set of rules with fixed premise membership functions, optimizing G_{ij} and h_{ij} in all consequent equations is a simple linear least square estimation problem. Chiu in [10] used this approach to optimize the rules obtained from the subtractive clustering method. Optimizing only the coefficients in consequent equations allows a significant degree of model optimization to be performed without adding much computational complexity. We have adopted the similar approach proposed by [10]. Subtractive clustering based FIS approach is used for external faults classification using genfis2 in MATLAB environment with own written codes. By applying subtractive clustering to each class of data sets, a set of rule can be obtained for identifying each class. The combined form of individual sets of rules form the rule base classifier. For example suppose we found 2 cluster centers in class C1 data sets and 3 centers in class C2 data sets the rule base will contain 2 rules that identify class C1 members and 3 rules that identify class C2 members [14].

FIS is used to collect all of fuzzy rule base to set up the crisp output. From these fuzzy rules, the membership of each data on each cluster can also be performed, and antecedent of each rule can be quantified. This quantification process for each rule produces weight for each fuzzy rule base to set the fuzzy output for each class. In this work, each class output for any pattern is calculated using weight-average method and the higher of class outputs represents final output condition for the any pattern [4].

A FIS is composed of inputs, outputs, and rules. Fig. 6.2 shows an example of subtractive clustering based FIS network for six class classification. Three phase voltages and three phase line currents are inputs to FIS and six output conditions (Five external faults and normal) constitute the output of FIS. Each input and output may have any number of membership functions (MFs) decided by clustering radius selected. Gaussmf is used as input variables MF for FIS. The rules dictate the behaviour of the fuzzy system based on inputs, outputs and MFs. The parameters of subtractive clustering were chosen as follows: squash factor 1.25, accept ratio 0.5 and rejection ratio 0.15. Clustering and FL together provide a simple and powerful means to model the fault relationship.

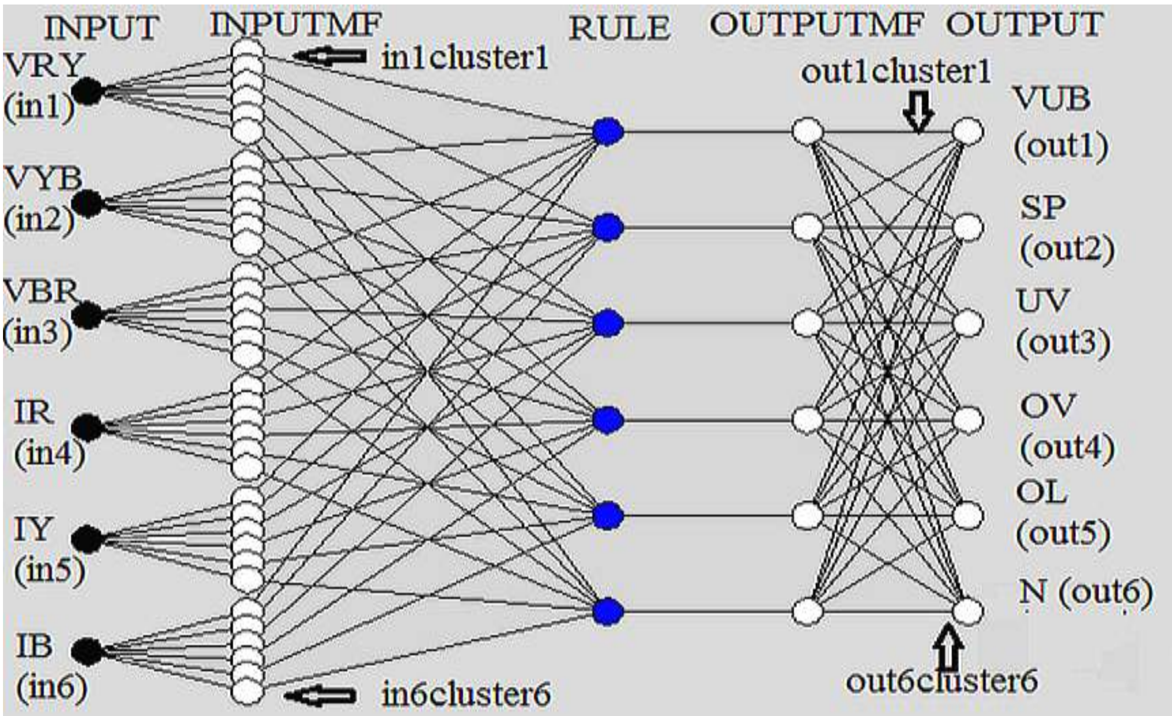


FIGURE 6.2

Subtractive Clustering Based FIS Network for Six Class Classification

6.2.4 Classification Results and Rules Obtained Using SC_FIS

We measure the performance of fault identification by 10-fold random subsampling cross validation of total train input 321 data sets. In 10-fold random subsampling cross validation method, data sets are splitted in train (75%) and test (25%) fold. Train data sets are used to construct FIS model and test data sets are used to evaluate the model. Table 6.1 shows target output assigned to FIS during training. Total average test classification accuracy and average RMSE of 10 fold testing data sets are used as performance measures to assess the performance of this method for different output condition classification.

The output layer of subtractive clustering based FIS system shown in Fig. 6.2 produce net output fuzzy vector for its each class of input. The maximum of the output fuzzy vector represents the final class as winner takes all condition. A compete transfer function is used which produces 1 for that class and 0 for other classes. The output of FIS for a particular condition (fault or normal) is close to 1.0 (usually in range of 0.5-1.0) while the other outputs are close to 0.0 (usually in range of 0.0 - 0.5). Table 6.3 shows total average classification accuracy and RMSE error obtained for train and test data sets of 10 times random subsampling cross validation data sets with different cluster radius. We have attempted to find best generalized configuration which have highest test classification accuracy and also least RMSE error and least possible rules. It is observed from Table 6.3 that the error with respect to experimentally obtained testing data diverges significantly when the cluster radius less than 0.2, showing model is overfitting.

TABLE 6.3

Total Average Classification Accuracy, Average RMSE Error and Rules for FISs Obtained With Different Cluster Radius for Practical Data Sets

Cluster Radius	Average Train Accuracy (%)	Average Test Accuracy (%)	Average Test RMSE Error	Average RMSE Train Error	No. of Rules
0.1	99.87	90	32300	0.1	22
0.2	98.17	92.75	498.32	0.1696	14
0.3	96.87	93.88	0.711	0.211	8
0.4	96.62	92.25	0.608	0.244	7
0.5	**96.63**	**94.37**	**0.4032**	**0.276**	**6**
0.6	96.42	93	0.703	0.291	5
0.7	96.46	90.88	0.492	0.294	5
0.8	96.67	92.12	0.74	0.3	5
0.9	96.5	91.87	0.54	0.3	5

The best generalized FIS is obtained for practical data sets is with cluster radius 0.5 that have highest average test classification accuracy and least RMSE test error. The FIS obtained have 6 clusters, 6 premise MFs and 6 rules. A fuzzy classification rule R_i which describes the relation between the input feature space and classes obtained as,

R_i: If X_{p1} is φ_{i1} and _ _ and X_{pj} is φ_{ij} and_ _ and X_{pn} is φ_{in} then Out_{p1} is φ_{i1} and_ _ and Out_{ps} is φ_{iv} and _ _Out_{pm} is $\varphi_{im.}$ ……………………………………………………………...(6.11)

Where X_{pj} denotes the j^{th} input variable of p^{th} sample; n represents the number of inputs; φ_{ij} denotes the fuzzy set of the j^{th} variable in the i^{th} rule and characterized by the appropriate membership function. Out_{ps} represents s^{th} output class of p^{th} sample; m represents number of output class; φ_{iv} denotes the fuzzy set of the v^{th} output class in i^{th} rule. Similarly the rule R1 of Fig. 6.3 obtained for practical data is as follows.

R1: If (VRY is in1cluster1) and (VYB is in2cluster1) and (VBR is in3cluster1) and (IR is in4cluster1) and (IY is in5cluster1) and (IB is in6cluster1) then (VUB is out1cluster 1) and (SP is out2cluster1) and (UV is out3cluster1) and (OV is out4cluster1) and (OL is out5cluster1) and (N is out6cluster1)……(6.12)

Fig. 6.3 shows the ruleviewer of FIS obtained with cluster radius 0.5 for experimentally obtained real time data sets. Table 6.4 shows the six cluster centers and spread coefficients (standard deviation) of six clusters obtained for the six FIS inputs having 0.5 cluster radius through subtractive clustering. The six output classes each have six linear MF (with consequent parameter).

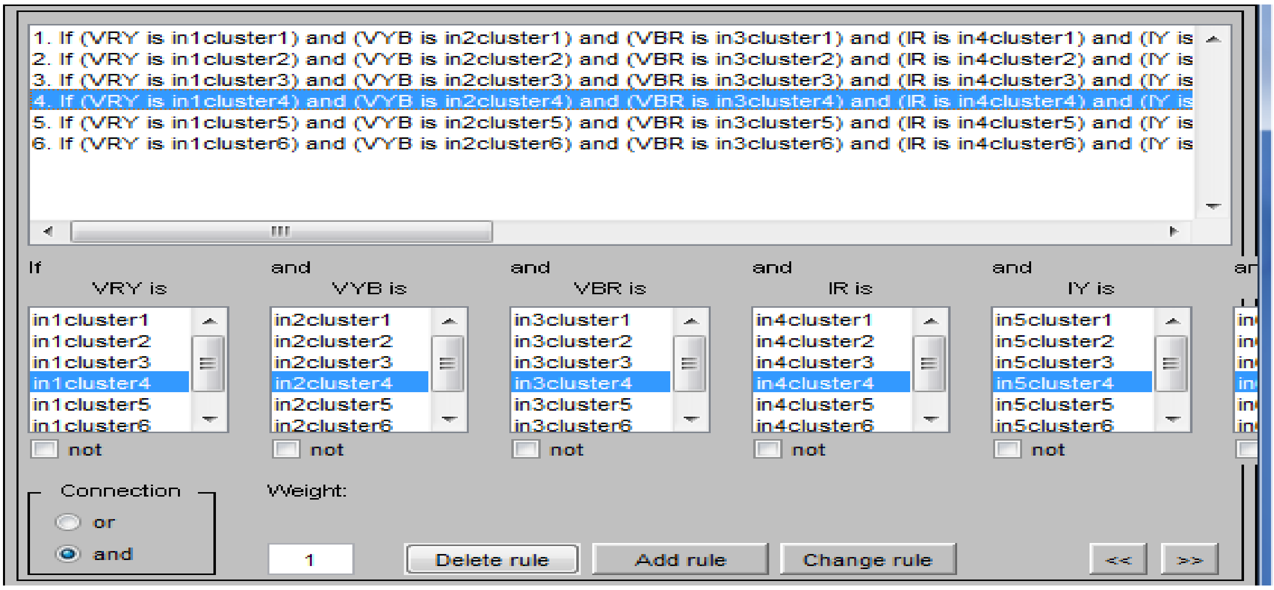


FIGURE 6.3

FIS (Obtained With Cluster Radius 0.5) Rules

TABLE 6.4

Cluster Centers, Standard Deviation and Rules Obtained Through Subtractive Clustering for Experimentally Obtained Data Sets FIS with Cluster Radius 0.5

Cluster no. obtained for six inputs and Rule no.	Spread coefficient [36.23] and cluster centre for VRY (In)	Spread coefficient [30.11] and cluster centre for VYB (In 2)	Spread coefficient [39.42] and cluster centre for VBR (In 3)	Spread coefficient [1.51] and cluster centre for IR (In 4)	Spread coefficient [2.05] and cluster centre for IY (In 5)	Spread coefficient [2.08] and cluster centre for IB (In 6)	Dominant Rules for condition
1	412.695	409.701	408.191	4.036	3.886	4.2	N
2	393.351	405.581	397.726	3.859	4.623	4.132	VUB
3	460.389	457.805	459.98	4.459	4.227	4.514	OV
4	370.578	367.789	368.838	4.718	4.623	4.827	UV
5	410.059	408.063	409.24	5.195	5.059	5.345	OL
6	451.638	448.9	450.64	4.527	4.35	4.595	N

The final output class (condition) for a particular pattern appears to FIS (fault or normal) is close to 1.0 (usually in range of 0.5-1.0) while the other outputs are close to 0.0 (usually in range of 0.0 - 0.5). If, a data sets point with strong membership to the first cluster is fed to FIS, then rule1 will fire with more firing strength than the other rules. Similarly, if an input have strong membership to two clusters then that two cluster related rules fire with more strength than other rules. Rules with lesser weights count for less in the final output. It is observed through subtractive clustering that the dominant rules obtained for normal conditions are 1 and 6, for overvoltage 3, for overload 5, for undervoltage 4 and for voltage unbalance conditions 2. As single phasing is worst case of voltage unbalance, here we obtain a common FIS rule for voltage unbalance and single phasing identification in best generalized FIS for real time data sets. The Rule R1 can be interpreted as

R1: If (VRY is in1cluster1) and (VYB is in2cluster1) and (VBR is in3cluster1) and (IR is in4cluster1) and (IY is in5cluster1) and (IB is in6cluster1) then condition is normal……………………………………………………………………………….(6.13)

The other rules can be interpreted in similar way and shown in Table 6.4. The generalized FIS does monitor and detect faults for train patterns and (unseen) test inputs accurately.

Experimental results show that FIS is able to detect test input with good amount of total overall classification accuracy of 94.4% respectively using proposed approach. The best FIS configuration is used for comparison with other classifier using independent test data sets in chapter 7. The output fuzzy status shows the relative output of six class conditions for train and test patterns. So monitoring of FIS output results is also possible and helpful in maneuver the motor operation is an additional advantage.

Fig 6.5 to Fig. 6.7 show the best generalized FIS configuration Fig 6.4 output results for some example independent test patterns of Table 5.2

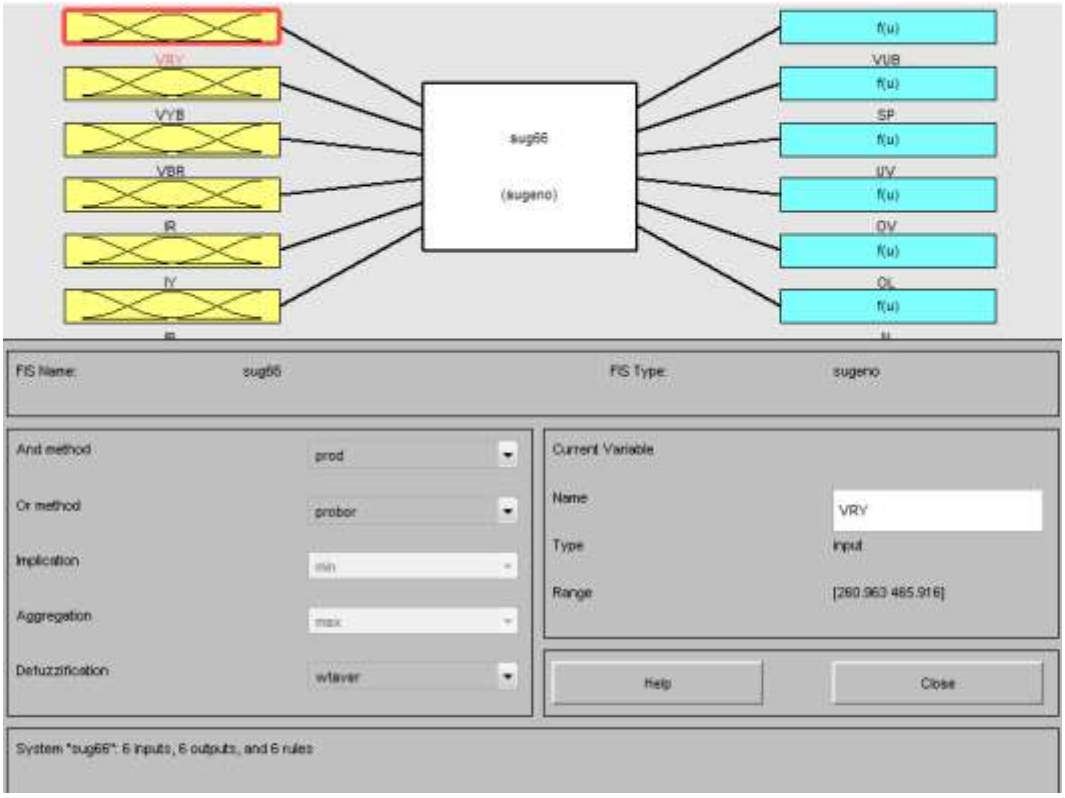


FIGURE 6.4

FIS

.

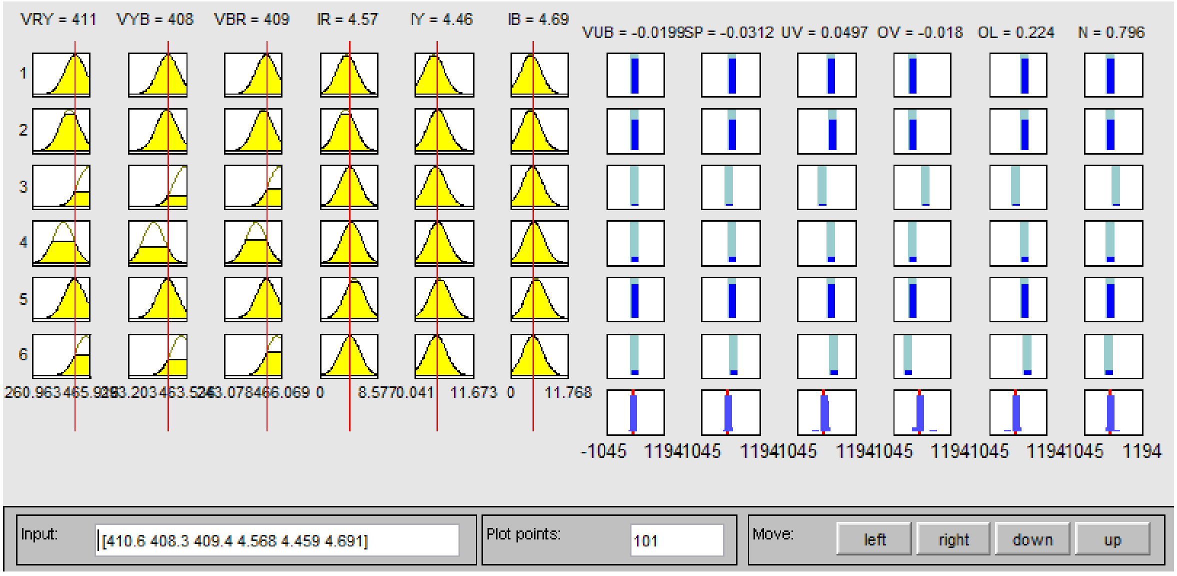


FIGURE 6.5

FIS Ruleviewer for Normal condition Sr. No. 1 of Table 5.2

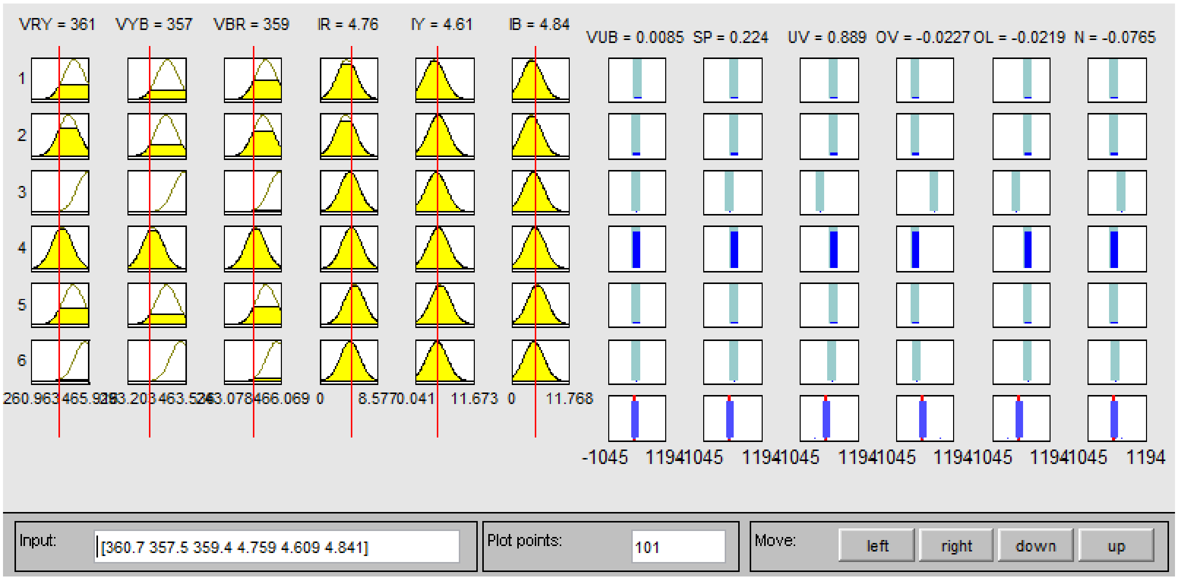


FIGURE 6.6

FIS Ruleviewer for UV condition Sr. No. 5 of Table 5.2

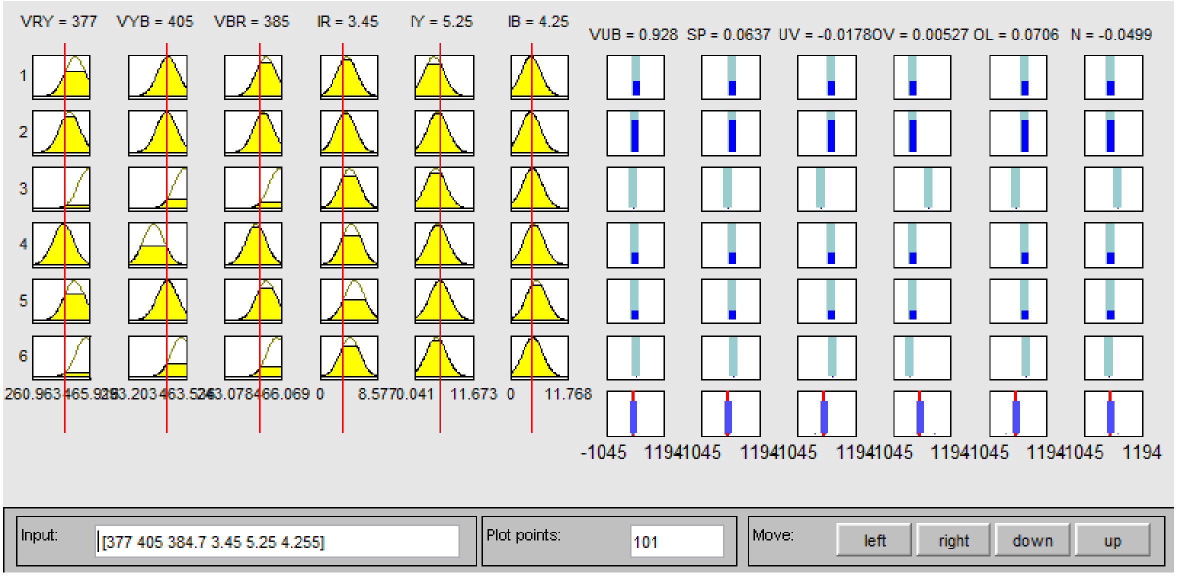


FIGURE 6.7

FIS ruleviewer for VUB condition Sr. No. 9 of Table 5.2

6.3 External faults identification Using PNN

6.3.1 Probabilistic Neural Network (PNN)

The PNN was first presented by D. F. Spechet is a feedforward network formulation of probability density estimation and competitive learning. It provides a general solution to pattern classification problems and based on stastical approach of bayseian classifier. The network paradigm also uses Parzen estimators which were developed to construct probability density function (pdf) required by bayes theory. The PNN used a supervised training set to develop distribution functions within a pattern layer. These functions are used, in recall mode, to estimate the likehood of an input feature vector being part of learned category or class. The learned pattern can also combined or weighted with the prior probability of each class to determine the most likely class for a given input vector. If the prior probability is unknown then all class can be assumed to be equally likely and the determination of class is solely based on the closeness of the input feature vector to the distribution function of a class [15].

If the pdf of each of the population is known then an unknown x belong to class i according to Bayes optimal decision rule is given by

$h_i c_i f_i > h_j c_j f_j$, for all $j \neq i$ ……………………………………………………………(6.14)

f_k is the pdf for class k. The other parameter h is the prior probability and c is misclassification cost which expresses the cost of incorrectly classifying an unknown [16].

In many situations, the loss functions and prior probabilities can be considered equal. So the using decision rule given by equation above is to estimate the probability density functions from the training patterns [17] [12].

In case of inputting network a vector the pdf for a single sample will be given as

$$f(x)=\frac{1}{(2\pi)^{m/2}\,(\sigma)^m} \exp\left(-\left(x-x_j\right)^T\left(x-x_j\right)\middle|(2\sigma)^2\right) \quad (6.15)$$

m is the input space dimension, x_j is the j^{th} sample number and σ is an adjustable smoothing parameter. pdf for a single population is calculated from the parzen's pdf estimator as

$$f_i(x)=\frac{1}{(2\pi)^{m/2}\,(\sigma)^m\, n_i} \sum_{j=1}^{n_i} \exp\left(-\left(x-x_j\right)^T\left(x-x_j\right)\middle|(2\sigma)^2\right) \quad (6.16)$$

Which is the average of the pdf's for n_i samples in the i^{th} population. The classification criteria in this case of multivariate input will be expressed as follows

$f_i(x) > f_j(x)$, for all $j \neq i$

$$f_i(x)=\frac{1}{n_i} \sum_{j=1}^{n_i} \exp\left(-\left(x-x_j\right)^T\left(x-x_j\right)\middle|(2\sigma)^2\right) \quad (6.17)$$

Which eliminate the common factors and absorb the '2' into σ.

PNN architecture learning speed is very fast which makes it capable to adapting its learning in real time, deleting or adding training data as new condition arises. PNN can be shown to always converge to the Bayes optimal solution as the number of training samples increase. PNN belongs to family of radial basis function NN which due to their robustness widely used in pattern classification problems. PNN handle data that has spikes and points outside the norm better than other neural networks. It requires large space in memory [18].

6.3.2 PNN Architecture

The structure used in this study as shown in Fig. 6.8 has multilayer structure consisting of a single radial basis function hidden layer of locally tuned neurons which are fully interconnected to an output competitive layer of six neurons. In this system real valued input vector is feature vector consist values of three phase RMS voltage and currents and six outputs are index of six classes. All hidden neurons simultaneously receive six dimensional real valued input vectors via neuron weights. The hidden layer consists of a set of same type of radial basis function (Gaussian) and associated with j^{th} hidden unit is parameter vector called c_j a center. The first layer input weights IW1,1 are set to the transpose of the matrix formed for the Q training pairs P'. The hidden layer node calculate the Euclidean distance between centre and new input vector and produce a vector whose elements indicate how close the input to the vectors of training set. These elements are multiplied element by element by the bias and sent to the radial basis transfer function. An input vector close to a training vector is represented by a number close to 1 in the output vector a1. If input is close to several of a single class; it is represented by several elements of a1 that are close to 1 [12].

a^1= radbas (‖IW1,1-p‖.*b) ……………………………………………………………(6.18)

Where radbas is radial basis transfer function and can be given as

$\mathrm{radbas}(n)=\exp(-n^2)$

Here,

n= (‖IW1,1-p‖.* b) …………………………………………………………………… (6.19)

The second layer weights LW 2,1 are set to the matrix of target vectors. Each vector is set to the matrix T of target vectors. Each vector has 1 only in the row associated with that particular class of input and 0's elsewhere. The multiplication of T and a^1 sums the elements of a1 due to each of the k input classes. Finally second layer transfer function, compete produce 1 corresponding to target element of n2 and 0's elsewhere [12]. In our case the hidden neurons are 321 same as number of patterns.

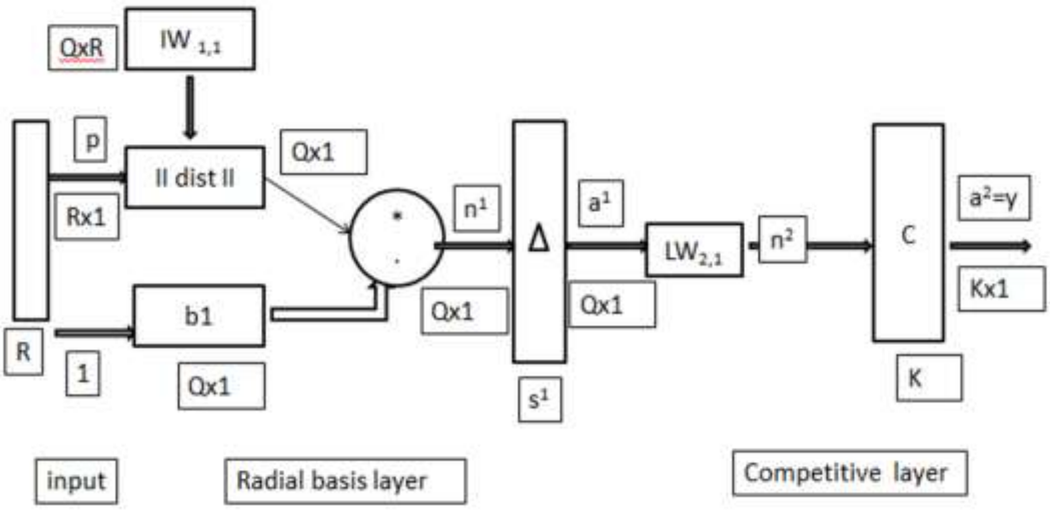


FIGURE 6.8

PNN Architecture [12]

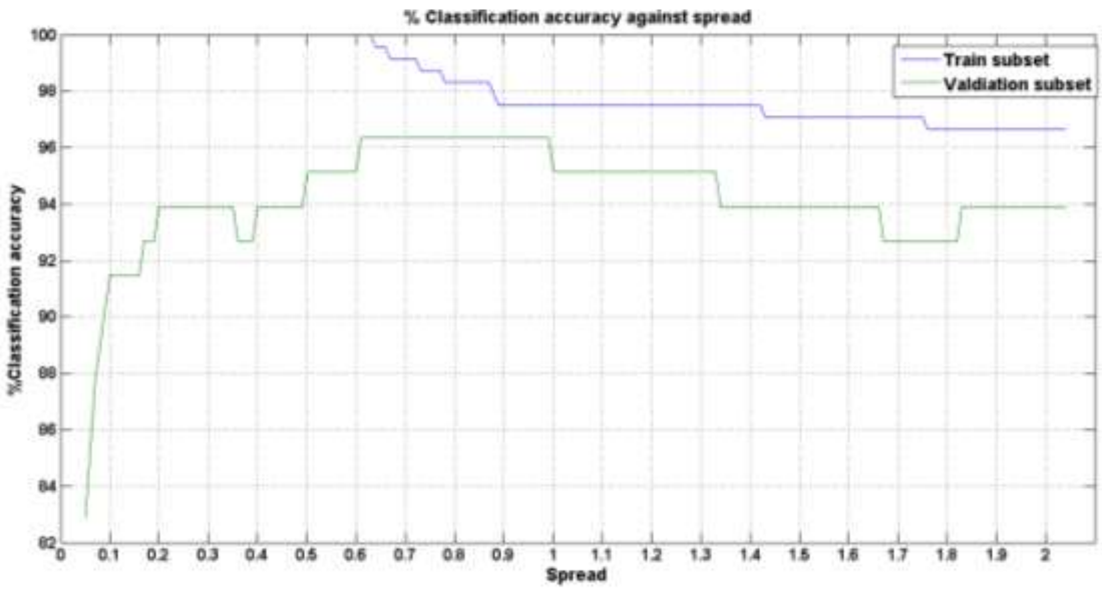


FIGURE 6.9

Total Classification Accuracy Vs. Spread for PNN Training Subsets.

6.3.3 Classification Results Obtained Using PNN

We have used 321 total train input data sets by splitting in training (75%) and validation subsets (25%) for each class. Validation subsets are used to test PNN for each spread after training. The output assign according to Table 6.1 for training. Fig. 6.9 shows the results of PNN train and validation (subset of total train data sets) data sets classification accuracy against radial basis function spread. It is found from validation data sets classification accuracy results that PNN gives most generalized results between spread 0.6 and 1. The results are obtained in MATLAB environment using newpnn and own written codes. The train and independent test data sets results of PNN obtained with 0.62 spared is considered for further comparison with other classifier performance using 72 independent test data sets in chapter 7**.**

6.4 External Fault Identification Using ANFIS

6.4.1 Introduction

Neural Network and fuzzy set theory, which are termed soft computing techniques, are tools of intelligent system. Fuzzy system does not usually learn and adapt from environment, whereas ANN has the capacity of on line adaption and learning. Neuro-fuzzy system is the combination of neural network and fuzzy inference system. ANFIS is the fuzzy logic based paradigm that uses the learning ability of ANN to enhance intelligent system's performance using prior knowledge. Fundamentally a neuro-fuzzy system is a fuzzy network that not only includes a fuzzy inference system but can also overcome some limitations of neural networks and fuzzy systems as it can learn and able to represent knowledge in interpretable form. The problem of selecting suitable MF values like in Mamdani fuzzy system can be avoided and that offers the possibility of solving tuning problems and design difficulties of fuzzy logic in most cases [19]. The basic structure of the classic FIS is a model that maps input characteristic to input MFs, Input MF to rules, rules to set of output characteristics, output MF to a single valued output or decision associated with the output. All these process are developed using fixed MFs. The neuro-adaptive learning method works similar that of neural networks and provide a method for fuzzy modelling procedure to gather information about a dataset. Then fuzzy logic computes the MF parameters that best allow the associated FIS to track the given input/output data. It is able to construct an input-output mapping based on both human knowledge and simulated input-output data pairs. Fuzzy classification is the task of partitioning a feature space into fuzzy classes. It can be possible to describe feature space with fuzzy regions and control each region with fuzzy rules. It is possible to optimize MF parameters with neural networks. As a result fuzzy classification systems and NN can be combined which is named as adaptive neuro-fuzzy inference system [19]. ANFIS proposed by Jang [19] is shown in APPENDIX F.

6.4.2 Fault Identification Using ANFIS

ANFIS is used for further premise parameter and consequent parameter optimization after generating subtractive clustering based FIS (as shown in FIGURE 6.2) with only one column output. ANFIS can be used for online and batch learning paradigm. Each epoch of batch learning composed of a forward pass and a backward pass. As in [20], in forward pass the antecedent parameters are fixed and the consequence parameters are optimized in least square estimation. Once the optimum consequence parameters are found the backward pass stage starts. In this stage gradient decent used to optimally adjust the antecedent membership parameters corresponding fuzzy sets in the input domain. The output of the ANFIS calculated by fixing the consequence parameters to the values found in the forward pass. The output error of the ANFIS is used to adapt the antecedent parameters using a standard backpropagation algorithm. ANFIS is a fuzzy Sugeno model used with adaptive network to facilitate learning and adaption. ANFIS can construct an input – output mapping based on both human knowledge and input output data observations. We have used Matlab ANFIS function and own written codes for fault diagnosis using ANFIS.

6.4.3 ANFIS Architecture

For a fuzzy inference system with six inputs VRY,VYB,VBR,IR,IY AND IB and one output Z with the first order Sugeno model fuzzy rules set can be written as

If VRY is A_{i1} and VYB is A_{i2} and VBR is A_{i3} and IR is A_{i4} and IY is A_{i5} and IB A_{i6} then class $C_1 = p_1 VRY + q_1 VYB + r_1 VBR + s_1 IR + t_1 IY + u_1 IY + v_1$ ……………… (6.20)

Where (p_1, q_1, r_1, s_1, t_1, u_1, v_1) are parameter of output functions. A_{ij} is the exponential membership function as shown in (6.9).

Layer 0: it consists of plain input variable set. In this case it is VRY, VYB, VBR, IR, IY and IB.

Layer 1: The node function of every node i in this layer take the form as

$O_i^1 = \mu A_i(VRY)$ …………………………………………………………………(6.21)

Where x is the input to node i, and μA_i is the membership function which can be Gaussian, triangular or other shapes of the linguistic label A_i associated with this node. We have used Gaussian-shaped MFs defined as

$$\mu A_i(VRY)=\exp\left(\frac{(x-c_i)^2}{2\sigma_i^2}\right) \quad \ldots\ldots (6.22)$$

Where $\{c_i, 2\sigma_i^2\}$ are the parameters of MF governing Gaussian functions. The parameters in this layer are referred as premise or antecedent parameters.

Layer 2: Every node in this layer multiplies incoming signals from layer 1 and send product out as follows

$$O_i^2= w_i= \mu A_i(VRY)^*\ \mu B_i(VYB)^*\_\ \_ \quad \ldots\ldots (6.23)$$

Layer 3: Every node i in this layer determines the ratio of the i^th rule's firing strength to the sum of all rules firing strength as

$$O_i^3= \overline{w_i}= \frac{w_i}{w_1+w_2+\_\ \_} \quad i = 1, 2,\_\ \_ \quad \ldots\ldots (6.24)$$

Output of this layer represents the normalized firing strengths.

Layer 4: Every node in this layer is an adaptive node with a node function of the form

$$O_i^4= \overline{w_i}\, f_i = \overline{w_i}\,(p_i VRY+q_i\ VYB+ r_i\ VBR+s_i IR+t_i\ IY+ t_i\ IB+ v_i) \quad \ldots\ldots (6.25)$$

Where $\overline{w_i}$ is the output of layer 3, and $\{p_i, q_i, r_i,s_i,t_i,u_i,v_i\}$ is the parameters set and parameters are referred as consequent parameters.

Layer 5: The single node in this layer is a circle node labelled ∑ that computes the overall output as summation of all incoming signals, i.e.

$$O_1^5 = \text{overall output} = \sum_i \overline{w_i}\, f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad \ldots\ldots (6.26)$$

ANFIS is composed of inputs, outputs and rules. Each input and output may have any number of membership functions decided by clustering radius selected. Gaussmf is used as input variables membership function. The rules dictate the behaviour of the fuzzy system based on inputs, outputs and membership functions. The parameters of subtractive clustering were chosen as follows: squash factor 1.25, accept ratio 0.5 and rejection ratio 0.15. Clustering and FL together provide a simple yet powerful means to model the fault relationship that we want to develop.

6.4.4 Classification Results and Rules Obtained Using ANFIS

Three phase RMS voltages and current values are used as the inputs to ANFIS and six conditions constitute the output of ANFIS. The output assign for normal condition is 1(0.5-1.5), 2 (1.5 - 2.5) for OL, 3 (2.5 – 3.5) for OV, 4 (3.5-4.5) for UV, 5 (4.5-5.5) for SP and 6 (5.5-6.5) for VUB. We have attempted to obtain best generalized ANFIS configuration by comparing test classification accuracy of different cluster radius ANFISs. A large cluster radius generally results in fewer clusters and hence a coarser model, while a small cluster radius can produce excessive number of clusters and model that does not generalize well. Cluster radius is an approximate specification of the desired resolution of the model which can be adjusted based on resultant complexity and generalization ability of the model [10]. We have taken 321 data sets for ANFIS training and 72 data sets for testing FIS with independent test (unseen) input. The test data sets are also used as checking data alongwith ANFISs training for preventing overfitting for same number of epoch training.

TABLE 6.5

Test and Train Data Sets Total Classification Accuracy of ANFISs obtained With Different Subtractive Cluster Radius for Practical Data Sets

Sr. No	**Cluster Radius**	**% Total Classification Accuracy (Test Data sets)**	**% Total classification Accuracy (Train Data Sets)**	**Rules**
1	0.09	87.5	98.1	21
2	0.1	89	95.3	20
3	0.11	83.33	90.97	17
4	0.12	89	93.3	15
5	**0.13**	**91.7**	**95.95**	**14**
6	0.14	90.3	93.77	13
7	0.15	89	91	13
8	0.16	73.6	86	12
9	0.17	77.88	88.16	12
10	0.18	77.78	86	12
11	0.19	75	85.05	9
12	0.2	72.22	83.5	9

Table 6.5 shows the results of train and test total classification accuracy of ANFISs obtained with difference cluster radius for practical data sets. The ANFIS configuration obtained with cluster radius 0.13 have highest test classification accuracy and less rules and chosen as the best generalized ANFIS configuration. The best ANFIS configuration is obtained with 14 clusters, 14 membership functions and 14 rules. The dominant rules obtained for different conditions are shown in Table 6.6. As single phasing is worst case of voltage unbalance, here we obtain a common FIS rule for voltage unbalance and single phasing identification in best generalized FIS for real time data sets same as SC_FIS. The output of ANFIS diagnosed as, for example,

if (VRY is in cluster1) and (VYB is in cluster1) and (VBR is in cluster1) and (IR is in cluster1) and (IY is in cluster1) and (IB is in cluster1) then output is OV………..(6.27)

TABLE 6.6

Cluster Centers, Standard Deviation and Rules Obtained Through Subtractive Clustering for ANFIS with Cluster Radius 0.13

Rules	Spread Coefficient [9.4] and Cluster Centre for VRY (In 1)	Spread Coefficient [9.075] and Cluster Centre for VYB (In 2)	Spread Coefficient [10.24] and Cluster Centre for VBR (In 3)	Spread Coefficient [0.39] and Cluster Centre for IR (In 4)	Spread Coefficient [0.54] and Cluster Centre for IY (In 5)	Spread Coefficient [0.54] and Cluster Centre for IB (In 6)	Output Condition
1	460.389	457.805	459.98	4.459	4.227	4.514	OV
2	410.008	407.807	409.01	5.059	4.923	5.209	OL
3	410.571	408.677	409.65	4.5	4.377	4.636	N
4	394.067	409.906	399.722	3.695	4.705	4.05	VUB
5	448.644	446.035	447.902	4.5	4.309	4.568	N
6	370.578	367.789	368.838	4.718	4.623	4.827	UV
7	416.405	414.614	415.842	3.791	3.695	3.886	N
8	412.029	404.021	398.57	4.036	3.695	4.595	VUB
9	382.169	379.355	380.685	4.623	4.514	4.732	N
10	429.633	427.202	429.019	4.5	4.309	4.568	N
11	396.728	393.76	395.372	3.968	3.873	4.009	N
12	408.089	406.426	406.989	2.591	2.427	2.509	N
13	376.054	372.446	374.032	4.064	4.009	4.105	UV
14	414.793	413.181	414.153	5.727	5.632	5.959	OL

The total train 321 and independent test 72 data sets results of ANFIS obtained with 0.13 spared is considered for further comparison using with other classifier performance in chapter 7. Fig 6.10 shows the best generalized ANFIS configuration. Fig 6.11 to Fig. 6.13 show the output results for some example independent test patterns of Table 5.2

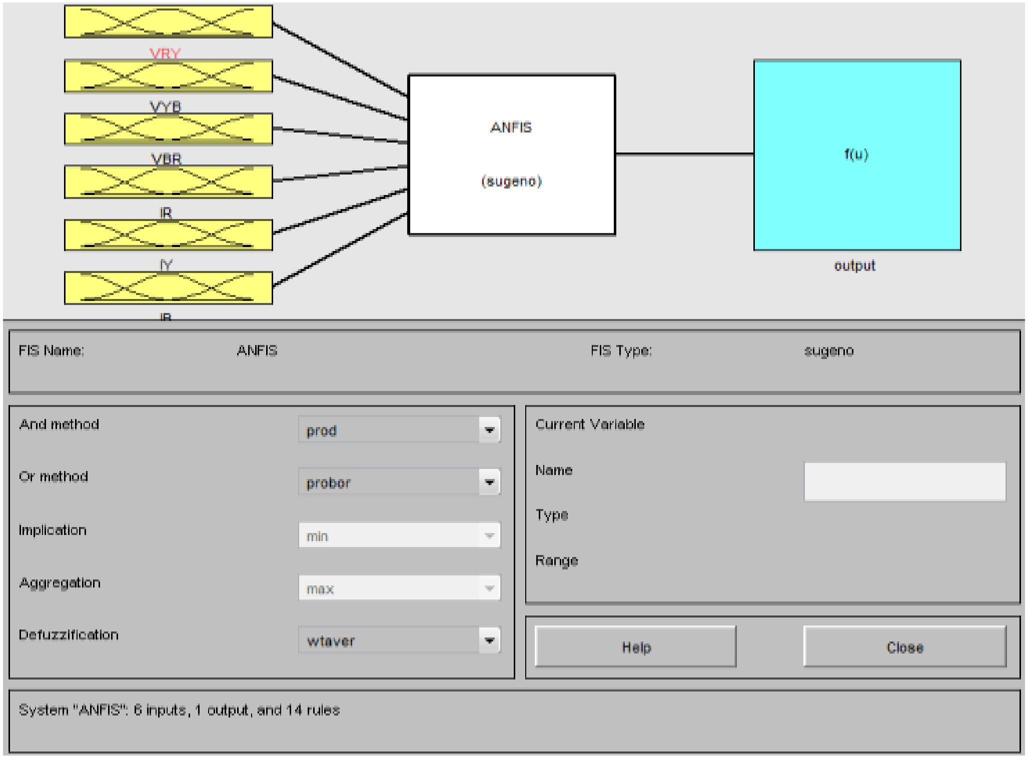


FIGURE 6.10

ANFIS

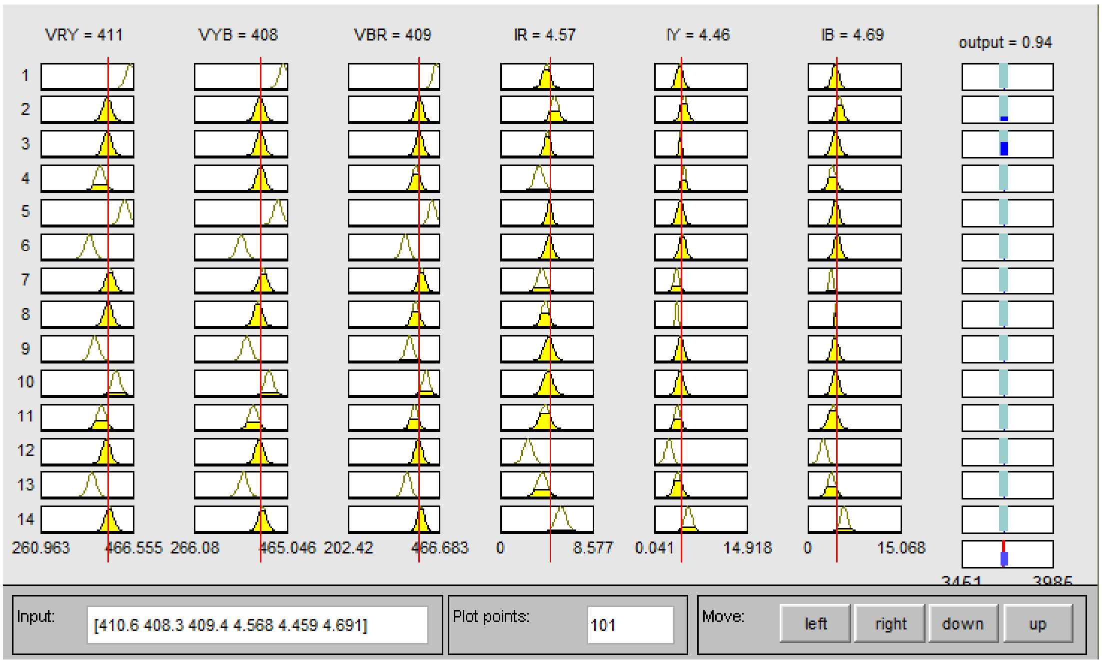


FIGURE 6.11

ANFIS ruleviewer for Normal condition Sr. No. 1 of Table 5.2

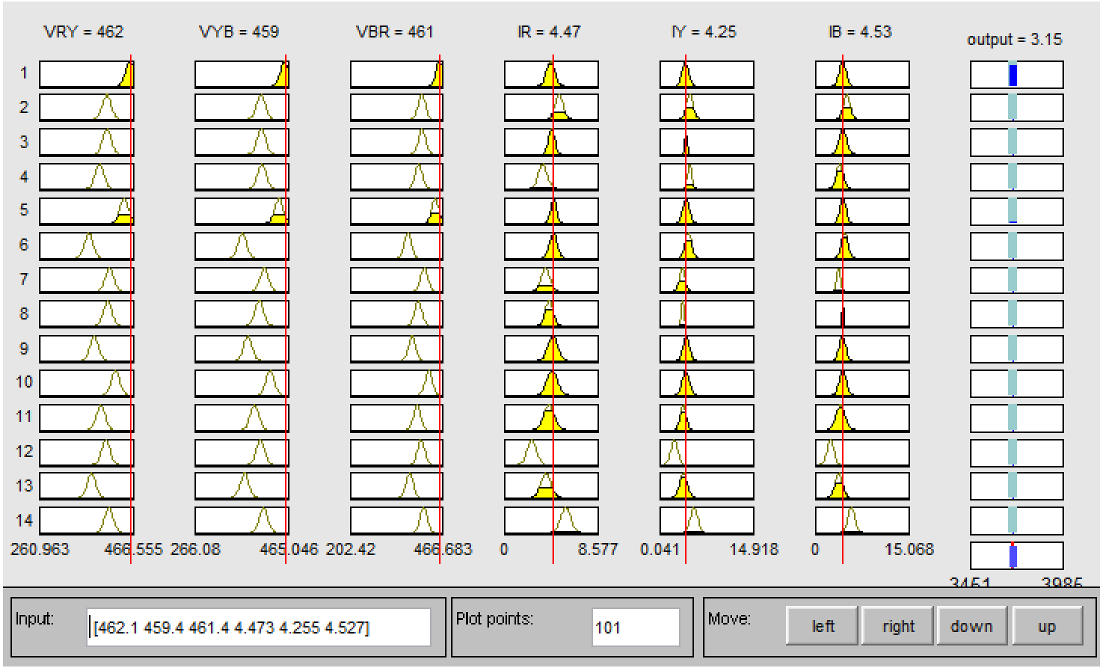


FIGURE 6.12

ANFIS ruleviewer for OV condition Sr. No. 4 of Table 5.2

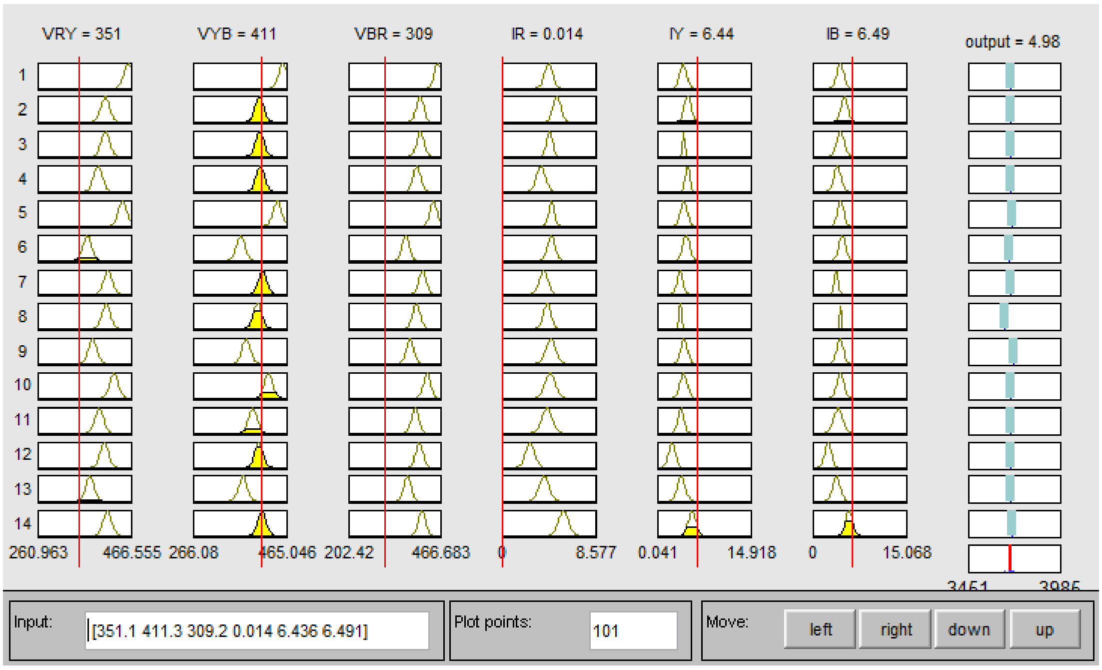


FIGURE 6.13

ANFIS ruleviewer for SP condition (R phase) Sr. No. 6 of Table 5.2

6.5 External Fault Identification Using NBC

6.5.1 NBC

The Naïve bayes model is a highly simplified and practical bayseian probability model. The Naïve bayes classifier depends on strong independence assumption so the probability of one attribute does not affect the probability of other. The Naïve bayes classifier advantage over bayes net is 2*n number of parameters for modeling require instead of $2*(2^n-1)$ for n number of variables.

The Naïve bayes classifier applies to learning task where each instance or sample input vector X= $\{X_1,X_2, \dots, X_n\}$ is described by a conjunction of variables values and target function take any value from some finite set C, where class labels C= $\{C_1,C_2, \dots C_j \dots Cm\}$. Cm is number of classes. Naïve bayes classifier identify a new test instance described by tuple of input variables $\{X_1,X_2, \dots, X_n\}$ based on most portable target class using posterior probability.

Naïve bayes classifier used for prediction of class for new instance is based on bayes theorem.

$$P(C_j \mid X) = \frac{P(C_j)P(X \mid C_j)}{P(X)}$$

…………………………………………………………….(6.28)

In (6.28) P (C_j) is the prior probability of class C_j, $P(X/C_j)$ is class conditional density or likehood, P(X) can be ignored as it is same for all class. Because of Naive assumptions the class conditional density can be estimated using

$$P(X / C_j) = \prod_{i=1}^{n} (P(X_i \mid C_j) \qquad \text{……………………………………………………… (6.29)}$$

The class is decided for X is based on bigger posterior probability and given as

$$C = arg \max_{j=1 \dots m} P(C_j) \prod_{i=1}^{n} P(X_i \mid C_j) \text{ ……………………………………………………… (6.30)}$$

We have used sampled data of continuous signal of three phase voltages and currents as input data using logging device and used nonparametric normal kernel density estimation for finding class conditional probability. Conditional probability densities of each class can be calculated as

$$P_s(X) = \frac{1}{S.h^d}\sum_{t=1}^{S} k\,\frac{X - x_t}{h} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (6.31)$$

Wherein (6.31) S is number of data belonging to class, k is kernel function with its bandwidth or smoothing parameter h and d is number of dimension. For training purpose the data set is defined as x_t. Class conditional density of testing data applied to a trained classifier is calculated as

$$p\left(X_i = x_{iy}^{tst} \big/ C = j\right) = \frac{1}{N_{trj}}\sum_{t=1}^{Ntrj}\frac{1}{\sqrt{2\pi\sigma}}\,e^{-\frac{(x_{iy}^{tst} - x_{it}^{tr})^2}{2\sigma^2}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ (6.32)$$

Wherein (6.32) X_i is ith variable, j is determined class, x_{iy}^{tst} is yth tested data and x_{it}^{tr} is training data that used. The gaussian kernel has width σ around each of the N_{trk} training pattern of class C_j [21] [22].

6.5.2 External faults Results Using NBC

10-fold cross-validation is used and training data sets sets are divided in 10 equal folds of training and testing subset to find the average test classification accuracy. The average 10-fold train classification accuracy obtained is 87.54%. The average 10-fold test classification accuracy obtained is 86.11%. The output assign for normal condition is 1, 2 for OL, 3 for OV, 4 for UV, 5 for SP and 6 for VUB. Naïve bayes classifier is further used for comparison with other classifiers with 72 independent test data sets in chapter 7**.**

6.6 External Faults Results Using LDA

10-fold cross-validation is used and training data sets are divided in 10 equal folds of training and testing subset to find the average test classification accuracy. The average 10-fold train classification accuracy obtained is 71.96 %. The average 10-fold test classification accuracy obtained is 76.39 %. The output assign for normal condition is 1, 2 for OL, 3 for OV, 4 for UV, 5 for SP and 6 for VUB. LDA is used for comparison with other classifiers with 72 independent test data sets in chapter 7.

References

1. Orhan U, Hekim M, Ozer Mahumut (2011) EEG signals classification using the k-means clustering and a perceptron neural network model, Expert system with applications, Vol. 38, pp. 13475-13481, ISSN: 0957-4174

2. Haykin S. (2008) *Neural Networks and learning machines (3rd ed.)* Pearson Education, New Jersy, USA.

3. Zidani F, Benbouid, MEH, Diallo D, Nai said M (2003) 'Induction Motor stator faults diagnosis by a current concordia pattern based fuzzy decision system', IEEE Transactions on Energy Conversion, Vol. 18, No. 4, pp. 469-475, ISSN: 0885-8969.

4. Keshavarzi A, Sarmadian F, Rahmani A, Ahmadi A, Labbafi R, Iqbal M (2012) 'Fuzzy clustering analysis for modeling of soil cation exchange capacity', Australian Journal of Agriculture Engineering, Vol. 3, No. 1, pp. 27-33, 2012, ISSN: 1836-9448.

5. Rajasekaran S, Vijayalakshmi Pai GA (2007) Neural Networks, Fuzzy logic, and Genetic Algorithms Synthesis and Applications. Prentice Hall of India, New Delhi.

6. Ross TJ, Booker JM, Parkinson WJ (2002) eds. Fuzzy Logic and Probability Applications: Bridging the Gap [online], SIAM, Philadelphia, ASA, Alexandria, VA, 2002. pp. 41-42. Available: https://books.google.co.in/books?id=s9NT6Fz

7. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani (1997) *Neuro-Fuzzy and soft computing: A computational approach to learning and machine intelligence*, Prentice Hall, NJ, 1997.

8. Takagi T, Sugeno M (1985) 'Fuzzy identification of systems and its applications to modeling and control', IEEE Transactions on systems, man and Cybernetics, Vol. SMC-15, (1), pp. 116-132, ISSN**:** 2168-2216.

9. Hammouda K, and Fakhreddine, K. (1997) 'A Comparative study of data clustering techniques', University of waterloo, Ontario, Canada, vol. 13, issue 2-3, Nov. 1997, pp. 149-159.

10. Chiu, S.L. (1994) 'Fuzzy model identification based on cluster estimation'. *Journal of Intelligent Fuzzy systems*, Vol. 2, pp. 267-278, ISSN: 1064-1246**.**

11. Yager R.R., and Filev D.P. (1994) 'Generation of fuzzy rules by mountain clustering', *Journal of intelligent and fuzzy system'*, Vol.2, pp. 209-219, ISSN: 1064-1246.

12. Matlab 7.10, Mathworks Inc. 2010.

13. Chiu, SL (1995) 'Extracting fuzzy rules for pattern classification by cluster estimation', *Proc. IFSA*, pp. 1-4, 199.

14. Chopra S, Mitra R, and Kumar V, ‘Identification of rules using subtractive clustering with application to fuzzy Controllers’, *Proceedings of 2004 International Conference on Machine Learning and Cybernetics,* Vol.7, pp. 4125- 4130, 2004, ISSN**:** 2168-2216.

15. Araghi L, Kahaloozade H, Reza M (2009) ‘Ship identification using probabilistic neural networks’, Proceedings of the International Multiconference of engineers and computer scientists (IMECS 2009), Hong Kong, China, pp. 18-20.

16. M.M.El Emary I and Ramakrishnan (2008) On the application of various probabilistic neural networks in solving difference pattern classification problems. World Applied Science journal, Vol. 4, No. 6, pp. 772-780, ISSN: 1818-4952.

17. Swathi S, Rizwana S, Anjan Babu G, Santosh Kumar P, Sarma P (2012) Classification for neural network structures for breast cancer diagnosis. International Journal of Computer Science and Communication, Vol. 3, No. 1, pp. 227-231, ISSN: 2249-5789.

18. Wang C, Yang W, Chen J, Liao G (2009) Power disturbance recognition using probabilistic neural networks, Proceedings of the international multi conference of engineers and computer scientists (IMECS), Hong Kong, China, pp 1573-1577.

19. Jang JSR (1993) ‘ANFIS: Adaptive Network based Fuzzy Inference Systems’, IEEE Transactions on systems, Man and Cybernetics, Vol. 23, No 3, pp. 665-685, ISSN: 2168-2216.

20. Ahmed Taher Azar, Shaimaa Ahmed El-said (2013) ‘Superior neuro-fuzzy classification systems’, Neural Computing and Applications, 23 (Suppl 1), S55-S72, ISSN: 1433-3058.

21. Ren J, LEE SD, Chen X, kao B, Cheng R, Cheung D (2009) Naive bayes classification of uncertain data. Proceedings of the 9th IEEE international conference on data mining (ICDM), Miami, FL, USA, pp. 944-949.

22. Asfani D, Purnomo M, Sawitri D (2013) ‘Naïve Bayes classifier for temporary short circuit fault detection in stator winding’, Proceedings of 9th IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives (SDEMPED), Valenica, Spain, pp 288-29.

CHAPTER - 7

Comparison between MLPNN, PNN, SC_FIS, ANFIS, NBC and LDA for Induction Motor External Faults Identification

7.1 Measures of Performance Evaluation

The following stastical measures to evaluate performance of classifiers for multiclass identification.

- Total classification Accuracy: It is the total number of correct decisions to the total number of decisions.

- Confusion matrix: A confusion matrix contains information about output and target classifications done by a classification system.

- Sensitivity (Recall or True positive rate (TPR)): It is the ratio number of true positives (TP) to number of actual positives. FN in (7.1) are false negatives.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP+FN}} \quad (7.1)$$

- Specificity: It is the number of true negatives (TN) decisions to number of actual negative cases. False Alarms (FPR) can be obtained by subtracting specificity from 1. FP in (7.2) are false positives.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN+FP}} \quad (7.2)$$

- Precision: It is a measure of the accuracy provided that a specific class has been predicated.

$$\text{precision} = \frac{\text{TP}}{\text{TP+FP}} \qquad \text{(7.3)}$$

- F-measure: It can be used as single measure of performance and it is harmonic mean of precision and sensitivity.

$$\text{F-measure} = \frac{2 \times \text{precision} \times \text{sensitivity}}{\text{precision} + \text{sensitivity}} \qquad \text{(7.4)}$$

7.2. Results and Discussions

7.2.1 Classifier comparison Using total classification accuracy for total train 321 and 72 independent test data sets

We have used 321 data sets for training and independent 72 data sets for testing of the classifier. The details of data sets are given in Table 5.1.

TABLE 7.1

Fault Classification Accuracy Results of Classifiers

Sr. No.	Classifier	% Total Classification Accuracy (Training Input Data Sets)	% Total Classification Accuracy (Independent Testing Data Sets)	Average classification Accuracy
1	LDA	71.96	76.39	74.18
2	NBC	87.54	86.11	86.83
3	PNN	99.07	93.06	96.1
4	MLPNN	100	98.61	99.3
5	SC_FIS	96.57	97.2	97
6	ANFIS	95.95	91.7	93.83

Table 7.1 shows the total classification accuracy for train and independent test data sets of LDA, NBC, PNN, MLPNN, SC_FIS, and ANFIS classifiers. It is observed that neural network, SC_FIS and adaptive neurofuzzy classifier results are found more accurate and quite better than conventional Well-known LDA and simple probabilistic approach Naïve bayes classifier with respect to total classification accuracy of train and independent test data sets. All classifier are further compared with other statistical measures like sensitivity, specificity, precision and overall F-measure using total train (321) and independent (72)

test data sets. For that in following section confusion matrix for all classifiers are shown for training and independent testing data sets.

Total classification accuracy sample calculation for the training data sets of SC_FIS based on confusion matrix (TABLE 7.4) is shown in APPENDIX G. Sensitivity, specificity, precision and F-measure sample calculations for VUB and Normal condition output (N) for SC_FIS training data sets (TABLE 7.4) are also shown in APPENDIX G.

7.2.2 Confusion Matrix for 321 Total Training and 72 Independent Test Data Sets for MLPNN

TABLE 7.2

Confusion Matrix for 321 Total Training Data Sets for MLPNN

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	50	0	0	0	0	0
SP	0	41	0	0	0	0
UV	0	0	30	0	0	0
OV	0	0	0	30	0	0
OL	0	0	0	0	30	0
N	0	0	0	0	0	140

TARGET CLASS

TABLE 7.3

Confusion Matrix for 72 Independent Test Data Sets for MLPNN

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	13	0	0	0	0	0
SP	0	8	0	0	0	0
UV	0	0	10	0	0	0
OV	0	0	0	8	0	0
OL	0	1	0	0	10	0
N	0	0	0	0	0	22

TARGET CLASS

7.2.3 Confusion Matrix for 321 Total Training and 72 Independent Test Data Sets for SC_FIS

TABLE 7.4

Confusion Matrix for 321 Total Training Data Sets for SC_FIS

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	50	0	0	0	0	3
SP	0	41	0	0	0	0
UV	0	0	30	0	0	4
OV	0	0	0	29	0	1
OL	0	0	0	0	30	2
N	0	0	0	1	0	130

TARGET CLASS

TABLE 7.5

Confusion Plot for 72 Independent Test Data Sets for SC_FIS

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	13	0	0	0	0	1
SP	0	9	0	0	0	0
UV	0	0	10	0	0	1
OV	0	0	0	8	0	0
OL	0	0	0	0	10	0
N	0	0	0	0	0	20

TARGET CLASS

7.2.4 Confusion Matrix for 321 Total Training and 72 Independent Test Data Sets for PNN

TABLE7.6

Confusion Matrix for 321 Total Training Data Sets for PNN

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	50	0	0	0	0	0
SP	0	41	0	0	0	0
UV	0	0	29	0	0	0
OV	0	0	0	30	0	0
OL	0	0	0	0	30	2
N	0	0	1	0	0	138

TARGET CLASS

TABLE 7.7

Confusion Matrix for 72 Independent Test Data Sets for PNN

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	12	1	0	0	0	1
SP	1	8	0	0	0	0
UV	0	0	10	0	0	0
OV	0	0	0	8	0	0
OL	0	0	0	0	10	2
N	0	0	0	0	0	19

TARGET CLASS

7.2.5 Confusion Matrix for 321 Total Training and 72 Independent Test Data Sets for ANFIS

TABLE 7.8

Confusion Matrix for 321 Total Training Data Sets for ANFIS

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	49	0	0	0	0	0
SP	1	41	1	0	0	0
UV	0	0	26	0	0	0
OV	0	0	2	28	0	1
OL	0	0	1	2	30	5
N	0	0	0	0	0	134

TARGET CLASS

TABLE 7.9

Confusion Matrix for 72 Independent Test Data Sets for ANFIS

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	12	0	0	0	0	0
SP	1	9	0	0	0	0
UV	0	0	10	0	0	1
OV	0	0	0	8	0	1
OL	0	0	0	0	8	1
N	0	0	0	0	2	19

TARGET CLASS

7.2.6 Confusion Matrix for 321 Total Training and 72 Independent Test Data Sets for NBC

TABLE 7.10

Confusion Matrix for 321 Total Training Data Sets for NBC

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	48	0	0	0	0	21
SP	2	41	0	0	0	0
UV	0	0	30	0	0	8
OV	0	0	0	30	0	6
OL	0	0	0	0	30	3
N	0	0	0	0	0	102

TARGET CLASS

TABLE7.11

Confusion Matrix for 72 Independent Test Data Sets for NBC

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	11	0	0	0	0	3
SP	2	9	0	0	0	0
UV	0	0	10	0	0	2
OV	0	0	0	8	0	1
OL	0	0	0	0	10	2
N	0	0	0	0	0	14

TARGET CLASS

7.2.7 Confusion Matrix for 321 Total Training and 72 Independent Test Data Sets for LDA

TABLE 7.12

Confusion Matrix for 321 Total Training Data Sets for LDA

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	42	6	0	0	0	12
SP	0	25	0	0	0	0
UV	2	10	30	0	0	18
OV	0	0	0	30	0	29
OL	0	0	0	0	30	7
N	6	0	0	0	0	74

TARGET CLASS

TABLE 7.13

Confusion Matrix for 72 Independent Test Data Sets for LDA

OUTPUT CLASS

	VUB	SP	UV	OV	OL	N
VUB	11	1	0	0	0	2
SP	0	4	0	0	0	0
UV	1	4	10	0	0	3
OV	0	0	0	8	0	3
OL	0	0	0	0	10	2
N	1	0	0	0	0	12

TARGET CLASS

7.2.8 Classifiers Performance Comparison Using Sensitivity, Specificity, Precision and F-measure

MLPNN, SC_FIS, PNN, ANFIS, NBC and LDA are further compared with sensitivity, specificity, precision and F-measure statistical measures using confusion matrix.

Table 7.14 and Table 7.15 show sensitivity, specificity, precision and F- measure comparison of all classifiers for total train and independent test data sets. Results shows that MLPNN, PNN, SC_FIS and ANFIS achieve impressive results for train data sensitivity, specificity, precision and F-measure. Sensitivity, specificity and most importantly overall F-measure values of PNN and ANFIS are near or more than 90% for four test output conditions and comparable to MLPNN and SC_FIS, but MLPNN and SC_FIS performance are better in for all conditions. The PNN requires more hidden nodes than the MLPNN to reach comparable performance; this is because training of hidden nodes in PNN is unsupervised. The most advantage of SC_FIS and ANFIS faults identification over MLPNN is that faults heuristics extraction is also possible with good statistical measures.

The MLPNN outperforms the other classifiers with respect to train and independent test data sets classification accuracy, sensitivity, false alarms, specificity and F-measure. The advantage of soft computing based fault identification classifier over prevalent conventional thermal based fault identification protection scheme is that it can detect any unseen external faults with high accuracy and produce better generalized results.

TABLE 7.14

Statistical Parameters Comparison for Train Input Data Sets

Output Condition	LDA				NBC				PNN				MLPNN				SC_FIS				ANFIS			
	Sensitivity (%)	Specificity (%)	Precision (%)	F-measure	Sensitivity (%)	Specificity (%)	Precision (%)	F-measure	Sensitivity (%)	Specificity (%)	Precision (%)	F-measure	Sensitivity (%)	Specificity (%)	Precision (%)	F-measure	Sensitivity (%)	Specificity (%)	Precision (%)	F-measure	Sensitivity (%)	Specificity (%)	Precision (%)	F-measure
VUB	84	91.3	70	**0.76**	96	91.73	69.6	**0.81**	100	100	100	**1**	100	100	100	**1**	100	98.86	94.3	**0.97**	98	100	100	0.99
SP	61	100	100	**0.76**	100	99.17	95.4	**0.98**	100	100	100	**1**	100	100	100	**1**	100	100	100	**1**	100	99.63	95.3	0.98
UV	100	87	50	**0.67**	100	96.91	79	**0.88**	96.67	100	100	**0.98**	100	100	100	**1**	100	98.6	88.2	**0.96**	87	100	100	0.93
OV	100	87.4	50.8	**0.67**	100	97.67	83.3	**0.91**	100	100	100	**1**	100	100	100	**1**	96.7	99.64	96.7	**0.97**	93.3	98.94	90.3	0.92
OL	100	97.7	81.1	**0.9**	100	98.82	90.9	**0.95**	100	99.7	93.8	**0.97**	100	100	100	**1**	100	99.29	93.8	**0.97**	100	96.68	79	0.88
N	52.9	96.3	92.5	**0.67**	72.9	82.7	100	**0.84**	98.57	99.5	99.3	**0.99**	100	100	100	**1**	92.9	99.45	99.2	**0.97**	95.7	100	100	0.98

TABLE 7.15

Statistical parameters Comparison for Independent Test Data Sets

Output Condition	**LDA**				**NBC**				**PNN**				**MLPNN**				**SC_FIS**				**ANFIS**			
	Sensitivity (%)	**Specificity (%)**	**Precision (%)**	**F-measure**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-measure**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-measure**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-measure**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-measure**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**	**F-measure**
VUB	84.6	93.6	78.6	**0.82**	84.6	94.44	78.6	**0.82**	92.3	96.6	85.7	**0.89**	92.3	100	100	**0.96**	100	98.28	92.9	**0.96**	92.3	100	100	**0.96**
SOP	44.4	100	100	**0.62**	100	96.36	81.8	**0.9**	88.9	98.4	88.9	**0.89**	100	100	90	**0.95**	100	100	100	**1**	100	98.28	90	**0.95**
UV	100	84.91	55.6	**0.72**	100	96.3	83.3	**0.91**	100	100	100	**1**	100	100	100	**1**	100	98.36	90.9	**0.95**	100	98.24	91	**0.95**
OV	100	94	72.7	**0.84**	100	98.18	88.9	**0.94**	100	100	100	**1**	100	100	100	**1**	100	100	100	**1**	100	98.31	89	**0.94**
OL	100	95.74	83.3	**0.91**	100	96.3	83.3	**0.91**	100	96.8	83.3	**0.91**	100	98.38	100	**1**	100	100	100	**1**	80.8	98.31	89	**0.85**
N	54.5	97.72	92.3	**0.69**	63.6	100	100	**0.78**	86.4	100	100	**0.93**	100	100	100	**1**	90.9	100	100	**0.95**	86	95.92	91	**0.88**

As in [1] neural networks are a promising alternative to various conventional classification methods. Since any classification procedure seeks a functional relationship between the group membership and the attributes of the object, accurate identification of this underlying function is must needed. Neural networks are flexible in modelling of real world complex relationships because of they can give extremely good nonlinear input output mapping. The major strength with neural network is its ability to extract the patterns and irregularities as well as detecting multi-dimensional nonlinear connection in data.

References

1. Bangal CB (2009) Automatic Generation Control of Interconnected Power Systems Using Artificial Neural Network Techniques. Ph.D. Thesis. Bharath University, Chennai, pp 30- 45, Available: http://shodhganga.inflibnet.ac.in/handle/10603/48
[Accessed 15 June 2015]

CHAPTER - 8

Conclusions and Future Scope

8.1 Conclusions

A brief chapter-wise summary of the contents of this thesis is as follows.

This study has focused on induction motor external faults identification using ANN and Fuzzy soft computing techniques and presented the need and advantages of such techniques in induction motor external fault identification in chapter 1.

Chapter 2 has presented literature survey mainly related with induction motor faults identification and all work are broadly classified in three categories ANN based, fuzzy logic based and hybrid and other approaches.

We have considered the most probable external faults OL, OV, UV SP and VUB in this study and all external faults are discussed in chapter 3. Three phase RMS voltage and RMS currents values are obtained using induction motor external faults simulation in MATLAB/SIMULINK environment at varying supply voltage and load. 174 training and 46 testing data sets (patterns) are prepared for six output (five external faults and normal) conditions. Scatter plot visualization of the training data sets show that the problem is linearly nonseperable and complex. We have also used conventional and widely used LDA for external faults identification. The training and testing (unseen) classification accuracy obtained with LDA is 70.11% and 73.9% respectively.

MLPNN and LM algorithm is used for the induction motor external faults identification in chapter 4. Three phase RMS voltages and RMS currents values obtained through simulation used as training and testing of MLPNN. Several MLPNN configurations are tested with growing neuron phenomena and the best generalized and well trained MLPNN configuration is evaluated using early stopping for generalisation. The total training and independent test data sets classification accuracies 98.9% and 97.8% are obtained respectively using MLPNN.

Real time input data sets are obtained practically for five external fault conditions simulated on operational induction motor using laboratory experimental setup and are discussed in chapter 5. OL, OV, UV, SP of any phase, VUB and normal conditions are practically created with varied operating voltage and load. RMS values of three phase voltages and currents are logged for classifiers training and testing. A representative set of 321 training and 72 independent test data sets is prepared for different classifier training and testing.

SC_FIS, ANFIS, PNN and NBC are explained in detail in chapter 6. External faults identification using MLPNN, SC_FIS, PNN, ANFIS, LDA and NBC is discussed using experimentally obtained data sets.

Chapter 7 has presented the performance comparison of MLPNN, SC_FIS, PNN, ANFIS, NBC and LDA classifiers using total classification accuracy, separate test data sets classification accuracy and average classification accuracy. Total classification accuracy for total training data sets and independent test data sets, and average classification accuracy obtained with MLPNN are 100%, 98.61, and 99.3% respectively. All classifiers are also compared using statistical measures like sensitivity, specificity, precision and F-measure for train and independent test data sets and discussed. Besides generalized and accurate fault identification advantage of SC_FIS and ANFIS is that it allows insights in the form of linguistically interpretable rules which is not possible with conventional fault identification schemes. MLPNN outperforms all others in terms of classification accuracy (Table 7.1) and overall F-measure for training and testing data sets (Tables 7.14 & 7.15).

The major contributions are briefly summarized as follows:

- This study evaluates the potential of mainly ANN and fuzzy logic techniques for induction motor external faults identification.
- Induction motor external faults simulation is used to simulate external faults alongwith normal operating conditions for varying operating voltage and load.
- Scatter plot visualization of the obtained train data sets is done for different output classes using six input variables (three phase RMS voltages and currents). Plot displays input variable relations with respect to six output classes and found the six classes linearly non separable, overlapping and complex. Classification results are

also obtained with conventional LDA technique.

- Real time data sets are obtained for five external faults and normal conditions at varying operating conditions for operational 3kW induction motor using experimental setup.
- This study used MLPNN for the induction motor accurate fault identification with fast LM BP algorithm and early stopping for generalization. To find the best generalized and well trained MLPNN configuration validation subset accuracy, independent test set and train subset classification accuracy alongwith validation subset error are considered. Different MLPNN configurations are tested using growing neuron phenomena and the best generalized well trained MLPNN configuration.
- Subtractive clustering based fuzzy inference system (SC_FIS) is used for external faults identification and the rules responsible for the five external faults and normal conditions are obtained. Best generalized FIS configuration with least rules is found by comparing average total classification accuracy and average test RMSE error using 10-times random subsampling for different FISs obtained with different cluster radius. High classification accuracy results are obtained for training as well as unseen patterns using SC_FIS without need of any iterative nonlinear parameter optimization like ANFIS with least rules.
- This study attempted to find the best generalized and well trained neural network PNN configuration for induction motor external faults detection and classification. The results of PNN train and validation (subset of total train data sets) data sets classification accuracy are used against radial basis function spread to find the best generalization spread for PNN.
- ANFIS is used for external faults identification. Best generalized ANFIS configuration with least rules is obtained using independent test data sets as checking data sets.
- This study also attempted conventional LDA and probabilistic NBC for external faults identification alongwith neural network and fuzzy classifiers for statistical performance comparison.
- This study has compared faults classification performance of classifiers using train and test classification accuracy and other statistical measures like sensitivity, specificity, precision and F-measure.

Induction Motors are a major part of industrial load and appears to various faults and abnormalities. Induction motor external faults identification is a complex and linearly non separable problem. Conventional fault identification schemes suffer from inaccurate fault identification. In conventional protective schemes relays, applied for one hazard may operate for others as some overlap found particularly in OL versus faults, unbalance voltages/currents and SP etc. It is also difficult to estimate negative sequence current for negative sequence protection. The current development of computer software based on intelligent systems components leads attention of relay engineers to use them in the diagnosis of faults in power system components such as induction motors. Neural network provide a natural framework for fault identification and it can approximate abnormal behaviour of dynamial systems through learning approach. Fuzzy logic can be used to provide a general heuristic solution to a particular problem. It can provide a heuristic output as a result of some complex computations by quantifying the actual numerical data into heuristic and linguistic terms.

Real time input data sets for classifiers training are obtained through various external fault conditions practically simulated on induction motor. OL, OV, UV, SP of any phase, VUB and normal condition were practically created with varied operating voltage and load. RMS values of three phase voltage and currents were logged as feature vector of classifier. Performance of MLPNN PNN, SC_FIS, ANFIS, NBC and LDA classifiers were compared using total classification accuracy (Table 7.1), sensitivity, specificity, precision and F-measure for training data sets (Table 7.14) and independent test data sets (Table 7.15). It is observed that MLPNN, SC_FIS, PNN and ANFIS results are found more accurate and better than conventional Well-known LDA and probabilistic approach Naïve bayes classifier. MLPNN and PNN show most impressive results with respect to training data sets accuracy 100 % & and 99.07% respectively. PNN can identify external faults with good amount of training and testing accuracy but requires as many hidden neurons as training patterns. Soft computing classifiers SC_FIS and ANFIS can provide heuristics behind faults in terms of rules besides high fault identification accuracy and other statistical performance measures. MLPNN and SC_FIS generalization performance found better based on independent test data sets classification accuracies (98.61% and 97.2% respectively) and other statistical measures for all six output conditions. It has been observed MLPNN performance outperforms to others in terms of all statistical performance measures for training and independent test data sets.

8.2 Future Scope

1. Till the date there is no AI based single methodology is available as in authors knowledge to detect all faults (external and internal), a comprehensive soft computing based fault identification scheme should be developed for identification for all external as well as internal faults at different load and operating voltage levels.

2. The proposed fault identification scheme to be developed and test with instantaneous values of three phase and voltages,

3. ANN or fuzzy based fault identification system to be develop for other power system components like, transformer, synchronous generator etc; using simple input variables like RMS or instantaneous values of three phase voltages and currents which are readily available through measurement instruments.

4. The reason of ANFIS little lower performance may be because of many nonlinear parameters are to modify and also the ratio of number of training patterns to total parameters is less. External faults identification using ANFIS can be tested with selected features or other Adaptive neurofuzzy systems and with large data sets.

APPENDICES

APPENDIX A: Training Data sets (Simulation)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
1	402.7	402.9	403.2	7.75	7.75	7.75	N
2	399.9	400	400.3	7.78	7.78	7.78	N
3	410.2	410.3	410.4	7.68	7.68	7.68	N
4	413.8	414	414.1	7.65	7.65	7.65	N
5	420	420.1	420.2	7.6	7.6	7.6	N
6	423.4	423.8	424.1	7.6	7.58	7.58	N
7	428.3	428.7	429	7.54	7.53	7.53	N
8	432	432.4	432.7	7.52	7.51	7.51	N
9	434.5	434.8	435	7.49	7.49	7.49	N
10	438.3	438.5	438.6	7.47	7.47	7.47	N
11	395.5	395.6	395.6	7.83	7.83	7.83	N
12	391.7	391.9	392.2	7.87	7.86	7.87	N
13	386.7	387	387.3	7.93	7.92	7.92	N
14	381.9	382.1	382.3	7.99	7.97	7.99	N
15	377.1	377.3	377.3	8.1	8.1	8.1	N
16	367.4	367.4	367.4	8.2	8.2	8.2	N
17	372	372.3	372.7	8.14	8.14	8.14	N
18	363.4	363.8	364.1	8.29	8.27	8.28	N
19	430.9	431.1	431.3	7.52	7.53	7.52	N
20	366.1	366.2	366.3	8.22	8.22	8.22	N
21	404	404.1	404.3	8.61	8.61	8.61	N
22	406.4	406.7	406.8	8.57	8.57	8.57	N
23	398.9	399.3	399.6	6.5	6.5	6.5	N
24	399	399.3	399.5	7.13	7.12	7.12	N
25	399	399.2	399.5	7.57	7.56	7.56	N
26	403	410.3	407.3	7.15	7.94	8.09	N
27	395.6	388.2	392.6	8.51	7.6	7.53	N
28	405.7	397.7	401.7	7.55	7.41	8.37	N
29	402.5	397.9	406	7.11	8.02	8.19	N
30	403.6	407.6	411.8	7.5	7.33	8.33	N
31	399.6	400	400.3	9	9	9	OL
32	399.1	399.2	399.4	9.2	9.2	9.2	OL
33	399.2	399.3	399.3	9.7	9.7	9.7	OL
34	398.9	399.3	399.8	9.95	9.95	9.96	OL
35	399.6	400.3	400	10.46	10.45	10.45	OL
36	399.7	402.4	400	10.2	10.2	10.2	OL
37	399.6	400.3	400	10.72	10.71	10.71	OL
38	399.7	402.4	400	11	11	11	OL
39	403.9	404.2	404.5	9.86	9.86	9.86	OL
40	391.6	392	392.2	10.15	10.13	10.14	OL

APPENDIX A: Training Data Sets (Simulation)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
41	396.6	397.1	396.8	9.24	9.24	9.25	OL
42	406.3	406.9	406.7	8.63	8.61	8.62	N
43	399.7	400.3	400	11.25	11.25	11.25	OL
44	399.8	400.2	400	11.56	11.55	11.56	OL
45	400	399.6	400	11.84	11.82	11.83	OL
46	399.7	400.3	400.1	12.11	12.11	12.11	OL
47	399.8	400.2	400	12.4	12.4	12.4	OL
48	400	399.7	400.3	12.67	12.66	12.67	OL
49	399.9	400.1	400	12.95	12.95	12.96	OL
50	399.7	400	400.3	9.32	9.32	9.32	OL
51	441.8	442.1	442.4	8.02	8.03	8.02	OV
52	443.1	443.3	443.6	8.01	8.02	8.01	OV
53	444.8	445.2	445.6	8	8	8	OV
54	446.8	447	447.2	7.98	7.99	7.98	OV
55	444.2	444.6	445	7.44	7.444	7.43	OV
56	451.8	451.9	452.1	7.38	7.4	7.4	OV
57	456.5	456.8	457.2	7.35	7.35	7.37	OV
58	471.5	471.5	471.6	7.34	7.25	7.36	OV
59	474.8	475.2	475.5	7.37	7.26	7.37	OV
60	444.3	444.6	444.8	7.44	7.44	7.44	OV
61	455.2	455.6	455.9	7.39	7.39	7.38	OV
62	468.9	469.1	469.2	7.34	7.34	7.34	OV
63	477.2	477.7	478.1	7.33	7.32	7.31	OV
64	479.6	480.1	480.5	7.32	7.31	7.31	OV
65	489.6	489.9	490.2	7.93	7.86	7.84	OV
66	452.8	453.2	453.5	6.38	6.4	6.4	OV
67	452.7	453.2	453.5	9.12	9.17	9.12	OV
68	446.6	447	447.4	7.42	7.43	7.42	OV
69	453.1	453.2	453.2	8.13	8.12	8.12	OV
70	443.2	443.3	443.6	6.4	6.4	6.4	OV
71	439.7	440	440.2	7.46	7.46	7.46	OV
72	442	442.1	442.3	7.46	7.46	7.46	OV
73	445	445.8	446.2	7.43	7.43	7.43	OV
74	449.1	449.5	449.8	7.41	7.41	7.41	OV
75	452.8	453.2	453.5	7.39	7.38	7.39	OV
76	458	458.1	458.2	7.37	7.36	7.38	OV
77	461.3	461.7	462.1	7.37	7.35	7.37	OV
78	467.4	467.4	467.9	7.37	7.33	7.35	OV
79	472.6	472.8	472.8	7.36	7.34	7.35	OV
80	482.3	482.5	482.8	7.34	7.34	7.3	OV

APPENDIX A: Training Data Sets (Simulation)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
81	361.1	361.3	361.4	8.3	8.3	8.3	UV
82	357.3	357.6	357.9	8.37	8.35	8.36	UV
83	352.4	352.7	353	8.48	8.46	8.47	UV
84	342.6	343	343.2	8.67	8.66	8.67	UV
85	336.7	336.8	337	8.8	8.8	8.8	UV
86	330.4	330.7	330.9	8.9	8.91	8.92	UV
87	323.2	323.3	323.4	9.13	9.124	9.13	UV
88	318.2	318.4	318.6	9.24	9.23	9.23	UV
89	312.1	312.3	312.5	9.44	9.43	9.43	UV
90	306	306.2	306.3	9.6	9.61	9.6	UV
91	293.7	294	294.1	10.1	10.1	10.1	UV
92	355	355.2	355.4	8.41	8.4	8.41	UV
93	345.4	345.4	345.4	8.6	8.6	8.6	UV
94	350.2	350.3	350.3	7.93	7.93	7.93	UV
95	340.3	340.5	340.6	9.76	9.75	9.76	UV
96	330.4	330.7	330.9	9.9	9.89	9.9	UV
97	352.4	352.7	353	9.32	9.31	9.32	UV
98	306.1	306.2	306.3	13.25	13.24	13.25	UV
99	342.8	343	343	6.94	6.94	6.94	UV
100	355	355.2	355.4	10.16	10.15	10.15	UV
101	284.4	400	362.8	0	16.75	16.75	SP
102	286.8	402.9	365.3	0	16.65	16.65	SP
103	305.4	416.4	377.8	0	16.13	16.13	SP
104	273	391.9	355.2	0	17	17	SP
105	256.6	379.7	343	0	17.45	17.45	SP
106	268.4	400	361.5	0	18.83	18.83	SP
107	252.2	400	358.9	0	21	21	SP
108	302.2	400	363.6	0	14.1	14.1	SP
109	369.2	399.3	400	11.7	0	11.7	SP
110	371.1	305.1	400.1	14.12	0	14.12	SP
111	371.1	275.5	400.2	18.7	0	18.7	SP
112	370.3	260.9	400.2	20.84	0	20.84	SP
113	371.4	288.1	400.2	16.73	0	16.73	SP
114	374	291.2	403.1	16.67	0	16.67	SP
115	386.1	277.9	392.1	16.99	0	16.99	SP
116	352.2	263.3	379.9	17.28	0	17.28	SP
117	399.7	368.2	283.1	17.26	17.26	0	SP
118	402.6	371.1	287.1	17.2	17.2	0	SP
119	416.1	384.1	305.6	16.7	16.7	0	SP
120	391.6	360	272.7	17.46	17.46	0	SP

APPENDIX A: Training Data Sets (Simulation)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
121	379.5	347.4	256.9	17.77	17.77	0	SP
122	399.7	366.9	268.4	19.34	19.34	0	SP
123	399.8	364.8	252.7	21.45	21.45	0	SP
124	399.7	368.5	319.1	12.14	12.14	0	SP
125	399.9	368.9	361.7	14.61	14.61	0	SP
126	409.6	389.5	400	9.46	7.25	6.96	VUB
127	407.5	400	391.9	8.41	8.65	6.46	VUB
128	409.6	400	389.5	8.54	8.9	6.14	VUB
129	411.5	395.9	391.7	9.06	8.46	6.1	VUB
130	411.4	391.7	396.1	9.41	7.83	6.45	VUB
131	407.6	407.8	383.1	7.92	10	6.06	VUB
132	407.5	395.9	396.2	8.74	7.99	6.75	VUB
133	406.7	406.4	383.5	7.95	9.86	6.11	VUB
134	409.7	392.8	390.4	9.2	8.4	6.11	VUB
135	383.7	391.6	398.1	7.36	9.09	7.34	VUB
136	394.1	399.7	405.9	7.25	8.77	7.4	VUB
137	411.2	400	387.6	8.78	9.14	5.82	VUB
138	413	385.1	400.2	10.14	7.13	6.75	VUB
139	409.3	385.2	404.3	9.9	6.12	7.38	VUB
140	416.5	391.7	389.8	9.91	8.62	5.49	VUB
141	390	400	390.5	7.02	8.62	8	VUB
142	360.5	400	361	5.07	11.53	9.19	VUB
143	351.1	351.3	400.1	9.74	4.64	12.57	VUB
144	399.9	370.4	370.5	10.54	8.72	5.62	VUB
145	364.3	364.6	400.4	9.01	5.27	11.14	VUB
146	357.1	399.7	356.9	11.95	9.41	4.88	VUB
147	354.1	364.6	390.3	8.1	6.31	11.17	VUB
148	351	320.2	370.6	11.62	4.57	10.93	VUB
149	364.6	392.9	371.8	9.91	9.13	5.73	VUB
150	389.5	361.5	372.1	10.47	7.75	6.73	VUB
151	368.5	387.5	356.2	6.82	10.76	7.66	VUB
152	383.9	361.2	317.8	7	7.71	9.99	VUB
153	385.7	367.5	373.8	9.6	7.8	7.1	VUB
154	404.4	349.2	387.3	12.72	6.48	7.61	VUB
155	367.4	373.7	349.3	7.96	10.32	6.99	VUB
156	358.9	382.8	343.3	6.74	11.61	7.83	VUB
157	414.1	400	414.7	8.99	6.65	7.65	VUB
158	426.8	400	426.9	10.06	5.86	7.71	VUB
159	399.9	414.4	414.5	6.66	7.64	8.98	VUB
160	399.8	423.2	423.3	6.1	7.67	9.73	VUB

APPENDIX A: Training Data Sets (Simulation)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
161	410.9	411.3	400.3	7.66	8.71	6.88	VUB
162	446.7	426.8	420.9	9.15	8.7	5.24	VUB
163	440.9	433.1	408.3	8.3	10.04	5.16	VUB
164	414.2	426.8	441.2	7.2	6.31	9.74	VUB
165	429.8	423.8	453.3	8.82	4.95	9.36	VUB
166	436.1	462.4	426.9	5.18	10.36	7.94	VUB
167	420.3	408.3	429	8.92	5.98	8.24	VUB
168	418.8	434.8	425.2	6.23	8.28	8.34	VUB
169	425.6	419.6	441.1	8.51	5.72	8.72	VUB
170	427.3	433.7	447.3	7.48	6.26	9.06	VUB
171	425.9	420.1	432.7	6.6	7.93	8.09	VUB
172	395.5	394	382.3	8.1	8.95	6.83	VUB
173	397.7	362.1	404	11.25	4.98	8.95	VUB
174	427.6	399.2	408.2	10.03	7.67	6.02	VUB

APPENDIX B: Testing Data Sets (Simulation)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
1	389.4	389.5	389.6	7.9	7.9	7.9	N
2	384.2	384.6	384.9	7.96	7.94	7.9	N
3	379.6	379.7	379.7	8	8	8	N
4	405.1	405.4	405.6	7.74	7.72	7.73	N
5	398.3	394.4	396.1	8.15	7.72	7.6	N
6	369.7	369.9	370	8.16	8.16	8.16	N
7	399.7	400	400.2	9.45	9.43	9.44	OL
8	399.7	400	400.2	9.57	9.56	9.57	OL
9	399.7	400	400.3	9.82	9.81	9.81	OL
10	399.6	400	400.3	10.1	10.1	10.1	OL
11	399.7	400	400	10.33	10.32	10.33	OL
12	399.8	400	400.2	9.19	9.19	9.19	OL
13	443.2	443.4	443.5	7.44	7.45	7.44	OV
14	448.1	448.2	448.5	7.42	7.42	7.42	OV
15	450.3	450.7	451.1	7.42	7.41	7.41	OV
16	459.6	460	460.2	7.38	7.37	7.37	OV
17	470.2	470.3	470.3	7.34	7.34	7.34	OV
18	465	465.4	465.7	7.36	7.35	7.35	OV
19	347.6	347.8	348	8.55	8.54	8.54	UV
20	325.5	326.8	326	9.04	9.03	9.04	UV
21	332.8	333.1	333.4	8.86	8.85	8.86	UV
22	340.4	340.5	340.6	8.7	8.7	8.7	UV
23	320.6	320.9	321.1	9.17	9.16	9.1	UV
24	296.6	410.3	372.2	0	16.4	16.4	SP
25	292.1	406.6	368.8	0	16.54	16.54	SP
26	250.4	374.8	338.3	0	17.66	17.66	SP
27	300.2	412.7	374.6	0	16.26	16.26	SP
28	258	400	360	0	20.26	20.26	SP
29	370.6	265.7	400.4	20.1	0	20.1	SP
30	380.8	301.1	410.5	16.4	0	16.4	SP
31	377.3	295.3	406.8	16.5	0	16.5	SP
32	368.5	284	397	16.84	0	16.84	SP
33	361.5	274.8	389.7	17.1	0	17.1	SP
34	410	378.3	297.1	17	17	0	SP
35	392.5	361.5	276.8	17	17	0	SP
36	384.3	352.4	263.6	17.6	17.6	0	SP
37	399.8	368.7	292.7	15.9	15.9	0	SP
38	399.8	365	255.3	21	21	0	SP
39	385.5	412.9	400	8.98	9.4	5.45	VUB
40	407.9	405.8	385.2	9.67	6.14	8.01	VUB

APPENDIX B: Testing Data Sets (Simulation)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
41	377.5	400	378.1	6.11	9.8	8.4	VUB
42	365.9	389.9	355.5	6.4	11.04	8.02	VUB
43	355.1	376.6	358.7	6.53	9.81	8.8	VUB
44	410.9	411.3	400.3	7.66	8.71	6.88	VUB
45	430.2	420.4	450.1	5	9.14	8.98	VUB
46	423.6	438.5	433.8	6.37	7.82	8.52	VUB

APPENDIX C: Training Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
1	410.161	407.756	408.882	4.636	4.527	4.759	N
2	409.854	407.577	408.549	4.623	4.541	4.759	N
3	410.213	407.833	408.908	4.623	4.527	4.759	N
4	410.213	407.756	408.882	4.677	4.568	4.814	N
5	410.187	408.063	409.138	4.568	4.445	4.705	N
6	410.571	408.677	409.65	4.5	4.377	4.636	N
7	414.051	411.799	413.027	4.5	4.377	4.609	N
8	419.68	417.351	418.784	4.486	4.364	4.595	N
9	424.311	421.957	423.671	4.5	4.336	4.582	N
10	425.002	422.955	424.49	4.486	4.309	4.582	N
11	424.746	423.057	424.362	4.473	4.309	4.595	N
12	427.765	425.207	427.1	4.5	4.309	4.568	N
13	429.633	427.202	429.019	4.5	4.309	4.568	N
14	433.215	430.682	433.113	4.5	4.295	4.582	N
15	434.623	431.987	434.162	4.5	4.295	4.555	N
16	436.516	433.957	435.723	4.486	4.323	4.568	N
17	439.177	436.67	438.384	4.5	4.309	4.568	N
18	444.295	441.762	443.45	4.5	4.323	4.582	N
19	443.911	441.506	443.092	4.5	4.323	4.582	N
20	444.653	442.299	443.988	4.5	4.309	4.582	N
21	445.548	442.913	444.653	4.5	4.323	4.568	N
22	447.032	444.525	446.111	4.5	4.336	4.595	N
23	450.845	448.44	450.103	4.527	4.336	4.595	N
24	451.51	448.772	450.538	4.527	4.35	4.595	N
25	451.638	448.9	450.64	4.527	4.35	4.595	N
26	453.378	450.589	452.483	4.541	4.35	4.609	N
27	454.427	451.92	453.915	4.555	4.35	4.609	N
28	448.644	446.035	447.902	4.5	4.309	4.568	N
29	442.248	439.74	441.736	4.486	4.282	4.555	N
30	420.089	416.968	419.143	4.486	4.309	4.527	N
31	420.115	416.814	419.091	4.5	4.309	4.527	N
32	417.428	414.793	416.609	4.473	4.295	4.541	N
33	412.618	410.699	411.825	4.459	4.336	4.595	N
34	412.311	410.699	411.876	4.459	4.323	4.595	N
35	412.388	410.801	411.978	4.459	4.323	4.609	N
36	406.861	404.609	406.016	4.473	4.336	4.582	N
37	403.355	401.104	401.897	4.459	4.364	4.595	N
38	402.409	400.08	400.873	4.473	4.377	4.595	N
39	395.986	393.351	394.656	4.514	4.405	4.609	N
40	392.276	389.717	390.997	4.541	4.432	4.65	N

APPENDIX C: Training Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
41	389.18	386.365	387.517	4.568	4.473	4.664	N
42	388.617	385.291	386.672	4.595	4.5	4.664	N
43	387.107	384.242	385.623	4.582	4.473	4.677	N
44	385.291	382.476	384.088	4.609	4.486	4.705	N
45	384.472	381.402	383.014	4.609	4.5	4.691	N
46	382.169	379.355	380.685	4.623	4.514	4.732	N
47	380.404	377.768	378.92	4.623	4.527	4.732	N
48	379.508	376.642	378.05	4.636	4.527	4.745	N
49	378.382	375.491	376.847	4.664	4.555	4.759	N
50	378.178	375.363	376.591	4.65	4.555	4.759	N
51	442.478	439.663	441.736	4.35	4.145	4.405	N
52	444.064	441.199	443.092	4.35	4.173	4.391	N
53	446.751	443.757	446.035	4.364	4.159	4.405	N
54	448.184	444.883	447.416	4.377	4.159	4.405	N
55	450.973	447.775	450.129	4.391	4.186	4.432	N
56	452.738	450.001	452.175	4.391	4.2	4.473	N
57	439.791	436.926	439.228	4.336	4.132	4.377	N
58	436.26	433.395	435.544	4.309	4.132	4.364	N
59	431.22	428.738	430.861	4.295	4.118	4.377	N
60	431.143	428.891	430.938	4.295	4.118	4.377	N
61	401.052	397.777	399.415	4.295	4.186	4.364	N
62	401.308	398.033	399.62	4.309	4.2	4.377	N
63	401.283	397.982	399.594	4.309	4.186	4.377	N
64	401.18	397.88	399.543	4.323	4.2	4.377	N
65	426.614	424.465	426.64	3.968	3.805	4.05	N
66	426.537	424.337	426.563	3.982	3.818	4.05	N
67	403.458	401.385	402.895	3.927	3.832	4.036	N
68	396.728	393.76	395.372	3.968	3.873	4.009	N
69	394.425	391.022	392.609	3.982	3.9	3.995	N
70	389.82	386.493	388.105	3.995	3.914	4.023	N
71	387.082	383.372	385.137	4.023	3.927	4.036	N
72	384.139	380.864	382.399	4.036	3.941	4.05	N
73	378.766	375.363	376.975	4.05	3.968	4.077	N
74	377.052	374.212	375.593	4.677	4.568	4.773	N
75	377.436	374.109	375.721	4.677	4.568	4.745	N
76	377.461	374.237	375.619	4.677	4.582	4.745	N
77	447.109	444.346	447.135	4.05	3.832	4.077	N
78	444.806	442.555	444.499	3.995	3.832	4.064	N
79	444.551	441.429	444.039	4.036	3.832	4.036	N
80	423.364	421.42	422.264	3.886	3.791	3.982	N

APPENDIX C: Training Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
81	422.904	421.445	422.443	3.886	3.791	4.009	N
82	416.405	415.304	416.149	3.818	3.723	3.968	N
83	417.531	415.458	416.763	3.573	3.477	3.655	N
84	417.3	415.407	416.712	3.573	3.477	3.655	N
85	417.275	415.381	416.661	3.559	3.464	3.655	N
86	416.405	414.614	415.842	3.791	3.695	3.886	N
87	416.686	414.869	416.123	3.573	3.464	3.668	N
88	416.865	414.972	416.302	3.573	3.464	3.668	N
89	416.942	414.972	416.379	3.586	3.464	3.668	N
90	416.865	414.869	416.354	3.559	3.45	3.641	N
91	417.428	415.484	417.07	3.273	3.15	3.327	N
92	418.375	416.302	417.889	2.632	2.495	2.632	N
93	410.494	408.191	409.317	4.895	4.786	5.045	N
94	410.724	408.37	409.522	4.895	4.773	5.032	N
95	410.468	408.345	409.343	4.868	4.745	5.018	N
96	454.504	451.05	453.992	4.5	4.241	4.5	N
97	454.785	451.843	454.785	4.445	4.173	4.486	N
98	455.22	451.408	454.734	4.527	4.241	4.5	N
99	455.297	451.306	455.092	4.555	4.241	4.5	N
100	455.451	451.51	454.964	4.541	4.241	4.5	N
101	406.707	410.213	407.347	3.995	4.214	4.118	N
102	411.876	408.217	405.709	4.05	3.886	4.295	N
103	406.886	411.031	407.807	3.941	4.214	4.064	N
104	412.695	409.701	408.191	4.036	3.886	4.2	N
105	412.311	409.343	407.859	4.241	4.091	4.391	N
106	412.618	411.134	410.903	4.486	4.391	4.514	N
107	412.234	410.98	410.776	4.473	4.405	4.5	N
108	407.219	405.888	405.658	4.473	4.391	4.514	N
109	407.296	406.221	405.709	4.323	4.268	4.377	N
110	407.705	405.146	403.023	4.2	4.105	4.377	N
111	407.577	405.76	405.735	4.173	4.077	4.214	N
112	407.577	405.658	405.658	4.173	4.105	4.227	N
113	406.707	404.302	404.481	2.918	2.836	2.986	N
114	407.04	404.532	404.686	2.918	2.836	3	N
115	406.375	402.05	398.468	2.864	2.686	3.177	N
116	406.042	404.123	404.072	2.905	2.85	2.973	N
117	419.859	418.375	420.064	2.7	2.55	2.605	N
118	419.833	418.324	419.91	2.714	2.536	2.605	N
119	421.676	420.115	421.778	2.414	2.264	2.305	N
120	414.383	413.078	413.974	2.659	2.523	2.55	N

APPENDIX C: Training Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
121	408.089	406.426	406.989	2.591	2.427	2.509	N
122	407.142	404.865	405.198	2.591	2.4	2.523	N
123	404.507	402.588	403.56	2.223	2.059	2.073	N
124	412.336	410.776	411.39	2.291	2.155	2.168	N
125	403.611	401.411	402.588	2.577	2.386	2.468	N
126	402.895	399.927	401.155	2.673	2.441	2.564	N
127	401.922	399.338	400.31	2.768	2.55	2.673	N
128	395.219	392.506	394.144	4.036	3.886	3.777	N
129	395.116	392.378	393.99	4.036	3.886	3.75	N
130	396.37	394.528	395.858	3.941	3.777	3.709	N
131	401.667	399.799	401.052	3.927	3.777	3.695	N
132	408.831	406.912	408.268	3.927	3.777	3.695	N
133	412.899	411.031	412.285	3.914	3.764	3.695	N
134	414.767	413.206	414.255	3.914	3.764	3.695	N
135	415.97	414.204	415.407	3.927	3.764	3.709	N
136	407.859	405.633	406.835	3.914	3.777	3.682	N
137	404.404	403.048	403.816	3.9	3.764	3.709	N
138	416.021	414.153	415.739	2.536	2.318	2.345	N
139	416.891	415.535	417.07	2.373	2.127	2.223	N
140	408.242	407.04	408.754	2.236	2.018	2.127	N
141	409.547	407.347	408.575	4.991	4.841	5.141	OL
142	409.65	407.372	408.473	4.964	4.841	5.127	OL
143	410.468	407.526	409.138	4.991	4.841	5.1	OL
144	410.494	407.91	409.317	4.977	4.841	5.1	OL
145	409.803	407.833	408.959	4.964	4.827	5.127	OL
146	410.034	407.91	408.908	4.964	4.841	5.127	OL
147	409.854	407.833	408.78	4.964	4.841	5.141	OL
148	409.522	407.577	408.549	5.127	5.005	5.318	OL
149	409.445	407.5	408.524	5.291	5.168	5.482	OL
150	409.266	407.424	408.242	5.277	5.168	5.482	OL
151	409.061	407.193	408.063	5.291	5.168	5.482	OL
152	409.471	407.424	408.396	5.291	5.168	5.468	OL
153	409.931	407.91	408.857	4.964	4.841	5.114	OL
154	409.215	407.372	408.242	4.95	4.827	5.114	OL
155	409.522	407.372	408.549	4.964	4.827	5.1	OL
156	409.547	407.372	408.575	4.95	4.827	5.1	OL
157	409.547	407.296	408.524	4.964	4.827	5.086	OL
158	409.419	407.193	408.37	4.95	4.827	5.1	OL
159	409.24	407.245	408.294	4.936	4.814	5.086	OL
160	409.496	407.449	408.601	4.95	4.814	5.086	OL

APPENDIX C: Training Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
161	409.599	407.398	408.626	4.964	4.814	5.086	OL
162	410.059	408.063	409.24	5.195	5.059	5.345	OL
163	410.008	407.807	409.01	5.059	4.923	5.209	OL
164	415.356	413.795	414.614	5.305	5.223	5.523	OL
165	415.228	413.616	414.511	5.414	5.318	5.632	OL
166	415.125	413.513	414.409	5.536	5.441	5.768	OL
167	414.972	413.437	414.255	5.536	5.441	5.768	OL
168	414.818	413.257	414.102	5.536	5.441	5.768	OL
169	414.793	413.181	414.153	5.727	5.632	5.959	OL
170	414.793	413.104	414.051	5.727	5.645	5.959	OL
171	458.828	456.679	458.265	4.568	4.377	4.664	OV
172	459.007	456.576	458.342	4.582	4.377	4.65	OV
173	458.572	455.809	458.009	4.582	4.35	4.623	OV
174	461.361	458.265	461.054	4.555	4.268	4.582	OV
175	462.231	459.033	461.643	4.555	4.295	4.582	OV
176	464.764	461.796	464.534	4.595	4.309	4.609	OV
177	462.487	459.033	462.231	4.568	4.282	4.568	OV
178	462.717	459.135	462.206	4.595	4.295	4.595	OV
179	455.86	453.352	455.451	4.405	4.2	4.486	OV
180	456.909	454.248	456.679	4.432	4.214	4.5	OV
181	458.521	455.144	457.856	4.459	4.227	4.486	OV
182	459.238	456.116	458.623	4.459	4.227	4.5	OV
183	460.107	457.523	459.852	4.459	4.227	4.514	OV
184	460.363	457.805	460.005	4.459	4.227	4.527	OV
185	460.619	458.214	460.338	4.459	4.227	4.541	OV
186	460.594	458.112	460.235	4.459	4.241	4.527	OV
187	460.389	457.805	459.98	4.459	4.227	4.514	OV
188	460.261	457.651	459.8	4.459	4.241	4.514	OV
189	460.312	457.702	459.749	4.459	4.241	4.514	OV
190	460.875	457.856	460.466	4.486	4.227	4.514	OV
191	461.719	458.7	461.157	4.486	4.241	4.527	OV
192	462.103	459.365	461.412	4.473	4.255	4.527	OV
193	462.513	459.314	461.899	4.5	4.241	4.5	OV
194	457.37	456.193	457.856	4.077	3.914	4.227	OV
195	457.958	456.781	458.393	4.105	3.927	4.227	OV
196	464.995	462.922	465.276	4.214	3.995	4.282	OV
197	465.916	463.536	466.069	4.227	3.982	4.282	OV
198	465.711	463.383	465.993	4.227	3.995	4.282	OV
199	457.139	453.352	456.73	4.555	4.255	4.527	OV
200	460.415	457.063	459.749	4.555	4.282	4.555	OV

APPENDIX C: Training Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
201	375.951	373.649	374.391	4.036	3.995	4.145	UV
202	376.028	373.725	374.365	4.036	3.995	4.145	UV
203	376.054	372.446	374.032	4.064	4.009	4.105	UV
204	376.719	374.212	375.44	4.664	4.568	4.786	UV
205	376.131	373.29	374.314	4.05	3.995	4.132	UV
206	376.233	372.6	374.288	4.077	3.995	4.105	UV
207	373.265	370.962	371.96	4.691	4.595	4.827	UV
208	373.521	371.013	372.19	4.691	4.582	4.814	UV
209	373.674	371.039	372.395	4.705	4.595	4.8	UV
210	373.7	371.192	372.472	4.691	4.582	4.814	UV
211	373.495	371.064	372.19	4.691	4.595	4.814	UV
212	373.418	370.757	371.985	4.677	4.582	4.8	UV
213	373.316	370.706	371.832	4.691	4.595	4.814	UV
214	370.578	367.789	368.838	4.718	4.623	4.827	UV
215	366.433	363.695	364.796	4.745	4.664	4.882	UV
216	363.593	361.162	362.211	4.786	4.691	4.936	UV
217	363.849	361.623	362.825	4.786	4.677	4.95	UV
218	363.67	361.572	362.876	4.786	4.664	4.95	UV
219	370.45	367.38	368.736	4.745	4.65	4.841	UV
220	376.77	373.086	374.647	4.677	4.582	4.732	UV
221	362.8	359.371	361.367	4.732	4.609	4.8	UV
222	362.723	358.834	361.111	4.745	4.609	4.786	UV
223	362.467	358.936	361.111	4.732	4.595	4.8	UV
224	362.365	358.808	361.111	4.745	4.609	4.8	UV
225	362.646	359.09	361.367	4.732	4.609	4.8	UV
226	374.877	371.397	372.446	4.05	4.036	4.118	UV
227	371.832	368.429	369.427	4.077	4.036	4.132	UV
228	365.051	362.109	362.953	4.118	4.091	4.2	UV
229	364.949	361.853	362.749	4.132	4.105	4.214	UV
230	373.239	370.783	371.934	4.05	3.995	4.145	UV
231	349.699	402.127	337.264	0.027	4.105	4.036	SP
232	367.687	416.149	359.448	0.491	3.955	4.009	SP
233	350.057	388.975	308.018	1.418	6.668	6.668	SP
234	323.191	386.98	258.481	0.041	8.468	8.536	SP
235	355.303	399.62	317.945	0.955	6.027	6.082	SP
236	339.157	398.647	290.849	0	6.805	6.859	SP
237	339.029	398.519	290.67	0.014	6.805	6.859	SP
238	344.71	403.227	298.013	0.055	6.641	6.695	SP
239	343.609	403.125	297.425	0.014	6.682	6.736	SP
240	377.308	412.797	349.981	1.582	5.264	5.277	SP

APPENDIX C: Training Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
241	351.209	411.543	309.527	0.014	6.45	6.491	SP
242	362.416	419.936	339.362	0.041	5.1	5.155	SP
243	362.262	419.833	339.413	0	5.1	5.141	SP
244	362.288	419.808	339.234	0	5.1	5.155	SP
245	322.398	402.306	243.078	0.668	11.673	11.768	SP
246	407.731	343.43	354.484	4.036	0.041	4.145	SP
247	407.5	345.989	356.173	4.023	0.491	4.159	SP
248	407.731	343.43	354.484	4.036	0.041	4.145	SP
249	407.5	345.989	356.173	4.023	0.491	4.159	SP
250	262.754	326.005	388.924	8.468	0.136	8.318	SP
251	260.963	324.88	388.643	8.577	0.123	8.441	SP
252	297.246	344.044	400.694	6.818	0.314	6.682	SP
253	327.157	361.162	406.042	6.068	1.132	5.932	SP
254	299.702	345.452	405.198	6.845	0.095	6.723	SP
255	300.06	345.528	405.325	6.845	0.095	6.709	SP
256	311.6	353.025	413.641	6.668	0.095	6.505	SP
257	355.891	375.209	422.111	4.909	0.859	4.773	SP
258	340.897	363.951	421.65	5.318	0.068	5.182	SP
259	348.189	357.785	411.211	4.145	4.105	0.055	SP
260	324.035	345.58	397.752	4.609	4.555	0.068	SP
261	323.677	345.273	397.445	4.609	4.541	0.068	SP
262	392.353	293.945	343.277	7.514	7.336	0.941	SP
263	391.201	266.08	327.669	8.455	8.25	0.027	SP
264	401.462	327.285	361.827	6.014	5.85	1.282	SP
265	400.234	293.228	340.897	7.036	6.859	0.014	SP
266	400.31	293.203	340.872	7.036	6.845	0.014	SP
267	406.349	302.44	347.115	6.832	6.641	0.027	SP
268	406.477	302.235	346.654	6.859	6.682	0	SP
269	414.46	326.415	362.851	6.273	6.082	0.723	SP
270	413.897	312.879	353.384	6.668	6.477	0.014	SP
271	421.829	351.516	371.653	5.073	4.909	0.559	VUB
272	375.184	409.01	376.642	3.218	5.686	4.841	VUB
273	397.726	410.238	402.409	3.791	4.582	4.05	VUB
274	397.547	410.289	402.204	3.777	4.595	4.064	VUB
275	396.856	410.187	401.641	3.75	4.609	4.05	VUB
276	395.423	410.136	400.694	3.723	4.65	4.064	VUB
277	394.067	409.906	399.722	3.695	4.705	4.05	VUB
278	394.246	410.008	399.927	3.695	4.705	4.064	VUB
279	384.139	409.88	384.037	3.259	5.1	4.527	VUB
280	358.117	409.087	362.237	2.632	6.232	4.936	VUB

APPENDIX C: Training Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
281	375.542	409.701	383.73	3.191	5.441	4.309	VUB
282	386.417	410.366	393.862	3.464	5.005	4.118	VUB
283	388.08	410.034	395.193	3.518	4.923	4.091	VUB
284	388.182	410.008	395.219	3.532	4.923	4.091	VUB
285	388.694	410.366	395.551	3.532	4.909	4.105	VUB
286	388.719	410.392	395.577	3.518	4.909	4.091	VUB
287	388.515	410.213	395.474	3.518	4.909	4.091	VUB
288	412.157	407.577	404.609	4.064	3.859	4.377	VUB
289	412.029	404.021	398.57	4.036	3.695	4.595	VUB
290	412.336	404.276	398.698	4.036	3.695	4.609	VUB
291	412.26	405.402	400.49	4.036	3.75	4.541	VUB
292	412.285	406.119	401.641	4.036	3.777	4.486	VUB
293	412.439	407.142	403.714	4.036	3.805	4.405	VUB
294	412.797	406.4	402.076	4.036	3.764	4.486	VUB
295	398.084	410.417	402.69	3.736	4.5	4.009	VUB
296	396.216	410.417	401.564	3.682	4.568	3.995	VUB
297	395.884	410.571	401.232	3.682	4.595	4.009	VUB
298	395.014	410.622	400.643	3.641	4.623	4.009	VUB
299	393.555	410.673	399.671	3.6	4.677	4.009	VUB
300	391.867	410.392	398.289	3.573	4.732	4.023	VUB
301	390.946	410.52	397.726	3.545	4.773	4.023	VUB
302	412.311	405.684	401.308	4.009	3.709	4.459	VUB
303	412.515	404.225	398.724	3.995	3.641	4.555	VUB
304	411.492	389.692	380.992	4.132	3.123	5.168	VUB
305	410.724	368.045	362.109	4.691	2.482	5.945	VUB
306	410.238	361.904	357.324	4.923	2.345	6.205	VUB
307	411.85	397.342	391.432	4.132	3.45	4.841	VUB
308	411.978	402.076	395.705	4.186	3.791	4.841	VUB
309	407.065	399.85	394.425	4.173	3.886	4.705	VUB
310	406.912	401.001	396.319	4.173	3.941	4.636	VUB
311	406.707	396.958	390.894	4.173	3.805	4.814	VUB
312	406.17	391.79	384.216	4.227	3.627	5.059	VUB
313	406.093	388.438	379.917	4.268	3.518	5.236	VUB
314	407.27	395.679	388.361	4.186	3.709	4.95	VUB
315	395.73	405.658	399.082	3.9	4.514	4.118	VUB
316	393.351	405.581	397.726	3.859	4.623	4.132	VUB
317	392.557	405.607	397.035	3.818	4.65	4.118	VUB
318	389.282	405.633	394.732	3.75	4.773	4.145	VUB
319	381.657	405.274	388.745	3.559	5.059	4.186	VUB
320	373.521	404.686	381.657	3.368	5.359	4.295	VUB
321	372.165	404.686	380.48	3.341	5.427	4.323	VUB

APPENDIX D: Testing Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
1	410.571	408.268	409.445	4.568	4.459	4.691	N
2	421.215	419.373	420.447	4.473	4.336	4.595	N
3	431.296	429.045	430.785	4.486	4.309	4.582	N
4	446.802	444.295	445.983	4.5	4.336	4.582	N
5	453.583	450.717	452.534	4.555	4.364	4.609	N
6	388.873	385.495	386.852	4.582	4.486	4.65	N
7	377.436	374.263	375.644	4.677	4.568	4.759	N
8	426.435	424.439	426.486	3.968	3.805	4.05	N
9	403.637	401.462	403.099	3.941	3.818	4.036	N
10	381.862	378.74	379.866	4.023	3.982	4.077	N
11	422.75	421.317	422.239	3.873	3.791	4.009	N
12	417.223	415.304	416.891	3.341	3.205	3.382	N
13	418.298	416.302	417.863	2.618	2.495	2.645	N
14	410.187	408.012	409.164	4.855	4.732	5.005	N
15	410.366	407.961	409.24	4.895	4.773	5.045	N
16	454.734	450.845	454.632	4.541	4.214	4.5	N
17	400.055	405.94	401.769	4.036	4.391	4.173	N
18	406.886	403.586	402.409	2.905	2.782	3.082	N
19	407.168	404.711	404.839	2.932	2.836	3	N
20	406.937	404.788	405.53	2.591	2.414	2.495	N
21	403.56	402.562	403.048	3.886	3.764	3.709	N
22	411.134	410.571	411.85	2.277	2.086	2.182	N
23	410.059	407.91	409.036	4.964	4.841	5.127	OL
24	409.394	407.398	408.319	5.291	5.168	5.482	OL
25	409.752	407.526	408.729	4.964	4.827	5.1	OL
26	409.496	407.449	408.575	4.95	4.827	5.086	OL
27	409.65	407.577	408.703	4.936	4.814	5.086	OL
28	409.599	407.398	408.549	5.291	5.168	5.468	OL
29	414.818	413.309	414.025	5.536	5.441	5.768	OL
30	414.895	413.257	414.102	5.755	5.659	5.986	OL
31	415.049	413.283	414.23	5.768	5.659	5.986	OL
32	415.1	413.283	414.46	5.741	5.632	5.945	OL
33	458.803	456.602	458.163	4.555	4.377	4.65	OV
34	464.713	461.489	464.381	4.582	4.295	4.609	OV
35	460.44	457.856	460.031	4.459	4.241	4.527	OV
36	462.103	459.365	461.412	4.473	4.255	4.527	OV
37	466.555	465.046	466.683	4.186	4.009	4.323	OV
38	466.223	464.688	465.941	4.173	4.036	4.295	OV
39	466.325	464.713	465.993	4.173	4.023	4.295	OV
40	462.769	460.952	462.462	4.132	3.968	4.241	OV

APPENDIX D: Testing Data Sets (Experiment)

Sr. No.	VRY	VYB	VBR	IR	IY	IB	Target Output
41	376.284	373.981	374.544	4.023	3.995	4.145	UV
42	373.495	371.013	372.011	4.677	4.595	4.814	UV
43	373.444	370.757	371.985	4.705	4.595	4.814	UV
44	363.618	361.597	362.825	4.786	4.664	4.95	UV
45	362.723	359.013	361.29	4.745	4.609	4.8	UV
46	360.702	357.452	359.448	4.759	4.609	4.841	UV
47	360.957	357.631	359.499	4.745	4.623	4.841	UV
48	367.61	364.616	365.486	4.105	4.064	4.186	UV
49	364.207	361.239	362.109	4.132	4.091	4.2	UV
50	373.342	370.834	372.088	4.05	3.995	4.145	UV
51	338.953	398.391	291.668	0.232	6.805	6.886	SP
52	351.106	411.287	309.22	0.014	6.436	6.491	SP
53	292.358	399.44	202.42	0.041	14.918	15.068	SP
54	292.64	340.82	400.387	6.995	0.095	6.859	SP
55	311.498	352.897	413.488	6.655	0.095	6.505	SP
56	341.102	364.105	421.829	5.318	0.068	5.182	SP
57	391.534	277.518	332.684	8.264	8.073	0.927	SP
58	414.127	313.11	353.639	6.668	6.477	0.014	SP
59	421.497	341.179	363.798	5.345	5.195	0.014	SP
60	347.371	407.654	341.639	2.1	6.818	5.605	VU B
61	396.984	410.341	401.769	3.764	4.609	4.064	VU B
62	395.73	410.264	400.976	3.736	4.664	4.064	VU B
63	388.336	410.161	395.398	3.518	4.909	4.091	VU B
64	393.632	410.366	399.287	3.668	4.718	4.077	VU B
65	412.26	404.225	398.775	4.036	3.695	4.595	VU B
66	412.746	406.375	401.974	4.036	3.764	4.486	VU B
67	390.715	410.341	397.675	3.545	4.773	4.023	VU B
68	410.034	355.405	353.256	5.127	2.155	6.382	VU B
69	412.106	402.127	395.781	4.2	3.791	4.868	VU B
70	406.784	400.592	396.191	4.173	3.927	4.623	VU B
71	407.219	395.526	388.259	4.2	3.709	4.95	VU B
72	376.975	405.018	384.651	3.45	5.25	4.255	VU B

APPENDIX E: Single Hidden Layer MLPNN Structure Used in Simulation

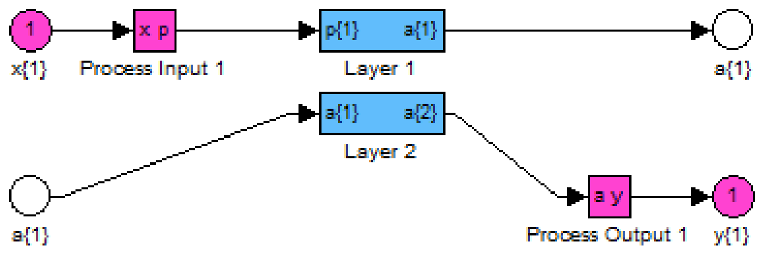


FIGURE E.1

Single Hidden Layer MLPNN Structure Used in Simulation

APPENDIX E1: MLPNN Layer-1(Hidden Layer) Diagram

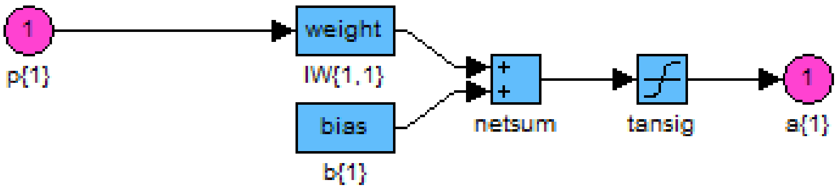


FIGURE E.2

MLPNN Layer-1 (Hidden Layer) Diagram. IW{1,1} inner Weights Subblock (weights between input and hidden layer), b{1}: bias of layer 1, P{1}: input to layer 1, a{1}: output of layer 1

APPENDIX E2: MLPNN Layer-2 (Output Layer) Diagram

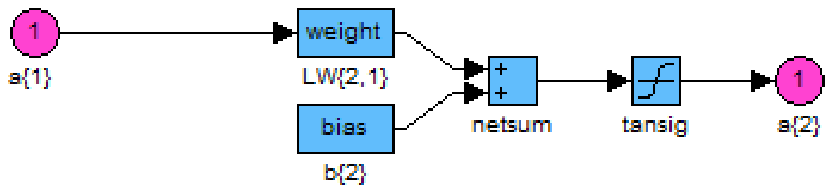


FIGURE E.3

MLPNN Layer-2 (Output Layer) Diagram. LW: Layer Weights Subblock (Weights between hidden and output layer, b {2}: bias of layer 2.a{1}: input to layer 2 , a{2}: output of layer 2

APPENDIX F: ANFIS Architecture

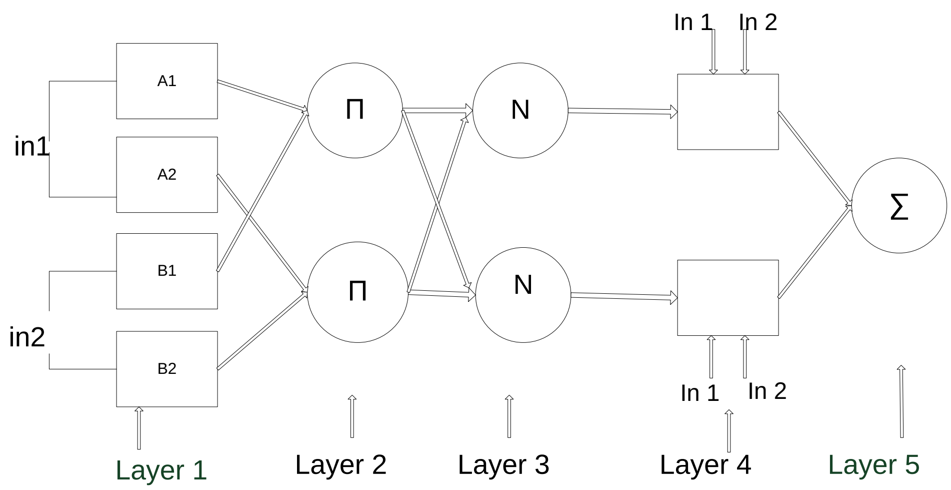


FIGURE F

ANFIS Architecture

APPENDIX G: Statistical Measurers Sample Calculations Based on SC_FIS Training Data Sets Confusion Matrix (TABLE 7. 4)

Total classification accuracy of SC_FIS (%) = total number of correct decisions / total number of decisions = 310 / 321 = 96.57%

Sensitivity, specificity, precision and F-measure for VUB condition and Normal condition is given below.

VUB Condition:

True positive (TP) for this case, refers to the VUB instances which are correctly identified by the classifier as VUB condition.

True negative (TN) for this case, refers to the other five conditions instances are correctly identified as that respective condition.

False positives (FP) for this case, refers to the any other conditions instances that are incorrectly identified as VUB

False negative (FN) for this case, refers to the VUB instances are misclassified as any other conditions.

TP = 50, TN = 260, FP = 3, FN = 0

Sensitivity (%) = TP/ (TP+FN) x 100 = (50 / (50 + 0)) x 100 = 100%

Specificity (%) = TN / (TP + FP) = (260 / (260 + 3)) x 100 = 98.8%

Precision (%) = TP / (TP +FP) = (50 / (50 + 3)) x 100 = 94.3%

F- measure = 2 x precision x sensitivity / (precision + sensitivity)

= (2 x 0.94 x 1) / (0.943 +1) = 0.97

Similarly statistical measures calculated for normal condition are also calculated.

Normal condition (N):

TP = 130, TN = 180, FP = 1, FN = 10.

Sensitivity (%) = TP/ (TP+FN) x 100 = (130/ (130+ 10)) X 100 = 92.9%

Specificity (%) = TN / (TN + FP) = (180 / (180 + 1)) x 100 = 99.4%

Precision (%) = TP / (TP +FP) = (130 / (130 + 1)) x 100 = 99.2%

F- measure = 2 x precision x sensitivity / (precision + sensitivity)

= (2 x 0.94 x 1) / (0.943 +1) = 0.97

List of Publications

1. Chudasama KJ, Shah VA (2012) 'Induction Motor Noninvasive Fault Diagnostic Techniques: A Review', International Journal of Engineering Research & Technology, 1(5), 1-7, ISSN: 2278-0181.
2. Chudasama KJ, Shah VA (2013) 'Noninvasive External Faults Detection of Induction Motor using Feedforward Neural Network', International Journal of Current Engineering and Technology, 3(2), 307-315, INPRESSCO, USA, ISSN: Electronic- 2277 – 4106, Print-2347-5161.
3. Chudasama KJ, Shah VA, Shah S (2016) 'Induction Motor Relaying Scheme for External Faults Detection and Classification using Subtractive Clustering based Sugeno Fuzzy Inference System', Electrical Power Component and Systems, 44(10), 1149-1162, Taylor and Francis, ISSN: Online-1532-5016, Print: 1532-5008.