

MACHINE LEARNING BASED INVERSION STUDIES IN EDDY CURRENT NON DESTRUCTIVE EVALUATION

By

SHUAIB AHMED S
(Enrollment No: ENGG02200904019)

INDIRA GANDHI CENTRE FOR ATOMIC RESEARCH
KALPAKKAM

*A thesis submitted to the
Board of Studies in Engineering Sciences*

*In partial fulfillment of requirements
for the Degree of*

DOCTOR OF PHILOSOPHY

of

HOMI BHABHA NATIONAL INSTITUTE



NOVEMBER, 2014

Homi Bhabha National Institute

Recommendations of the Viva Voce Board

As members of the Viva Voce Board, we certify that we have read the dissertation prepared by S. **Shuaib Ahmed** entitled “**Machine Learning Based Inversion Studies in Eddy Current Non Destructive Evaluation**” and recommend that it may be accepted as fulfilling the thesis requirement for the award of Degree of Doctor of Philosophy.

Chairman - **Dr. T. Jayakumar**

Date:

Guide / Convener - **Dr. B. Purna Chandra Rao**

Date:

Examiner

Date:

Member 1 - **Prof. Krishnan Balasubramanian**

Date:

Member 2-**Dr. K. Velusamy**

Date:

Final approval and acceptance of this thesis is contingent upon the candidate's submission of the final copies of the thesis to HBNI.

I hereby certify that I have read this thesis prepared under my direction and recommend that it may be accepted as fulfilling the thesis requirement.

Date:

Place: Kalpakkam

Guide

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at Homi Bhabha National Institute (HBNI) and is deposited in the library to be made available to borrowers under rules of the HBNI.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgement of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the Competent Authority of HBNI when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

(S. SHUAIB AHMED)

Date :

Place : Kalpakkam

DECLARATION

I, hereby declare that the investigation presented in the thesis has been carried out by me. The work is original and has not been submitted earlier as a whole or in part for a degree / diploma at this or any other Institution / University.

(S. SHUAIB AHMED)

Date :

Place : Kalpakkam

LIST OF PUBLICATIONS ARISING FROM THE THESIS

JOURNALS

1. S. Shuaib Ahmed, S. Thirunavukkarasu, B.P.C. Rao and T. Jayakumar, 'Defect Sizing Using Multidimensional Radial Basis Function'. Non-Destructive Testing and Evaluation (In review).
2. S. Shuaib Ahmed, B.P.C. Rao and T. Jayakumar, 'Greedy Breadth-First Search Based Chain Classification for Nondestructive Sizing of Subsurface Defects'. Applied Soft Computing (communicated).
3. S. Shuaib Ahmed, B.P.C. Rao and T. Jayakumar, 2013 'A Framework for Multidimensional Learning Using Multilabel Ranking'. International Journal of Advanced Intelligence Paradigms Vol: 5(4), pp 299-318.
4. S. Shuaib Ahmed, B.P.C. Rao and T. Jayakumar, 2012. 'Application of Multidimensional Chain Classifiers to Eddy Current Images for Defect Characterization'. Research Journal of Applied Sciences, Engineering and Technology, Vol:4 (24), pp 5544 5547.
5. Shuaib Ahmed, S.Thirunavukkarasu, B.P.C. Rao and T. Jayakumar, 2012, 'Competitive Learning on Cosine Similarity Method for Classification of Defects in Two-Layered Metallic Structures', Studies in Applied Electromagnetics and Mechanics Vol:36 pp 183 191.
6. S. Shuaib Ahmed, B.P.C. Rao, S. Thirunavukkarasu and T. Jayakumar, 2011. 'Indirect Methodologies for Inversion of Eddy Current NDE Data'. Journal of Non-destructive Testing & Evaluation, Vol:10 (3) pp 60-67.

CONFERENCES

1. Shuaib Ahmed S, B.P.C. Rao and T. Jayakumar, Radial Basis Function for Multidimensional Learning with an Application to Nondestructive Sizing of Defects. In IEEE-SSCI 2013, IEEE Symposium on Foundations of Computational Intelligence, Singapore, pp 38-43. 2013.

DEDICATED TO MY BELOVED PARENTS

ACKNOWLEDGEMENTS

I owe my deep sense of gratitude to my research supervisor and guide Dr. B. Purna Chandra Rao, Head, Nondestructive Evaluation Division (NDED), Indira Gandhi Centre for Atomic Research (IGCAR) for his guidance, consistent encouragement and all round support that led to the successful completion of this work. I have benefited immensely from many fruitful discussions with him and I would cherish the lessons learnt from him during my entire career, in the years to come.

I express my sincere gratitude to Dr. T. Jayakumar, Director, Metallurgy and Materials Group, IGCAR and the Chairman of my doctoral committee, for his inspiration, encouragement and most importantly providing direction during the course of thesis work. I would like to extend my heartfelt thanks to the former directors of IGCAR, Dr. Baldev Raj and Shri S. C. Chetal and the present Director, Dr. P. R. Vasudeva Rao, for permitting me to pursue research in this reputed centre. I thank eminent members of my doctoral committee Prof. Krishnan Balasubramaniam, IIT Madras and Dr. K. Velusamy, IGCAR, for their regular appraisal, kind advice and suggestions to improve the quality of this research work.

I thank IGCAR, Department of Atomic Energy, Government of India for providing me fellowship to carry out this research work in this esteemed organization.

My sincere thanks to Dr S. Thirunavukkarasu, NDED, IGCAR, for providing me technical support and guidance during the entire research work. I wish to acknowledge Smt B. Sasi, and Shri S. Mahadevan, NDED, IGCAR, for providing their valuable suggestions at various occasions. I would like to express my special thanks of gratitude to all my teachers right from early childhood education, especially Smt. Shaheen, Smt Jagadeeshwari, Prof. Sutanu Chakraborty, Shri Valsakumar, and Dr. Kanmani without their guidance, support, and hard

work this work would not have been possible. I take this opportunity to thanks Prof. Nafeesa Begum for her constant support and encouragement.

My thanks are also due to my fellow colleague Shri Anil Kumar Soni DGFS-PhD, IGCAR for his help in carrying out eddy current experiments. I would like to thank other colleagues Shri Ponseenivasan, Shri Samba Siva Rao, and Shri Arjun, and other NDED staff members for all their help, discussions, suggestions, and support.

I thank my friends Shashwat Kumar Swain, Kalyan Phani, Priyadarshini, Haripriya, Navneel Kumar, Subhra Patra, Dhanasekaran, Masilamani, Jothi kumar, Surajit Brojabasi, Ravi Kirana, Srinivasan, and others for all the fun times we had. I would like to thank my friends Mohammed Inayathullah, Kalarani, and Shitala Prasad for just being there for me.

This work would not have been possible but for the love, sacrifice, patience and support of my parents Shri Shameel and Smt Nayeem Banu. I am also very grateful to my brother, Mohammed Ovaize and my sister, Najeebuth Tarfeen for their understanding, affection, and support. I also sincerely thank my relatives and friends who were accommodating and supporting during the course of this work.

I would like to thank almighty Allah for His blessings to complete this research work.

(S. SHUAIB AHMED)

TABLE OF CONTENTS

ABSTRACT	V
LIST OF FIGURES	IX
LIST OF TABLES	XIII
LIST OF ALGORITHMS.....	XV
1 INTRODUCTION TO EDDY CURRENT INVERSION FOR DEFECT SIZING	1
1.1 NONDESTRUCTIVE EVALUATION.....	2
1.2 EDDY CURRENT TESTING	4
1.2.1 Principle of eddy current testing.....	4
1.2.2 Eddy current signal.....	7
1.2.3 Eddy current imaging	9
1.2.4 Multi-frequency eddy current testing.....	10
1.2.5 Defect sizing using eddy current testing.....	11
1.3 INVERSE PROBLEMS	12
1.3.1 Eddy current inversion.....	14
1.3.2 Direct inversion method	14
1.3.3 Empirical inversion method.....	16
1.4 MACHINE LEARNING FOR EMPIRICAL INVERSION	19
1.5 MACHINE LEARNING ALGORITHMS	21
1.5.1 Artificial neural networks.....	22
1.5.2 Support vector machine	27
1.6 SUMMARY	30
2 EMPIRICAL INVERSION	31
2.1 LITERATURE SURVEY	31
2.2 MOTIVATION.....	42
2.3 OBJECTIVE OF THE THESIS	44

3	MULTIDIMENSIONAL RADIAL BASIS FUNCTION ALGORITHM FOR SIMULTANEOUS SIZING OF DEFECT CHARACTERISTICS	47
3.1	MULTIDIMENSIONAL LEARNING	47
3.1.1	Multidimensional classification	48
3.2	MULTIDIMENSIONAL RADIAL BASIS FUNCTION NEURAL NETWORK	49
3.2.1	Representation of the prototype vectors.....	51
3.2.2	Weight optimization.....	52
3.2.3	Output.....	53
3.3	TRAINING AND EVALUATION OF MD-RBF NEURAL NETWORK ...	54
3.3.1	Dataset generation.....	55
3.3.2	Feature extraction.....	62
3.3.3	Performance evaluation metrics.....	71
3.3.4	Cross validation.....	74
3.3.5	Parameter optimization	75
3.4	PERFORMANCE EVALUATION OF MD-RBF NEURAL NETWORK ...	76
3.4.1	Performance evaluation of MD-RBF neural network on modeled images	76
3.4.2	Performance evaluation of the MD-RBF neural network on experimental images of rectangular defects.....	79
3.4.3	Performance evaluation of MD-RBF neural network on modeled images of other defects.....	81
3.5	SUMMARY	84
4	INFLUENCE OF SCAN PITCH ON DEFECT SIZING	87
4.1	SCAN PITCH	87
4.2	INFLUENCE OF SCAN PITCH ON SIZING LENGTH OF DEFECTS.....	89
4.3	INFLUENCE OF SCAN PITCH ON SIZING WIDTH OF DEFECTS	91
4.4	INFLUENCE OF SCAN PITCH ON SIZING DEPTH OF DEFECTS	92
4.5	SUMMARY	93
5	CHAIN CLASSIFICATION BASED ON GREEDY BREADTH-FIRST-SEARCH FOR DEFECT SIZING	95
5.1	CHAIN CLASSIFICATION	95

5.2	GREEDY BREADTH-FIRST-SEARCH (GBFS) ALGORITHM FOR DETERMINATION OF AN OPTIMAL SEQUENCE	97
5.3	RESULTS AND DISCUSSION	102
5.3.1	Independent classification	102
5.3.2	GBFS algorithm for an optimal sequence identification	105
5.3.3	Chain classification on test data	113
5.3.4	Testing the robustness of the chain classification	115
5.4	SUMMARY	116
6	GENERALIZED WRAPPER FRAMEWORK TO INCORPORATE MULTILABEL ALGORITHMS FOR DEFECT SIZING	119
6.1	MULTILABEL LEARNING	119
6.1.1	Multilabel classification	119
6.1.2	Multilabel ranking	120
6.2	WRAPPER FRAMEWORK FOR MULTIDIMENSIONAL SUPERVISED LEARNING	121
6.3	DATASET DESCRIPTION	124
6.4	RESULTS AND DISCUSSION	126
6.4.1	Real world benchmark multilabel data	126
6.4.2	Eddy current testing data	128
6.5	SUMMARY	129
7	CONCLUSION	131
7.1	TECHNICAL AND SCIENTIFIC CONTRIBUTIONS	134
8	FUTURE WORKS	135
	REFERENCES	139
	APPENDIX	145

ABSTRACT

The inverse problem of eddy current (EC) non-destructive evaluation (NDE) is described as the task of sizing defects from EC signals and images. Sizing of defects with respect to determination of length, width, depth, and height is essential for fracture mechanics based studies and for structural integrity assessment of various engineering components. The inverse problem of EC NDE is ill-posed, essentially, due to the diffusive and divergent propagation of electromagnetic fields in the test object. This demands the use of machine learning and artificial intelligence techniques for automated and operator independent sizing defects. Machine learning algorithms such as artificial neural networks, radial basis function neural networks, and support vector machines are used for sizing surface breaking defects. However, EC inversion of subsurface defects using machine learning is challenging, as it involves sizing depth locations below surface additionally. Studies to develop machine learning algorithms for simultaneous determination of all the four important defect characteristics, viz. length, width, depth, and height are very limited in literature. The limitation with the conventional machine learning algorithms is their inability to incorporate dependency among various defect characteristics and they demand the use of several learning algorithms for sizing different defect characteristics.

This thesis focuses development of novel machine learning algorithms for automated sizing of subsurface defects in metallic materials using EC images. It follows a new concept in machine learning, called multidimensional learning, involving prediction of multiple class variables. This thesis proposes a novel multidimensional radial basis function (MD-RBF) neural network to address dependency among the four defect characteristics and to simultaneously size all four defect characteristics in a single learning algorithm fed with

input features obtained from EC images. An extensive study has carried out relating 17 extracted features from EC images and 4 defect characteristics. It is also established that the extracted features are noise tolerant and rotation invariant and they form very good input features for the proposed MD-RBF neural network. Using these extracted features, it has been established from the experiments that the dependency can be effectively incorporated by the MD-RBF neural network. Training of the MD-RBF neural network has been carried out using numerically modeled EC images, while experimentally obtained EC images have been evaluated. The MD-RBF neural network has achieved a global accuracy of 83% with a relatively less number of computations as compared to the global accuracy of 76% produced by the conventional RBF neural networks.

Information content in EC images depends on the scan pitch i.e. the distance between two consecutive line scans during raster scan imaging. Influence of scan pitch on defect sizing performance by the proposed MD-RBF neural network has been systematically studied. From this study, it has been identified that the MD-RBF neural network is robust in defect sizing from EC images obtained even at higher scan pitches (lower resolutions) than that used for training. The most and least influencing defect characteristics have been identified, for the first time. This study has revealed a limit scan pitch for each defect characteristic below which defect sizing is accurate, however, at the least possible time than that of training.

This thesis proposes a novel chain classification coupled with greedy breadth-first-search (GBFS) algorithm to study the influence of dependency among the defect characteristics. The optimal sequence that strongly indicates the dependency structure during sizing has been identified, for the first time, using the GBFS algorithm. The efficacy of the chain classification has been demonstrated using numerically modeled as well as experimentally obtained EC images of defects.

In order to expand the scope of empirical inversion by incorporating dependency for other robust learning algorithms, a novel generalized MDLearn wrapper framework has been proposed. From the studies using benchmark real world dataset and EC images of defects, the framework has been validated. Studies clearly established that the proposed MDLearn framework can be effectively used for inversion.

The work presented in this thesis has significantly improved the capabilities of inversion for accurate and automated sizing of defects from features extracted from EC images. It has also provided better insights into the existence of dependencies among the defect characteristics and utilization of dependency enhanced sizing accuracy of inversion. Besides EC inversion, the multidimensional learning algorithms proposed in this thesis can be applied to any machine learning applications with multiple class variables.

LIST OF FIGURES

Figure 1.1 A generic nondestructive evaluation system.....	3
Figure 1.2 EC testing of an electrically conducting plate for detection of surface and subsurface defects.....	5
Figure 1.3 Typical EC testing (line scan) on a plate having three surface defects and a subsurface defect.....	7
Figure 1.4 Typical EC signals for four different defects (width: 1.0 mm and height: 1.5 mm) shown in Figure 1.3.....	8
Figure 1.5 Typical raster scanning of an EC probe and different EC images obtained for a defect.....	9
Figure 1.6 Typical multi-frequency EC response of a defect at four different excitation frequencies.....	11
Figure 1.7 Flowchart of a direct inversion method for solving inverse problems.....	15
Figure 1.8 Steps followed in a learning system.....	20
Figure 1.9 Architecture of a typical artificial neural network.....	23
Figure 1.10 Architecture of a probabilistic neural network.....	24
Figure 1.11 Architecture of a RBF neural network.....	25
Figure 1.12 Various RBF activation functions.....	26
Figure 1.13 Typical decision boundary learned by support vector machine.....	28
Figure 2.1 Crack depth determined by swept-frequency eddy current NDE for a series of rectangular and semi-elliptical EDM slots in Al alloy plates [55].....	33
Figure 2.2 The restored image of a notch (length 4.0 mm, width 0.3 mm, depth 0.4 mm) superimposed (after assigning a gray-level value of 0 for the sake of visual convenience) a) on the height-profile image and b) on the magnitude image [59].....	36
Figure 2.3 3-D image of a notch (length 6.0 mm, width 0.3 mm, depth 0.4 mm) in a stainless steel plate after fusion of depth-profile image and restored image [59].....	36
Figure 2.4 True and reconstructed profiles of natural cracks [62].....	38

Figure 2.5 Scheme of inverse dynamic ANN models with a moving window for flaw profile estimation [63].....	38
Figure 2.6 Estimated profiles of the defects: LPF is the identification using the filtered signals, X-APP identification using the approximated signals [63].	39
Figure 3.1 The architecture of the conventional RBF neural networks and the proposed MD-RBF neural network.....	50
Figure 3.2 Schematic of a subsurface defect with four defect characteristics.	51
Figure 3.3 Proposed approach for sizing of defects.....	54
Figure 3.4 Graphical user interface of CIVVA software.....	56
Figure 3.5 Cross sectional view of the EC probe used for modeling.....	57
Figure 3.6 Block diagram of the EC measurement system used for generation of EC images.	59
Figure 3.7 Photograph of the a) EC instrument and b) probe used for test data generation.....	59
Figure 3.8 CIVVA modeled (a and c) and experimental (b and d) EC images of a notch (depth: 2.0 mm, height: 3.0 mm, length: 25.0 mm, width: 2.0 mm) at 5 kHz and 10 kHz.....	60
Figure 3.9 CIVVA modeled (a and c) and experimental (b and d) EC images of a notch (depth: 3.0 mm, height: 2.0 mm, length: 25.0 mm, width: 2.0 mm) at 5 kHz and 10 kHz.....	61
Figure 3.10 Description of the features extracted from an EC image at one excitation frequency.....	64
Figure 3.11 Comparison of length of defect versus features a) F1 and b) F2. (○: width = 1.0 mm, depth = 0.0 mm, height = 1.0 mm; □: width = 2.0 mm, depth = 1.0 mm, height = 1.0 mm; Δ: width = 1.0 mm, depth = 2.0 mm, height = 1.0 mm).....	65
Figure 3.12 Comparison of depth of defect versus features a) F7 and b) F8. (○: length = 20.0 mm, width = 1.0 mm, height = 3.0 mm; □: length = 25.0 mm, width = 2.0 mm, height = 3.0 mm; Δ: length = 30.0, width = 2.0 mm, height = 3.0 mm).....	66

Figure 3.13 Comparison of height of defect versus features a) F7 and b) F16. (○:length = 25.0 mm, width = 2.0 mm, depth = 0.0 mm; □:length = 30.0 mm, width = 2.0 mm, depth = 0.0 mm; Δ: length =35.0, width = 3.0 mm, depth = 0.0 mm).	66
Figure 3.14 Comparison of height of defect versus feature F9. (○: length = 25.0 mm, width = 2.0 mm, depth = 0.0 mm; □: length = 30.0 mm, width = 2.0 mm, depth = 0.0 mm; Δ: length = 35.0 mm, width = 3.0 mm, depth = 0.0 mm).	67
Figure 3.15 EC images with different noise for a Defect-T6.	69
Figure 3.16 EC images of a defect (length: 20.0 mm, width: 1.0 mm, depth: 3.0 mm, and height: 2.0 mm) oriented with respect to surface of the testing specimen.	70
Figure 3.17 An illustration to compare the three metrics for performance evaluation of the proposed MD-RBF neural network.	74
Figure 3.18 10-fold cross validation performance for classification of a) RBF neural networks and b) MD-RBF neural network.	76
Figure 3.19 Schematic of the ellipsoidal defect modeled in CIVA.	81
Figure 3.20 Dimensions of a) Defect-E1, b) Defect-E2, and c) rectangular defect modeled by CIVA.	82
Figure 3.21 EC images of a) Defect-E1, b) Defect-E2, and c) rectangular defect at 5 kHz.	83
Figure 4.1 Comparison of two different scan pitches.	88
Figure 4.2 Comparison of CIVA modeled images obtained at two different scan pitches.	88
Figure 4.3 Time taken for EC imaging of a defect at different scan pitches for an area of 70 mm × 60 mm.	89
Figure 4.4 Influence of scan pitch on sizing length.	90
Figure 4.5 Influence of scan pitch on length related features.	91
Figure 4.6 Influence of scan pitch on sizing.	92
Figure 5.1 Typical chain classification scheme for subsurface defect sizing.	97
Figure 5.2 A random directed acyclic graph with 5 vertices.	97
Figure 5.3 Classifier chain tree and sequence of expansion of nodes using BFS and GBFS.	100

Figure 5.4 10-fold cross validation performance of RBF neural networks for classification of a) length and b) width for different α and μ	103
Figure 5.5 10-fold cross validation performance of RBF neural networks for classification of a) depth and b) height for different α and μ	103
Figure 5.6 10-fold cross validation performance of SVM for classification of all defect characteristics for different polynomial kernel parameter, d.	104
Figure 5.7 GBFS outcome using RBF neural network as the classification algorithm with length as the root node of the search tree.....	105
Figure 5.8 GBFS outcome using RBF neural network as the classification algorithm with width as the root node of the search tree.....	106
Figure 5.9 GBFS outcome using RBF neural network as the classification algorithm with depth as the root node of the search tree.....	107
Figure 5.10 GBFS outcome using RBF neural network as the classification algorithm with height as the root node of the search tree.	107
Figure 5.11 GBFS outcome using SVM as the classification algorithm with length as the root node of the search tree.	108
Figure 5.12 GBFS outcome using SVM as the classification algorithm with width as the root node of the search tree.	109
Figure 5.13 GBFS outcome using SVM as the classification algorithm with depth as the root node of the search tree.	109
Figure 5.14 GBFS outcome using SVM as the classification algorithm with height as the root node of the search tree.	110
Figure 5.15 EC images of a) surface and b) near-surface defect (length: 30.0 mm, width: 2.0 mm and height: 1.0 mm) at 5 kHz.	115
Figure 6.1 A wrapper framework for multidimensional supervised learning.....	122
Figure 6.2 Examples of multilabel images.....	126

LIST OF TABLES

Table 1.1 Comparison of direct inversion and empirical inversion methods.....	18
Table 1.2 Types of function and their representations.....	21
Table 1.3 Examples of functions and learning algorithms.....	22
Table 3.1 Training dataset for conventional machine learning and multidimensional learning.....	48
Table 3.2 Dimensions of EDM notches specified for fabrications.....	62
Table 3.3 Description of classes for defect sizing.....	62
Table 3.4 The features extracted from an EC image for a defect.....	63
Table 3.5 A set of 10 typical training defects.....	68
Table 3.6 Typical features of the defects in Table 3.5.....	68
Table 3.7 Effect of noise on the input features.....	69
Table 3.8 Effect of orientation of the extracted features.....	71
Table 3.9 Comparison of performance of the conventional RBF neural networks and MD-RBF neural network for 10-fold cross validation.....	77
Table 3.10 Comparison of performance of the RBF neural networks and MD-RBF neural network by 10-fold cross validation on training data using evaluation metrics.....	77
Table 3.11 Comparison of performance of the MD-RBF neural network following the dependency sequence.....	78
Table 3.12 Performance of the MD-RBF neural network on a set of defects from training data.....	79
Table 3.13 Performance of MD-RBF neural networks on experimental images of EDM notches.....	79
Table 3.14 Performance of the individual RBF neural networks on a set of defects from training data.....	80
Table 3.15 Performance of individual RBF neural networks on experimental images of EDM defects.....	80
Table 3.16 Performance evaluation of MD-RBF neural network for the ellipsoidal defects.....	83

Table 3.17 Extracted EC image features relevant to length for defects smaller than the probe diameter.....	84
Table 4.1 Predictions of MD-RBF neural network for different scan pitches.	92
Table 4.2 Feature value at different scan pitches for a defect (length: 25.0 mm, width: 1.0 mm, depth: 1.4 mm, height: 1.0 mm).	93
Table 5.1 Comparison of performance of independent RBF neural networks and SVM for defect sizing on training data.	105
Table 5.2 Performance of RBF neural network in chain sequence with different defect characteristics as root node of the GBFS.	108
Table 5.3 Performance of SVM in chain sequence with different defect characteristics as root node of the GBFS.	110
Table 5.4 Comparison of number of node expansion required and number of nodes expanded by GBFS.	111
Table 5.5 Comparison of performance of RBF neural networks and SVM in chain sequence with different defect characteristics as root node of the GBFS.	111
Table 5.6 Comparison of performance of independent as well as chain classification.	112
Table 5.7 Performance of the RBF neural network in chain classification.	113
Table 5.8 Performance of the SVM in chain classification.	114
Table 5.9 Performance of RBF neural network using independent classification.	114
Table 5.10 Performance of SVM using independent classification.	115
Table 5.11 Extracted features for a surface defect (depth: 0.0 mm) and near surface defect (depth: 0.2 mm).	116
Table 6.1 performance evaluation (mean \pm std. deviation) for ten-fold cross validation on benchmark multilabel dataset.	127
Table 6.2 Overall performance of all evaluation metrics based on two-tailed paired t -test at 5% significant level ($MDLearn_{ML-RBF} \rightarrow A$, $ML-RBF \rightarrow B$, $MDLearn_{ML-KNN} \rightarrow C$, $ML-KNN \rightarrow D$)	127
Table 6.3 Comparison of results for hamming loss reported in [67,74,75] and proposed framework.	127
Table 6.4 Accuracy of MDLearn and conventional uni-dimensional algorithm for individual dimension of EC data.	128
Table 6.5 Performance of MDLearn and conventional uni-dimensional algorithm for EC data.	128

LIST OF ALGORITHMS

Algorithm 3.1 Multidimensional radial basis function neural network.....	53
Algorithm 5.1 Greedy breadth-first-search	101
Algorithm 6.1 MDLearn wrapper - class transformation.....	122
Algorithm 6.2 MDLearn wrapper - class prediction.	123

NOMENCLATURE

LIST OF SYMBOLS

dB	decibel
$f(\cdot)$	Function representation
J	Current density
kHz	kilo Hertz
m/s	metre per second
mHz	milli Hertz
mm	millimeter
mV	milli volts
V	Electric scalar potential or volts
W	(Bold capital letters) Matrices
x	(Bold small letters) Vectors
<i>y</i>	(Italic small letters) Scalars
Z	Impedance
δ	Standard depth of penetration
λ	Limit scan pitch
μ	Magnetic permeability
μ_r	Relative magnetic permeability
σ	Electrical conductivity
ϕ	Phase angle
Ω	Ohm
\mathcal{D}	Training data

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
AISI	American Iron and Steel Institute
ANN	Artificial Neural Network
AWGN	Additive White Gaussian Noise
BFS	Breadth First Search
DAG	Directed Acyclic Graph
DAQ	Data Acquisition
EC	Eddy Current
ECT	Eddy Current Testing
EDM	Electric Discharge Machining
ENDE	Electromagnetic Non-Destructive Evaluation
FEM	Finite Element Method
GBFS	Greedy Breadth-First-Search
IACS	International Annealed Copper Standard
ID	Inner Diameter
IR	Infra Red
KNN	K-Nearest Neighbor
LMSE	Least Mean Square Error
LPF	Low Pass Filter
MAE	Mean Absolute Error
MDLearn	Multidimensional Learning
MD-RBF	Multidimensional Radial Basis Function
ML-KNN	Multilabel K-Nearest Neighbor
MLP	Multilayer Perceptron
ML-RBF	Multilabel Radial Basis Function
NDE	Non-Destructive Evaluation
OD	Outer Diameter
ODE	Ordinary Differential Equation
PC	Personal Computer
PD	Potential Drop

PDE	Partial Differential Equation
PEC	Pulsed Eddy Current
PNN	Probabilistic Neural Network
PSF	Point Spread Function
RBF	Radial Basis Function
ROI	Region of Interest
SNR	Signal-to-Noise Ratio
SS	Stainless Steel
SVD	Singular Value Decomposition
SVM	Support Vector Machine
UAF	Universal Approximation Function

1

INTRODUCTION TO EDDY CURRENT INVERSION FOR DEFECT SIZING

The scope of this chapter is to introduce the context of inverse problems in eddy current non-destructive evaluation (NDE). The chapter starts with a brief introduction to NDE. The principle of eddy current NDE along with the characteristics of eddy current signals and images are highlighted. The inverse problems and their importance are discussed in detail with an emphasis on eddy current inversion. The basics of machine learning employed for solving inverse problems are also discussed.

Rapid advances in computing science and robotics have enabled automation of various industrial processes. With automation increased throughput, consistency of output, robustness of the processes are possible and this will improve customer satisfaction, in turn, profitability for the industry. Automation completely relies on development of expert systems that are capable of taking decisions on their own. Artificial intelligence (AI) plays a key role in the expert systems, which replace human role for decision making. It exploits the computing power that is exponentially increasing in the past few decades without involving additional costs. Developing AI based expert systems often involves automated planning, automated reasoning, image analysis, and solving mathematical inverse problems. Example tasks that deal with automation using expert systems are automated manufacturing [1], automated diagnosis [2], and various other engineering and manufacturing areas [3] including structural health assessment of engineering components and structures [4].

1.1 NONDESTRUCTIVE EVALUATION

Nondestructive evaluation is an interdisciplinary science that deals with assessment of soundness and integrity of components or structures through detection and quantification of discontinuities viz. defects, microstructure variations and stresses without causing any damage to the components or structures [5,6]. NDE is an essential ingredient of modern engineering practice contributing significantly to quality, safety, and reliability of critical components in nuclear power plants, transportation, aerospace, petrochemical, and other industries, essentially because failures can affect the plant availability, productivity, and profitability. Any discontinuity that creates a substantial chance of failure of a component or structure in service is commonly called a *defect*. Common types of defects are cracks, delaminations, cavities, material loss, porosity, pitting, etc. Defects can form in a material or component during manufacturing process (process induced defects) such as casting, welding, rolling, forging, and machining; and defects can also form during service life (service induced defects) like creep cavities, fatigue cracks, hydrogen or helium embrittlement, and stress corrosion crack.

Stainless steels are one of the major structural materials used in various industries, due to their good corrosion resistance and better mechanical properties (yield, ductility, and toughness). NDE of stainless steels for detection of defects is important. In these steels, defects form in components due to exposure to high temperatures, pressures, irradiation, and hostile corrosive media. Defect detection and sizing are essential to ensure safety and reliability of components made of stainless steels.

A few widely used NDE techniques for detection of discontinuities include ultrasonic testing, radiography, eddy current testing, liquid penetrant testing, magnetic flux leakage, infrared thermography, and other optical techniques [7,8]. The schematic representation of a generic NDE system is shown in Figure 1.1. It consists of a source input e.g. electromagnetic energy in eddy current testing, to introduce energy into test object using an exciter. The transmitted energy

in the test material is modified depending on the discontinuity present in the test material. The modified reflected or transmitted energy is picked up by a receiver or sensor. This response is processed and displayed in the form of a signal or an image to extract information about the defect or discontinuity.

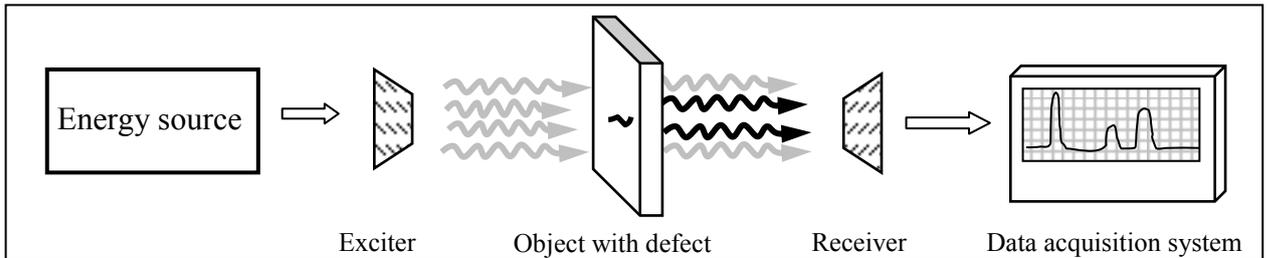


Figure 1.1 A generic nondestructive evaluation system.

Among the host of NDE techniques, electromagnetic NDE (ENDE) techniques are popular for testing thin (thickness ≤ 5 mm) electrically conducting materials such as stainless steel, aluminum, brass, titanium, and Inconel. Electromagnetic NDE techniques include a number of techniques that are based on electromagnetic field interaction with electrically conducting materials. These techniques include eddy current testing, magnetic flux leakage, magnetic Barkhausen emission and potential drop (PD) testing.

The ENDE techniques use static (steady state) and low frequency (diffusive) electromagnetic fields to interrogate the material. They use coils as well as solid state sensors to detect the manifestations. The ENDE techniques are influenced by several physical variables such as electrical conductivity, magnetic permeability, dielectric permittivity, excitation frequency, etc. Deterministic modeling of the ENDE techniques is possible, which means the result of performing an ENDE technique can be predicted by analytical and numerical models by solving the governing partial differential equations (PDE). These techniques are mostly non-contact. These techniques are sensitive to changes in electrical conductivity and magnetic permeability of the material being tested and are also sensitive to microstructural changes that effect

conductivity and permeability [9]. ENDE techniques are useful for measurement of coating thickness and conductivity measurements as well as detection and sizing of defects and characterization of microstructures [10,11].

1.2 EDDY CURRENT TESTING

Eddy current (EC) testing is an important ENDE technique widely used in aerospace, nuclear, petrochemical, and other industries. Almost all heat exchangers (boilers, steam generators, condensers) and aircrafts are inspected using this technique. The main reasons behind this widespread use are excellent sensitivity to surface as well as subsurface defects in metallic materials. Besides higher sensitivity, other attractive features of this technique include ease of operation, versatility, extremely high testing speeds (up to 10 m/s), data storage possibility and repeatability. This technique can detect wall thinning, cracks, pitting, stress corrosion cracking, hydrogen embrittlement, denting and deposits. The most popular applications of this technique include detection of defects in plates, tubes, rods, bars, multi-layer structures, discs, welds, blades, and other regular as well as irregular geometries; material sorting; heat treatment adequacy assessment; proximity sensing; and coating thickness measurements [12].

1.2.1 *Principle of eddy current testing*

Eddy current testing works on the principle of electromagnetic induction. In this technique, a coil (also called probe or sensor) placed over an electrically conducting material, e.g. stainless steel, aluminum, etc., as shown in Figure 1.2 is excited with a sinusoidal alternating current. Following the Ampere's law, this current generates a primary magnetic field in the vicinity of the coil. When an electrically conducting material is brought close to this coil, eddy currents are induced in the material according to the Faraday's law. These eddy currents also generate a secondary magnetic field, but in the opposite direction to the primary magnetic field following the Lenz's law and this field in turn, change the coil impedance, \mathbf{Z} , which is a complex quantity with

resistance (real component) and inductive reactance (imaginary component). Defects such as cracks, voids, inclusions, corrosion, wall loss etc. cause distortion of the eddy currents and they also change the coil impedance. The impedance change for defect-free and defective regions are different and this enables one to detect the presence of a defect [13].

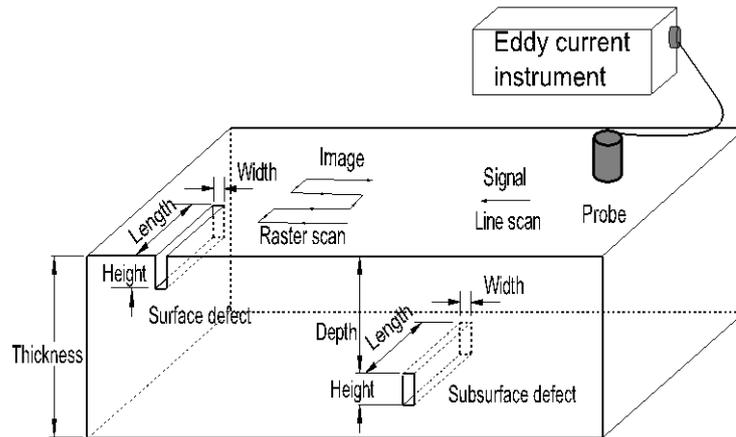


Figure 1.2 EC testing of an electrically conducting plate for detection of surface and subsurface defects.

The governing differential equation for coil excited with current density \mathbf{J} and producing eddy currents in a homogeneous isotropic electrically conducting material is derived from the Maxwell's curl equations [14] as.

$$\nabla \times \mathbf{E} = -\frac{d\mathbf{B}}{dt} \quad (1.1)$$

$$\nabla \times \mathbf{B} = \mu\mathbf{J} \quad (1.2)$$

where \mathbf{E} is the electric field, \mathbf{B} is the magnetic field vector, \mathbf{J} is the current density and μ is the magnetic permeability of the material. The governing PDE of eddy current testing for the current density in a homogenous medium is given as

$$\nabla^2 \mathbf{J} = j\omega\mu\sigma\mathbf{J} \quad (1.3)$$

Solving Equation (1.3) provides the distribution of the eddy current density in the lateral and depth directions of the excitation coil [15]:

$$\mathbf{J}_z = \mathbf{J}_0 e^{-\{z\beta\}} \sin(\omega t - \beta) \quad (1.4)$$

where \mathbf{J}_z is the current density anywhere along the thickness of the material (z-axis) and \mathbf{J}_0 is the current density at the surface of the material and β is defined as follows:

$$\beta = \sqrt{\pi f \mu \sigma} \quad (1.5)$$

The solution obtained in Equation (1.4) contains the magnitude (exponential) and phase terms, depicting the fact that flow of eddy currents in material is not uniform in the depth and lateral directions. The magnitude term describes the exponential attenuation of the eddy currents along the thickness of the material. The phase term β implies the linear variation of the phase of the eddy current sinusoids with thickness (depth). The depths at which the current density falls to $1/e$ times the surface current density \mathbf{J}_0 in a material is called standard depth of penetration denoted as δ [14]:

$$\delta = \frac{1}{\sqrt{\pi f \mu \sigma}} \quad (1.6)$$

where f is the excitation frequency, μ is the magnetic permeability and σ is the electrical conductivity. As can be seen, the standard depth of penetration of eddy currents in the test object decreases with increase in σ , μ and f . Thus, the eddy currents are quite dense at the surface as compared to that at deep inside and this phenomenon is generally known as *skin effect*.

Selection of frequency for excitation of an EC probe is very important in EC testing. A simpler way to determine the excitation frequency range involves, assuming values 1 and 2 for δ in equation (1.6) and calculating the extreme frequencies upon substituting σ and μ values of the test material.

The electrical conductivity is usually expressed as percentage IACS (International Annealed Copper Standard) in which the electrical conductivity of pure copper at 25°C is taken as 5.8×10^7 Siemens/meter. The IACS% value of austenitic stainless steel (SS) type 304 is 2.5 with an absolute electrical conductivity of 1.45×10^6 .

The impedance change response from an EC probe during a line scan on material surface is usually represented in form of EC signals. Visualization of defects in form of EC images obtained from several line scan EC signals has also attracted considerable interest for better representation and interpretation [15].

1.2.2 Eddy current signal

The locus of changes in impedance of an EC probe, during the movement of the probe over a test material as illustrated in Figure 1.3 is called an EC signal. The signal consists of two components viz. resistance and inductive reactance as a function of distance or time, as shown in Figure 1.4a. The EC signal is also visualized as a Lissajous figure on complex (impedance) plane as shown in Figure 1.4b with resistance, X , as abscissa and inductive reactance, Y , as ordinate. Figure 1.4 typically shows time domain as well as impedance plane EC signals for four defects; D1 with 20.0 mm length, D2 with 25.0 mm length, and D3 and D4 with 30.0 mm length. Defect D3 is a surface defect, while D4 is a subsurface defect located at a depth of 3.0 mm from surface.

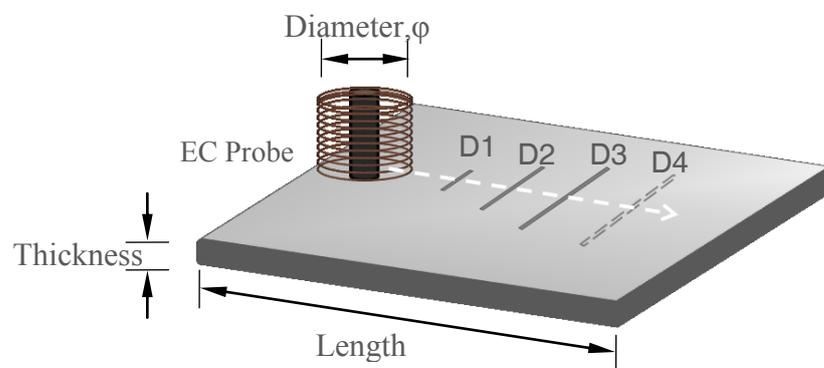
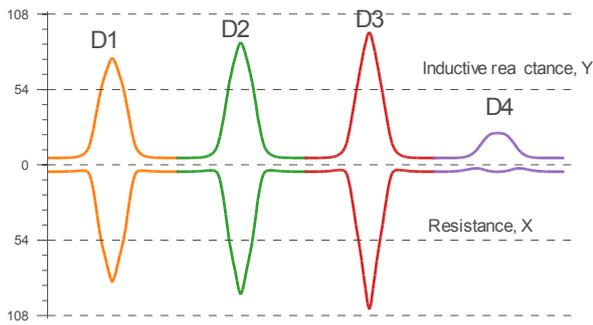
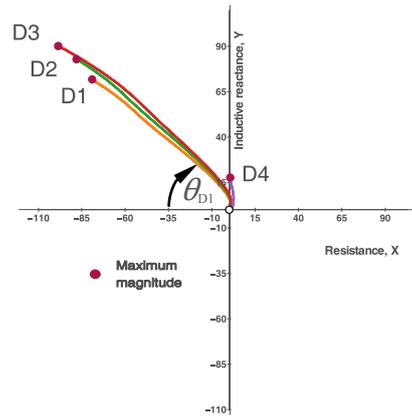


Figure 1.3 Typical EC testing (line scan) on a plate having three surface defects and a subsurface defect.



a) Time-domain signals



b) Impedance plane signals

Figure 1.4 Typical EC signals for four different defects (width: 1.0 mm and height: 1.5 mm) shown in Figure 1.3.

The maximum magnitude of EC signal corresponds to the maximum interaction region of the probe with a defect and this provides information about the defect severity. On the other hand, the signal phase angle provides information about the defect height, as can be seen from Figure 1.4b. In the impedance plane, magnitude and phase can be seen, however, the signal extent or defect length cannot be visualized. As can be observed from Figure 1.4b, the defect D3 and D4 has different maximum magnitude, however, at different phase angles. On the contrary, in the time domain signals with time or distance as abscissa, phase angle information that is essential for sizing is absent.

In practical inspections, EC signals are usually seen with noise which arises from disturbing variables such as surface roughness, probe tilt, and variation in *lift-off* between the probe and test material. In order to suppress the noise and to enhance information from defects, various signal processing methods are usually employed. These include the use of filters, Fourier analysis [16], and wavelet analysis [17].

1.2.3 Eddy current imaging

Using EC signals, one can detect defects and can get information about height of defects. However, defects which are described as three-dimensional functions of the space co-ordinates, cannot be completely reconstructed by merely scanning an EC probe over a defect in one direction, even using sophisticated signal processing methods. A definite benefit exists if a series of parallel line scans (raster scan) are made and the resulting eddy current data viz. resistance, inductive reactance, magnitude, and phase are presented in the form of a pseudo color or a gray level image [18,19,20]. This often referred as EC imaging, automates the measurement process. Typical resistance image, inductive reactance image, magnitude image, and phase image obtained for a defect (length: 20.0 mm, width: 2.0 mm, and height: 1.5 mm) in an SS plate with 5.0 mm thickness at 5 kHz are shown in Figure 1.5.

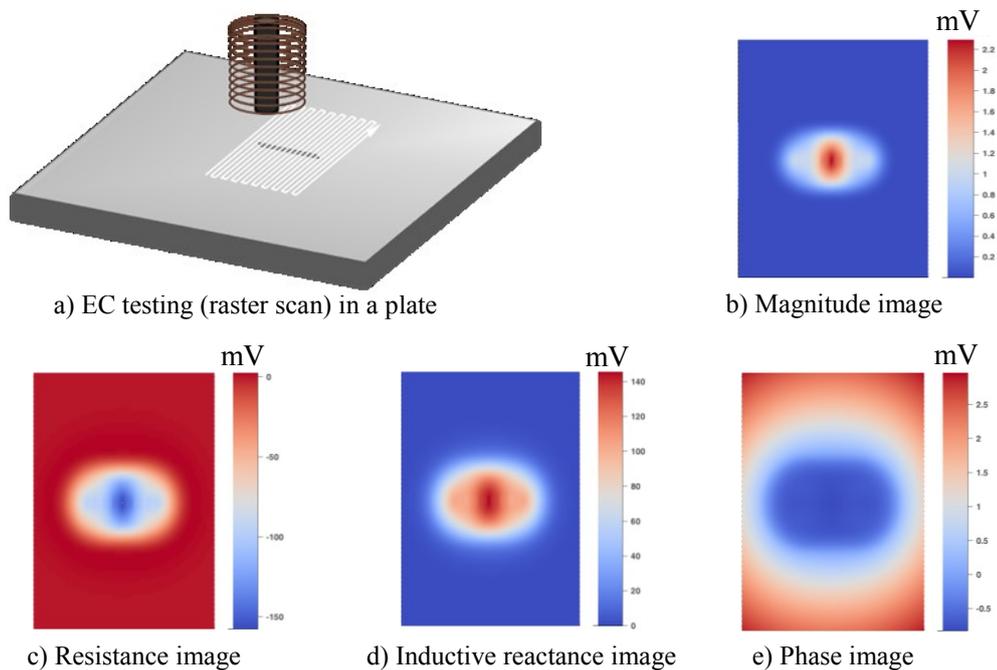


Figure 1.5 Typical raster scanning of an EC probe and different EC images obtained for a defect.

As can be seen from Figure 1.5, EC images are more convenient to interpret and are useful to obtain the spatial extent such as length and width of a defect. In addition to detection of defect, it

is also possible to quantify the defect height e.g. from the phase angle image. However, major problems that arise by EC imaging are:

- 1) EC images are blurred due to convolution of probe *footprint* point spread function (PSF) with defect.
- 2) The defect profile is a mere mapping of conductivity profile in the test material and cannot always be directly correlated with defect height or location from the surface.
- 3) EC images have the influence of disturbing variables, similar to signals.
- 4) EC imaging is a time intensive process.

1.2.4 Multi-frequency eddy current testing

Interestingly, it is also possible to simultaneously excite an EC probe coil with several frequencies by suitable instrumentation. This multi-frequency EC testing is very useful and is applied to cancel out unwanted noise to improve the signal-to noise ratio (SNR) for better detection of defects. Multi-frequency EC testing is also very important for sizing of defects, especially the subsurface defects, as one can obtain several responses of a defect from different interrogating depths (excitation frequencies). Multi-frequency EC response of a defect with length 20.0 mm, width 2.0 mm, depth 2.0 mm, and height 1.0 mm in a 5.0 mm thick SS plate at 1, 5, 10 and 25 kHz are shown in Figure 1.6.

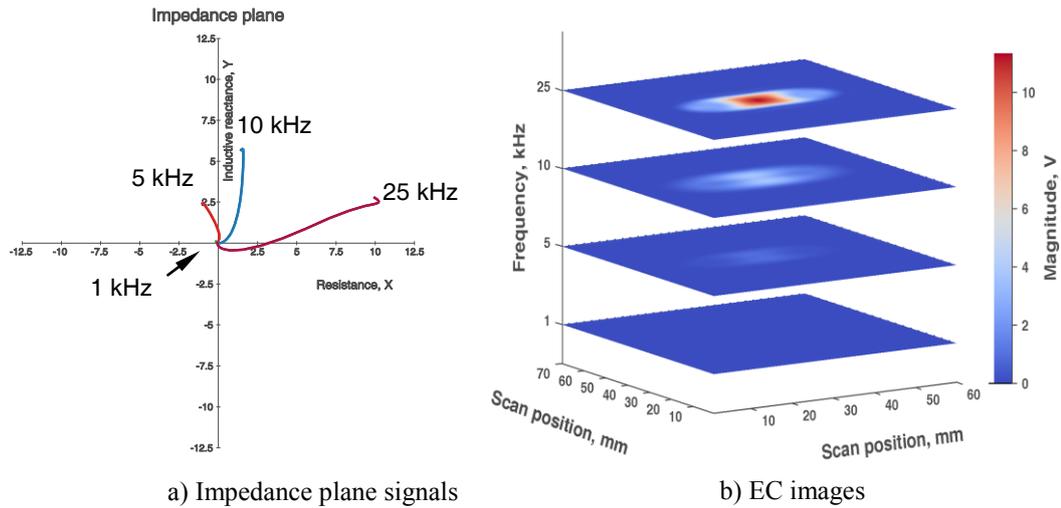


Figure 1.6 Typical multi-frequency EC response of a defect at four different excitation frequencies.

Instead of continuous alternating current, if the source coil is excited with a repetitive broadband pulse, such as a square wave, conventional EC technique becomes pulsed eddy current (PEC) technique. PEC induces transient eddy currents associated with highly attenuated magnetic pulses, to propagate through the test material at a large range of excitation frequencies. The response from defects in the test material is picked up by a probe. At each probe location, a series of voltage-time data pairs are produced as the induced fields decay. Defects close to the surface will produce eddy current response earlier in time than that of deep-surface defects. The response obtained from PEC testing is rather more complex than multi-frequency ECT. PEC has an inadequate dynamic range limited by sharing of driver power to entire frequency spectrum and slow repetition rate limited by settling time of the PEC waveform.

1.2.5 Defect sizing using eddy current testing

Determination of shape, size, location, and orientation of defects in a component is essential for fracture mechanics based studies and for structural integrity assessment of components. EC probe merely maps the electrical conductivity profile at the point of interrogation in a test material. Thus, the measurements made are relative and not absolute. As a result, calibration or

reference standards consisting of artificial defects are used for comparison and for interpretation of the measured EC signal or image and in turn, for defect sizing [21]. Calibration standards are made from a specimen having identical geometry, material properties and other metallurgical conditions such as ageing, etc. as that of the component being inspected. Artificial defects such as saw cuts, flat bottom holes, and electro-discharge machining (EDM) notches are often used to represent the real-world linear as well as volumetric defects, for EC instrument calibration and defect sizing. There are a number of standards such as ASME, ASTM, RCCMR, BS and BIS, which provide the standard guidelines for preparation of EC test procedures and for carrying out EC tests on products and components.

As many disturbing variables change the probe coil impedance, detection and sizing of defects is not straight forward. In EC testing, defect sizing is an inverse problem of electromagnetism and this requires mathematical modeling, optimization, artificial intelligence, and machine learning.

1.3 INVERSE PROBLEMS

One of the scientific procedures to study any physical system is by representing the system as a mathematical model and analyze. It can be assumed that the underlying fundamental physics of the system are well understood, so a function, G , relating the model parameters and observations of the system may be specified as

$$G(\mathbf{m}) = \mathbf{d} \quad (1.7)$$

where \mathbf{m} , is the vector of parameters characterizing the model and \mathbf{d} is the vector of observable data. With this definition of the mathematical model of a physical system, two major classes of problems can be defined: 1) Forward problem and 2) Inverse problem. Forward problem is defined as prediction of the observable data \mathbf{d} , given the model parameters \mathbf{m} . Practically, \mathbf{d} may be a function of time or space or collection of discrete observations. The forward mapping $G: \mathcal{M} \rightarrow \mathcal{D}$ is well defined. Given \mathbf{m} , G is the function which can take many forms. In some

cases, G is an ordinary differential equation (ODE) or partial differential equations (PDE). In some other cases, it is a linear or non linear system of algebraic equations. Given the model parameters, \mathbf{m} , the observable or measured data, \mathbf{d} , are uniquely identified due to the causality in the physical system. Forward mapping is usually many-to-one mapping. Solving the forward problem helps in understanding the underlying physics of the system and enables prediction of observable data for a known model parameters.

Inverse problems are defined as the use of observed data, \mathbf{d} , to infer the model parameters, \mathbf{m} . The function space $G^{-1}: \mathcal{D} \rightarrow \mathcal{M}$ defines inverse mapping with the function $\mathbf{m} = G^{-1}(\mathbf{d})$. The function G^{-1} is far more than computing simply a mathematical function. In many situations inverse problems are ill-posed. A function is said to be ill-posed if it does not possess any of the three following properties:

1. Existence of solution
2. Uniqueness of solution
3. Stability of solution

Inverse problems are ill-posed because there may be no model that exactly fits the data. This can occur in practice because mathematical model is an approximate or the data contains noise which complicates both inverse and forward problems. In contrast to forward mapping, inverse mapping is one-to-many and many models can adequately fit the data. Thus, there is a lack of uniqueness. The inverse problems are numerically unstable, i.e. small changes in the measurement data may lead to large changes in the model parameters. Therefore, inverse problems do not have a solution or if they have a solution, it may not be unique or might not depend continuously on the model or both [22]. Inverse problems are popular in many branches of science and mathematics, including computer vision (determination of objects from photographic images), medical imaging (detection of tumors), geophysics (understanding density

and velocity of a propagating wave), remote sensing (determination of objects on the Earth), astronomy (understanding properties of objects in the universe) and nondestructive testing (estimation of electrical conductivity, magnetic permeability, and sizing of defects) [23,24]. In order to solve the ill-posed inverse problems, several procedures such as regularization [25], Bayesian inversion [23], and empirical inversion [26] are reported in open literature.

1.3.1 Eddy current inversion

Eddy current inverse problem can be described as the task of reconstruction of an unknown defect from eddy current signals or image. Its success will have a significant impact on accurate sizing of defects in engineering components. The forward model of EC testing abstractly can be defined in a function space as \mathcal{G} :

$$\mathcal{G}(\mathbf{d}) = \mathbf{s} \quad (1.8)$$

The function \mathcal{G} can be stated as a map from defect space (\mathbb{D}) to observed signal/image space (\mathbb{S}):

$$\mathcal{G}: \mathbb{D} \rightarrow \mathbb{S}$$

With this definition two types of solution for inverse problems exist: Direct inversion and Empirical inversion.

1.3.2 Direct inversion method

Direct inversion defines an initial defect profile \mathbf{d}_0 , and modifies the defect profile iteratively, such that the error e is minimized, where error e is the difference between the actual defect profile \mathbf{d} and the modified defect profile \mathbf{d}' .

$$\begin{aligned}
 e &= \mathcal{G}(\mathbf{d}) - \mathcal{G}(\mathbf{d}') \\
 &= |\mathbf{s} - \mathcal{G}(\mathbf{d}')|
 \end{aligned}
 \tag{1.9}$$

where \mathbf{s} is the variable representing observed signal or image itself or features derived from it. In general, defect profile \mathbf{d}' is accepted, for any arbitrary ϵ , if $e \leq \epsilon$. If ϵ is too large, convergence to a solution will be faster and the solution requires less number of iterations but the accuracy will be compromised. If ϵ is too small, convergence will be slow and the solution will require more number of iterations, but the accuracy of the predicted defect profile will be reliable.

The task of modifying the defect profile can be performed by using traditional continuous optimization methods such as steepest descent and conjugate gradient methods [27]. Discrete heuristics search and swarm intelligence strategies for optimization such as genetic algorithm and particle swarm optimization can also be used for error minimization. Typical flowchart for a direct inversion method is shown in Figure 1.7.

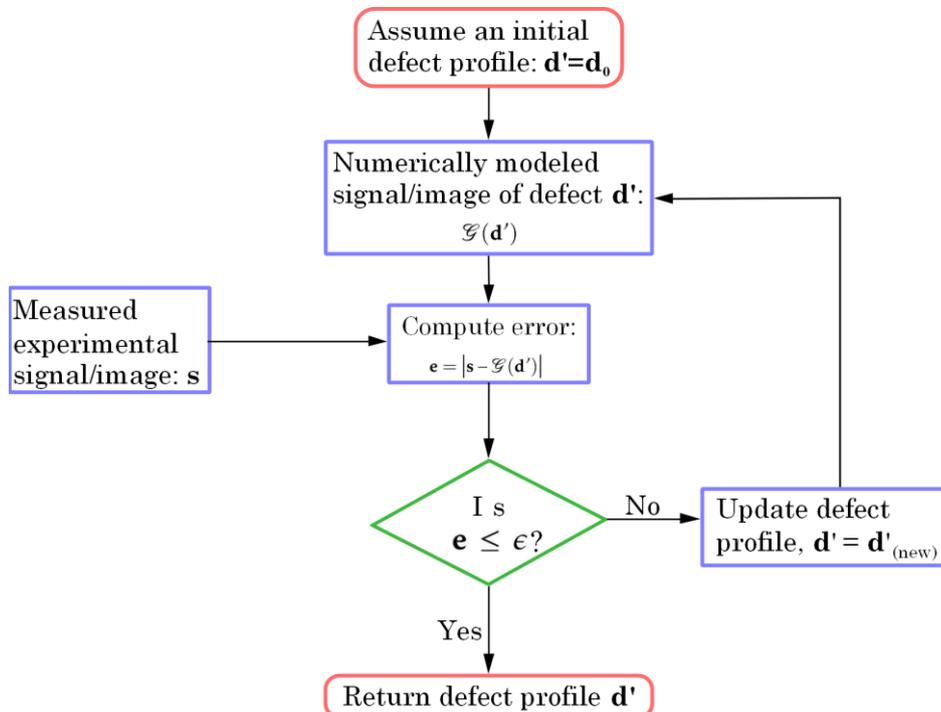


Figure 1.7 Flowchart of a direct inversion method for solving inverse problems.

Convergence to a solution depends on the initial defect profile used for inversion. In many practical situations, the initial defect profile will be random in nature. For accurate estimation, ϵ should be essentially a small value and hence this will be computationally intensive and therefore, has limited practical application [28].

In eddy current NDE, direct inversion is complicated, due to the diffusive nature of the electromagnetic fields. The interactions are exponential and different in depth and lateral directions, depending on the PSF of the EC probe and excitation frequency. Several variables simultaneously affect the probe coil impedance change. Further, various defect characteristics and noise influence the impedance change differently. The above mentioned processes are non-linear. Hence, direct inversion to obtain defect dimensions from EC signals or images is very cumbersome.

1.3.3 Empirical inversion method

Empirical inversion is the problem of computing an inverse function for the forward model \mathcal{G} .

$$\begin{aligned} \text{Forward model } \mathcal{G}: \mathbb{D} &\rightarrow \mathbb{S} \\ \Rightarrow \mathcal{G}(\mathbf{d}) &= \mathbf{s} \end{aligned} \tag{1.10}$$

$$\begin{aligned} \text{Inverse model } \mathcal{G}^{-1}: \mathbb{S} &\rightarrow \mathbb{D} \\ \Rightarrow \mathcal{G}^{-1}(\mathbf{s}) &= \mathbf{d} \end{aligned} \tag{1.11}$$

Since, \mathcal{G}^{-1} is not a well defined function in mathematical sense, in empirical inversion it is approximated with a known function \mathcal{F} such that

$$\mathcal{G}^{-1}(\mathbf{s}) \approx \mathcal{F}(\mathbf{s}) \tag{1.12}$$

such that

$$\mathcal{F}(\mathbf{s}) = \mathbf{d} \tag{1.13}$$

The known function \mathcal{F} is determined from a set of universal approximation functions (UAF).

Any function can be approximated as accurately as desired by a function in UAF. The class of

functions such as multilayer perceptron with one or more hidden layers, the class of radial basis functions are the example of UAF [29]. Functions from the set of UAF are parametric in nature and it is essential to find optimal parameters. The task of finding the optimal parameters is usually carried out by learning algorithms. The learning algorithms generally require similar data with known outputs for optimization called training data, which is obtained either by forward modeling or by systematic experimentation and covering all possible defect characteristics. Empirical inversion has been reported for reconstruction of defects from eddy current data [30,31,32,33,34]. The direct and empirical inversion methods are compared in Table 1.1.

Table 1.1 Comparison of direct inversion and empirical inversion methods.

CRITERIA	DIRECT INVERSION	EMPIRICAL INVERSION
Optimization parameter	Defect profile \mathbf{d}' . Such that $\mathcal{G}(\mathbf{d}') = \mathbf{s}$	Inverse function \mathcal{G}^{-1} . Such that $\mathcal{G}^{-1}(\mathbf{s}) \approx \mathcal{F}(\mathbf{s})$ and $\mathcal{F}(\mathbf{s}) = \mathbf{d}$
Preparation time of algorithm	<ul style="list-style-type: none"> • Training of data to the algorithm is not required and hence, no initial preparation time. 	<ul style="list-style-type: none"> • Initial training of data is required to find the optimal parameters for the approximation function. • Time depends on size of training data and complexity of function and learning algorithm.
Running time of algorithm	<ul style="list-style-type: none"> • Depends on initial defect profile and value assigned to ϵ • Number of iterations required for convergence. • Repetition of whole process required for each new signal/image. 	<ul style="list-style-type: none"> • Once trained, running time is small. • Can be deployed any number of times with same trained function.
Accuracy	<ul style="list-style-type: none"> • Depends on initial defect profile and value assigned to ϵ. • Exactness of the physical process. 	<ul style="list-style-type: none"> • Depends on training data. • Generalization capability of approximation function and learning algorithm.
Search space	Infinite	Finite
Search methodology	Incomplete search in the infinite space	Complete generalization in the finite space

1.4 MACHINE LEARNING FOR EMPIRICAL INVERSION

In NDE, automated detection and sizing of defects got richly benefitted in the recent years by machine learning, especially to obtain operator independent and consistent information of defects. Machine learning was applied for classification of different types of welding defects using data from ultrasonic testing [35] and radiography [36]. Apart from defect detection, machine learning was also applied to characterize metal loss from magnetic flux leakage signals [37]. Defect detection and sizing height of defects in composite materials and complex shapes were also reported using machine learning algorithms [38,39].

Machine learning is a branch of artificial intelligence with concepts influenced from different fields of science, including mathematics, computer science, statistics, biology, and economics. It involves development of computer programs/algorithms which improve the performance at certain task through experience [40]. In the past decade, machine learning was successfully implemented for performance of various real world tasks other than NDE, such as medical decision support [41], automated image classification [42], and speech recognition [43].

Machine learning can be defined as a function map. Given a dataset \mathcal{D} containing N samples of $\{(\mathbf{x}_n, y_n)\}_{n=1}^N$ pairs called training data. This training data is the experience provided to the machine learning algorithm. Training data is a set of past observations with $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^d$ as the input and its outcome $y \in \mathcal{Y}$. The goal of machine learning is to *learn* a function f that maps the input space \mathcal{X} to output space \mathcal{Y} :

$$\begin{aligned} f: \mathcal{X} &\rightarrow \mathcal{Y} \\ \Rightarrow y &= f(\mathbf{x}) \end{aligned} \tag{1.14}$$

This function $f(\cdot)$ is capable of predicting the outcome y_{pred} for any new input \mathbf{x}_{new} . Based on the nature of output variable y , machine learning is classified into different types. If the function

map is a continuous value in certain range, it is described as regression. On the other hand, if it is discrete, it is termed as a classifier or a classification function [44,45]. Learning can be unsupervised (y is unknown even for training data) or semi-supervised (y is partially known). The general steps followed in a learning system are shown in Figure 1.8. The training data in the raw form will be unstructured or have more dimensions like raw eddy current image of a defect. Hence, it is required to extract certain features from the training data which have certain relation to the output. Feature extraction facilitates to structure and produce relatively low dimensional data which have non redundant information about the output. These features are served as input to a machine learning algorithm whose output is a learned function f_d , that can be used to predict the output for any unknown input. Machine learning algorithm has two steps 1) identification of a functional representation and 2) learning process corresponding to the identified function [46].

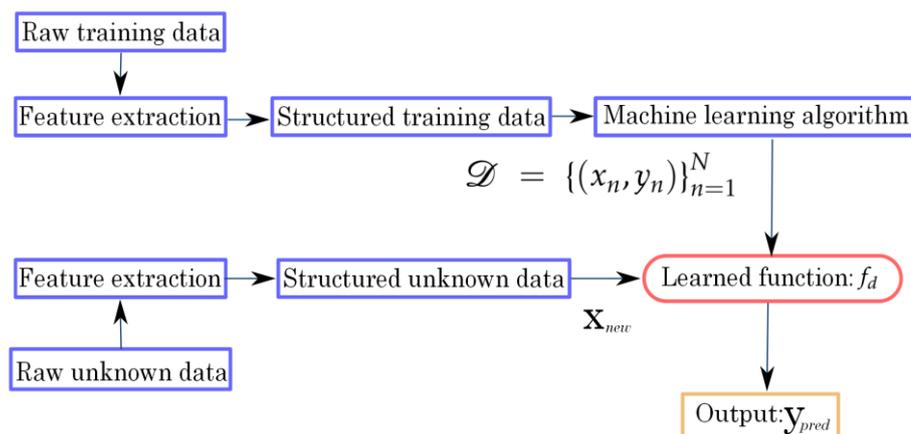


Figure 1.8 Steps followed in a learning system.

Functions must be represented in a formal language that can be computed. The representation decides a set, over which the learning algorithm searches a function. The set is called as *hypothesis space*. If a function which provides better solution for \mathbf{x}_{new} is not present in the hypothesis space, it cannot be learned by any learning algorithm [46].

Any set of machine learning functions representation is usually parametric in nature and required to find the optimal parameters that can predict the output accurately for unknown input. The optimal parameters can be obtained by a learning algorithm. The learning algorithm can be defined as a map \mathcal{L} from training dataset \mathcal{D} to a function $f_d: \mathcal{L}: \mathcal{D} \rightarrow F_d$. The objective of learning is to predict y accurately for a new \mathbf{x} , not present in dataset \mathcal{D} using the learned function f_d from \mathcal{D} such that $y_{\text{pred}} = f_d(\mathbf{x}_{\text{new}})$. There are many issues with finding the optimal parameters such as the existence of local minima, the reachability of global minimum and generalization to unseen data.

1.5 MACHINE LEARNING ALGORITHMS

Advancements in the field of machine learning have produced many practical algorithms, for addressing the issues mentioned in Section 1.4. Some of the function representations and learning algorithm commonly used in machine learning are summarized in Table 1.2 and Table 1.3 respectively.

Table 1.2 Types of function and their representations.

REPRESENTATION	FUNCTION
Instance based	<ul style="list-style-type: none"> • K-nearest neighbors • Support vector machine
Hyperplanes	<ul style="list-style-type: none"> • Naïve Bayes
Axis parallel hyperplanes	<ul style="list-style-type: none"> • Decision trees
Neural nets	<ul style="list-style-type: none"> • Multilayer perceptrons • Radial basis function neural nets
Graphical models	<ul style="list-style-type: none"> • Bayesian networks

Table 1.3 Examples of functions and learning algorithms.

FUNCTION	LEARNING ALGORITHMS
Decision trees	<ul style="list-style-type: none"> • Information Gain • Gini Index
Support vector machine	<ul style="list-style-type: none"> • Quadratic programming • Sequential minimal optimization
Multilayer perceptron	<ul style="list-style-type: none"> • Back-propagation • Quick propagation • Resilient back propagation
Radial basis function neural networks	<ul style="list-style-type: none"> • Orthogonal least squares

The representation and learning algorithms for different types of machine learning relevant to this thesis viz. ANN and SVM are discussed in this section.

1.5.1 Artificial neural networks

Artificial neural networks are based on emulation of biological nervous systems. They combine a large amount of simple processing elements which are highly interconnected. The processing elements are generally called neurons and they are structured in layers. Graphically, architecture of a typical ANN is shown in Figure 1.9. The first layer is called the input layer and the last layer is termed as the output layer. The layers of neurons, which are interconnected between input and output layers, are called hidden layers. Mathematically, a neuron function $g(x)$ is a composition of other functions which are usually a nonlinear weighted sum: $g(x) = h(\sum_j \mathbf{w}_j x)$, where $h(\cdot)$ is a predefined non-linear activation function. Some of the commonly used activation functions are: Sigmoidal including logistic, hyperbolic tangent and arctangent in multilayer perceptron (MLP) neural networks while Gaussian and multiquadric in radial basis function (RBF) neural networks.

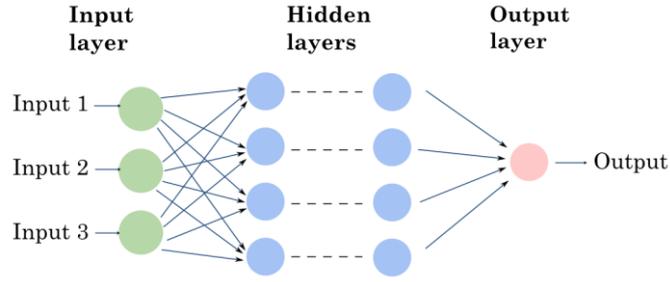


Figure 1.9 Architecture of a typical artificial neural network.

1.5.1.1 Multilayer perceptron

A neural network with sigmoidal units of neurons in the hidden layer is commonly called multilayer perceptron neural network [47]. The nodes produce the following output function based on logistic sigmoidal function:

$$g(\mathbf{x}, \mathbf{w}_j) = \frac{1}{1 + e^{-(w_{j0} + \sum_{i=1}^K w_{ji} \cdot x_i)}} \quad (1.15)$$

The commonly used learning algorithm for MLP neural networks is back-propagation. The MLP neural networks have an important property: the family of real functions that represent those networks can approximate any function with enough precision, if requisite number of hidden nodes are selected. However, due to large number of variables, tuning of MLP neural networks is a time consuming process. Moreover, it considers the entire input data as a single entity and local information about the input space are not considered. Local information about the input data is very important for the eddy current inverse problems because, it cannot be represented as a mathematical function.

1.5.1.2 Probabilistic neural networks

A probabilistic neural network (PNN) is a four layered feed forward network [48] viz. input layer, pattern layer, summation layer and output layer as shown in Figure 1.10.

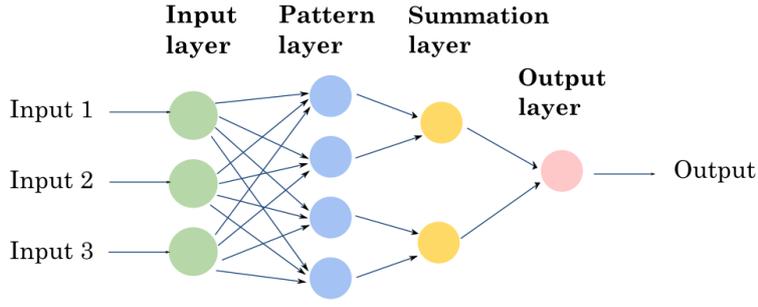


Figure 1.10 Architecture of a probabilistic neural network.

The pattern layer accepts the weighted sum of inputs and applies Gaussian function as

$$g(z_i) = \exp\left(\frac{z_i - 1}{s^2}\right) \quad (1.16)$$

where z_i can be defined as follows

$$z_i = \sum_j x_j \cdot w_{ji} \quad (1.17)$$

The summation layer sums the outputs from the pattern layer nodes corresponding to a specific class. The output layer determines the output using user defined weight for each class as

$$f_x(x) = c_k \sum_i g(z_i) \quad (1.18)$$

The training process of PNN is essentially determining the value of s and this does not involve any iterations. Often the values of s and c_k are user defined parameters and the user need to determine them based on the relevant importance of individual classes. In general, training of PNN is faster than MLP. Training samples can be added or removed without extensive retraining. PNN has large memory requirements and as a result, slow in execution. Moreover, to achieve a good classification, large training data set is required, even more than that of MLP neural networks and other types of neural networks. PNN completely relies on *a priori* information from the training data.

1.5.1.3 Radial basis function neural network

A radial basis function (RBF) is a real-valued function whose value depends on distance from the origin or any other points called *centroids*. This RBF function allows data to contribute the neural network but not equally and it is based on its distance from the centroids. The contribution of the data decays with increase in distance from the centroids. RBF neural networks are represented as three layer neural network and have RBF nodes in the hidden layer. A general architecture of a RBF neural network is given in Figure 1.11.

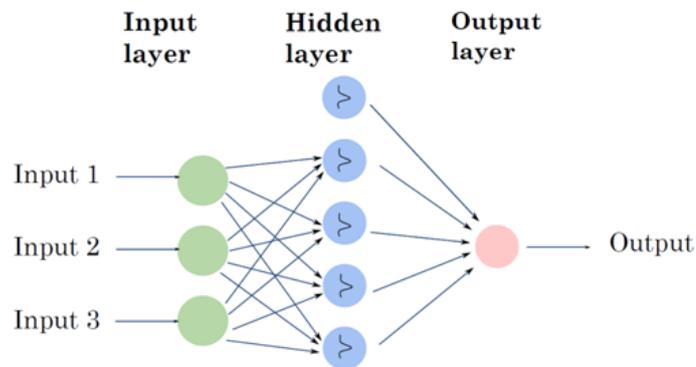


Figure 1.11 Architecture of a RBF neural network.

Each hidden node makes an independent approximation of the input space, typically by an activation function. There are different types of activation functions which include:

GAUSSIAN FUNCTION:
$$\Omega(\mathbf{x}) = e^{-\left(\frac{\|\mathbf{x}, \mathbf{v}\|_2^2}{2\sigma^2}\right)} \quad (1.19)$$

MULTIQUADRIC FUNCTION:
$$\Omega(\mathbf{x}) = \sqrt{1 + \frac{\|\mathbf{x}, \mathbf{v}\|_2^2}{2\sigma^2}} \quad (1.20)$$

INVERSE MULTIQUADRIC
FUNCTION:

$$\Omega(\mathbf{x}) = \frac{1}{\sqrt{1 + \frac{\|\mathbf{x}, \mathbf{v}\|_2^2}{2\sigma^2}}} \quad (1.21)$$

INVERSE QUADRIC
FUNCTION:

$$\Omega(\mathbf{x}) = \frac{1}{1 + \frac{\|\mathbf{x}, \mathbf{v}\|_2^2}{2\sigma^2}} \quad (1.22)$$

WINDOW FUNCTION:

$$\Omega(\mathbf{x}) = \begin{cases} 1 & x \leq 1 \\ 0 & x > 1 \end{cases} \quad (1.23)$$

where \mathbf{x} is the input, \mathbf{v} is the centroid, $\|\cdot\|_2$ is the ℓ^2 norm, and σ is the *spread* parameter. The different activation functions with a constant spread parameter σ , can be visualized as given in Figure 1.12.

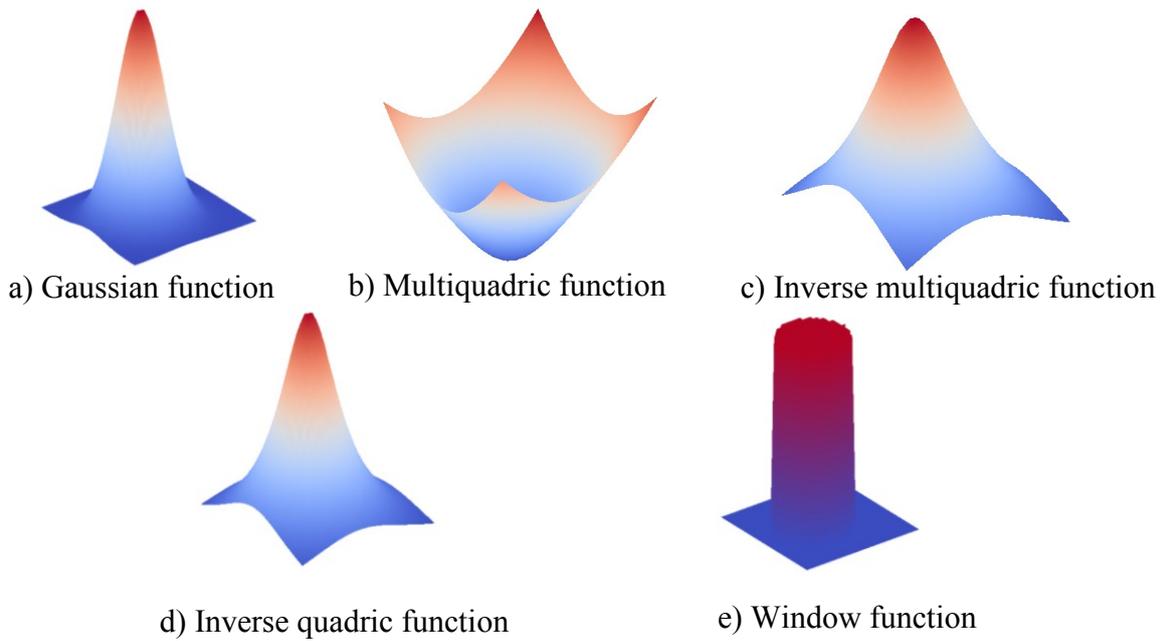


Figure 1.12 Various RBF activation functions.

The output layer of a RBF neural network combines the effect of all the hidden nodes by weighted sum of each obtained value in the hidden layer. The key aspect is that each node is

placed in a region in the input space i.e. centroids and locally they represents the whole input space by a specific radius or *spread* parameter, σ for generalization.

The learning process for RBFs consists of determining and tuning the hidden layer nodes and σ in the input space such that entire input space is mapped. Generally, unsupervised learning algorithms such as k-means algorithm and its variants and Kohonen self organizing maps are employed for the purpose. The learning process also involves determination of weights from the hidden layer to the output layer. The weight determination is usually carried out using singular value decomposition (SVD) or orthogonal least squares in order to minimize the error between the hidden and the output layer mapping:

$$E = \frac{1}{2} \sum_{i=1}^m (f(\mathbf{x}_i) - t_i)^2 \quad (1.24)$$

where t_i is the true output.

RBF neural networks have been proved to be universal approximation functions. Compared to the MLP neural networks, RBF neural networks have the advantage of utilizing local information about the input space in the function and as a consequence, only some neurons based on their vicinity in the input space will be activated for a specific input. This facilitates the training process, and both local optima and error surface complexity are reduced during weight optimization. The MLP training generally is a single phase learning process, while it is two phase for the RBF neural networks: determination of centroids by unsupervised learning and weight optimization by supervised learning.

1.5.2 Support vector machine

Support vector machine (SVM) is a non-probabilistic binary linear classifier [49]. It discriminates different categories of training data by a linear separator \mathbf{w} , such that they are divided by a clear gap that is as wide as possible, as shown in Figure 1.13. Let the training data

$\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^m$ is the input data and $y_i \in \{-1, +1\}$ is the output value. N is the size of training data. The general form of SVM prediction function:

$$f(\mathbf{x}) = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle + b \quad (1.25)$$

where $\langle \cdot, \cdot \rangle$ is the dot product in Ω and the function ϕ is the mapping $\phi: \mathcal{X} \rightarrow \Omega$. The goal of SVM is to determine the values of \mathbf{w} and b .

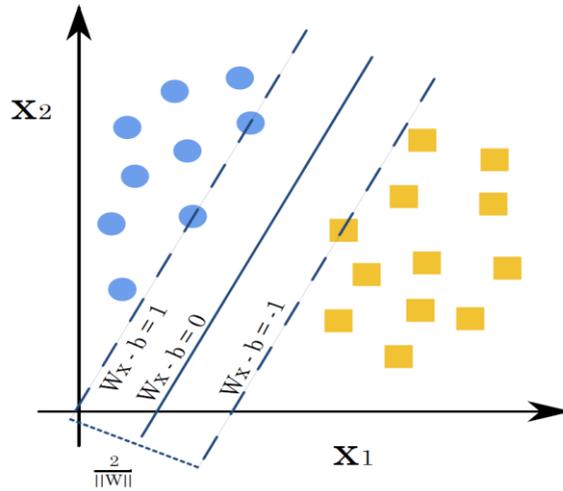


Figure 1.13 Typical decision boundary learned by support vector machine.

The basic requirement of a SVM is to find a decision boundary in the form of hyperplane that bisects the training data into separate classes. A constraint is posed to the decision boundary that it should contain a large margin between the classes. Formally, it leads to the convex optimization problem:

$$\min_{\mathbf{w}} \frac{\|\mathbf{w}\|^2}{2} + C \left(\sum_{i=1}^N \xi_i \right)^k \quad (1.26)$$

$$\text{subject to: } \begin{aligned} y_i(\mathbf{w} \cdot \mathbf{x}_i + b) &\geq 1 - \xi_i \\ \xi_i &\geq 0 \end{aligned}$$

where ξ is the slack variable added to tolerate error. It is possible to construct Lagrangian for the optimization problem and this problem is more solvable in its dual form. After several reductions of the optimization problem in its dual form, the prediction function $f(\mathbf{x})$ becomes:

$$f(\mathbf{x}) = \sum_{i=1}^N \lambda_i y_i \mathbf{x}_i \cdot \mathbf{x} + b \quad (1.27)$$

The above function for classification is applied when the data is linearly separable. For nonlinear classification, kernel method is applied. The kernel method allows computation of dot products in high dimensional feature spaces, using simple functions defined on training data. It facilitates formulation of nonlinear variants of any algorithm that can be cast in terms of dot products. The basic motivation for working in high dimensional spaces is that it can help to linearly separate the data, which otherwise, not linearly separable in the input space. The prediction function $f(\mathbf{x})$ using kernel method is given by:

$$f(\mathbf{x}) = \sum_{i=1}^N \lambda_i y_i k(\mathbf{x}_i, \mathbf{x}) + b \quad (1.28)$$

where $k(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle$. The kernel is a symmetric function and the availability of such a function is ensured by the Mercer's theorem [49]. Some of the popular kernels include:

4. Polynomial kernel: $k(\mathbf{x}, \mathbf{y}) = (\langle \mathbf{x}, \mathbf{y} \rangle + c)^d$ (1.29)

5. Radial basis function kernel: $k(\mathbf{x}, \mathbf{y}) = e^{-\left(\frac{1}{\omega^2} \|\mathbf{x}, \mathbf{y}\|_2^2\right)}$ (1.30)

6. Sigmoidal kernel: $k(\mathbf{x}, \mathbf{y}) = \tanh(\beta \langle \mathbf{x}, \mathbf{y} \rangle + \gamma)$ (1.31)

where $c, d, \omega, \beta, \gamma$ are the parameters of the kernel.

Advantage of SVM includes scalability to large datasets, noise tolerance, and good generalization capability [45]. However, SVM is applicable only for binary classification. For multiclass classification, techniques such as one-against-one class and one-against-all classes

have to be applied [45]. In one-against-one class classification, each pair of classes is considered as a binary classification. In one-against-all class classification, a binary classification is performed for each possible class, where the class is coded as positive training data and all other classes are coded as negative. By voting, a possible class value is determined for both techniques.

1.6 SUMMARY

Eddy current testing is important for nondestructive detection of defects in metallic materials which are widely used in nuclear, aerospace, and other industries. Sizing defects (inversion) with respect to size and location is important for fracture mechanics based structural integrity assessment of engineering components and structures. Empirical inversion is a practically feasible technique for eddy current data and this thesis focuses on the development of novel machine learning algorithms for efficient sizing of defects from eddy current images.

2

EMPIRICAL INVERSION

This chapter covers the reported literature on defect sizing in eddy current inversion and related areas. It also identifies the gap areas which helped in setting the motivation and in identifying the objectives of the research work.

2.1 LITERATURE SURVEY

Copley (1983) [18] carried out eddy current imaging for defect sizing by superimposing a number of EC signals by generating parallel line scans for various rectangular and cylindrical surface defects in Rene 95 alloy. He noted that the length of surface defects showed a linear correlation with the geometric distance between the 10% maximum magnitudes towards the length direction. He also reported that the phase angle increases with defect height irrespective of the length. After systematic studies he discussed that first estimating height of surface defects and then finding length from the knowledge of magnitude change and defect height would be beneficial.

Udpa *et al* (1986) implemented MLP neural network for determination of type of defects in Inconel 600 tubes. The EC signals from defects in tubes were parameterized using Fourier descriptors [15], which are invariant under translation, rotation, and scaling, apart from data compression. Signals were obtained at 100 kHz from a differential EC probe. 4 different types of defects viz. through hole, axi-symmetric OD slot, and flat bottom hole were trained as four classes of MLP neural network. 8 Fourier descriptors of the EC signals from 24 different defects were fed as input to the MLP neural network. The results were compared

with conventional K-means algorithm and it was found that the MLP neural network achieved 100% classification for the 24 defects studied while K-means misclassified a through hole as an axi-symmetric slot [50].

Hedengren and Ritscher (1990) proposed three approaches to estimate the point spread function (PSF) of an EC probe by linear approximation [51] which can be used for defect sizing. The first approach considered a small hole (diameter: 0.254 mm) assumed as a point defect and its EC response was taken as the PSF. Second approach was based on the results of the first approximation where analytical model was used to produce a smooth approximation. The third approach minimizes the least mean square error (LMSE) between EC images and defect geometry of defects varying in lengths viz. 0.254, 0.508, 1.270, and 2.540 mm. It was reported that the LMSE method was better to estimate the PSF.

Mann *et al* (1991) [52] used EC signals from an uniform field EC probe at 10 different frequencies and determined the length and height of surface defects using MLP neural network. The network was trained using magnitude and phase information of 1000 simulated defects and MLP neural network was used to evaluate 8 defects of known dimensions. MLP was successful in determining the length and height, with an error of $\pm 10\%$.

Norton and Bowler (1993) proposed eddy current inversion studies using the direct iterative method [27]. In this method, a defect present in an electrically conducting material was represented in terms of electrical conductivity. An algorithm based on mean-square error formulated in terms of gradient of the impedance was implemented for inversion. The method derived the gradient analytically from a forward problem. It successfully determined a simulated crack of semi elliptical (length: 22.10 mm, height: 8.61 mm) and irregular shape (length: 49.78 mm, height: 8.94 mm) in a 24.00 mm thick aluminum alloy plate with an error of $\pm 4\%$ [53].

Burke (1994) proposed a method for estimating defect height using EC data [54]. This method was based on applying least square polynomial fit to the approximate solution of Maxwell's equation for nonmagnetic conducting materials. The method used computation of two functions: a normalizing function depends on probe parameters and *skin-depth*, and crack function depends on probe and defect height. It was implemented for EDM defects of varying heights from 1.0 mm to 12.0 mm on 3 different aluminum plates. It was found that the actual defect height and the estimated height from EC measurements were in close agreement with an error of $\pm 10\%$. Later, Burke (2001) applied the same algorithm using swept frequency data for a range of semi elliptical and rectangular defects [55]. It was determined that the heights of defects were estimated $\pm 15\%$ error as shown in Figure 2.1.

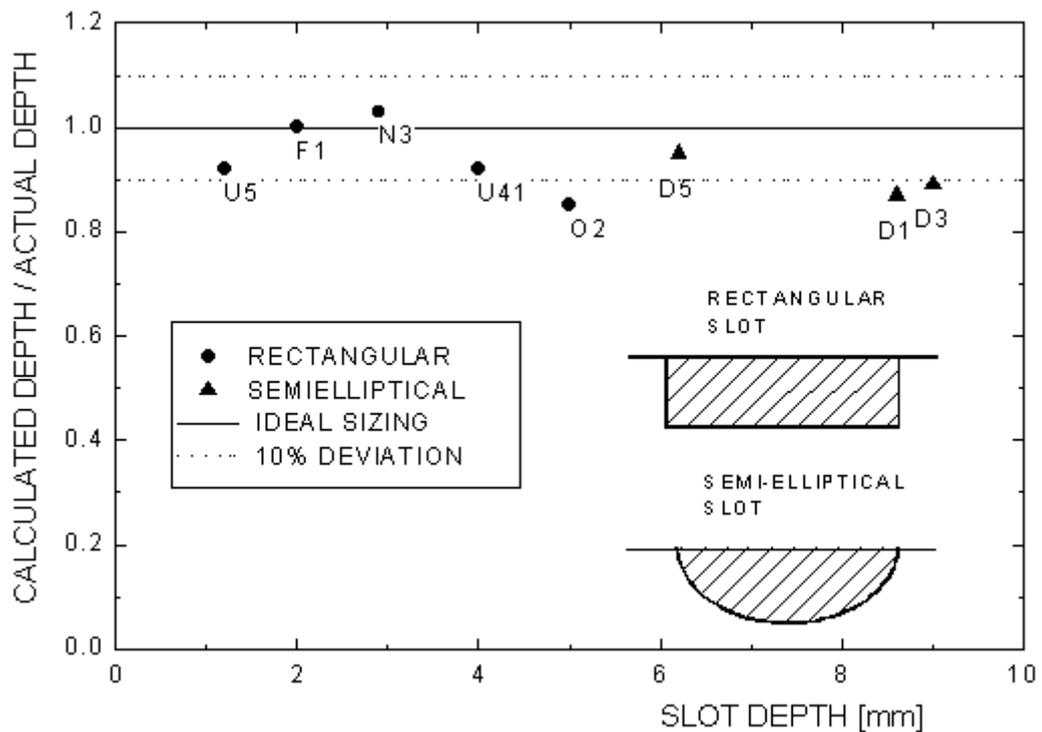


Figure 2.1 Crack depth determined by swept-frequency eddy current NDE for a series of rectangular and semi-elliptical EDM slots in Al alloy plates [55].

Shyamsunder *et al* (2000) [56] studied classification of different types of surface defects viz. partial/through thickness holes and notches of various dimensions, fatigue cracks, stress

corrosion cracks and wall thinning in stainless steel sheet of 1.0 mm thickness and stainless steel plate of 12.0 mm thickness. They used four different types of probes varying size and core, 8 different scanning patterns and three different frequencies viz. 100 kHz, 250 kHz, and 500 kHz. Results of classification using a total of 72 extracted features including pulse shape features, power features, and statistical descriptors were reported for 6 different classifiers viz. linear discriminant, minimum distance, empirical Bayesian, K-nearest neighbor, MLP neural network, and Kohonen self organizing map. They reported best results for MLP neural network which produced 98% accuracy for classification of holes and notches, and 100% accuracy for fatigue crack, stress corrosion cracks and wall thinning.

Song and Shin (2000) proposed empirical inversion method based on 2D finite element method (FEM) and MLP neural networks [57]. They considered axisymmetric defects in an 1.28 mm thick Inconel 600 tube. 4 types of defects were studied; 'I' and 'V' shaped present on inner diameter (ID) surface and in outer diameter (OD) surface of the tube. 200 defect samples were simulated using FEM at 100 kHz and 400 kHz. 3 artificial OD grooves were used with 75%, 58%, and 38% depth of tube wall thickness. Dataset was generated by extracting 22 features (11 from each frequency) from the defects. Based on a study related to performance and redundancy by individual feature, 10 features were selected. A probabilistic neural network (PNN) was used to classify the 4 defect types with 91% success. Further, four different MLPs were used for sizing type of defect and a separate MLP for sizing width and height. A linear correlation of 0.97 was reported between the actual and estimated size of width and height of defects. However, the methods used in this study required 6 different classification algorithms for determining the shape and width and height values. Training and parameter tuning of different classification algorithms are tedious and cumbersome. Efforts to develop a single algorithm that can simultaneous size of all the characteristics of defects are not found in literature.

Rao *et al* (2000) [58] proposed a scheme using artificial neural network and image processing method to address the three important aspects concerning sizing of defects from EC images viz. influence of disturbing variables, image blurring, and large imaging time. In this scheme, an MLP neural network was first developed to evaluate the critical and most important characteristic of defect i.e. the height of defects in the presence of disturbing variables. Using this neural network, raster scan imaging was carried out to form height-profile images, which were later subjected to an image processing method for restoration of length and width of the defects. The proposed scheme was implemented on stainless steel plates and welds. 56 EDM defects with heights ranging from 0.200 to 1.400 mm were considered, of which 24 were used for training the online neural network and 32 for testing. An absolute EC probe of 4.000 mm diameter was used with an excitation frequency of 75 kHz and 150 kHz. 12 features were used as input for online neural network including magnitude, phase, and their higher orders. With raster scan imaging, the maximum deviation using optimized and trained neural network for evaluation of height was found to be 0.055 mm. The height-profile images were subjected to image processing involving edge detection, filtering algorithms, determination of the effective diameter of the probe, and removal of blur for restoration of length and width. With processing of images the maximum deviation in length and width estimation was found to be 3 times the scan interval of raster scanning.

In order to realize fast imaging and sizing of defects a two step intelligent imaging scheme was proposed [59]. Binary images were obtained in a first step at coarse scan pitch of 1.0 mm and defect region were localized with defective and non defective region. This localized region was subjected to fine scan with a pitch of 0.3 mm to obtain accurate height-profile image using separate neural networks for linear and circular defects. Typical result obtained by superimposing binary image on height profile image and magnitude image were shown in

Figure 2.2. Localised binary image and height profile image were fused to form 3-dimensional sketch of a defect as shown in Figure 2.3.

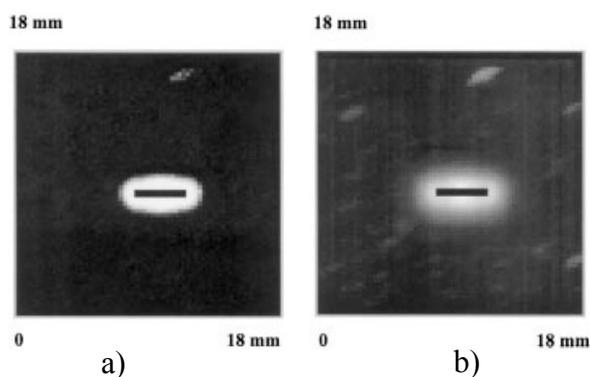


Figure 2.2 The restored image of a notch (length 4.0 mm, width 0.3 mm, depth 0.4 mm) superimposed (after assigning a gray-level value of 0 for the sake of visual convenience) a) on the height-profile image and b) on the magnitude image [59].

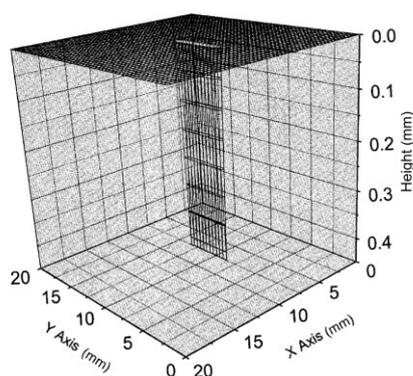


Figure 2.3 3-D image of a notch (length 6.0 mm, width 0.3 mm, depth 0.4 mm) in a stainless steel plate after fusion of depth-profile image and restored image [59].

This work was extended by Rao *et al* (2002) [60] for estimation of height of surface defects in stainless steel welds. 48 EDM defects with height in the range of 0.20 mm to 1.50 mm and length in the range of 0.20 mm to 1.50 mm were introduced in a weld region of a stainless steel plate. 24 defects were used for training of online neural network and another set of 24 were used for testing. They reported that the online neural network was able to evaluate the height with the maximum deviation of 0.08 mm.

Thirunavukkarasu *et al* (2004) proposed radial basis function neural network, for estimating the height of defects in stainless steel plates with magnetic permeability variations [61]. EC signals from EDM notches with depth ranging from 0.200 mm to 1.200 mm and length 3.000 mm, 5.000 mm, and 8.000 mm were used to train the RBF neural network. An absolute coil probe of 4.000 mm diameter was used at 75 kHz and 150 kHz excitation frequencies. RBF neural network was able to evaluate depth with the maximum deviation of 0.035 mm. The interpolation capability of RBF neural network was studied by excluding few intermittent defects and training extreme defects and the maximum deviation was found to be 0.050 mm. The extrapolation capability of RBF neural network was also studied by estimating the extreme defect such as with depth 1.200 mm using network trained with defects which were ≤ 0.800 mm and found the deviation to be 0.055 mm.

Yusa *et al* (2002) [62] proposed a generalized MLP neural network for simulation of mapping between eddy current signals and defect profiles in an Inconel 600 tube with outer diameter 11.23 mm and wall thickness 1.27 mm. Signals were experimentally obtained with a scan pitch of 0.50 mm using a four sensor probe and an excitation coil which was optimized for maximum detection sensitivity. 400 cases of defects were used to generate the training data using FEM. The area of inspection was 10.00 mm in length, 0.20 mm in width and was divided into $20 \times 1 \times 10$ cells. Since the output neuron of MLP neural network is a real number, Yusa *et al* took the degree of excitation of the most-excited neuron also into consideration, and proposed this parameter as a q parameter. EC signals from two real defects were used with trained MLP neural network for validation and the results are shown in Figure 2.4. It was revealed that Defect 1 was 3.00–4.00 mm in length and 60% in maximum depth. Defect 2 was also 3.00–4.00 mm in length and 53% in maximum depth (the solid line in Figure 2.4). Good agreement between reconstructed and real profiles of the cracks was reported.

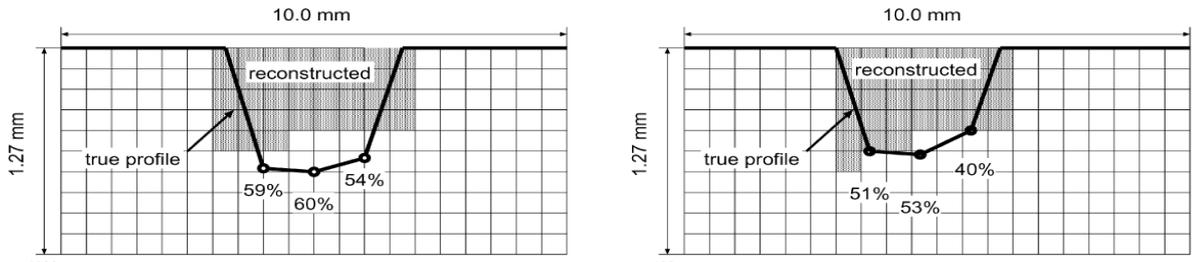


Figure 2.4 True and reconstructed profiles of natural cracks [62].

Chady *et al* (2007) [63] reported a multifrequency excitation and spectrogram (MFES) eddy current system and inverse model to detect defects in a 1.23 mm thick Inconel 600 plates. 15 defects of 3 different length viz. 2.00, 3.00, and 5.00 mm and 5 different heights ranging from 10% to 80% thickness of plate were studied at 15 different frequencies ranging from 36 to 204 kHz. An MLP neural network, combined with low pass filter (LPF) and signal approximation (X-APP) using a Gaussian function were used to estimate the defect profiles as shown in Figure 2.5. Frequency components of the sensor output signal, taken from a moving window of EC signal constituted the input. For estimation of defect length, 8 different combinations of LPF, X-APP, and frequency characteristic approximation were used.

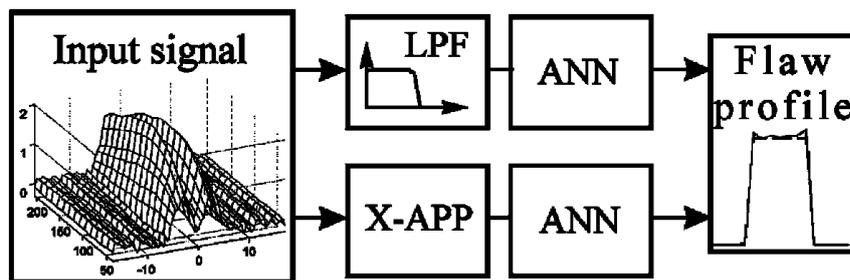


Figure 2.5 Scheme of inverse dynamic ANN models with a moving window for flaw profile estimation [63].

The algorithm was tested with signals of different SNR. All noise-free data and 20% of noisy data were used for training, 10% of noisy data were used for validation, and 70% for testing. It

was reported that better results (in the case of height and length estimation) were achieved using the approximation-based algorithms than that of the filtered signals, as typically shown in Figure 2.6.

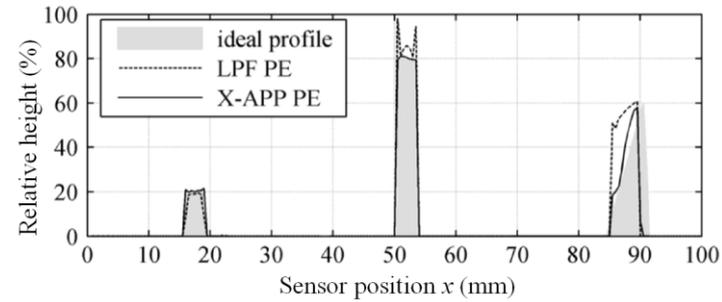


Figure 2.6 Estimated profiles of the defects: LPF is the identification using the filtered signals, X-APP identification using the approximated signals [63].

Though results of the methods proposed in [52] - [63] were encouraging, the reported studies were mostly concentrated on sizing height of the surface defects from EC signals. the applicability of these methods to detect and size subsurface and three dimensional defects involving length, width, depth, height, and orientation are neither known nor found in the open literature. For this, extensive study on feature extraction from EC images and development of novel inversion algorithms are necessary.

Ramos *et al* (2014) [64] studied SVM for estimation of thickness of plates from pulsed eddy current testing. EC signals of stainless steel plates with thickness ranging from 1.00 mm to 3.50 mm, Aluminum 1050 with thickness ranging from 1.00 mm to 5.00 mm, and Aluminum 3105 with thickness ranging from 1.00 mm to 7.00 mm were used for training the SVM. Raw signals along with 5 coefficients from discrete cosine transform of signals, sum of all the elements of the signal, and 4 values of auto regressive model of the signal were used as the features (2510 features). It was reported that during cross validation, the SVM was able to estimate the thickness of the plates with errors of 1.52% and maximum mean deviation around

0.05 mm. The use of SVM showed promising performance, however, the features used in [64] were found to be influenced by noise. Efficient feature extraction may be useful to handle the noise and efficient computation. The authors further suggested the use of other features such as principal component analysis for this purpose [64].

Most of the inversion studies reported in the literature [52] - [64], assumed that the EC response obtained is independent with respect to different defect characteristics. This assumption leads to the development of independent inversion algorithms for each defect characteristic. However, the electromagnetic fields from EC probe interact with all the defect characteristics simultaneously and the measured impedance change is a combined influence of all the perturbations in the defect region. In this context, there is a clear benefit if weightage is given to the use of image format and incorporate dependency among the defect characteristics. By studying the effect of dependency and implementing it during the inversion process the performance of inversion process is expected to enhance.

Bernieri *et al* (2008) [65] performed empirical inversion of subsurface defects in an aluminum plate of 2.00 mm thickness using two different machine learning algorithms viz. MLP neural network and SVM. The information of length was used to predict height of defect. Further, the information of length and height were used for predicting the depth of the defects. In this work, 200 numerically simulated subsurface defects (length: 3.00 mm, height: 1.00 mm, and depth: 1.00 mm) were used for sizing the length, depth, and height of defects. The influence of width of the defect was assumed to be negligible and width was kept constant at 0.50 mm. They used peak amplitude and peak distance as features for MLP neural network and SVM. They estimated the length with a mean absolute error (MAE) of 0.04 mm, depth and height with a MAE of 0.12 mm for a test set of 10 defects.

Recently, Bernieri *et al* (2014) [66] studied defect sizing using 550 numerically simulated subsurface defects (length: 1.00 mm to 9.00 mm, height: 0.20 mm to 1.80 mm, and depth: up to 1.60 mm) in an aluminum plate of 2.00 mm thickness. They used peak amplitude and peak distance obtained from 23 different excitation frequencies in the range 5 Hz to 5 kHz as features for SVM. They proposed a new architecture and in this the information of length was used to predict depth of the defect. Further, the information of length and depth were used for predicting the height of the defect. With the above approach, MAE values of 0.10 mm, 0.40 mm, and 0.08 mm were obtained for depth, length, and height respectively, for a test set of 12 defects.

Bernieri *et al* (2014) did not study the reason for variation in MAE by the two architectures and they did not compare the results with the existing algorithms in the literature. They carried out studies on aluminum plates which have high electrical conductivity. In such materials, the resolution of defect sizing along the depth direction was poor even at lower excitation frequencies. Moreover, with three different defect characteristics considered by Bernieri *et al*, there is the possibility of 13 different sequences, which were not systematically studied and optimized. In this context, it is beneficial, if a single machine learning algorithm is developed which incorporates the dependency implicitly. This will eliminate the optimization process to find the best sequence among the defect characteristics. Further, an extensive study to identify the most optimum sequence for defect sizing is important and essential.

Apart from the classical machine learning algorithms such as MLP, PNN, and RBF, there are some significant contributions to the literature of machine learning for better understanding of algorithms and their process. Bielza and Li (2011) studied machine learning algorithms capable of incorporating dependency [67]. They modified the classical probabilistic graphical Bayesian networks for machine learning problems with multiple outputs (multidimensional learning). It was implemented for dataset such as yeast gene expression analysis, natural scene construction from images and emotion analysis from music. However, learning with Bayesian networks is

strongly dependent on the quality of prior belief, i.e. statistical distribution of classes in a given dataset for classification. Further, it assumes the input features to be discrete. These methods cannot be applied for defect sizing from eddy current data, as it is difficult to obtain reliable *a priori* information from systematically generated data.

2.2 MOTIVATION

Stainless steels are one of the important materials in nuclear, petrochemical, and aerospace industries and these steels are subjected to eddy current testing. Eddy current testing is efficient in detecting surface and subsurface defects reliably. However, sizing of defects is very important for fracture mechanics based studies and for structural integrity assessment of components. Defect sizing from EC response in stainless steels is very limited in open literature. Being a low conducting material, use of high frequency EC testing will give the same *skin depth* as that of the high conducting materials, e.g. aluminum, but at a relatively higher SNR. This is encouraging for defect sizing in stainless steels, essentially because, for high SNR detection, highest possible frequency is preferred.

Inspired by encouraging performance of the empirical inversion in the literature, along with recent developments in the field of machine learning, studies can be focused on exploring empirical inversion for sizing of defects. The EC empirical inversion studies available in the literature, however, are limited to surface breaking planar cracks. But, formation of subsurface defects in components is a possibility during manufacturing and service life and their sizing is very important. Although, sizing of subsurface defects includes determination of orientation, multiple defect reconstruction and spacing between defects, the most important characteristics for subsurface defect sizing are length, width, depth, and height.

Eddy current empirical inversion of subsurface defects using machine learning is challenging, as it involves sizing of additional variable, i.e. depth, in contrast to surface defects extensively

studied in literature. It is also expected that the response from these defects is feeble as compared to that of surface defects, following the *skin effect* phenomenon. Attempts to develop machine learning algorithms for determination of all the four defect characteristics, viz. length, width, depth, and height are very limited in the literature.

Inversion algorithms proposed in the literature are concentrated of sizing of height of surface defects from EC signals. However, defects are three-dimensional functions of the space coordinates and they cannot be completely reconstructed by EC signals from a line scan. Due to limited information content available in the EC signals, use of EC images is advantageous for sizing defect characteristics. The two major challenges which directly influence the performance of machine learning algorithms when EC images are used for defect sizing are feature extraction and scan pitch (scan step size during raster imaging). Extraction of features from EC images relevant to each defect characteristic is essential and information on this for subsurface defects is very limited in open literature. Further, no literature is found concerning the influence of scan pitch of EC images on the performance of inversion algorithms for sizing. A study on determining the optimum scan pitch may be useful and this can reduce the time required EC imaging without compromising the defect sizing performance.

Traditional eddy current empirical inversion assumes that the EC response is dependent on one or two defect characteristics. However, it can be easily understood from the electromagnetic interactions that EC image is dependent on all the defect characteristics. The interactions are different in depth and length directions. It is evident from the literature that use of one defect characteristics as input to evaluate other defect characteristics is beneficial. It is attractive to develop learning algorithms that are capable of incorporating dependency in some way. This will fully exploit the expressive power of the learning algorithms and in turn, the algorithms will be capable of producing enhanced performance than the conventional algorithms.

All the inversion algorithms proposed in the literature implement separate learning algorithms for sizing of each defect characteristic. Training several algorithms and their parameter tuning is time consuming and cumbersome. Hence, it is beneficial, if research is focused to develop a single multidimensional learning algorithm that can simultaneously size all the defect characteristics.

RBF neural network is one of the most studied neural network structures in NDE and it is also proved to be a universal approximation function. Compared to other neural network architectures, RBF neural networks use fewer parameters, less complex, and more efficient in using local data information for generalization.

A clear benefit exists with an extensive study, to identify the most optimum sequence for defect sizing, where the outputs of known defect characteristics are fed as input to estimate the other unknown defect characteristics. The studies reported in this direction are incomplete and inconsistent. An extensive study concerning of determination of optimum sequence is expected to bring out the sequence of most dependent among the defect characteristics for sizing and it will enable accurate sizing. It is also worth exploring the prospects of developing a generalized framework with which any robust machine learning algorithm that incorporates dependency can be used for eddy current inversion, in particular, other NDE techniques, in general.

2.3 OBJECTIVE OF THE THESIS

The primary objective of the thesis is to develop efficient machine learning algorithms for empirical inversion of eddy current images to size defects in stainless steel plates. The objectives of the thesis are detailed below:

1. To develop a single multidimensional machine learning algorithm for sizing subsurface defects by incorporating dependency implicitly among the defect characteristics using features from eddy current images.

2. To study various features from eddy current images and identify features relevant to each of the defect characteristics.
3. To study the influence of imaging scan pitch on defect sizing and identify, if there is any limit scan pitch, for defect sizing.
4. To study the influence of dependency on defect sizing and to develop machine learning algorithms for sizing defects by incorporating dependency among different defect characteristics.
5. To evolve a generalized framework with which any other robust machine learning algorithm that incorporates dependency can be utilized for defect sizing purpose in NDE.

3

MULTIDIMENSIONAL RADIAL BASIS FUNCTION ALGORITHM FOR SIMULTANEOUS SIZING OF DEFECT CHARACTERISTICS

In this chapter, a new multidimensional radial basis function (MD-RBF) neural network is proposed for simultaneous sizing of 4 defect characteristics viz. length, width, depth, and height in a single machine learning algorithm with input features related to the 4 defect characteristics. MD-RBF neural network is designed to address dependency among the 4 defect characteristics. The robustness of the proposed MD-RBF neural network to size surface and subsurface defects is systematically studied and evaluated. The performance of MD RBF neural network is also compared with that of conventional RBF neural networks in this chapter. In this thesis, defects and notches are used interchangeably, because the dimensions of notches considered are large, they are realistic defects.

3.1 MULTIDIMENSIONAL LEARNING

Most of the previous studies in machine learning, based on supervised classification assume that the input data has to be classified with a single class. However, defect sizing requires classification of multiple class variables such as length, width, depth, and height from input features of eddy current images of defects. The machine learning algorithms for defect sizing task require a learning function that maps a vector of input features into a vector of class values

(outputs). This is a recent concept in machine learning, called multidimensional learning. In contrast to conventional machine learning, the dataset \mathcal{D} used for defect sizing is of the form that can be represented as $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$ as shown in Table 3.1.

Table 3.1 Training dataset for conventional machine learning and multidimensional learning

SAMPLE NUMBER	INPUT	OUTPUT	
		CONVENTIONAL LEARNING	MULTIDIMENSIONAL LEARNING
1	$x_1^1, x_1^2, \dots, x_1^m$	y_1	$y_1^1, y_1^2, \dots, y_1^d$
2	$x_2^1, x_2^2, \dots, x_2^m$	y_2	$y_2^1, y_2^2, \dots, y_2^d$
\vdots	\vdots	\vdots	\vdots
N	$x_N^1, x_N^2, \dots, x_N^m$	y_N	$y_N^1, y_N^2, \dots, y_N^d$

3.1.1 Multidimensional classification

The multidimensional classification attempts to learn a function $f(\cdot)$ that would assign a d dimensional vector of class variables $\mathbf{y}_i = \{y_i^j\}_{j=1}^d$ such that $y_i^j \in C^j = \{c_1^j, c_2^j, \dots, c_{k_j}^j\}$, $k_j \geq 2$ for an m dimensional input $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^m$:

$$\begin{aligned}
 f: \mathbb{R}^m &\rightarrow C^1 \times C^2 \times \dots \times C^d \\
 f(x^1, x^2, \dots, x^m) &= (y^1, y^2, \dots, y^d) \\
 \Rightarrow f(\mathbf{x}) &= (y^1, y^2, \dots, y^d)
 \end{aligned} \tag{3.1}$$

In the above mapping C^j represents the sample space of y^j and $C^1 \times C^2 \times \dots \times C^d$ denotes the space of their joint configuration. For defect sizing, the input \mathbf{x} is the features extracted from EC images and the output dimension d is the number of defect characteristics considered for sizing.

Certain intuitive approaches are possible for solving the multidimensional classification. One approach decomposes all the dimensions and solves each dimension like a conventional multiclass classification as follows:

$$f(\mathbf{x}) = (y^1, y^2, \dots, y^d) \Rightarrow \begin{cases} f_1(\mathbf{x}) = y^1 \\ f_2(\mathbf{x}) = y^2 \\ \vdots \\ f_d(\mathbf{x}) = y^d \end{cases} \quad (3.2)$$

However, by this approach, possible dependencies among the classes are not considered and the expressive power of the machine learning algorithms is restricted. Another approach builds a new set of classes by taking the Cartesian product of the classes belonging to all the dimensions. With this approach, dependency among classes is ensured. However, the number of new classes required for learning will increase exponentially, with the addition of each dimension and this will leave insufficient data to train each new class and with a high potential for overfitting.

Working in this direction, a new algorithm using radial basis function neural network is proposed. This is a single function capable of simultaneous classification of all the outputs, incorporating the dependency among them.

3.2 MULTIDIMENSIONAL RADIAL BASIS FUNCTION NEURAL NETWORK

Radial basis function neural network is one of the most studied ANN architecture, represented by three layers, input layer, hidden layer, and output layer. In general, the hidden layer is represented by prototype vectors formed from centroids, which are determined by performing cluster analysis on the input. The hidden layer is activated by a basis function usually of Gaussian type. In contrast to conventional neural network, the multidimensional radial basis function (MD-RBF) neural network is unique in a way that it takes the input of all defect characteristics in a single training and produces the output. This implicitly considers the dependency among the defect characteristics. The architecture of the conventional RBF neural networks and the proposed MD-RBF neural network are given in Figure 3.1.

The input layer nodes of MD-RBF neural network takes m dimensional input features \mathbf{x} extracted from EC images. The output layer nodes represents D dimensional output class

variable (defect characteristics), $\mathbf{y} = \{y^d\}_{d=1}^D$. Each class variable is discretized into few class values $y^d \in C^d = \{c_1^d, c_2^d, \dots, c_{k_d}^d\}$.

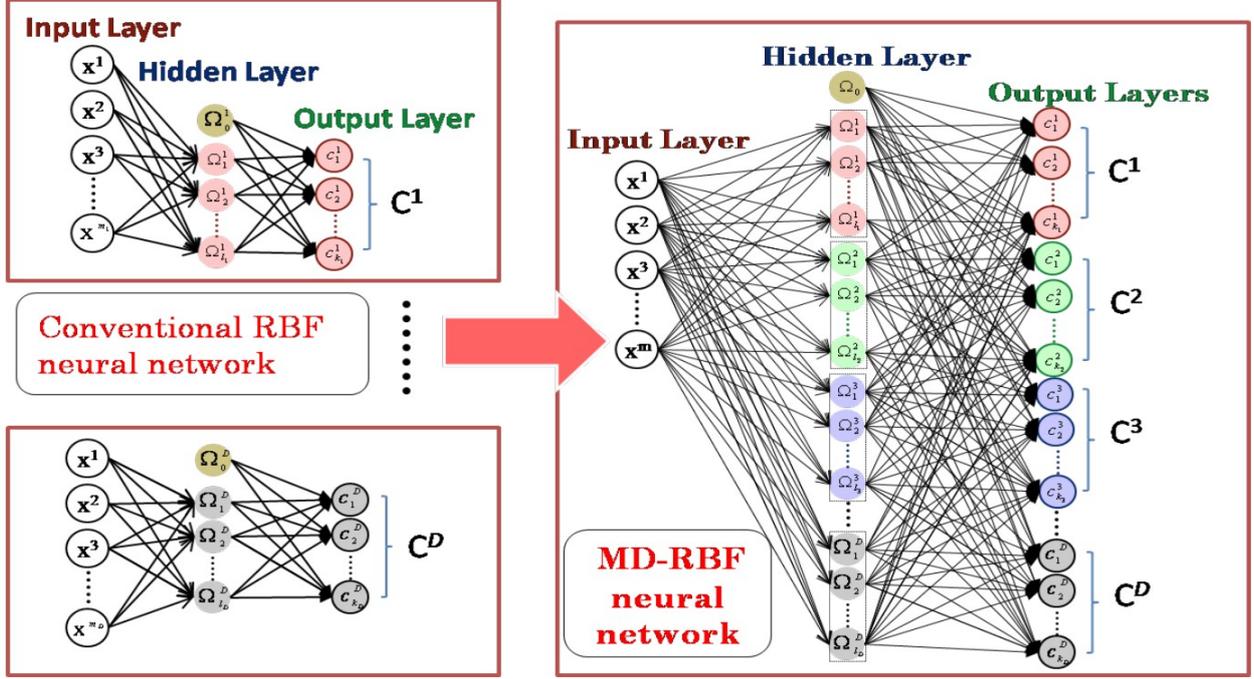


Figure 3.1 The architecture of the conventional RBF neural networks and the proposed MD-RBF neural network.

Determination of the hidden layer nodes from the input layer and its association with output layer requires training. Training of the MD-RBF neural network is carried out with a dataset of known input (EC image features) and output (defect characteristics) pairs.

Let this dataset be $\mathcal{S} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$ obtained from N instances of the EC images i.e. N defects. $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ be a matrix formed from vectors of m dimensional input features and $\mathbf{Y} = \{\mathbf{y}_n\}_{n=1}^N$ be a matrix of output vector formed from D dimensional class variables (defect characteristics) such that $\mathbf{y}_n = \{y_n^d\}_{d=1}^D$ associated with each input feature \mathbf{x}_n . In this thesis, four class variables, i.e. 4 geometric defect characteristics for a subsurface defect such as length, width, depth, and height are considered as schematically shown for a subsurface defect in Figure 3.2. The

characteristics for a surface defect are length, width, and height. As depth tends to 0 in Figure 3.2, the defect becomes a surface defect.

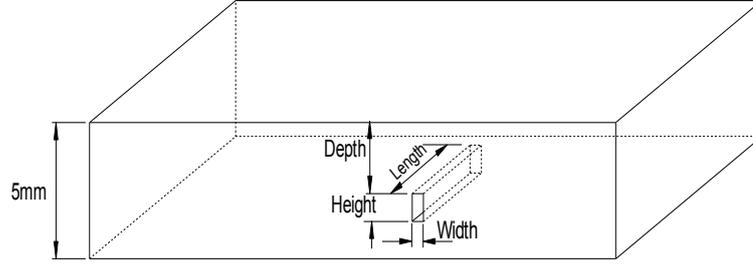


Figure 3.2 Schematic of a subsurface defect with four defect characteristics.

Using this formulation, training of the MD-RBF neural network is carried out in two phases:

- Representation of the prototype vectors
- Weight optimization of the hidden and output layers

The dependency among the output class variables is addressed by the way of representation of prototype vectors and determination of the output class values through weight optimization.

3.2.1 Representation of the prototype vectors

The prototype vectors in the proposed algorithm are formed from the centroids computed by performing supervised cluster analysis on each possible class value of every class variable i.e. on merged class space $\psi = \{c_1^1, c_2^1, \dots, c_{k_1}^1, \dots, c_1^D, c_2^D, \dots, c_{k_D}^D\}$ and $J = |\psi| = \sum_{d=1}^D k_d$. For each class value $c_a^d \in \psi$, clustering using k-means algorithm is carried out on the set of input instances \mathcal{U}_{ad} with class value c_a^d , i.e. $\mathcal{U}_{ad} = \{\mathbf{x}_n | (\mathbf{x}_n, \mathbf{y}_n) \in \mathcal{S}, y_n^d = c_a^d, c_a^d \in \psi\}$. Therefore, v_a^d clustered groups are formed for each class value c_a^d . The number of clusters is set as a fraction of the number of instances, α , in \mathcal{U}_{ad} , such that $v_a^d = \alpha \times \mathcal{U}_{ad}$. The centroids of v_a^d clustered groups form the prototype vectors $\{\mathbf{v}_1^1, \dots, \mathbf{v}_{l_1}^1, \mathbf{v}_1^2, \dots, \mathbf{v}_{l_2}^2, \dots, \mathbf{v}_1^D, \dots, \mathbf{v}_{l_D}^D\}$. Prototype vectors and their corresponding basis functions

$\{\Omega_1^1(\cdot), \dots, \Omega_{l_1}^1(\cdot), \Omega_1^2(\cdot), \dots, \Omega_{l_2}^2(\cdot), \dots, \dots, \Omega_1^D(\cdot), \dots, \Omega_{l_D}^D(\cdot)\}$ form the hidden layer, which are re-indexed from 1 through L thus $\{\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_L\}$ and $\Omega_l(\cdot)$ ($0 \leq l \leq L$). The basis function $\Omega(\cdot)$ for prototype vectors are Gaussian style activation as described below:

$$\Omega_l(x_n) = e^{-\frac{\|\mathbf{x}_n, \mathbf{v}_l\|_2^2}{2\sigma_l^2}} \quad (3.3)$$

where $\|\cdot\|_2$ is the ℓ^2 norm and $\Omega_0(\mathbf{x}_n)$ is kept constant at 1, σ is a factor of average Euclidean distance between the prototype vectors as defined in the following equation:

$$\sigma = \mu \times \left(\frac{\sum_{m=1}^{L-1} \sum_{n=m+1}^L \|\mathbf{v}_m, \mathbf{v}_n\|_2}{L(L-1)} \right) \quad (3.4)$$

where μ is the *spread* parameter, the factor of *scaling* σ . For better generalization and desired performance, it is important to optimize α and μ of the MD-RBF neural network.

3.2.2 Weight optimization

The mapping among the hidden layer and output layer is governed by a weight matrix \mathbf{W} of dimension $(L+1) \times J$. The matrix $\mathbf{W}=[w_{ij}]_{(L+1) \times J}$ is obtained by training which is performed by minimizing the sum of squares error, E , as given in equation (3.5) with required output for each class variable is coded as +1, if the class value is true class else it is set to -1.

$$E = \frac{1}{2} \sum_{n=1}^N \sum_{j=1}^J (f_j(\mathbf{x}_n) - t_n^j)^2 \quad (3.5)$$

where t_n^j is the true class. Differentiating the equation (3.5) with respect to w and setting the derivative to zero gives the following normal equation for linear sum-of-squares that can be used to compute weights:

$$(\mathbf{\Omega}^T \mathbf{\Omega}) \mathbf{W} = \mathbf{\Omega}^T \mathbf{T} \quad (3.6)$$

3.2.3 Output

Upon minimizing the error function E , a real value is obtained for each class of each dimension in merged class space $\mathbf{y}_i = \{C^1 = \{c_1^1, c_2^1, \dots, c_{k_1}^1\}, \dots, C^D = \{c_1^D, c_2^D, \dots, c_{k_D}^D\}\}$, the class which obtains the maximum real value for each class variable $C^d = \max(c_1^d, c_2^d, \dots, c_{k_d}^d)$ is declared as the predicted class. The pseudocode for computing the proposed MD-RBF neural network is given in Algorithm 3.1.

Algorithm 3.1 Multidimensional radial basis function neural network.

$Z = \text{train_MD-RBF}(\mathcal{S}, \alpha, \mu, z)$

Input:

\mathcal{S} : the multidimensional training set $\mathcal{S} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$ with D output dimensions
 α : the clustering parameter
 μ : the spread parameter
 z : the test instance

Output:

Z : predicted multidimensional classes of the test instance

Process:

foreach $d \in D$ **do**

set $\mathcal{U}_{ad} = \{\mathbf{x}_n | (\mathbf{x}_n, \mathbf{y}_n) \in \mathcal{S}, y_n^d = c_a^d, c_a^d \in \psi\}$
 compute l_d centroids $\{\mathbf{v}_1^d, \dots, \mathbf{v}_{l_d}^d\}$

end

form matrix of Ω using equation (3.5)

compute \mathbf{W} using equation (3.6)

foreach $d \in D$ **do**

$Z^d = \max(c_1^d, c_2^d, \dots, c_{k_d}^d)$

end

It can be noted that in the hidden layer itself, a partial classification is carried out by representing cluster centroids associated with all possible class values. Thus, the information about all the class variables is fully utilised for each class value during the optimization of weights. Thus, the MD-RBF neural network produces D predicted classes for each input \mathbf{x}_i , and this takes care of the dependency among the defect characteristics implicitly, from the eddy current images of defects.

3.3 TRAINING AND EVALUATION OF MD-RBF NEURAL NETWORK

The approach followed for training and evaluation of the proposed MD-RBF neural network is shown in Figure 3.3. Machining of defects and generation of EC images of defects in large numbers experimentally for training is time consuming and cumbersome. In order to train the proposed MD-RBF neural network, large amount of dataset covering defects of different sizes is required. It is also difficult to machine subsurface defects with known dimensions, e.g. 2.0 mm height notch located at 2.0 mm below surface in a 5.0 mm thick plate. Therefore, EC images obtained from numerical model are used for training purpose. Experimental EC images of defects of known size, fabricated using electric discharge machining (EDM) are used for testing and evaluation.

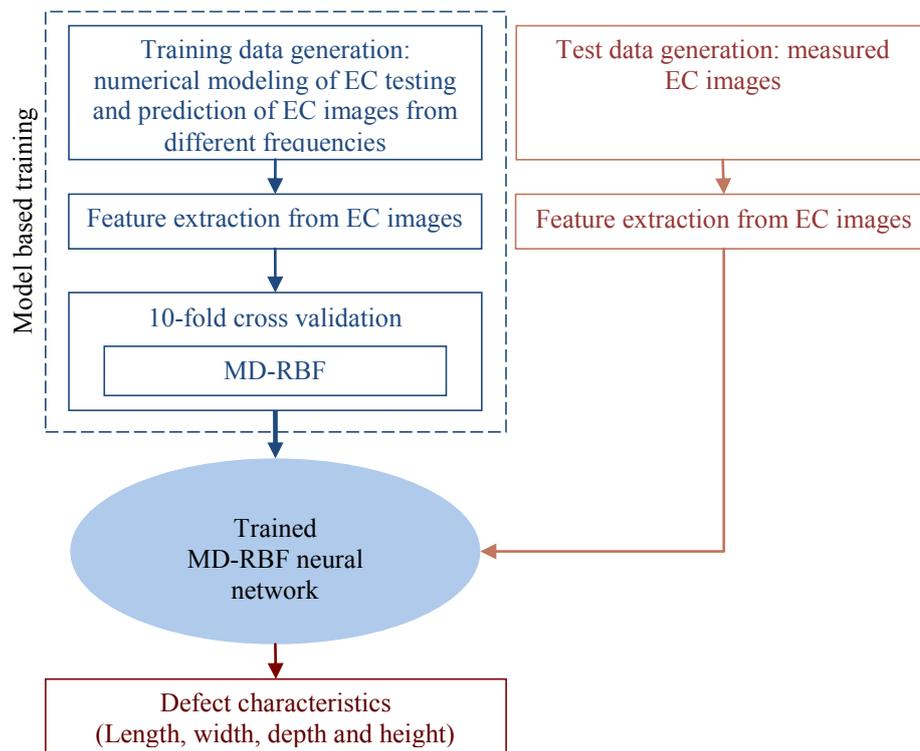


Figure 3.3 Proposed approach for sizing of defects.

As EC images are essentially high dimensional data, direct classification using whole image data is computationally expensive. Therefore, certain characteristic features are obtained for dimensionality reduction and for redundancy elimination. The overall training of the MD-RBF neural network and optimization of its parameter *viz.* α and μ is carried out using 10-fold cross validation process. In this process, the whole dataset is randomly permuted and partitioned into 10 mutually exclusive subsets of equivalent size. At a time, 9 subsets are used as training and the untrained subset is used for testing. The process is continued 10 times till all the subsets are tested. The cross validation can assess true performance and generalization capability of the learning algorithm. To minimize variability in results, 10-fold cross validation has been performed 10 times and the average has been used for the analysis.

3.3.1 Dataset generation

3.3.1.1 Model based generation of EC images

Eddy current images have been generated using CIVA modeling software version 9. CIVA is benchmarked software for numerical simulation of eddy current, ultrasonic, and radiography NDE techniques. The simulation in CIVA is based on semi-analytical methods using dyadic Greens function approach [68]. In this approach, the interaction between defect and electric field generated by the probe is described with an integral equation, which is derived from Maxwell's equations and solved numerically using the method of moments.

$$\mathbf{J}_\Omega(\mathbf{r}) = \mathbf{J}_0(\mathbf{r}) + j\omega\mu_0\sigma f(\mathbf{r}) \int_{\Omega} G_{\omega}^{ee}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{J}_\Omega(\mathbf{r}') d\mathbf{r}' \quad (3.7)$$

The unknown fictitious current density, \mathbf{J}_Ω is defined in the volume Ω containing the defect and depends on the total electric field. The solved current density is used for calculating the probe response or signal for a defect. The term \mathbf{J}_0 in Equation (3.7) is an excitation term that depends on the total primary electric field $\mathbf{E}_0(\mathbf{r})$ emitted by the probe in the region Ω containing the

defect. The dyad \mathbf{G}_{Ω}^{ee} links the fictitious current density to the electric field it creates inside Ω .

The contrast function $f(\mathbf{r})$ in equation (3.7) is defined by

$$f(\mathbf{r}) = \frac{\sigma_0 - \sigma(\mathbf{r})}{\sigma_0} \quad (3.8)$$

where σ_0 is the specimen conductivity and $\sigma(\mathbf{r})$ is the flaw conductivity.

CIVA eddy current module has been extensively validated by others through a series of experiments [68,69]. CIVA facilitates graphical user interface as typically shown in Figure 3.4.

The EC probe used for modelling is a transmit-receive type probe with one coil for AC excitation and the other coil for reception of the EC response. The excitation coil is shielded with cup core ferrite. The cross section of the probe used is given in Figure 3.5. In order to verify the proposed MD-RBF neural network, only one type and configuration of probe has been considered in this study.

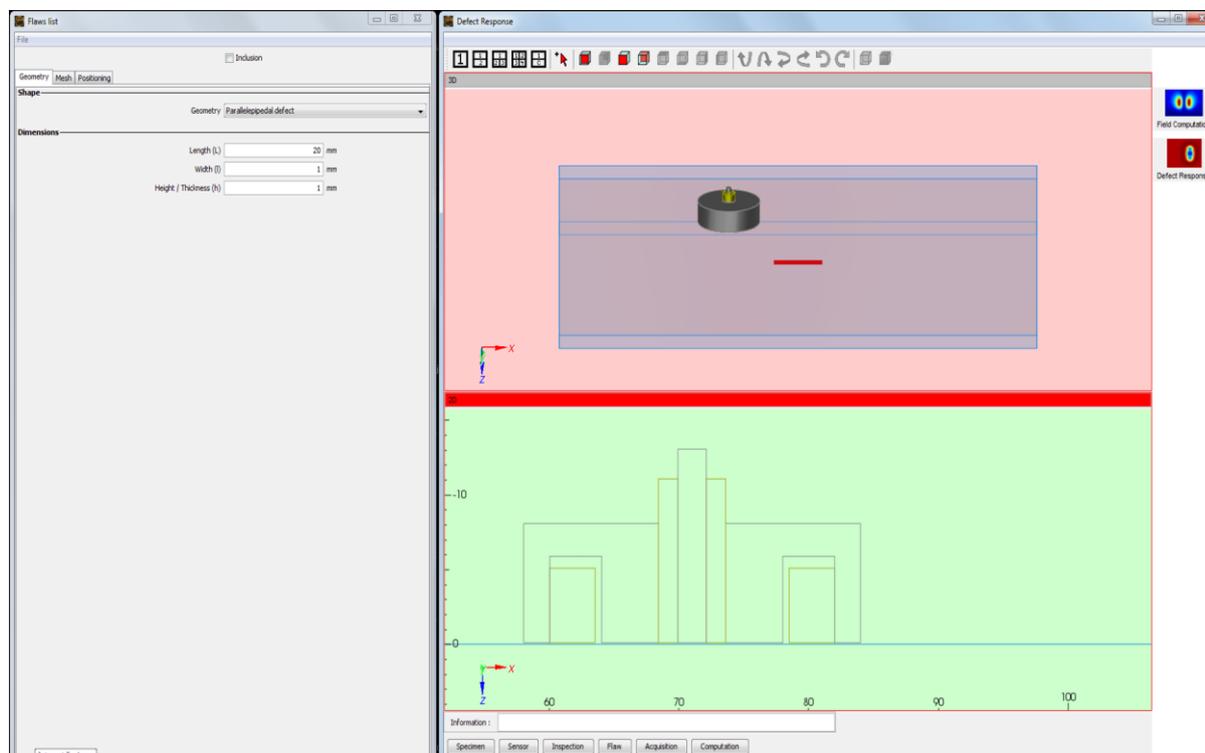


Figure 3.4 Graphical user interface of CIVA software.

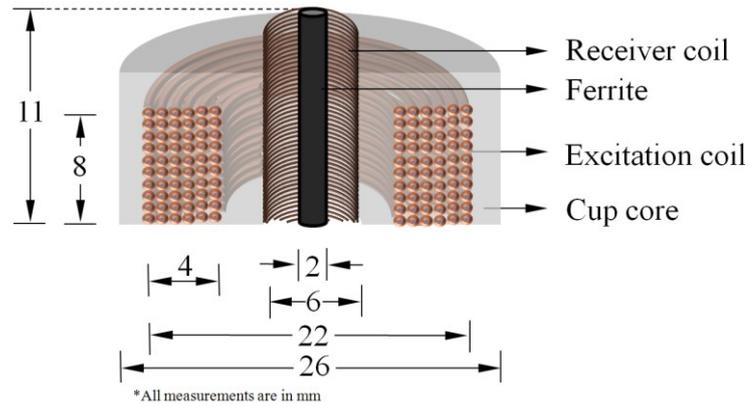


Figure 3.5 Cross sectional view of the EC probe used for modeling.

3.3.1.2 EC imaging experimental setup

To assess the performance of the MD-RBF neural network for sizing realistic defects, experimentally obtained EC images obtained from rectangular machined notches have been used. The block diagram of the experimental setup is shown in Figure 3.6. The EC instrument used for generation of images consists of three major systems as defined below:

- 1) **PROBE EXCITATION SYSTEM:** The probe excitation system consists of a sine wave generator and a power amplifier. The sine wave generator is capable of generating single frequency sine wave in the range of 500 Hz to 80 kHz. The sine wave generator output is fed to the power amplifier which drives the EC probe. A calibrated ammeter is attached to the power amplifier to monitor the current in the EC probe. The power amplifier is capable of driving up to a maximum current of 3A in to the excitation coil.

- 2) **SCANNING SYSTEM:** The scanning system essentially moves the EC probe over the surface of test specimen in a raster scanning pattern. The scanning system consists of two stages viz. X and Y. It also has a probe holder for mounting the transmit-receive EC probe onto it. The scanner has a maximum coverage area of 500 mm × 500 mm. Stepper motors are used to move the X and Y stages. The minimum possible scan pitch for both

X and Y stage is 0.01mm and the reposition accuracy is also 0.01 mm. The National Instrument PCI-7330 stepper motor controller interface is used to connect the scanner to PC and LabView software has been used to move the scanner in a raster pattern. Images of defects are obtained by automated raster scanning of the EC probe over test sample with a *lift-off* of about 0.1 mm from the specimen.

3) MEASUREMENT AND DATA ACQUISITION SYSTEM: This system consists of a 16 bit analog-to-digital converter card, digital signal processing based lock-in amplifier and a PC. The analog-to-digital converter card has 4 channels with 16 bit high dynamic resolution with a sampling frequency of 250 kHz. The Lock-in amplifier is used for phase lock-in measurement. Lock-in amplifier is used for measuring the in-phase and quadrature components of a sinusoid which is buried in noise. It selectively measures the response of the sinusoidal signal with respect to a reference frequency, thereby acting like a narrow band pass filter with a bandwidth of the order of mHz. The measured in-phase and quadrature components using the lock-in amplifier from 2 different frequencies are digitised and stored for further analysis. The movement of the scanner is synchronised with the data acquisition system to acquire data at discrete points during the raster scanning.

Photograph of the EC instrument and the probe used in the studies are shown in Figure 3.7.

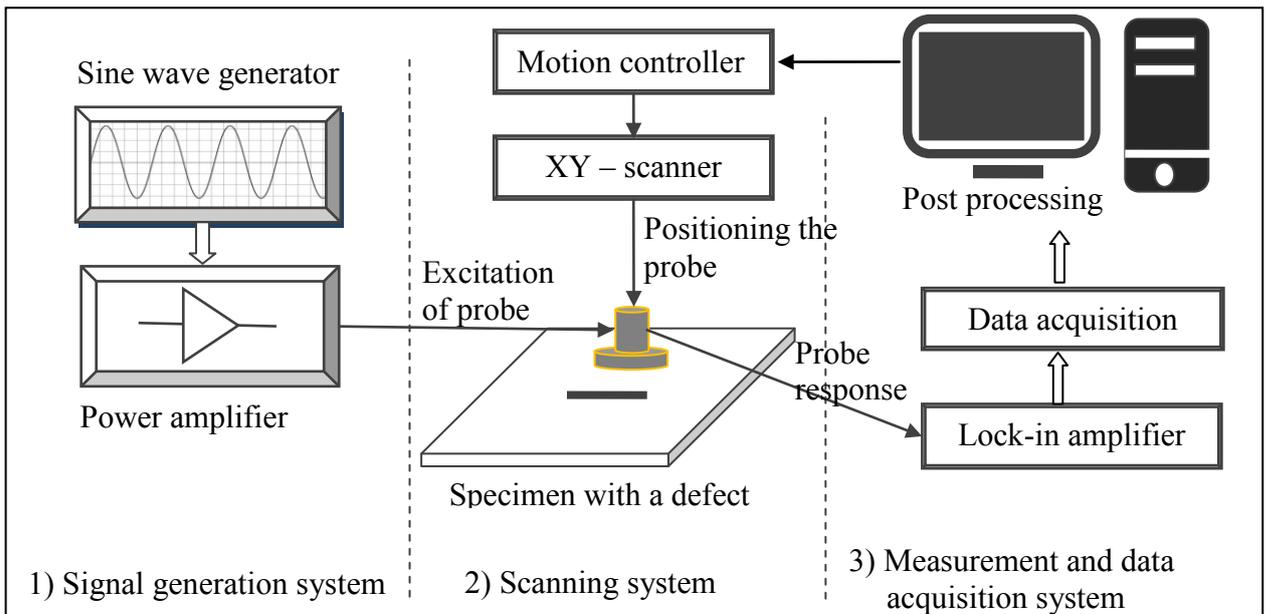
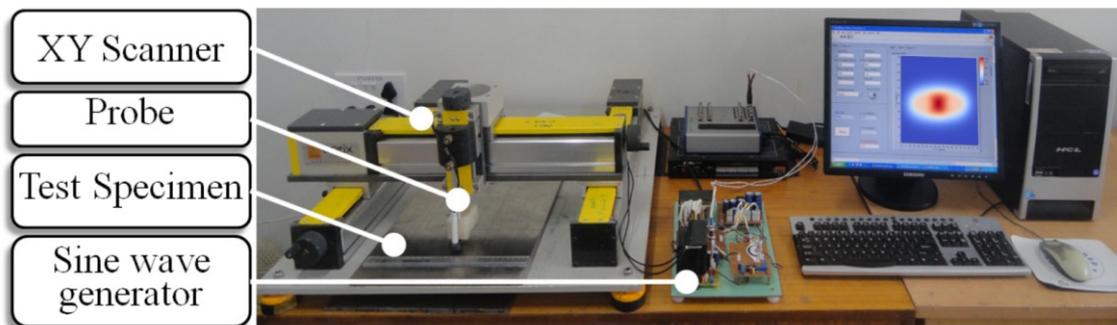


Figure 3.6 Block diagram of the EC measurement system used for generation of EC images.



a) EC experimental setup



b) EC probe

Figure 3.7 Photograph of the a) EC instrument and b) probe used for test data generation.

3.3.1.3 Validation of model

In order to validate, the model used for generation of EC images for training purpose, care has been taken that all the parameters in the model are nearly same as the experimental conditions. For imaging, centre of the EC image i.e. ROI is aligned with the centre of the defect. The scan pitch has been fixed as 1.0 mm along both the directions. The distance covered by the EC probe along the length direction is 60.0 mm while it is 70.0 mm across the defect. The region surrounding a detected feature in all four directions above a threshold of 10 mV is segmented and considered as ROI.

The typical CIVA modeled EC images of two defects at 5 kHz and 10 kHz and experimentally obtained images of these defects are shown in Figure 3.8 and Figure 3.9 respectively. As can be seen, a good agreement of <5% exists between the experimental and modelled EC images. This gives the confidence that the CIVA model correctly represents the experimental condition and hence CIVA model can be used to generate images of defects for training the proposed MD-RBF neural network.

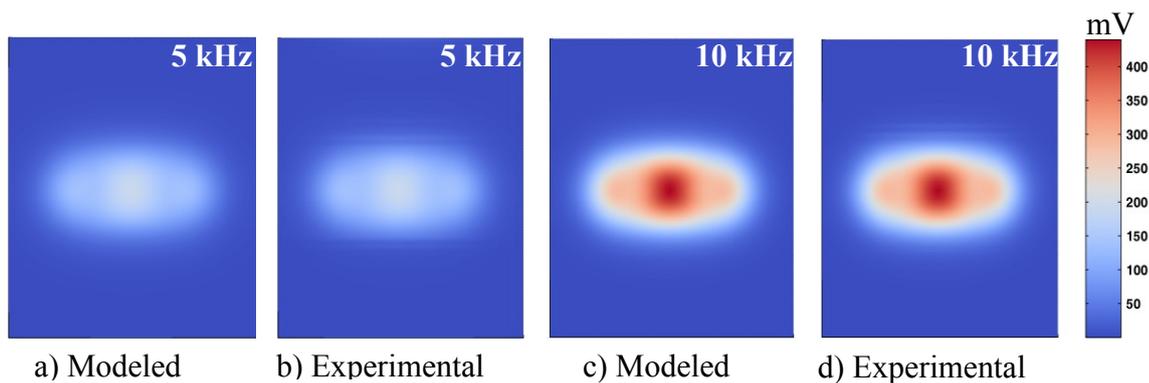


Figure 3.8 CIVA modeled (a and c) and experimental (b and d) EC images of a notch (depth: 2.0 mm, height: 3.0 mm, length: 25.0 mm, width: 2.0 mm) at 5 kHz and 10 kHz.

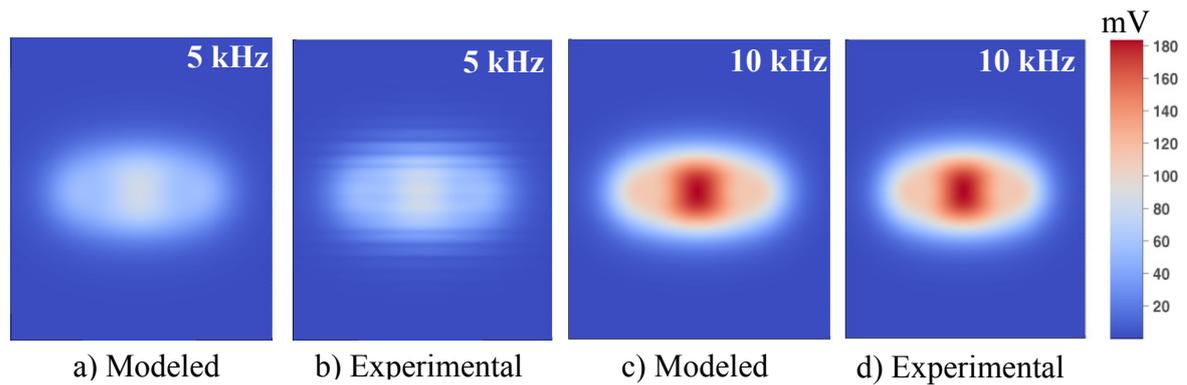


Figure 3.9 CIVA modeled (a and c) and experimental (b and d) EC images of a notch (depth: 3.0 mm, height: 2.0 mm, length: 25.0 mm, width: 2.0 mm) at 5 kHz and 10 kHz.

3.3.1.4 Generation of EC images for inversion

Notches of three different widths (1.0, 2.0, and 3.0 mm) in AISI type 304 austenitic stainless steel plate of 5.0 mm thickness have been modeled. Length of notches has been varied from 20.0 mm to 35.0 mm in steps of 5.0 mm. EC images of notches have been obtained at 5 kHz and 10 kHz. These frequencies ensure full penetration of eddy currents in the thickness direction and obtain redundant information of defects within a depth of 5.0 mm. The disturbing variables such as surface roughness, probe tilt, and variation in lift-off are not considered in the present study, hence, input features from two frequencies are assumed to be adequate. Only one defect is considered as ROI of 70×60 sized images have been obtained at an equal scan pitch of 1.0 mm along both the directions.

A total of 372 defects have been modeled with combinations of different depths (0.0, 1.0, 1.2, 1.3, 1.4, 1.7, 1.8, 2.0, 2.2, 2.6, 2.7, 3.0, 3.2, and 3.7 mm) and heights (0.5, 1.0, 1.4, 1.6, 1.8, 2.0, 2.2, 2.4, 2.8, 3.0, 3.3, 3.7, 3.8, and 4.0 mm). Among the modeled defects, 300 are subsurface defects and 72 are surface defects (depth: 0.0 mm). The dataset also includes defects with length shorter (20.0 and 25.0 mm) as well as longer (30.0 and 35.0 mm) than the probe diameter (26.0 mm).

For evaluation, ten notches as detailed in Table 3.2 have been specified for electric discharge machining on a 5.0 mm thick stainless steel plate. The error in machining for notches is specified with a tolerance of ± 0.05 mm.

Table 3.2 Dimensions of EDM notches specified for fabrications.

DEFECT	LENGTH, mm	WIDTH, mm	DEPTH, mm	HEIGHT, mm
Defect-1	25.0	2.0	2.0	3.0
Defect-2	25.0	2.0	3.0	2.0
Defect-3	25.0	3.0	2.0	3.0
Defect-4	30.0	2.0	3.0	2.0
Defect-5	30.0	2.0	2.0	3.0
Defect-6	30.0	3.0	2.0	3.0
Defect-7	35.0	2.0	3.0	2.0
Defect-8	30.0	2.0	1.0	4.0
Defect-9	35.0	2.0	2.0	3.0
Defect-10	25.0	2.0	0.0	2.0

The defects have been categorized into 5 depth classes, 4 length classes, and 3 classes for height and width as shown in Table 3.3. The defects belong to depth class D1 in Table 3.3 are surface defects while others are subsurface defects.

Table 3.3 Description of classes for defect sizing.

LENGTH, mm	CLASS	WIDTH, mm	CLASS	DEPTH, mm	CLASS	HEIGHT, mm	CLASS
17.5-22.5	L1	0.5-1.5	W1	0.0	D1	<2.0	H1
22.5-27.5	L2	1.5-2.5	W2	0.5-1.5	D2	2.0-3.5	H2
27.5-32.5	L3	2.5-3.5	W3	1.5-2.5	D3	>3.5	H3
32.5-37.5	L4			2.5-3.5	D4		
				>3.5	D5		

3.3.2 Feature extraction

There exists a definite benefit, if features extracted from EC signals as well as from EC images are used for defect sizing. Following this, interesting sets of features have been extracted from images as well as signals. A total of 40 features have been analyzed and the best 17 features that have information related to length, width, depth, and height have been chosen. These 17 features are enlisted in Table 3.4.

Table 3.4 The features extracted from an EC image for a defect.

FEATURE	DESCRIPTION
F1	Distance from peak magnitude to 25% of peak magnitude along length direction at 5 kHz
F2	Distance from peak magnitude to 50% of peak magnitude along length direction at 5 kHz
F3	Distance between extreme peak magnitudes along length direction at 5 kHz
F4	Total area of defect response covering within 25% of peak magnitude in the image at 5 kHz
F5	Total area of defect response covering within 50% of peak magnitude in the image at 5 kHz
F6	Distance between extreme peak magnitudes along width direction at 5 kHz
F7	Maximum magnitude in the EC image at 5 kHz
F8	Phase angle at the maximum magnitude in the EC image at 5 kHz
F9	Ratio of maximum magnitudes of EC image at 5 kHz and 10 kHz
F10	Distance from peak magnitude to 25% of peak magnitude along length direction at 10 kHz
F11	Distance from peak magnitude to 50% of peak magnitude along length direction at 10 kHz
F12	Distance between extreme peak magnitudes along length direction at 10 kHz
F13	Total area of defect response covering within 25% of peak magnitude in the image at 10 kHz
F14	Total area of defect response covering within 50% of peak magnitude in the image at 10 kHz
F15	Distance between extreme peak magnitudes along width direction at 10 kHz
F16	Maximum magnitude in the EC image at 10 kHz
F17	Phase angle at the maximum magnitude in the EC image at 10 kHz

The geometrical distance covered from maximum peak magnitude to degrade 25%, and 50% of peak magnitude has been taken as the features for classification of length of the defects (F1, F2, F10, and F11). Total area of defect response covering within 25%, and 50% of peak magnitude in the image and geometrical distance covered between two extreme peaks have been used as the features for length and width (F3, F4, F5, F6, F12, F13, F14, and F15). The magnitude and phase angles of signals have been extracted as the features for classification of depth and height (F7, F8, F16, and F17). The chosen features are pictorially represented in Figure 3.10.

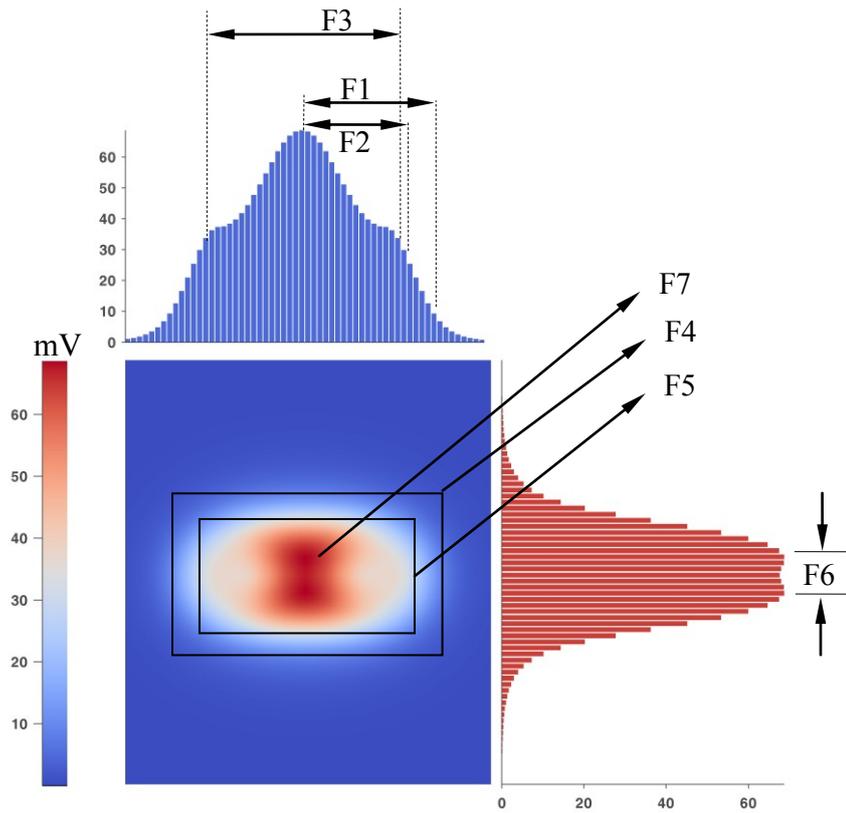


Figure 3.10 Description of the features extracted from an EC image at one excitation frequency.

Additionally, for defect height sizing purpose, a new feature proposed in this thesis, ratio of maximum magnitude at two frequencies (F9) has been used, for the first time. Features F4, F5, F13, and F14 are image features; Feature F9 is the derived feature from two frequencies while all other features are from EC signals. The relationship between these features and the defect characteristics has been studied extensively and discussed in the following subsections. These 17 features extracted from EC images have been used as the input for the proposed MD-RBF neural network.

3.3.2.1 Relationship between input features and defect characteristics

To confirm the effectiveness of the features extracted from the EC images, a systematic comparative study has been carried out. Length of the defect has been compared to the features

F1, F2, and F3. The features F1 and F3 have been plotted as a function of length varying from 5.0 mm to 40.0 mm and shown in Figure 3.11. Three different cases varying width, depth, and height of defects has been considered at 5 kHz frequency and are shown in Figure 3.11.

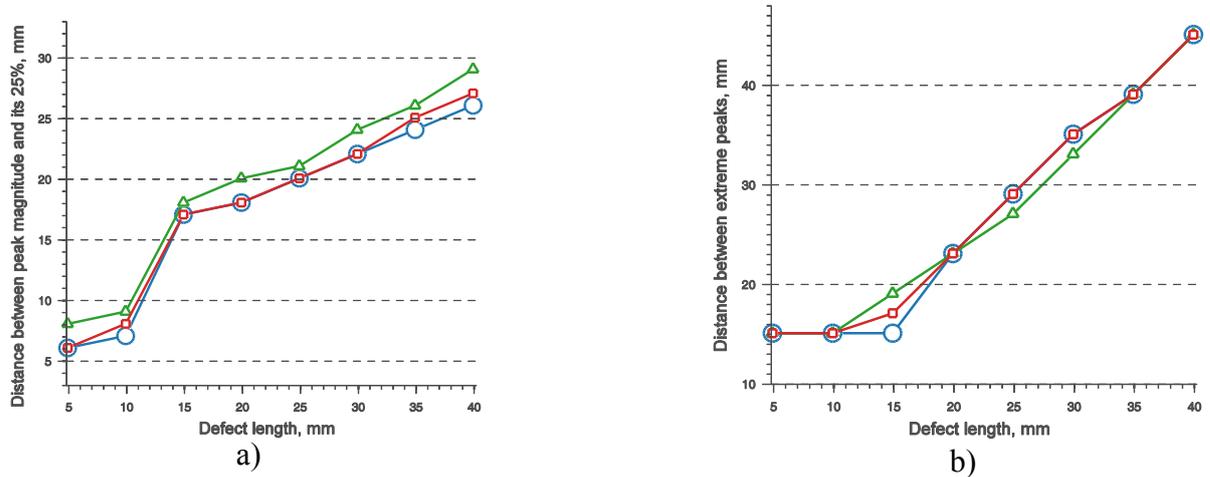


Figure 3.11 Comparison of length of defect versus features a) F1 and b) F2. (○: width = 1.0 mm, depth = 0.0 mm, height = 1.0 mm; □: width = 2.0 mm, depth = 1.0 mm, height = 1.0 mm; △: width = 1.0 mm, depth = 2.0 mm, height = 1.0 mm).

A monotonic increase in feature F1 has been observed for all the lengths considered in each case, while the feature F3 has shown a monotonic increase from length above 15.0 mm. This has confirmed the strong correlation between the chosen features and defect length. Similarly, a very good correlation has been observed between defect width and features F4, F5, and F6. Features F7 and F8 have been studied and the results are plotted as the function of depth varying from 0.0 to 2.0 mm in Figure 3.12. Three different cases have been studied at 5 kHz frequency and the results are shown in Figure 3.12.

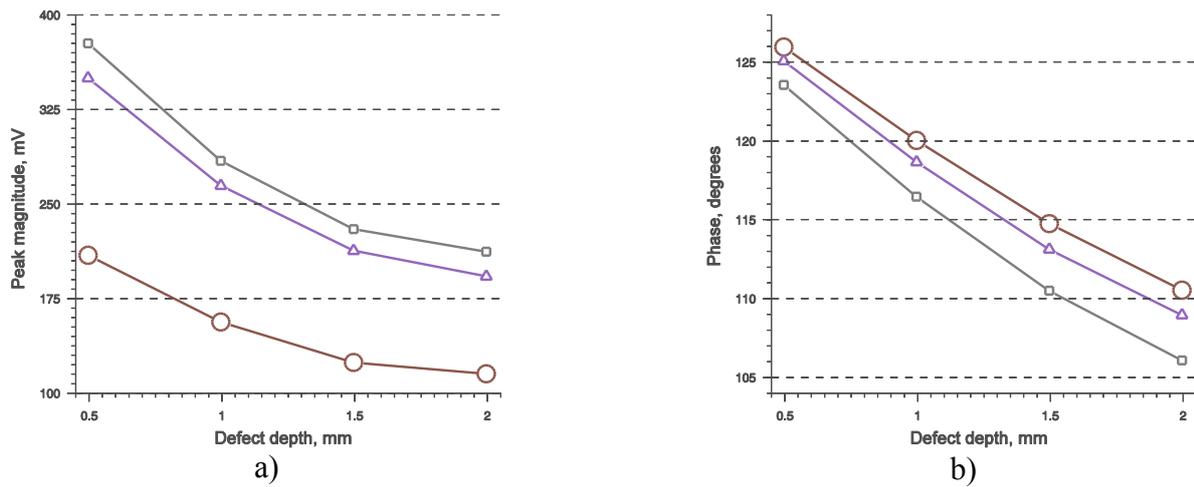


Figure 3.12 Comparison of depth of defect versus features a) F7 and b) F8. (\circ : length = 20.0 mm, width = 1.0 mm, height = 3.0 mm; \square : length = 25.0 mm, width = 2.0 mm, height = 3.0 mm; \triangle : length = 30.0, width = 2.0 mm, height = 3.0 mm).

It can be seen from Figure 3.12 that both the features decrease monotonically with increase in defect depth provided all other defect characteristics are constant. This monotonic increase points out the suitability of the choice of features for sizing defect depth effectively. Features F7 and F16 have been plotted as a function of height in the range of 0.5 to 4 for three different cases in Figure 3.13.

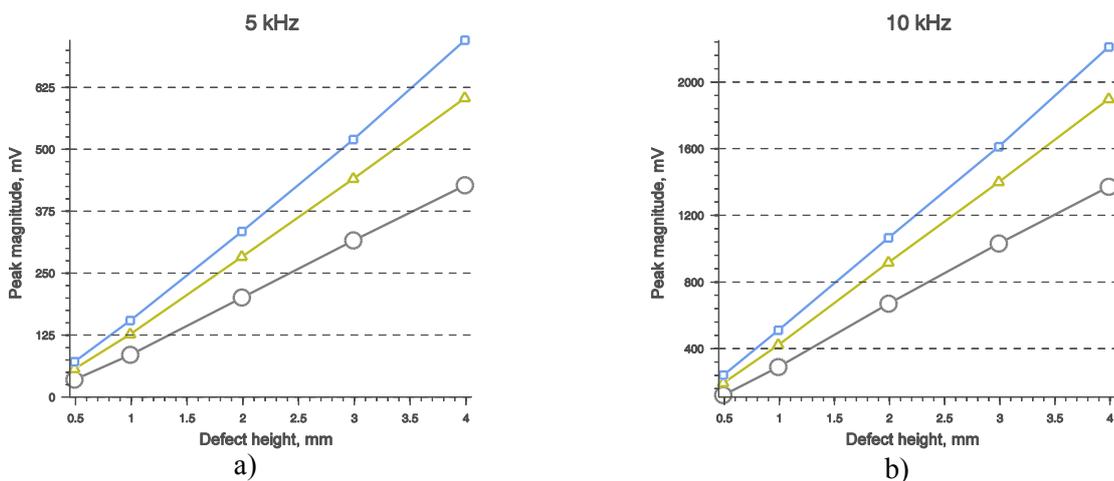


Figure 3.13 Comparison of height of defect versus features a) F7 and b) F16. (\circ : length = 25.0 mm, width = 2.0 mm, depth = 0.0 mm; \square : length = 30.0 mm, width = 2.0 mm, depth = 0.0 mm; \triangle : length = 35.0, width = 3.0 mm, depth = 0.0 mm).

Figure 3.14 displays the results for feature F9 for sizing height of defect. Feature F9 shows constant variability between different cases than that of features F7 and F16, which show a monotonic increase with increase in height. From this study it can be confirmed that the Feature F9 has strong relationship with that of the height of defect.

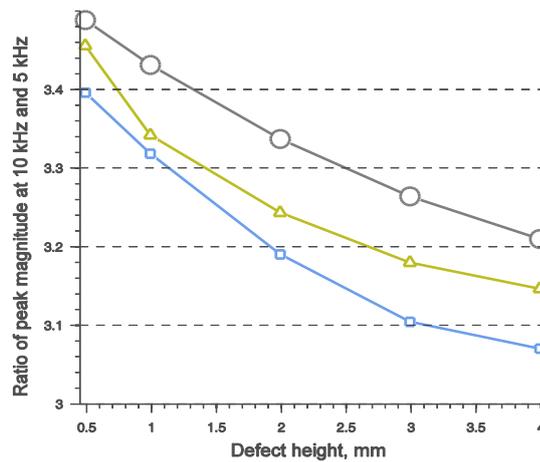


Figure 3.14 Comparison of height of defect versus feature F9. (○: length = 25.0 mm, width = 2.0 mm, depth = 0.0 mm; □: length = 30.0 mm, width = 2.0 mm, depth = 0.0 mm; △: length = 35.0 mm, width = 3.0 mm, depth = 0.0 mm).

The study clearly brings out the existence of relationship between each of the defect characteristics and the features extracted from the EC images. However, there exists variability in the feature values with different cases which pose difficulty for simultaneous sizing. The plots given in Figure 3.11 to Figure 3.14 illustrate the variations in features with respect to individual defect characteristics. These plots cannot be directly used for complete sizing of all defect characteristics. Error in sizing will be large when other characteristics vary. This results in large scatter in sizing. These difficulties demand the use of robust inversion algorithms. Table 3.5 gives a typical set of 10 defects chosen from training data. The extracted features for these defects are given in Table 3.6.

Table 3.5 A set of 10 typical training defects.

DEFECT	LENGTH, mm	WIDTH, mm	DEPTH, mm	HEIGHT, mm
Defect-T1	25.0	3.0	0.0	3.0
Defect-T2	25.0	2.0	1.0	1.4
Defect-T3	30.0	1.0	1.0	3.7
Defect-T4	25.0	1.0	1.8	2.4
Defect-T5	30.0	2.0	2.2	2.8
Defect-T6	20.0	1.0	3.0	2.0
Defect-T7	25.0	1.0	2.7	1.6
Defect-T8	20.0	1.0	3.7	1.0
Defect-T9	20.0	2.0	3.7	1.0
Defect-T10	25.0	1.0	3.7	1.0

Table 3.6 Typical features of the defects in Table 3.5.

DEFECT	F1	F2	F3	F4	F5	F6	F7	F8	F9
Defect-T1	21.00	18.00	26.00	1008.00	648.00	0.00	635.80	-0.76	2.60
Defect-T2	22.00	33.00	26.00	1968.00	1188.00	2.00	119.76	-0.56	2.45
Defect-T3	24.00	20.00	32.00	1056.00	640.00	0.00	267.54	-0.41	2.28
Defect-T4	22.00	18.00	26.00	1056.00	648.00	0.00	101.16	-0.34	2.28
Defect-T5	25.00	21.00	30.00	1300.00	840.00	0.00	180.94	-0.25	2.20
Defect-T6	21.00	18.00	20.00	1092.00	720.00	0.00	49.39	-0.24	2.27
Defect-T7	22.00	18.00	26.00	1144.00	720.00	0.00	42.71	-0.22	2.21
Defect-T8	41.00	33.00	22.00	1804.00	1056.00	6.00	13.76	-0.14	2.22
Defect-T9	39.00	31.00	16.00	1716.00	992.00	8.00	22.78	-0.14	2.20
Defect-T10	45.00	37.00	26.00	2340.00	1480.00	2.00	16.52	-0.10	2.14
DEFECT	F10	F11	F12	F13	F14	F15	F16	F17	
Defect-T1	21.00	18.00	26.00	924.00	648.00	0.00	1650.63	-0.41	
Defect-T2	22.00	31.00	26.00	1008.00	680.00	0.00	292.85	-0.06	
Defect-T3	24.00	20.00	26.00	1012.00	640.00	0.00	587.55	0.18	
Defect-T4	21.00	17.00	26.00	924.00	612.00	0.00	230.74	0.32	
Defect-T5	24.00	20.00	30.00	1152.00	720.00	0.00	397.57	0.45	
Defect-T6	20.00	17.00	20.00	960.00	612.00	0.00	112.26	0.54	
Defect-T7	21.00	17.00	26.00	1008.00	680.00	0.00	94.21	0.54	
Defect-T8	39.00	31.00	20.00	1716.00	1116.00	4.00	30.51	0.68	
Defect-T9	38.00	30.00	18.00	1480.00	812.00	8.00	50.11	0.69	
Defect-T10	43.00	35.00	26.00	1144.00	720.00	0.00	35.40	0.74	

3.3.2.2 Effect of noise on input features

Noise is inevitable in EC imaging due to disturbing variables such as surface roughness, probe tilt, and variation in lift-off between EC probe and test material. To understand the effect of noise on the extracted features for defect sizing, a study has been carried out. Since there are several random variables involved, following the central limit theorem, the convolution of them converges to Gaussian distribution. These do not require the individual random variables to have any particular

distribution, or even that the random variables belong to the same distribution. Three defects viz. Defect-T1, Defect-T4, and Defect T6 (Table 3.5) have been considered to study the effect of noise. Random additive white Gaussian noise (AWGN) up to 10 dB has been added to the EC images of defects and the resulting EC images for Defect-T6 are shown in Figure 3.15. The features values from EC images with AWGN are given in Table 3.7.

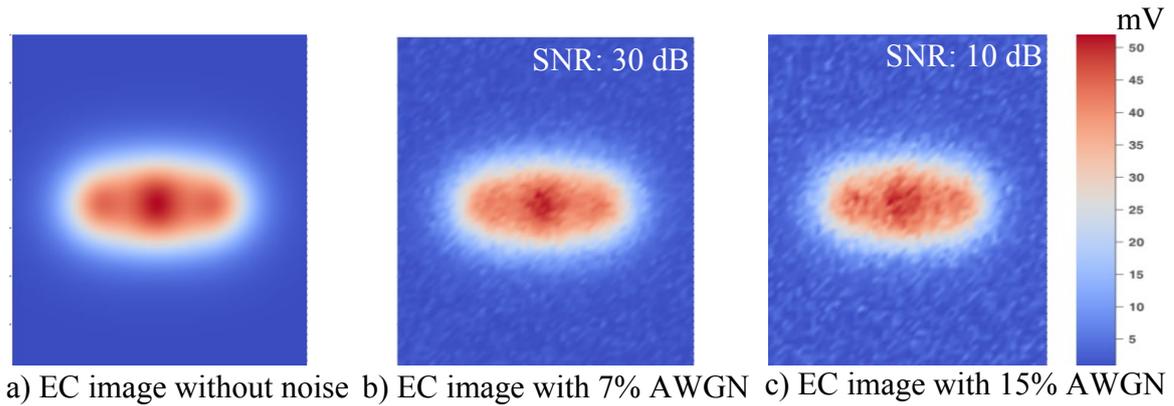


Figure 3.15 EC images with different noise for a Defect-T6.

Table 3.7 Effect of noise on the input features.

FEAT- URE	DEFECT-T1			DEFECT-T4			DEFECT-T6		
	0%	7%	15%	0%	7%	15%	0%	7%	15%
F1	21.00	22.00	22.00	22.00	24.00	23.00	21.00	20.00	20.00
F2	18.00	19.00	19.00	18.00	21.00	18.00	18.00	17.00	17.00
F3	26.00	26.00	25.00	26.00	25.00	25.00	20.00	20.00	20.00
F4	1008.00	1056.00	792.00	1056.00	1056.00	1196.00	1092.00	960.00	880.00
F5	648.00	684.00	608.00	648.00	756.00	792.00	720.00	612.00	544.00
F6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
F7	635.81	655.75	639.77	101.17	104.92	103.05	49.40	50.98	55.41
F8	-0.77	-0.75	-0.78	-0.35	-0.31	-0.32	-0.24	-0.23	-0.23
F9	2.60	2.59	2.81	2.28	2.30	2.37	2.27	2.25	2.07
F10	21.00	21.00	20.00	21.00	21.00	22.00	20.00	19.00	21.00
F11	18.00	17.00	17.00	17.00	17.00	17.00	17.00	16.00	17.00
F12	26.00	26.00	25.00	26.00	26.00	25.00	20.00	20.00	20.00
F13	924.00	880.00	880.00	924.00	924.00	836.00	960.00	1092.00	880.00
F14	648.00	612.00	612.00	612.00	360.00	448.00	612.00	720.00	680.00
F15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
F16	112.27	112.16	114.45	197.34	198.04	204.62	321.78	319.93	339.06
F17	0.54	0.53	0.52	0.44	0.44	0.45	-0.49	-0.49	-0.45

It can be observed from Table 3.7 that the features extracted from EC images with noise are nearly same as that of the features from noise-free EC images. Small variations in the features are expected to be tolerated by the Gaussian basis function present in the hidden layer of the proposed MD-RBF neural network with an optimized σ . Thus, the chosen 17 features are considered to be noise tolerant.

3.3.2.3 Effect of defect orientation on input features

In realistic EC test situations, orientation of a defect with respect to the measurement surface is non deterministic. It is essential to identify the angle of orientation of the defect for feature selection and image rotation. By assuming the maximum magnitude of an EC image as the reference origin, the energy of a line (signal) in all directions can be computed. The angle with maximum energy can be taken as the angle of orientation of the defect and the image can be rotated to make the angle of orientation parallel to the standard X-axis.

To study the effect of orientation on the features, three defects (Defect-T1, Defect-T4, and Defect T6) have been considered and their dimensions are given in Table 3.5 and the images for different orientations viz, 30°, 45°, and 60° for Defect-T6 are shown in Figure 3.16.

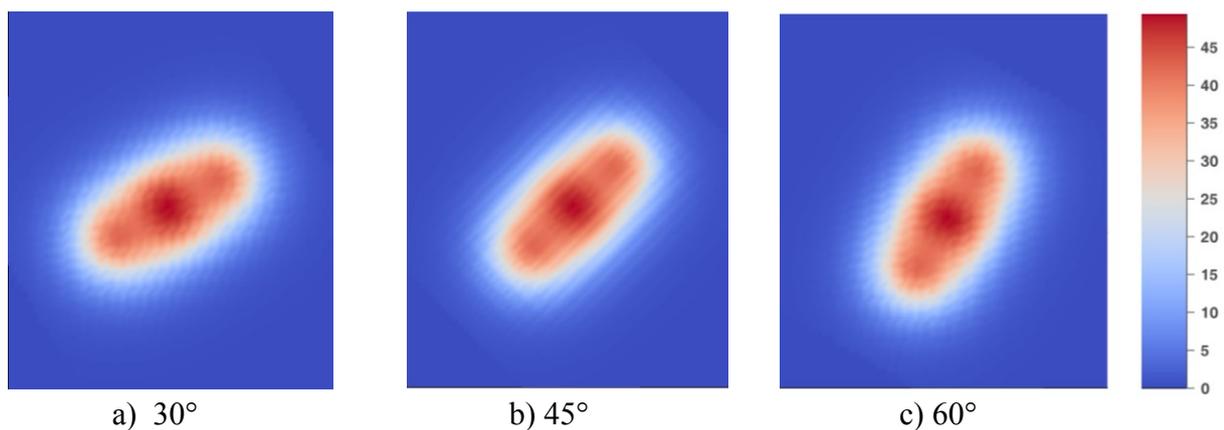


Figure 3.16 EC images of a defect (length: 20.0 mm, width: 1.0 mm, depth: 3.0 mm, and height: 2.0 mm) oriented with respect to surface of the testing specimen.

Features have been extracted from the EC images are given in Table 3.8. As can be observed, the defect orientations have no influence on the extracted features. Thus, the extracted features are considered rotation invariant.

Table 3.8 Effect of orientation of the extracted features.

FEAT- URE	DEFECT-T1			DEFECT-T4			DEFECT-T6		
	30°	45°	60°	30°	45°	60°	30°	45°	60°
F1	21.00	21.00	21.00	22.00	22.00	22.00	21.00	21.00	21.00
F2	18.00	18.00	18.00	18.00	18.00	18.00	18.00	18.00	18.00
F3	26.00	26.00	26.00	26.00	26.00	26.00	20.00	20.00	20.00
F4	1008.00	1008.00	1008.00	1056.00	1056.00	1056.00	1092.00	1092.00	1092.00
F5	648.00	648.00	648.00	648.00	648.00	648.00	720.00	720.00	720.00
F6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
F7	635.81	635.81	635.81	101.17	101.17	101.17	49.40	49.40	49.40
F8	-0.77	-0.77	-0.77	-0.35	-0.35	-0.35	-0.24	-0.24	-0.24
F9	2.60	2.60	2.60	2.28	2.28	2.28	2.27	2.27	2.27
F10	21.00	21.00	21.00	21.00	21.00	21.00	20.00	20.00	20.00
F11	18.00	18.00	18.00	17.00	17.00	17.00	17.00	17.00	17.00
F12	26.00	26.00	26.00	26.00	26.00	26.00	20.00	20.00	20.00
F13	924.00	924.00	924.00	924.00	924.00	924.00	960.00	960.00	960.00
F14	648.00	648.00	648.00	612.00	612.00	612.00	612.00	612.00	612.00
F15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
F16	1650.63	1650.63	1650.63	230.74	230.74	230.74	112.27	112.27	112.27
F17	-0.41	-0.41	-0.41	0.32	0.32	0.32	0.54	0.54	0.54

3.3.3 Performance evaluation metrics

To evaluate the performance of the proposed MD-RBF neural network, normalised accuracy in the range of 0 to 1 has been considered as the metric for individual defect characteristics. Apart from evaluation for individual class variable, it is also possible to evaluate the overall performance of the learning algorithm for all the class variables together. Performance evaluation metrics for multidimensional learning are different from the metrics that are commonly used for single output classification, namely accuracy, precision, recall, F1-measure, etc. Multidimensional evaluation metrics are mean accuracy, global accuracy, and entropy of accuracy. These metrics are discussed in the following subsections.

Given a test set with N instances and D dimensional of input feature $\mathcal{S} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$ and a predicted set of class \mathbf{y}'_n , then the evaluation metrics are the following:

3.3.3.1 Mean accuracy

Mean accuracy is the average of accuracy predicted for each class variable. It can be stated as follows:

$$\text{Mean accuracy} = \frac{1}{N} \sum_{n=1}^N \frac{1}{D} \sum_{d=1}^D \delta_1(y'_{nd}, y_{nd}) \quad (3.9)$$

where $\delta_1(y'_{nd}, y_{nd}) = \begin{cases} 1, & \text{if } y'_{nd} = y_{nd} \\ 0, & \text{otherwise} \end{cases}$

3.3.3.2 Global accuracy

Global accuracy considers a set of classes associated with each instance as a single entity and it credits only if all classes, for an instance, are predicted accurately. Global accuracy is defined as follows:

$$\text{Global accuracy} = \frac{1}{n} \sum_{n=1}^N \delta_2(\mathbf{y}'_n, \mathbf{y}_n) \quad (3.10)$$

where $\delta_2(\mathbf{y}'_n, \mathbf{y}_n) = \begin{cases} 1, & \text{if the entire set } \mathbf{y}'_n = \mathbf{y}_n \\ 0, & \text{otherwise} \end{cases}$

3.3.3.3 Entropy of accuracy

The mean accuracy and global accuracy do not consider the bias of accuracy on few class variables. In certain situations, most of the predicted classes are accurate only on some particular subset of class variables. This situation can be handled effectively by entropy of accuracy which is defined as follows:

$$\rho_d = \sum_{n=1}^N \delta_1(y'_{nd}, y_{nd}) \quad (3.11)$$

$$\rho = \sum_{d=1}^D \rho_d \quad (3.12)$$

Entropy of accuracy

$$= \frac{\left(\sum_{d=1}^D \left(-\frac{\rho_d}{\rho} \log_2 \frac{\rho_d}{\rho} \right) \right) \times \left(\frac{1}{n} \sum_{n=1}^N \frac{1}{d} \sum_{d=1}^D \delta_1(y'_{nd}, y_{nd}) \right)}{\log_2 D} \quad (3.13)$$

The entropy of accuracy is a real valued factor in the range from 0 to 1 of mean accuracy and it is related to the extent of spread of predicted accuracy over the class variables. Entropy of accuracy is a newly proposed metric for multidimensional evaluation in this thesis. The higher the value of mean accuracy, global accuracy, and entropy of accuracy, the better is the performance of the MD-RBF neural network.

3.3.3.4 Comparison of multidimensional evaluation metrics

Each evaluation metric described above has unique characteristics. To understand them an example with 4 different cases are illustrated in Figure 3.17. Each case represents 10 instances of predictions of length, width, and height of a surface defect. The (✓) mark in Figure 3.17, indicates the correct predictions. Mean accuracy considers average of overall performance. It can be noted that for all the cases the mean accuracy is 0.3. This is because, on average, there are 9 correct predictions in all the cases. Global accuracy is a strict measure which credits correct prediction, only if all the outputs are predicted accurately. This is reflected in Figure 3.17, as expected for Case-1, all other cases have global accuracy as 0. Mean accuracy and global accuracy do not consider the bias of accuracy on any particular output. As a result, the mean

accuracy is higher and the global accuracy is 0, as depicted in Figure 3.17, for Case-2, Case-3, and Case-4. This situation is ably handled by the entropy of accuracy, which measures the spread of accuracy among the outputs. Case-2 has 3 correct predictions in each output, hence, the entropy of accuracy is equal to the mean accuracy. In Case-4, only height has given correct prediction, hence, the entropy of accuracy is equal to global accuracy. By using entropy of accuracy, it is possible to access the performance in comprehensive manner because it takes spread across the dimensions in to account.

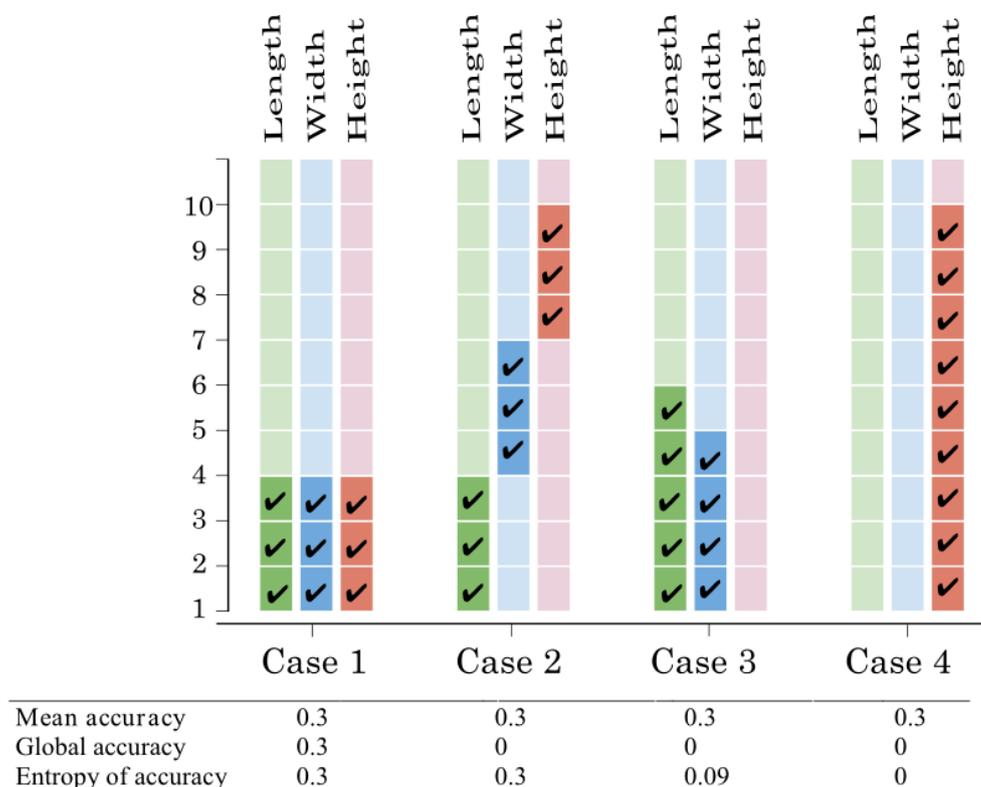


Figure 3.17 An illustration to compare the three metrics for performance evaluation of the proposed MD-RBF neural network.

3.3.4 Cross validation

The overall training of the MD-RBF neural network and optimization of its parameters viz. α and μ have been carried out using 10-fold cross validation process. In this process, the whole dataset is randomly permuted and partitioned into 10 mutually exclusive subsets of equivalent

size. At a time, 9 subsets are used as training and the untrained subset is used for testing. The process continues 10 times till all the subsets are tested. This 10-fold cross validation is able to assess the true performance and generalization capability of the MD-RBF neural network. To minimize the variability in the results, 10-fold cross validation has been performed 10 times and average data has been used for the analysis. Extensive analysis has been carried out by 10-fold cross validation to optimize the MD RBF neural network and the conventional RBF neural networks by varying the parameters α and μ . Mean accuracy has been chosen as the multidimensional evaluation metric for optimization purpose. It is important to note that four different trainings are required for conventional RBF neural network each of the defect characteristic. On the contrary, the proposed MD-RBF neural network requires only one training to estimate all the four defect characteristics.

3.3.5 Parameter optimization

The parameters of the conventional RBF neural networks and the MD-RBF neural network viz. α and μ have been optimized to produce maximum performance. The results of the optimization based on the performance of the RBF neural networks and the MD-RBF neural network are given in Figure 3.18. As can be observed, for constant μ , the accuracy initially improves significantly with increase in α and for further increase in α , the performance tends to remain nearly constant. On the contrary, when α is fixed, the curves get closer as μ increases. The important aspect to note from Figure 3.18 is the optimal range of clustering factor, α , for the RBF neural networks and the MD-RBF neural network. The optimal range of α is between 0.3 and 0.5 for the RBF neural networks while it is between 0.015 and 0.02 for the MD-RBF neural network. Based on these observations, μ has been selected as 1 for both the algorithms and α has been selected as 0.37 for the RBF neural networks and 0.0188 for the MD-RBF neural network. At this clustering factor, the number of clusters required for optimal sizing using MD-RBF neural network for each of the four characteristic of defects is 7. Hence, a total of about 28

clusters have to be identified for the hidden layer of the MD-RBF neural network. In contrast, the number of clusters required for one RBF for sizing of one characteristic of defects is 138. Hence, a total of 552 different clusters have to be identified for the hidden layer of the 4 different RBF neural networks for optimal performance. This exemplifies the superior capability of the MD-RBF neural network. The optimal performance of the MD-RBF neural network with a significantly less number of clusters relative to the conventional RBF neural networks can be attributed to the systematic identification of the clusters in the hidden layer.

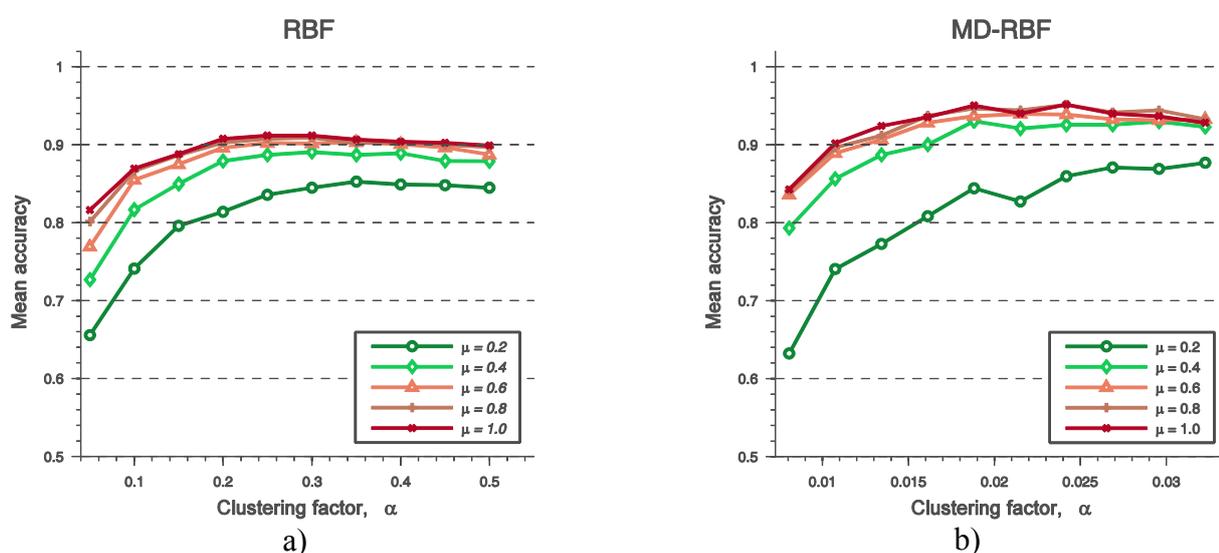


Figure 3.18 10-fold cross validation performance for classification of a) RBF neural networks and b) MD-RBF neural network.

3.4 PERFORMANCE EVALUATION OF MD-RBF NEURAL NETWORK

3.4.1 Performance evaluation of MD-RBF neural network on modeled images

The individual performance using accuracy as metric for 10-fold cross validation with optimal parameter settings for each defect characteristic is given in Table 3.9. It can be noted from Table 3.9 that even though MD-RBF neural network required comparatively less number of clusters than the conventional RBF neural networks, the performance of the MD-RBF neural network for

sizing all the 4 defect characteristics clearly shows promise for sizing. The performance for sizing the height of defects has substantially increased to 0.9435 from 0.9005. This enhancement can be attributed to addressing the dependency implicitly among the defect characteristics by the proposed MD-RBF neural network.

Table 3.9 Comparison of performance of the conventional RBF neural networks and MD-RBF neural network for 10-fold cross validation.

CLASSIFICATION ALGORITHMS	ACCURACY			
	LENGTH	WIDTH	DEPTH	HEIGHT
RBF neural networks	0.9634	0.9118	0.9462	0.9005
MD-RBF neural network	0.9710	0.9277	0.9637	0.9435

Table 3.10 tabulates the overall performance (in terms of mean \pm standard deviation) of the RBF neural networks and MD-RBF neural network for ten independent runs of 10-fold cross validation with the optimal parameter settings. As can be observed, the mean accuracy has increased to 0.9515 from 0.9305. The entropy of accuracy for the conventional RBF neural networks and MD-RBF neural network is similar to their corresponding mean accuracies. This indicates that the accuracy is not biased to a few defect characteristics. The MD-RBF neural network considers all the defect characteristics simultaneously, while RBF neural networks consider each defect characteristic as an independent entity.

Table 3.10 Comparison of performance of the RBF neural networks and MD-RBF neural network by 10-fold cross validation on training data using evaluation metrics.

CLASSIFICATION ALGORITHM	MEAN ACCURACY	GLOBAL ACCURACY	ENTROPY OF ACCURACY
RBF neural networks	0.9305 \pm 0.0053	0.7634 \pm 0.0221	0.9302 \pm 0.0053
MD-RBF neural network	0.9515 \pm 0.0027	0.8306 \pm 0.0100	0.9514 \pm 0.0027

Using the MD-RBF neural network, 83% of defects have been successfully classified all their defect characteristics accurately, as reflected by the global accuracy. This is against 76% that achieved by the conventional RBF neural networks. Similar performance has been observed by changing the number of classes for each defect characteristics. This is a significant new result.

To understand the factor of dependency addressed by the MD-RBF neural network, an analysis has been carried out by implementing one separate MD-RBF neural network for each of the defect characteristics. The result of this analysis is given in Table 3.11. {D, W} in Table 3.11 denotes implementation of a single MD-RBF neural network for sizing depth and width together and separate neural network for sizing height and length. {D, W, H, L} denotes a single MD-RBF neural network implemented for sizing depth, width, height, and length together. It can be observed from Table 3.11 that the mean accuracy has gradually increased to 0.9515 from 0.9360 for MD-RBF neural network that classified all 4 defect characteristics together. Upon comparison of the results, it has been established that the independent classification of each different defect characteristic may result in poor accuracy. This is directly attributed to the dependency effectively handled by the proposed MD-RBF neural network. To understand the dependency among the defect characteristics, extensive study has been carried out and discussed in Chapter 5.

Table 3.11 Comparison of performance of the MD-RBF neural network following the dependency sequence.

MD-RBF NEURAL NETWORK	ACCURACY				MEAN ACCURACY
	DEPTH	WIDTH	HEIGHT	LENGTH	
{D},{W},{H},{L}	0.9489	0.9140	0.9242	0.9570	0.9360
{D, W},{H},{L}	0.9513	0.9126	0.9242	0.9570	0.9362
{D,W,H},{L}	0.9637	0.9245	0.9435	0.9570	0.9471
{D, W, H, L}	0.9637	0.9277	0.9435	0.9710	0.9515

For illustration of the predictions made by the MD-RBF neural network, a set of 10 defects (Defect-T1 through Defect-T10) from training set given in Table 3.5 were considered and the classification results during cross validation are given in Table 3.12. The predicted value for depth and height are assumed to be the midpoint of the class range for each. It can be noted from Table 3.12 that the predictions of depth and height by the MD-RBF neural network closely agree with that of the actual dimensions, measured using optical method with an accuracy of 0.05 mm. The length and width are accurately predicted for all the defects.

Table 3.12 Performance of the MD-RBF neural network on a set of defects from training data.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-T1	25.0	25.0	3.0	3.0	0.0	0.0	3.0	2.8
Defect-T2	25.0	25.0	2.0	2.0	1.0	0.8	1.4	1.0
Defect-T3	30.0	30.0	1.0	1.0	1.0	0.8	3.7	4.3
Defect-T4	25.0	25.0	1.0	1.0	1.8	2.0	2.4	2.8
Defect-T5	30.0	30.0	2.0	2.0	2.2	2.0	2.8	2.8
Defect-T6	20.0	20.0	1.0	1.0	3.0	3.0	2.0	1.0
Defect-T7	25.0	25.0	1.0	1.0	2.7	3.0	1.6	1.0
Defect-T8	20.0	20.0	1.0	1.0	3.7	4.3	1.0	1.0
Defect-T9	20.0	20.0	2.0	2.0	3.7	4.3	1.0	1.0
Defect-T10	25.0	25.0	1.0	1.0	3.7	4.3	1.0	1.0

3.4.2 Performance evaluation of the MD-RBF neural network on experimental images of rectangular defects

Performance of the trained MD-RBF neural network with simulated EC images has been tested on experimental images of EDM notches, and the results are given in Table 3.13. As can be observed, the results of MD-RBF neural networks closely agree with the actual values for all the 4 defect characteristics.

Table 3.13 Performance of MD-RBF neural networks on experimental images of EDM notches.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-1	25.0	25.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-2	25.0	25.0	2.0	2.0	3.0	3.0	2.0	2.8
Defect-3	25.0	25.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-4	30.0	30.0	2.0	2.0	3.0	3.0	2.0	2.8
Defect-5	30.0	30.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-6	30.0	30.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-7	35.0	35.0	2.0	2.0	3.0	3.0	2.0	1.0
Defect-8	30.0	30.0	2.0	2.0	1.0	0.8	4.0	4.3
Defect-9	35.0	35.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-10	25.0	25.0	2.0	2.0	0.0	0.0	2.0	1.0

The results of classification during cross validation for individual RBF neural networks for defects given in Table 3.5 are tabulated in Table 3.14. The individual RBF neural networks have predicted the length of defects accurately, however, misclassified the width of Defect-8. It can

also be observed that height prediction by the MD-RBF neural network is closer to the actual height measured by using the optical method than the predictions by the individual RBF neural networks.

Table 3.14 Performance of the individual RBF neural networks on a set of defects from training data.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-T1	25.0	25.0	3.0	3.0	0.0	0.0	3.0	2.8
Defect-T2	25.0	25.0	2.0	2.0	1.0	0.8	1.4	1.0
Defect-T3	30.0	30.0	1.0	1.0	1.0	0.8	3.7	4.3
Defect-T4	25.0	25.0	1.0	1.0	1.8	2.0	2.4	2.8
Defect-T5	30.0	30.0	2.0	2.0	2.2	2.0	2.8	2.8
Defect-T6	20.0	25.0	1.0	1.0	3.0	3.0	2.0	1.0
Defect-T7	25.0	25.0	1.0	1.0	2.7	3.0	1.6	2.8
Defect-T8	20.0	20.0	1.0	2.0	3.7	4.3	1.0	2.8
Defect-T9	20.0	20.0	2.0	2.0	3.7	4.3	1.0	2.8
Defect-T10	25.0	25.0	1.0	1.0	3.7	4.3	1.0	1.0

The results of validation using experimental EC images by individual RBF neural networks are given in Table 3.15. As can be observed, the individual RBF neural networks misclassified the width of Defect-7. The height of Defect-2 and Defect-4 are not as close to the actual value. These are deeper defects considered in this study with depth ≥ 3.0 mm.

Table 3.15 Performance of individual RBF neural networks on experimental images of EDM defects.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-1	25.0	25.0	2.0	2.0	2.0	2.0	3.0	1.0
Defect-2	25.0	25.0	2.0	2.0	3.0	3.0	2.0	2.8
Defect-3	25.0	25.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-4	30.0	30.0	2.0	3.0	3.0	3.0	2.0	1.0
Defect-5	30.0	30.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-6	30.0	30.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-7	35.0	35.0	2.0	1.0	3.0	3.0	2.0	2.8
Defect-8	30.0	30.0	2.0	2.0	1.0	0.8	4.0	4.3
Defect-9	35.0	35.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-10	25.0	25.0	2.0	2.0	0.0	0.0	2.0	1.0

From comparison of the performance of MD-RBF neural network with the RBF neural networks, it can be clearly stated that simultaneous prediction of all the four defect characteristics by proposed MD-RBF neural network is more accurate than that of the individual RBF neural networks, essentially due to the effective incorporation of the dependency.

3.4.3 Performance evaluation of MD-RBF neural network on modeled images of other defects

To verify the performance of the MD-RBF neural network for non-rectangular shaped defects and for defects smaller than the probe diameter, studies have been carried out and reported in following subsections.

3.4.3.1 Performance evaluation of MD-RBF neural network for non-rectangular shaped defects

For non-rectangular shaped defects, the bottom curvature of the rectangular defects has been modified such that it resembles ellipsoidal shape. Two ellipsoidal defects (Defect-E1 and Defect-E2) having identical length, width, depth, and height have been modeled using CIVA. The dimensions of the ellipsoidal defects are described by four parameters, viz. semi-major axis (R1), semi-minor axis (R2), height, and width. The sectional view of a typical ellipsoidal defect is shown in Figure 3.19.

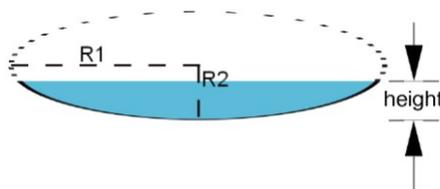


Figure 3.19 Schematic of the ellipsoidal defect modeled in CIVA.

The dimensions of the ellipsoidal defects are shown in Figure 3.20a and Figure 3.20b. The difference between defects E1 and E2 are their radii R1 and R2. The radii of defect E1 are R1-10.0 mm and R2-1.0 mm, while the radii of E2 are R1-20.0 mm, R2-7.5 mm. The semi-major axis, semi-minor axis, and the heights of Defect-E1 and Defect-E2 are chosen such that the length and width of the defects are 20.0 mm and 1.0 mm, respectively. Figure 3.20c shows the rectangular defects with the same dimensions (i.e. length: 20.0 mm, width: 1.0 mm, depth: 1.0 mm, height: 1.0 mm), for comparison purpose.

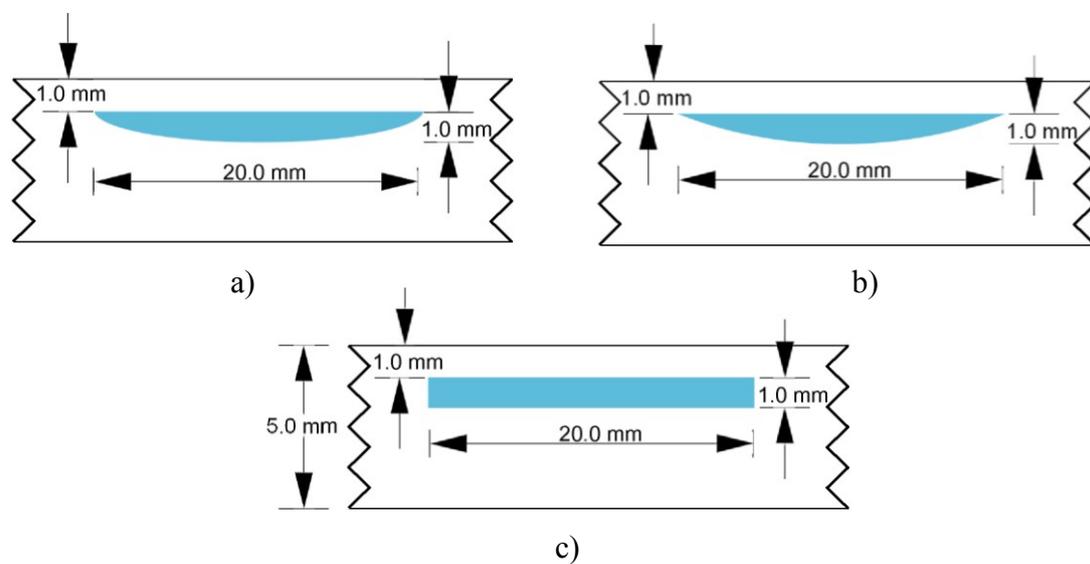


Figure 3.20 Dimensions of a) Defect-E1, b) Defect-E2, and c) rectangular defect modeled by CIVA.

The predicted EC images of these defects at 5 kHz are shown in Figure 3.21 and the evaluation results of the MD-RBF neural network are given in Table 3.16. As can be observed, the predictions of MD RBF neural network are identical, despite change in the shape of the defects. This brings out the capability of the proposed MD-RBF neural network for sizing non-rectangular realistic defects.

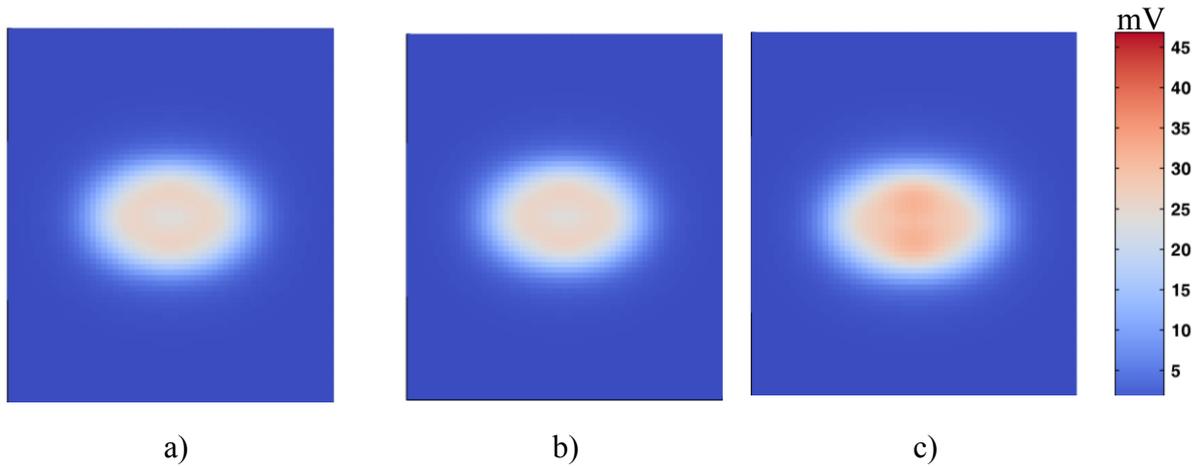


Figure 3.21 EC images of a) Defect-E1, b) Defect-E2, and c) rectangular defect at 5 kHz.

Table 3.16 Performance evaluation of MD-RBF neural network for the ellipsoidal defects.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-E1	20.0	20.0	1.0	1.0	2.0	2.0	1.0	1.0
Defect-E2	20.0	20.0	1.0	1.0	2.0	2.0	1.0	1.0

3.4.3.2 Performance assessment for defects smaller than EC probe diameter

In order to assess the capability of the proposed MD-RBF, defects that are smaller than the probe diameter of 26.0 mm have been considered. Two sets of defects with lengths of 5.0 mm and 10.0 mm have been modeled using CIVA. Set-1 contains defects with depth 3.0 mm, height 2.0 mm, and width 1.0 mm, and Set-2 contains defects with depth 2.0 mm, height 3.0 mm, and width 1.0 mm.

For these 4 notches, length was estimated as 20.0 mm. The extracted image features relevant to length (F1, F2, F3, F10, F11, and F12) are given in Table 3.17. As can be observed from Table 3.17, these 6 features have direct correlation with defect length. This result is encouraging and this increases the confidence that defects smaller than the probe diameter can be sized with reasonable error, although they are not used for training. Further, increasing the number of

classes for defect length, the error in length estimation can be reduced significantly without much penalty on computational resources.

Table 3.17 Extracted EC image features relevant to length for defects smaller than the probe diameter.

FEATURES	FEATURE VALUE		
	LENGTH: 5.0 mm	LENGTH: 10.0 mm	LENGTH: 20.0 mm
SET-1			
F1	17.0	19.0	21.0
F2	15.0	17.0	20.0
F3	14.0	16.0	20.0
F10	13.0	13.0	18.0
F11	11.0	13.0	17.0
F12	14.0	14.0	20.0
SET-2			
F1	17.0	19.0	21.0
F2	15.0	17.0	20.0
F3	14.0	14.0	20.0
F10	11.0	13.0	18.0
F11	11.0	13.0	17.0
F12	14.0	14.0	20.0

3.5 SUMMARY

A new multidimensional radial basis function (MD-RBF) neural network has been proposed for eddy current inversion. This incorporates dependency implicitly. It is capable of sizing simultaneously all the four characteristics of defects (i.e. length, width, depth, and height) considered with all the input features together through a single training. It overcomes the limitation of conventional RBF neural networks that require the selective input for each characteristic of a defect and a separate neural network for sizing each of the defect characteristic.

Features are extracted from EC images based on their relevance to defect characteristics. It has been established that the extracted features considered in this thesis show strong relationship with the defect characteristics. The extracted features proposed in this thesis are found to be noise tolerant (up to 10 dB) and they are also found to be rotation invariant.

The MD-RBF neural network has ensured a global accuracy of 83% with a relatively less number of computations as compared to the global accuracy of 76% achieved by the conventional RBF neural networks.

The proposed MD-RBF neural network has successfully sized defects that are located even 3.0 mm below the surface. For subsurface machined notches, no miss-classifications have been observed by the MD-RBF neural network, while three misclassifications have been observed by the conventional RBFs. Further, the MD RBF neural network has been able to size non-rectangular defects that have not been used during training.

The proposed MD-RBF neural network has shown promising performance for sizing 4 characteristics of rectangular defects from EC images obtained from modelling as well as experiments. The MD-RBF neural network ensured successful sizing of defects that are relatively larger than the probe diameter as well as defects slightly smaller than the probe diameter. It is able to size both surface and subsurface defects in stainless steel plates.

4

INFLUENCE OF SCAN PITCH ON DEFECT SIZING

Multidimensional learning has produced encouraging results for sizing defects. Probe scan pitch is an important variable in eddy current imaging and this is expected to influence defect sizing. This chapter is devoted to study the influence of scan pitch on defect sizing and to determine, if there is any limit of scan pitch for an EC probe below which the MD-RBF neural network ensures reliable sizing of defects.

4.1 SCAN PITCH

Unlike other imaging modalities, eddy current images are formed by collecting discrete raster scans. Thus, the quality of an EC image depends on the number of the data points acquired in ROI. Scan pitch is defined as the distance between two consecutive line scans during the raster scan imaging, as typically shown in Figure 4.1.

EC images obtained at smaller scan pitch produce high resolution images as typically shown in Figure 4.2a. High resolution EC images have higher probability of detection as well as better interpretation of defects. However, the time required for raster scan imaging is very high. On the contrary, EC images obtained at larger scan pitches produce low resolution images as shown in Figure 4.2b. Low resolution images have relatively limited information and they are not suitable for accurate sizing. Hence, it becomes paramount importance to study the influence of scan pitch on defect sizing.

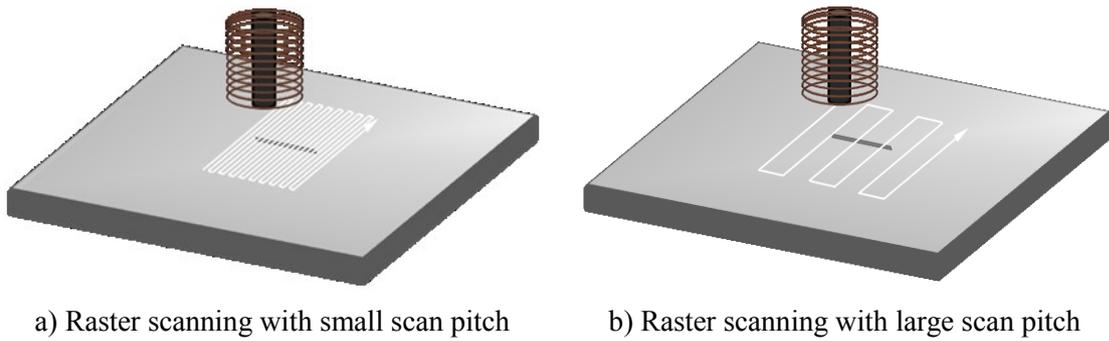


Figure 4.1 Comparison of two different scan pitches.

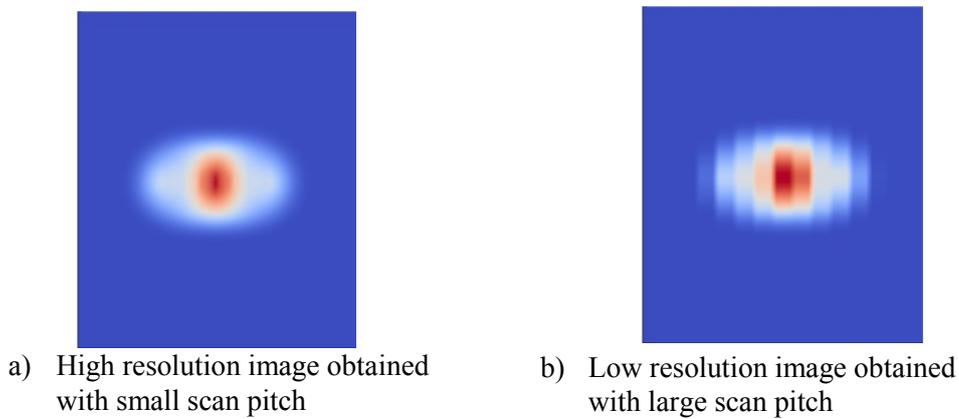


Figure 4.2 Comparison of CIVA modeled images obtained at two different scan pitches.

For analysis of influence of scan pitch on defect sizing, the proposed MD-RBF neural network has been trained with training dataset as described in Section 3.3.1.4 (Page no 61). It is important to note that these training defects are obtained at a scan pitch of 1.0 mm and the size of each image is 70×60 . The trained MD-RBF is tested with a set of EC images of defects which are not trained and obtained at higher scan pitches than those of the training EC images, but covering the same area i.e. ROI. In all the cases, it has been assumed that the center line scan passes through the center of defect.

The time taken for scanning at a point during experiment using the setup discussed in Section 3.3.1.2 has found to be 0.14 second while moving the probe 1.0 mm requires 0.03 seconds in

either of the direction. The time taken for EC imaging of a defect at various scan pitches for an area of 70 mm × 60 mm is given in Figure 4.3. It can be observed there is a near exponential fall in time taken with changes in scan pitch. Thus, it is beneficial to find a highest scan pitch below which defect sizing is accurate and consistent as to that of scan pitch 1.0 mm. This is called as limit scan pitch. The objective of this study is to identify limit scan pitch, λ , for each of the defect characteristics viz. λ_l , λ_w , and λ_d for length, width, and depth, respectively. This will reduce considerable time during EC imaging for sizing of different defect characteristics.

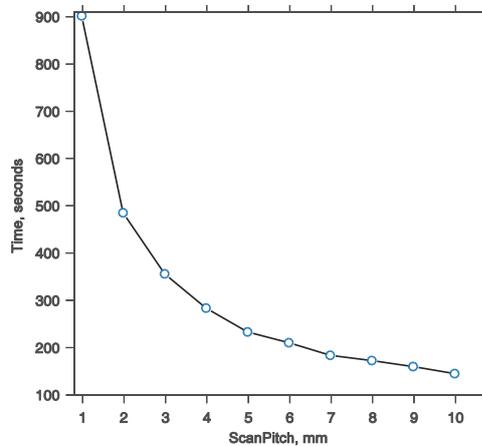


Figure 4.3 Time taken for EC imaging of a defect at different scan pitches for an area of 70 mm × 60 mm.

4.2 INFLUENCE OF SCAN PITCH ON SIZING LENGTH OF DEFECTS

The influence of scan pitch on the performance of sizing length of defects is studied using defect cases which are not used in the training and by varying the scan pitch in the range of 1.0 mm to 10.0 mm. The width, depth, and height of defects are kept constant, as 1.0 mm, 2.2 mm, and 1.0 mm respectively, for subsurface defects. For surface defects, width and height of the defects are kept constant, as 1.0 mm. The results of performance for scan pitch ranging from 6.0 to 10.0 mm for surface and subsurface defects are shown in Figure 4.4. The actual length of the defects considered is represented as the abscissa and the predicted length by the MD-RBF neural

network is represented as the ordinate. It is desired that the actual and the predicted lengths show good correlation. It can be observed from Figure 4.4, that the predicted and actual lengths are nearly same until a scan pitch of 6.0 mm for surface defects and 7.0 mm for subsurface defects. Increase in scan pitch has lead to underestimation of the larger surface defects, while overestimation has been observed for the subsurface defects.

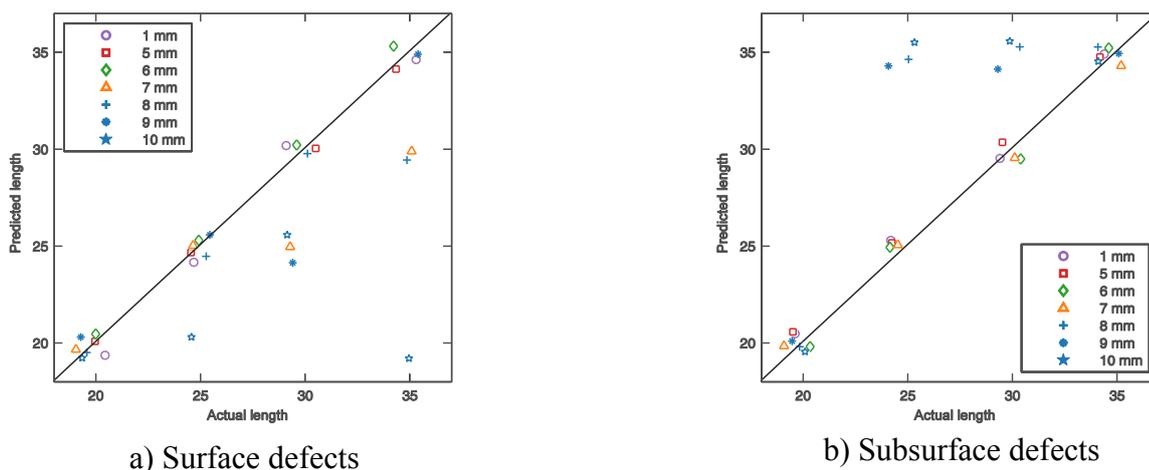


Figure 4.4 Influence of scan pitch on sizing length.

This can be better understood by studying the features relevant to length of defects. A comparison between various scan pitches and distance between extreme peaks along the length direction in EC image (Feature F8 and Feature F12) for surface defects is shown in Figure 4.5. It is clearly observed that the geometrical distance between the extreme peaks is constant until the scan pitch is 6.0 mm. These features lead to accurate prediction by MD-RBF neural network until a scan pitch of 6.0 mm. Similar observations can be made from Figure 4.5 for subsurface defects with a comparison between various scan pitches and the geometrical distance covered between peak magnitude and 25% of peak magnitude (F1). It is clearly observed that the Feature F1 changes significantly after the scan pitch is 6.0 mm for defects with length 20.0, 25.0, and 30.0 mm, thus causing erroneous predictions in length for a scan pitch above 6.0 mm. Hence, the

limit scan pitch λ_l is 6.0 mm for the EC probe chosen in the thesis. At this limit scan pitch the time taken for EC imaging is about 25% of the time required to obtain EC images at 1.0 mm scan pitch.

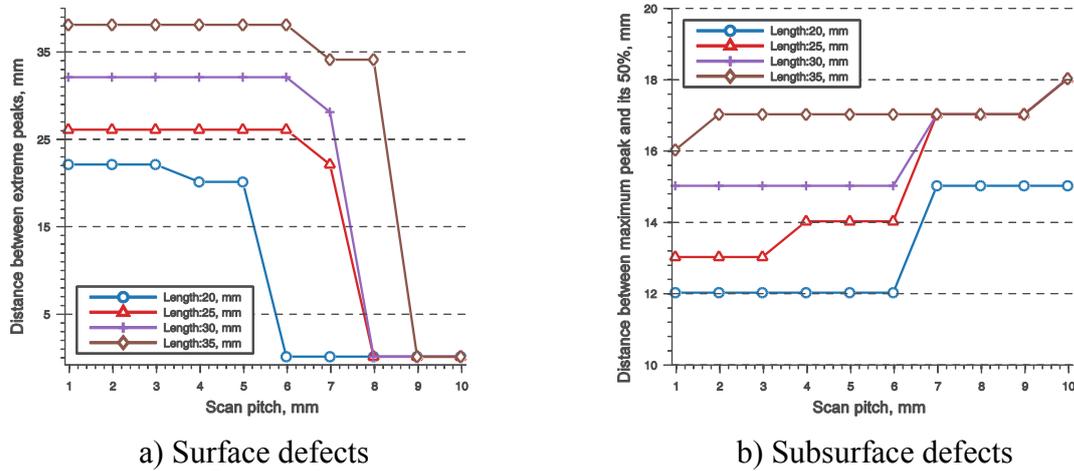


Figure 4.5 Influence of scan pitch on length related features.

4.3 INFLUENCE OF SCAN PITCH ON SIZING WIDTH OF DEFECTS

The influence of scan pitch on the performance of sizing width of defects is studied by varying the scan pitch in the range of 1.0 to 10.0 mm. The length, depth, and height of defects are kept constant as 25.0 mm, 2.2 mm, and 1.0 mm respectively, for subsurface defects. For surface defects, length and height of defects are kept constant, as 25.0 mm and 1.0 mm respectively. The comparison between the actual width and predicted width by the MD-RBF neural network for scan pitch in the range of 6.0 mm to 10.0 mm is shown in Figure 4.6. As can be observed from Figure 4.6, the predicted and actual widths show a good correlation until a scan pitch of 6.0 mm for both surface as well as subsurface defects similar to that of the length predictions. Thus, the limit scan pitch for width λ_w is found to be 6.0 mm for both surface defects and subsurface defects, for the probe chosen in the thesis.

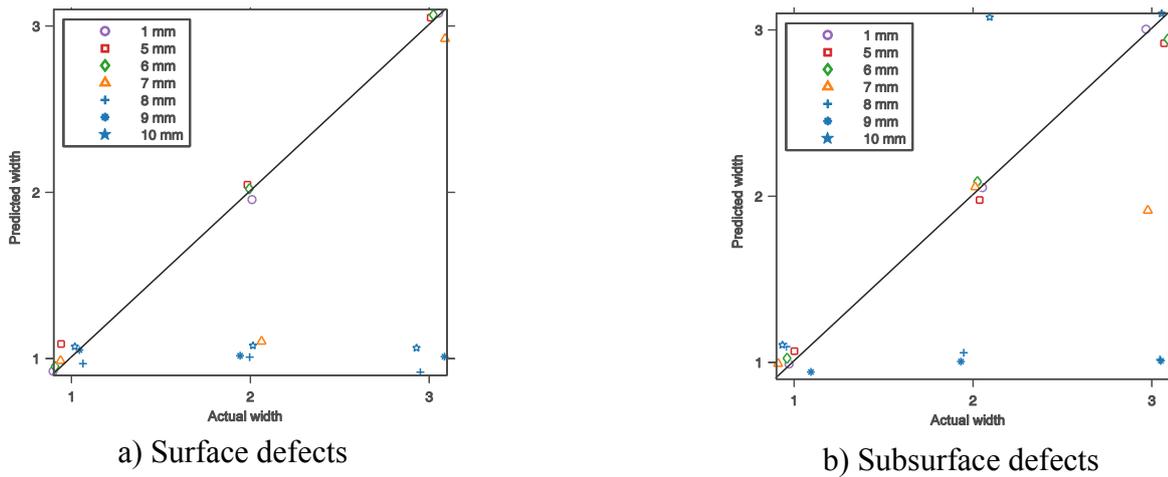


Figure 4.6 Influence of scan pitch on sizing.

4.4 INFLUENCE OF SCAN PITCH ON SIZING DEPTH OF DEFECTS

Influence of scan pitch on the performance of MD-RBF neural network for sizing depth of defects is investigated by varying the scan pitch from 1.0 to 30.0 mm. Other characteristics of defects such as length, width, and height have been kept constant, as 25.0 mm, 1.0 mm, and 1.0 mm, respectively. The depth of defect has been considered as 0.0 mm (surface), 1.4 mm, 1.8 mm, 2.2 mm, and 3.7 mm. The predicted value for depth is assumed to be the midpoint of the class range (refer Table 3.3). The results are given in Table 4.1.

Table 4.1 Predictions of MD-RBF neural network for different scan pitches.

ACTUAL DEPTH, mm	3.0 mm	4.0 mm	6.0 mm	8.0 mm	10.0 mm	15.0 mm	30.0 mm
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1.0	0.8	0.8	0.8	0.8	0.8	0.8	0.8
1.8	2.0	3.0	3.0	3.0	3.0	3.0	3.0
2.2	2.0	2.0	2.0	2.0	3.0	3.0	3.0
4.0	4.3	4.3	4.3	4.3	4.3	4.3	4.3

It is observed that the predictions of MD-RBF neural network for depth closely agree with that of the actual value measured by the optical method, even for 30.0 mm scan pitch. The 17 feature

values for one typical defect are given in Table 4.2. It is observed that the change with scan pitch in feature values relevant to depth such as maximum magnitude (F7/F16) is less than that of the change in distance between maximum magnitudes (F3/F12) even for 30.0 mm scan pitch, provided the center line scan of EC image and center line of defects are aligned. Thus, limit scan pitch for depth $\lambda_d = 30.0$ mm and this study clearly brings out an interesting observation that depth sizing is least influenced by the scan pitch. The time taken for depth sizing at this scan pitch is 95.61 seconds which is 10 times less than that of EC images obtained at scan pitch of 1.0 mm.

Table 4.2 Feature value at different scan pitches for a defect (length: 25.0 mm, width: 1.0 mm, depth: 1.4 mm, height: 1.0 mm).

FEATURE NO	3.0 mm	4.0 mm	6.0 mm	8.0 mm	10.0 mm	15.0 mm	30.0 mm
F1	21.0	21.0	21.0	21.0	21.0	21.0	21.0
F2	17.0	17.0	17.0	17.0	16.0	16.0	16.0
F3	26.0	28.0	28.0	0.0	0.0	0.0	0.0
F4	420.0	336.0	252.0	252.0	168.0	168.0	168.0
F5	272.0	272.0	204.0	136.0	128.0	128.0	128.0
F6	0.0	0.0	0.0	0.0	0.0	0.0	0.0
F7	43.4	43.5	42.9	42.9	39.9	39.9	39.9
F8	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
F9	2.4	2.4	2.4	2.4	2.3	2.3	2.3
F10	21.0	21.0	21.0	21.0	21.0	21.0	21.0
F11	17.0	17.0	17.0	17.0	15.0	15.0	15.0
F12	26.0	28.0	28.0	0.0	0.0	0.0	0.0
F13	420.0	336.0	252.0	252.0	160.0	160.0	160.0
F14	272.0	272.0	192.0	128.0	120.0	120.0	120.0
F15	0.0	0.0	0.0	0.0	0.0	0.0	0.0
F16	103.7	104.0	102.1	102.1	93.6	93.6	93.6
F17	0.1	0.1	0.1	0.1	0.1	0.1	0.1

4.5 SUMMARY

EC images obtained at smaller scan pitch produce high resolution images and better for defect sizing. It is essential to evaluate the performance of MD-RBF neural network with respect to change in scan pitch, in order to find the highest scan pitch below which defect sizing is

accurate, with least possible time. This has been studied in this chapter, for the first time. It has been observed that the performance of MD-RBF neural network is robust for EC images obtained at higher scan pitches than that of the training images, for the probe chosen in this thesis. It has been found, for the first time, that the limit scan pitch λ , varies for different defect characteristics ($\lambda_l = 6.0$ mm, $\lambda_w = 6.0$ mm, and $\lambda_d = 30.0$ mm). Sizing length and width of defects has more influence on the scan pitch while sizing depth of defects has found to be the least influence on the scan pitch. While the scan pitch has definite influence on the sizing accuracy, the performance of MD-RBF neural network has not significantly altered by the change in scan pitch up to 6.0 mm when MD-RBF neural network is trained with EC images of 1.0 mm scan pitch. For the chosen probe of 26 mm diameter, the limit scan pitch of 6 mm is about 25%. Thus when raster scan imaging is performed at a scan pitch of 6 mm the probability of accurate size estimation of length, width, depth, and height is very high by the chosen probe. The scan pitch is dependent on the diameter of the probe and hence, limit scan pitch required to be estimated for every probe used for inversion.

5

CHAIN CLASSIFICATION BASED ON GREEDY BREADTH-FIRST-SEARCH FOR DEFECT SIZING

The scope of this chapter is to extensively study the nature of dependency on sizing for each defect characteristics. The chapter discusses a new methodology called chain classification proposed to incorporate dependency among the 4 defect characteristics. A new algorithm called greedy breadth-first-search (GBFS) is also proposed to identify the sequence of dependency for chain classification. Based on the performance of the chain classification using GBFS, the dependency structure of the defect characteristics is identified.

5.1 CHAIN CLASSIFICATION

As discussed in Section 3.4.1, the proposed MD-RBF neural network incorporates the dependency implicitly. However, it is important to understand the explicit structure of the dependency among the defect characteristics. This will provide a better insight into the empirical inversion process for defect sizing. By incorporating the explicit nature of dependency, improvement in accuracy of defect sizing is also expected, as evident from the results discussed in Section 3.4.1. In this context, the explicit structure of dependency for defect sizing has been investigated systematically. For this, chain classification method recently proposed and applied for several real world datasets, [70,71] is employed.

For a dataset \mathcal{D} represented as $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$, the chain classification decomposes the dimensions of multidimensional data in such a way that previous predictions are used for present classification as:

$$f(\mathbf{x}) = (y^1, y^2, \dots, y^d) \Rightarrow \begin{cases} f_1(\mathbf{x}) = y^1 \\ f_2(\mathbf{x}, \hat{y}^1) = y^2 \\ \vdots \\ f_d(\mathbf{x}, \hat{y}^1, \hat{y}^2, \dots, \hat{y}^{d-1}) = y^d \end{cases} \quad (5.1)$$

In chain classification, d independent classification functions ($f_1(\cdot), f_2(\cdot), \dots, f_d(\cdot)$) are trained, representing d class variables. The process starts with the function $f_1(\{\mathbf{x}_i\}_{i=1}^N)$, where $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^m$ is a vector of input features, predicting class values $\{\hat{c}_i^1\}_{i=1}^N$. Then, during execution of the function $f_2(\cdot)$, along with \mathbf{x} , the class values \hat{c}^1 is also provided as input: $f_2(\{\mathbf{x}_i\}_{i=1}^N, \{\hat{c}_i^1\}_{i=1}^N)$, that predicts class values $\{\hat{c}_i^2\}_{i=1}^N$. In this way, the input feature \mathbf{x} along with all class values $\hat{c}^1, \hat{c}^2 \dots \hat{c}^{k-1}$ is provided to the function $f_k(\cdot)$. The chain process completes with the function $f_d(\{\mathbf{x}_i\}_{i=1}^N, \{\hat{c}_i^1\}_{i=1}^N, \dots, \{\hat{c}_i^{d-1}\}_{i=1}^N)$ predicting the class values $\{\hat{c}_i^d\}_{i=1}^N$. Thus, by including predictions from previous classification, the chain classification explicitly incorporates dependency.

Figure 5.1 shows an example of possible chain classification scheme for sizing of length, width, depth, and height of a subsurface defect from its EC images. The classification function $f(\cdot)$ can be any supervised classification algorithm such as multilayer perceptron, radial basis function neural network, SVM, decision trees, and KNN. The benefits of chain classification are: 1) It explicitly incorporates dependency among the class variables during classification and 2) The optimal sequence represents the dependency structure among the defect characteristics during sizing. In this study, two diverse representations and learning algorithms, namely, RBF neural network and SVM have been considered based on their good performance for defect sizing in literature for chain classification [61,64,65].

The central question that remains with chain classification is determination of an optimal sequence of class variables to incorporate in the chain. This is an optimization problem which aims at identifying a sequence that maximizes the performance of the chain.

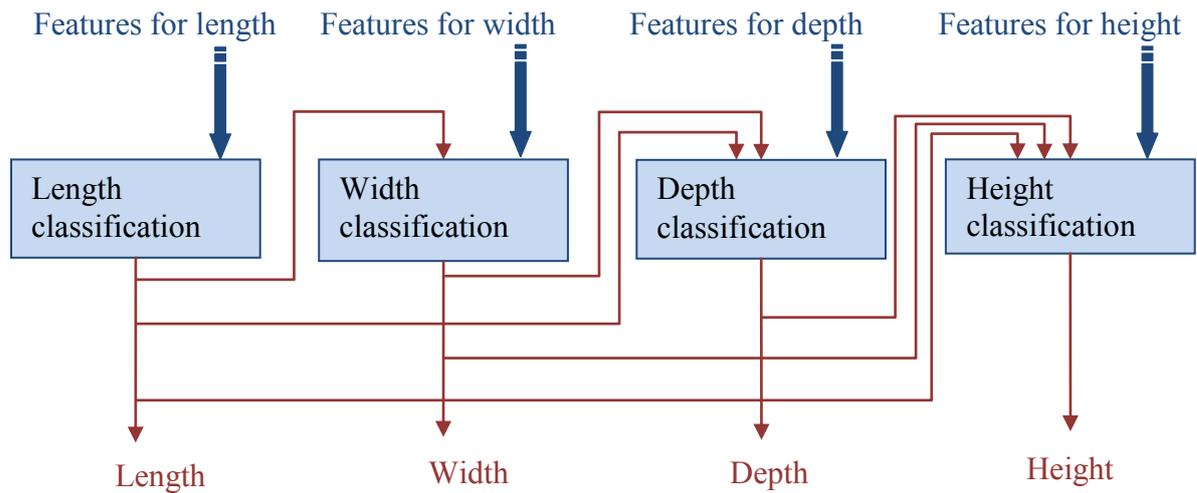


Figure 5.1 Typical chain classification scheme for subsurface defect sizing.

5.2 GREEDY BREADTH-FIRST-SEARCH (GBFS) ALGORITHM FOR DETERMINATION OF AN OPTIMAL SEQUENCE

If the D dimensional class variables are represented as a set of vertices of a graph $\mathcal{V} = \{\mathcal{V}_{C1}, \mathcal{V}_{C2}, \dots, \mathcal{V}_{Cd}\}$ then the sequence of class variables can be defined as a directed acyclic graph (DAG) $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the directed edges between the vertices. For chain classification in this thesis, the number of incoming (indegree) and outgoing (outdegree) edges from a node of the DAG is restricted to at most one. An example DAG randomly chosen with 5 vertices is shown in Figure 5.2.

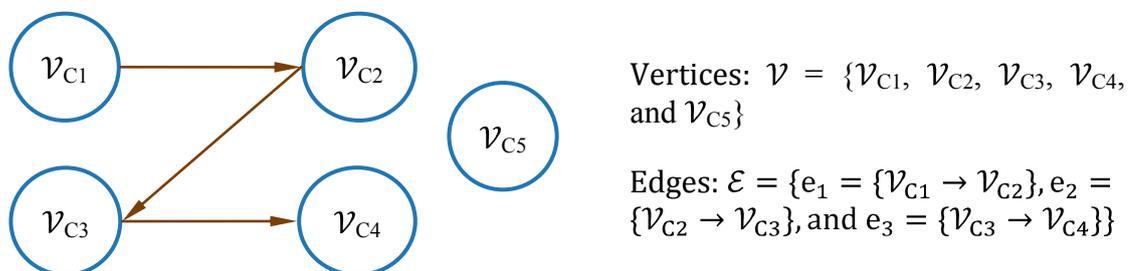


Figure 5.2 A random directed acyclic graph with 5 vertices.

Given a DAG \mathcal{G} it is easy to implement the chain classifier. As an example, following Figure 5.2, chain classifier for class variables of dimension $d = 5$ can be implemented as follows:

- Implement independent classifier with input features $\{\mathbf{x}^{(i)}\}_{i=1}^N$ to determine the class values $\{C_1^{(i)}\}_{i=1}^N$.
- Along with input feature $\{\mathbf{x}^{(i)}\}_{i=1}^N$, include the class values of $\{C_1^{(i)}\}_{i=1}^N$ to determine the class values of $\{C_2^{(i)}\}_{i=1}^N$.
- Along with input feature $\{\mathbf{x}^{(i)}\}_{i=1}^N$, include the class values of $\{C_2^{(i)}\}_{i=1}^N$ to determine the class values of $\{C_3^{(i)}\}_{i=1}^N$.
- Along with input feature $\{\mathbf{x}^{(i)}\}_{i=1}^N$, include the class values of $\{C_3^{(i)}\}_{i=1}^N$ to determine the class values of $\{C_4^{(i)}\}_{i=1}^N$.
- Implement independent classifier with input features $\{\mathbf{x}^{(i)}\}_{i=1}^N$ to determine the class values $\{C_5^{(i)}\}_{i=1}^N$.

But given the dimension D of class variables, it is difficult to determine the best possible sequence of dependency. Because all possible enumeration of a DAG for D dimension when the indegree and outdegree of each node restricted to 1 is

$$a(D) = D! \sum \left(\frac{1}{k!} \right) + 1 \quad (5.2)$$

where $a(D)$ is the number of possible DAGs for a given dimension D and $k = 0, 1, \dots, D-2$. It can also be represented as a recurrence relation as follows:

$$a(D) = (D \times (a(D - 1) + n - 2)) + 1, \quad (5.3)$$

for $D > 1, \quad a(1) = 1$

For $D = 1, 2, 3, 4, 5, 6 \dots$ possible enumerations are 1, 3, 13, 61, 321, 1951, ... Hence, it is difficult to implement the chain by an exhaustive search of all possible sequences and trace

the optimal sequence. The determination of an optimal sequence for the chain classification is an optimization problem which can be represented as a search tree and it is different from the traditional search strategies [72] in the following aspects:

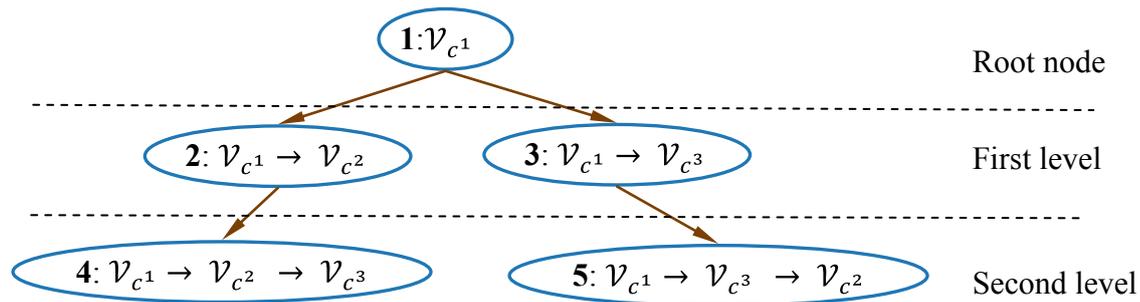
- Initially, the target (optimal sequence) is unknown.
- Performance evaluation (requires computation of classification algorithm) at each node (node expansion) of the search tree is computationally intensive than a mere visit to the node.
- Performance is a vector (a component for each class variable) and depending on the path and level of the search tree, the evaluation function modifies different components of the performance vector.
- Performance evaluation at a node of the search tree provides only partial information about the target.

To address the above optimization problem, a modified breadth-first-search (BFS) strategy, greedy breadth-first-search (GBFS) is proposed. GBFS can optimize the sequence rapidly than traditional uninformed search strategies. The objective of the GBFS is to trace the best sequences that possibly enhance the performance by propagating in the chain.

An example search tree for the GBFS is illustrated in Figure 5.3. Three class variables $\{C^1, C^2$ and $C^3\}$ are taken in the example with its nodes (vertices) represented as $\{\mathcal{V}_{c^1}, \mathcal{V}_{c^2}, \mathcal{V}_{c^3}\}$. \mathcal{V}_{c^1} is considered as the root node of the search tree and the first level and the second level chains are also displayed in Figure 5.3.

In the example, initially, \mathcal{V}_{c^1} is visited and expanded, followed by the visit and expansion of $\{\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2}\}$ and $\{\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^3}\}$. $\{\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2}\}$ represents the class values of C^1 that are included during the prediction of C^2 . The searching process continues, following the BFS. In the example, the performance of $\{\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2} \rightarrow \mathcal{V}_{c^3}\}$ is computed by classification, only if, the

performance of C^2 in $\{\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2}\}$ is greater or equal to the performance of C^2 during independent classification. Similarly, the performance of $\{\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^3} \rightarrow \mathcal{V}_{c^2}\}$ is computed by classification only if the performance of C^3 in $\{\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^3}\}$ is greater or equal to the performance of C^3 in $\{\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2} \rightarrow \mathcal{V}_{c^3}\}$, if it is expanded, otherwise it is compared to independent classification of C^3 . The pseudocode for GBFS is given in Algorithm 5.1.



Sequence of expansion for BFS and GBFS

BFS :

$1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 4 \Rightarrow 5$

GBFS :

Case 1: (if $performance(\mathcal{V}_{c^2}) > performance(\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2})$):

$1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 5$

{assuming $performance(\mathcal{V}_{c^3}) < performance(\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^3})$ }

Case 2: (if $performance(\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2} \rightarrow \mathcal{V}_{c^3}) > performance(\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^3})$):

$1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 4$

{assuming $performance(\mathcal{V}_{c^2}) < performance(\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2})$ }

Case 3: (if $performance(\mathcal{V}_{c^3}) > performance(\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^3})$):

$1 \Rightarrow 2 \Rightarrow 3$

{assuming $performance(\mathcal{V}_{c^2}) > performance(\mathcal{V}_{c^1} \rightarrow \mathcal{V}_{c^2})$ }

Figure 5.3 Classifier chain tree and sequence of expansion of nodes using BFS and GBFS.

Algorithm 5.1 Greedy breadth-first-search

```
GBFS(v)
    //Initialize bestPerformance as a vector of performance of independent classification
for    //each class variable
     $Q \leftarrow \emptyset$  // an empty queue
     $bestPath \leftarrow \emptyset$ 
     $Q.enqueue(v)$  // enqueue a vertex  $v$  onto  $Q$ 
    while  $Q \neq \emptyset$  loop
         $t \leftarrow Q.dequeue()$ 
        if  $parent(t) = \emptyset$  then
            for each  $children(t)$  loop
                 $u \leftarrow children(t)$ 
                 $Q.enqueue(u)$  // enqueue vertex  $u$  onto  $Q$ 
            end loop
            continue
        end if
        if  $performance(parent(t)) < bestPerformance$  then
            continue
        else
             $p \leftarrow computePerformance(t)$ 
            // using classification algorithm
            if  $p < bestPerformance$  then
                 $bestPath \leftarrow bestPath \cup parent(t)$ 
                continue
            else if  $children(t) = \emptyset$  then
                 $bestPath \leftarrow bestPath \cup t$ 
                continue
            else
                 $bestPerformance \leftarrow p$ 
                for each  $children(t)$  loop
                     $u \leftarrow children(t)$ 
                     $Q.enqueue(u)$  // enqueue vertex  $u$  onto  $Q$ 
                end loop
            end if
        end if
        return  $unique(bestPath)$ 
    end loop
end GBFS
```

Input to the GBFS algorithm is a class variable represented as a root node of the search tree. The children of this node are the first level chain, formed from all possible remaining class variables, and so on. The algorithm initiates with a queue, for implementation of the BFS, and a *bestPerformance* tracker, containing the performance of each class variable. Any comparison is carried out only to the concerned individual class variable. The tracker *bestPerformance* is

updated at each node expansion. A node is expanded using *computePerformance(.)* only if the performance of its parent is equal to the value of the *bestPerformance* tracker for the concerned class variable. At the end, all the paths that improved the performance over independent classification are returned.

5.3 RESULTS AND DISCUSSION

All the subsurface defects used in the training dataset described in Section 3.3.1.4 have been used for defect sizing using chain classification. For classification of four defect characteristics (length, width, depth, and height), different set of features have been used based on their relationship to the defect characteristics as discussed in Section 3.3.2. RBF neural network and SVM have been used as classification algorithms in chain classification. The clustering required for hidden layer determination of RBF neural networks has been carried out by the traditional k-means algorithm, while polynomial function has been used as the kernel of SVM.

5.3.1 *Independent classification*

In order to understand the sensitivity of the classification algorithms and optimize the performance, with respect to α and μ for RBF neural networks and d for the polynomial kernel of SVM, systematic analysis has been performed. The performance has been evaluated using accuracy on a scale of 0-1 during stratified 10-fold cross validation of training data, and the average of 10 independent runs is used. The higher the accuracy, the better the performance.

5.3.1.1 *Sensitivity analysis of RBF neural networks*

Figure 5.4 and Figure 5.5 illustrate the performance of the RBF neural networks under different parameter settings, namely, the clustering factor, α and the spread μ . α has been varied from 0.05 to 0.65 with an interval of 0.05 and the variation in accuracy has been studied for different fixed μ of 0.2, 0.4, 0.6, 0.8 and 1.0.

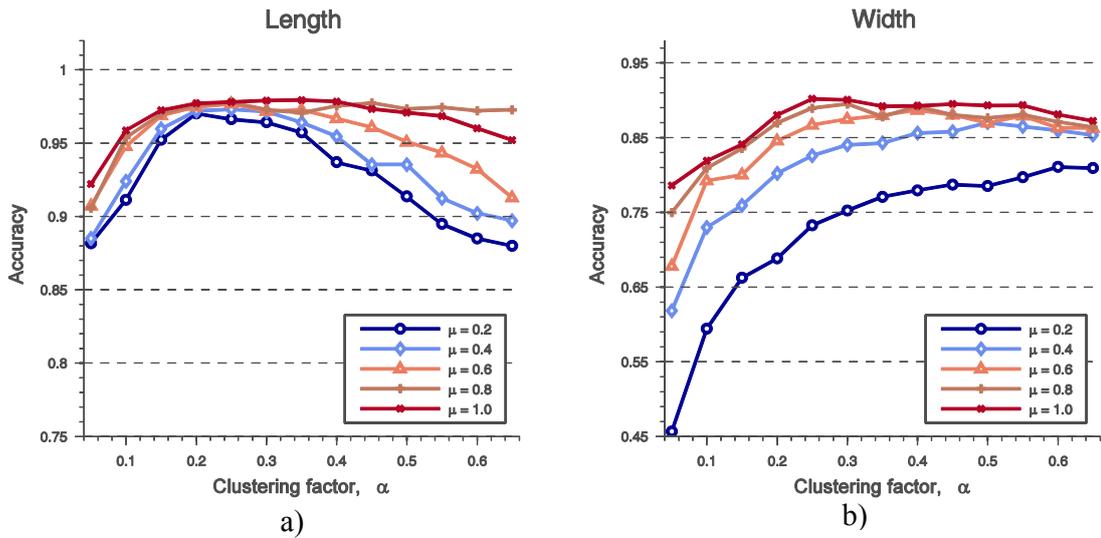


Figure 5.4 10-fold cross validation performance of RBF neural networks for classification of a) length and b) width for different α and μ .

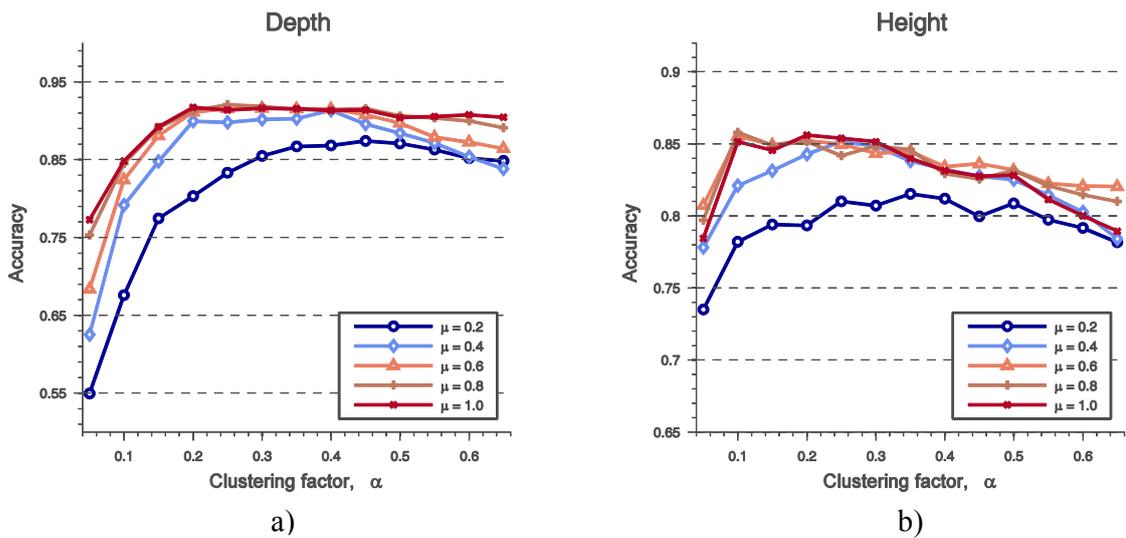


Figure 5.5 10-fold cross validation performance of RBF neural networks for classification of a) depth and b) height for different α and μ .

From Figure 5.4 and Figure 5.5, it can be observed that for constant μ , the initial performance improves significantly with increase in α . As α increases further, the accuracy remains nearly constant for width and depth, but it decreases for length and height. For height, even for $\mu = 1.0$, the performance decreases with increase in α after a short constant period. On the contrary, when α is fixed, the curves get closer to each other as μ increases, especially for α in the range between

0.2 and 0.3. It can be interpreted as the accuracy tends to produce stable performance as μ increases. Based on these observations, optimal α has been set to 0.25 and optimal μ has been set to 1.0. These parameter values have been used throughout the rest of the analysis.

5.3.1.2 Sensitivity analysis of SVM

Sensitivity analysis of SVM has been carried out by varying the polynomial kernel parameter, d and the results are shown in Figure 5.6. As can be observed, there is an initial increase in accuracy when d increases from 1 to 3. With further increase in d , the accuracy is found to be nearly constant until d reaches 6 and later it is decreased. Based on the accuracy, optimal value for d has been chosen as 3 and this has been used throughout the rest of the analysis.

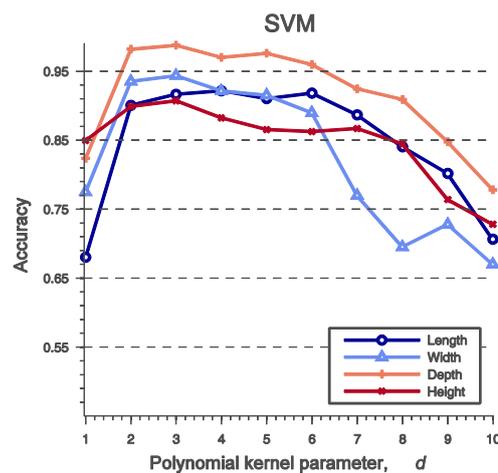


Figure 5.6 10-fold cross validation performance of SVM for classification of all defect characteristics for different polynomial kernel parameter, d .

With these optimal parameters of RBF neural networks and SVM, performance of the classification algorithms has been compared with mean accuracy and given in Table 5.1. It can be observed from Table 5.1, that the SVM has produced better mean accuracy of 0.9400 than that of the RBF neural networks which produced a mean accuracy of 0.9150.

Table 5.1 Comparison of performance of independent RBF neural networks and SVM for defect sizing on training data.

LEARNING ALGORITHM	ACCURACY				MEAN ACCURACY
	LENGTH	WIDTH	DEPTH	HEIGHT	
RBF neural networks	0.9793	0.9020	0.9207	0.8580	0.9150
SVM	0.9216	0.9436	0.9876	0.9072	0.9400

5.3.2 GBFS algorithm for an optimal sequence identification

Results of the GBFS algorithm for optimal sequence identification are represented in the form of a tree diagram, typically shown in Figure 5.7. L, W, D, and H indicate the classification results of tenfold cross validation on the training data for length, width, depth and height respectively. $L \rightarrow W$ represents the results, when length is assessed as an independent classifier and the results of length are given during the classification of width. Similarly, $L \rightarrow W \rightarrow D$ represents independent evaluation of length. And the results of length are given during the classification of width and further, results of length and width are given for the classification of depth.

5.3.2.1 GBFS algorithm using RBF neural network

Figure 5.7 demonstrates the results of chain classifiers using the RBF neural network as the classification algorithm for length as a root node of the search tree.

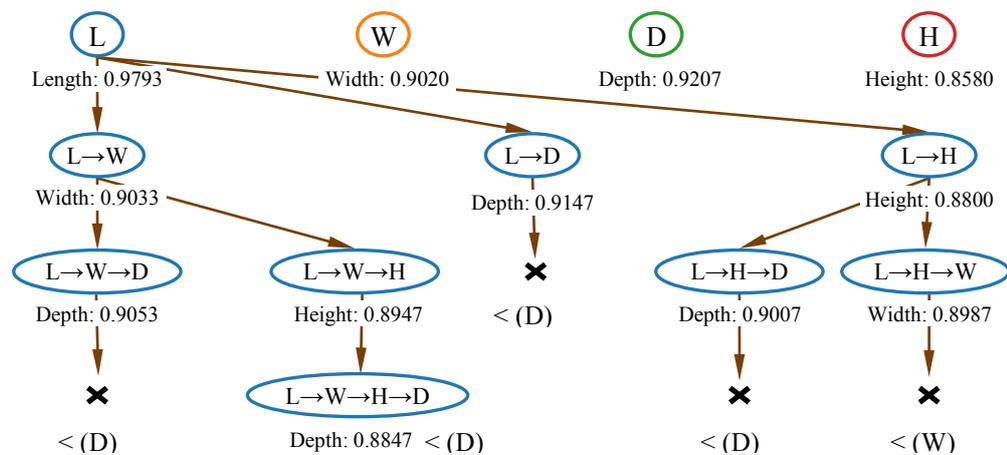


Figure 5.7 GBFS outcome using RBF neural network as the classification algorithm with length as the root node of the search tree.

As can be seen from Figure 5.7, the performance of the RBF neural network for width and height evaluation has been improved when length is added as input to them during classification than independent classification, thus the chain has expanded further. However, the performance for depth has reduced and further, the classification has been terminated, as it has not met the greedy requirement. It can also be noted from Figure 5.7 that none of the sequences improved the performance of all the defect characteristics. The best sequence obtained when length is considered as root node, which maximized the defect characterization, is $L \rightarrow W \rightarrow H$. This sequence resulted in a mean accuracy of 0.9244, as compared to the mean accuracy of 0.9150, achievable by the independent classification. Figure 5.8 shows the results of the chain classifiers using RBF neural network as the classification algorithm for width as the root node of the search tree. Here, no sequence improved the performance of the chain classification. The RBF neural network for chain classification using width as the root node is not promising than that of the independent classification.

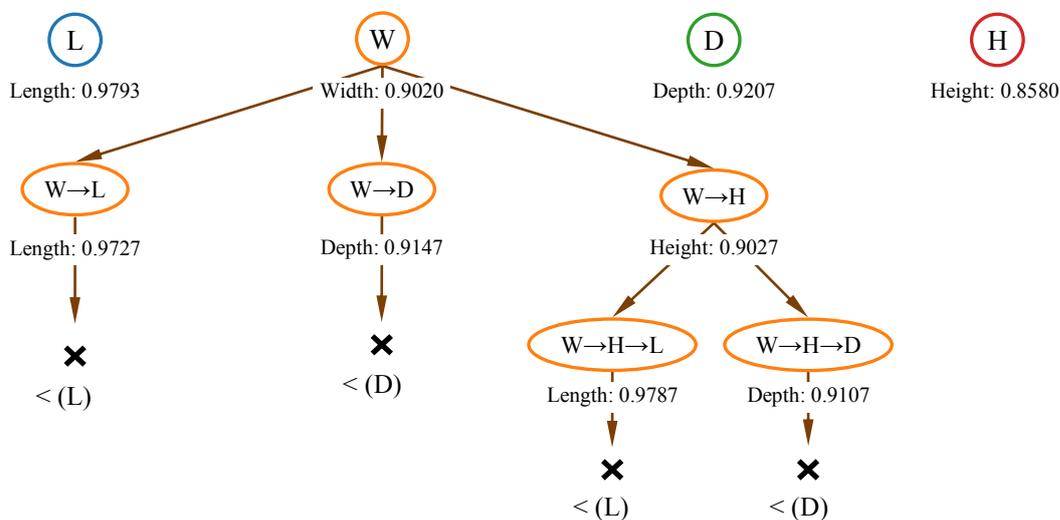


Figure 5.8 GBFS outcome using RBF neural network as the classification algorithm with width as the root node of the search tree.

Figure 5.9 gives the search tree with depth as the root node. From Figure 5.9, it can be observed that the depth as the root node has improved the classification performance for many sequences.

Thus, the most promising sequence is $D \rightarrow W \rightarrow H \rightarrow L$. This sequence achieved a mean accuracy of 0.9306 which is highest among any other sequence. When height is considered as the root node, the mean accuracy has reduced for all the sequences and this can be observed from Figure 5.10.

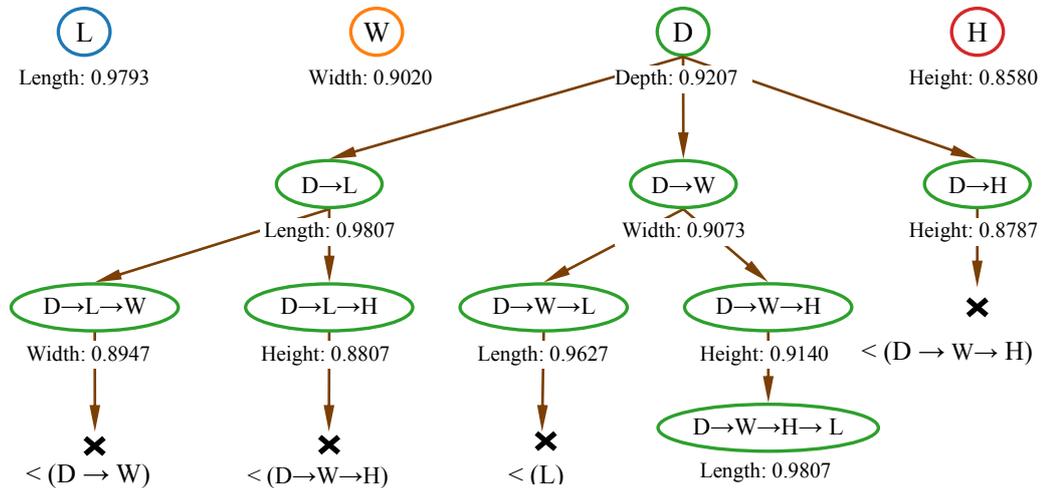


Figure 5.9 GBFS outcome using RBF neural network as the classification algorithm with depth as the root node of the search tree.

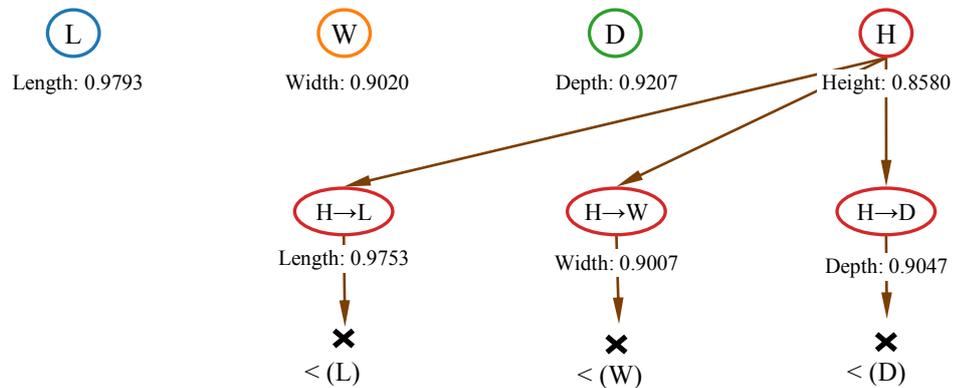


Figure 5.10 GBFS outcome using RBF neural network as the classification algorithm with height as the root node of the search tree.

The total number of nodes expanded by the GBFS for RBF neural network is 25 including 1 for independent classification for the root nodes. Table 5.2 summarizes the sequences using RBF

neural network that improved the performance of the chain classification over the independent classification.

Table 5.2 Performance of RBF neural network in chain sequence with different defect characteristics as root node of the GBFS.

SEQUENCE	MEAN ACCURACY
D→W→H→L	0.9306
D→W→H	0.9303
W→H	0.9261
L→W→H	0.9244
D→L→H	0.9209
L→H	0.9205
D→H	0.9201
D→W	0.9163
L→W	0.9153
D→L	0.9153
Independent	0.9150

5.3.2.2 GBFS algorithm using SVM

Figure 5.11 to Figure 5.14 illustrate the results of the chain classifiers using SVM as the classification algorithm for different defect characteristics as the root node of the search tree. The best sequence identified by SVM is $D \rightarrow W \rightarrow H \rightarrow L$. This sequence resulted in a mean accuracy of 0.9622.

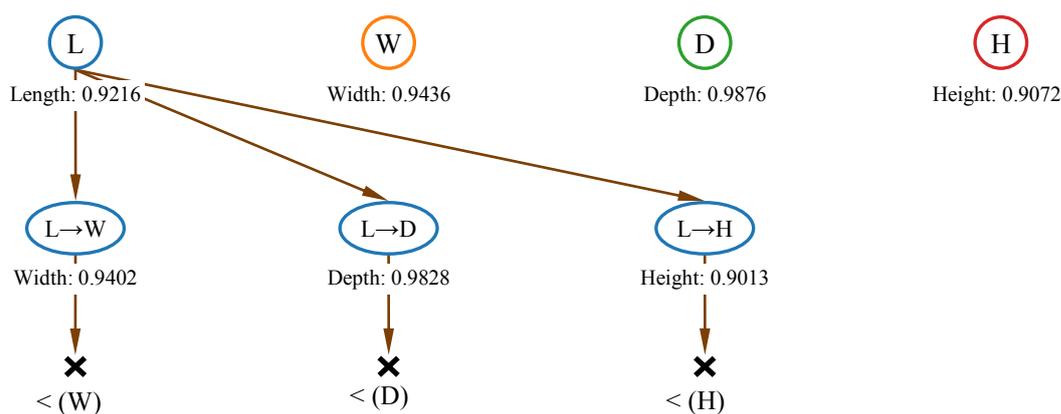


Figure 5.11 GBFS outcome using SVM as the classification algorithm with length as the root node of the search tree.

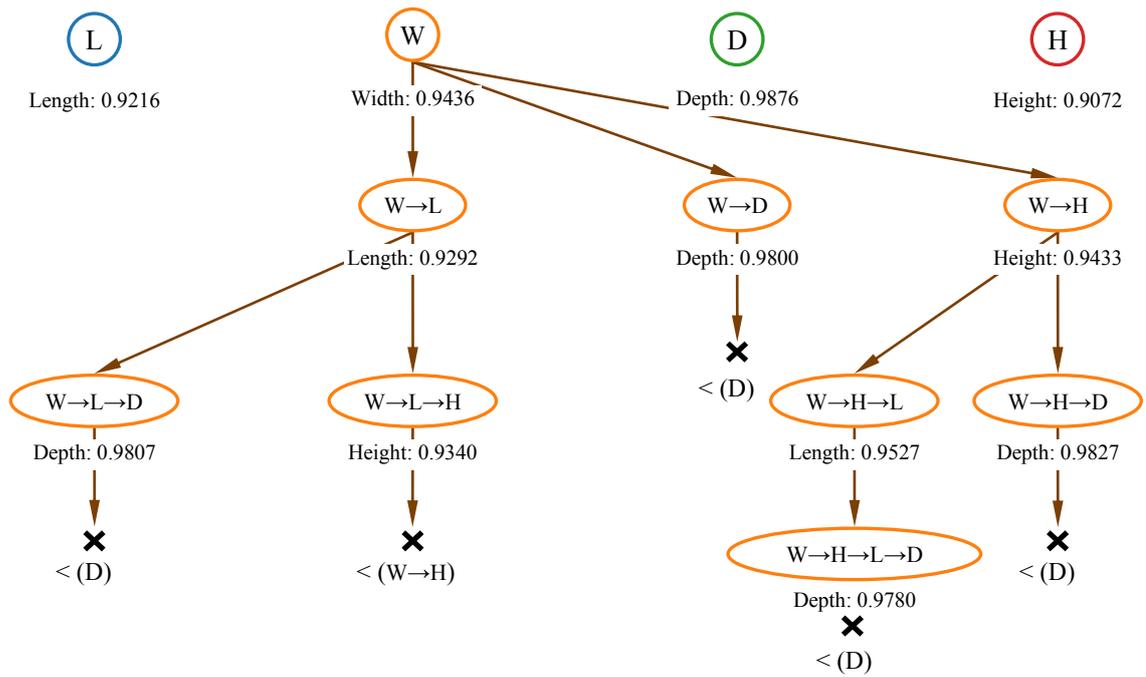


Figure 5.12 GBFS outcome using SVM as the classification algorithm with width as the root node of the search tree.

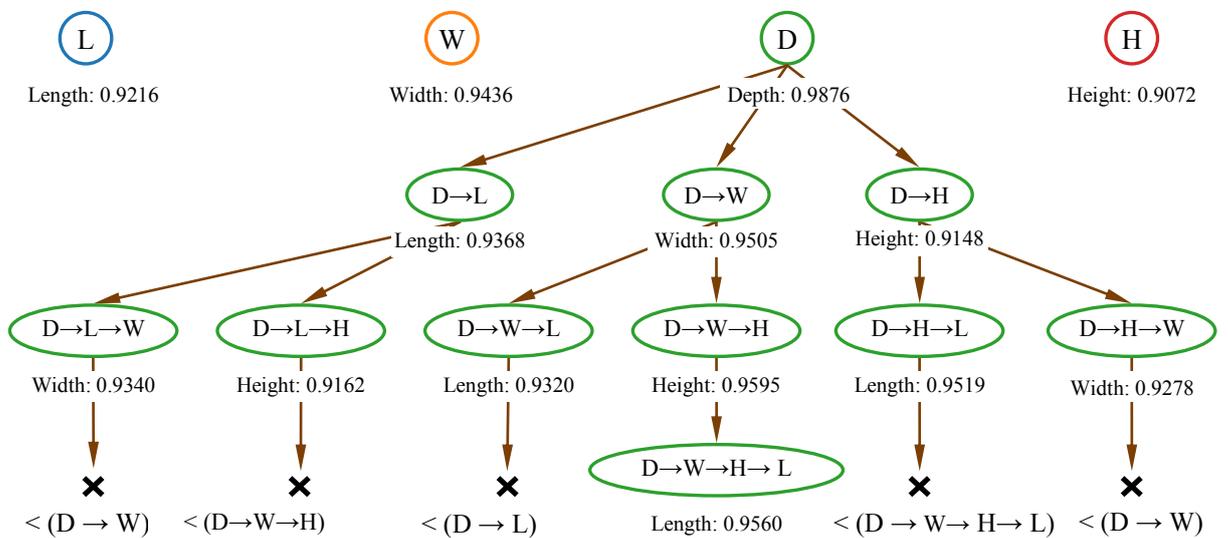


Figure 5.13 GBFS outcome using SVM as the classification algorithm with depth as the root node of the search tree.

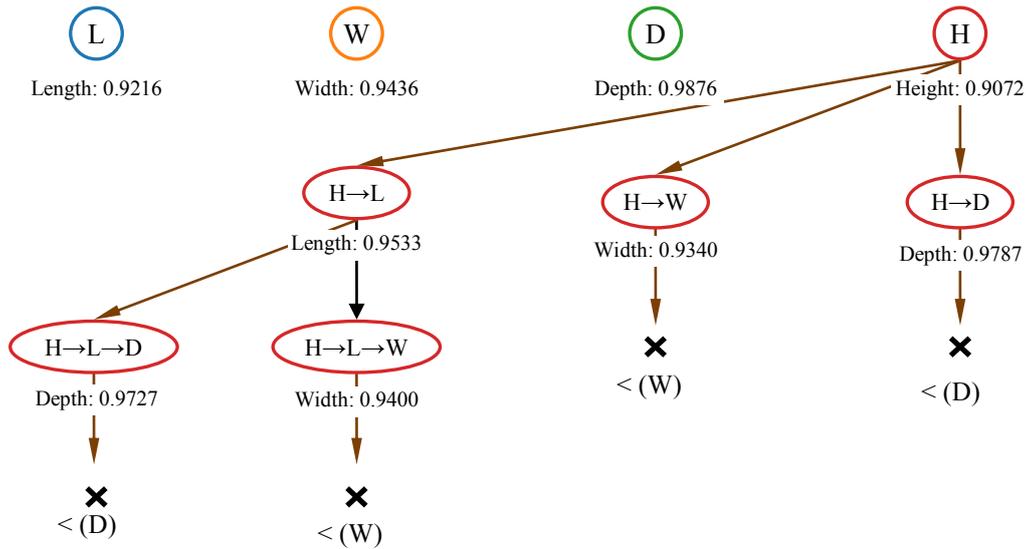


Figure 5.14 GBFS outcome using SVM as the classification algorithm with height as the root node of the search tree.

The total number of nodes expanded by the GBFS for SVM is 27. Table 5.3 summarizes the sequences using SVM that improved the performance of the chain classification over the independent classification.

Table 5.3 Performance of SVM in chain sequence with different defect characteristics as root node of the GBFS.

SEQUENCE	MEAN ACCURACY
D→W→H→L	0.9622
D→W→H	0.9547
W→H→L	0.9510
D→H→L	0.9494
W→H	0.9490
W→L→H	0.9486
H→L	0.9479
D→L→H	0.9460
D→W→L	0.9442
D→L	0.9438
W→L	0.9419
D→H	0.9419
D→W	0.9417
Independent	0.9400

Comparison of total number of nodes expanded for RBF neural network and SVM with each class variable as root node is given in Table 5.4. In total, the GBFS has terminated with

expansion of 25 nodes for RBF neural network while SVM terminated with 27 nodes. This includes 1 node expansion for independent classification. It would have required 61 expansions using either of the algorithms in brute force without greedy.

Table 5.4 Comparison of number of node expansion required and number of nodes expanded by GBFS.

CLASS VARIABLE OF ROOT NODE	NUMBER OF NODE EXPANSIONS REQUIRED	NUMBER OF NODES EXPANDED FOR GBFS	
		RBF NEURAL NETWORKS	SVM
Length	15	8	3
Width	15	5	8
Depth	15	8	10
Height	15	3	5
Total	60	24	26

Table 5.5 summarizes the sequences that improved the performance of the chain classification over the independent classification by either of the algorithm.

Table 5.5 Comparison of performance of RBF neural networks and SVM in chain sequence with different defect characteristics as root node of the GBFS.

DEFECT CHARACTERISTICS	SEQUENCE	MEAN ACCURACY	
		RBF NEURAL NETWORKS	SVM
	Independent	0.9150	0.9400
Length	L→W	0.9153	-
	L→W→H	0.9244	-
	L→H	0.9205	-
Width	W→L	-	0.9419
	W→H	0.9261	0.9490
	W→L→H	-	0.9486
	W→H→L	-	0.9510
Depth	D→L	0.9153	0.9438
	D→W	0.9163	0.9417
	D→H	0.9201	0.9419
	D→L→H	0.9209	0.9460
	D→W→L	-	0.9442
	D→W→H	0.9303	0.9547
	D→W→H→L	0.9306	0.9622
	D→H→L	-	0.9494
Height	H→L	-	0.9479

Interestingly, many sequences listed in Table 5.5, have improved the accuracy over the independent classification for RBF neural network as well as SVM. This can be interpreted as the class variables are dependent on one another and they cannot be isolated. However, D→W→H→L sequence has produced the best results with a mean accuracy of 0.9303 and 0.9622 for RBF neural networks and SVM, respectively.

Table 5.6 compares the performance of the chain classification and independent classification algorithms. It can be observed that the classification for length and width have produced a marginal improvement in accuracy to 0.9807 from 0.9793, and to 0.9073 from 0.9020 respectively, using the RBF neural network. Accuracy has increased to 0.9560 from 0.9216 and to 0.9505 from 0.9436 for length and width, respectively, using SVM.

Table 5.6 Comparison of performance of independent as well as chain classification.

LEARNING ALGORITHM	ACCURACY			
	LENGTH	WIDTH	DEPTH	HEIGHT
Independent RBF neural networks	0.9793	0.9020	0.9207	0.8580
Chain RBF neural networks (D→W→H→L)	0.9807	0.9073	0.9207	0.9140
Independent SVM	0.9216	0.9436	0.9876	0.9072
Chain SVM (D→W→H→L)	0.9560	0.9505	0.9876	0.9595

It can also be noted that the accuracy in length sizing is 0.9807 using RBF neural network is greater than that of the SVM (0.9560). Substantial improvement in classification has been achieved for height by both the classification algorithms. For height sizing the accuracy has improved to 0.9140 from 0.8580 for the RBF neural network and to 0.9595 from 0.9072 for the SVM. Thus, the results clearly establish, for the first time, that the chain classification results in superior performance for defect sizing as compared to independent classification. Following these observations, it can be stated that the SVM is better for classification of depth, width, and height while RBF neural network is better for classification of length. A hybrid chain classifier following D→W→H→L sequence consisting SVM for depth, width, and height and RBF neural network for length is able to produce the highest mean accuracy of 0.9694.

The optimal sequence D→W→H→L can also be explained from the eddy current test phenomenon. Defect depth influences the probe impedance change (magnitude and phase) more predominantly than that of length, width, or height. By determining the class of depth first, the algorithm has the tendency to classify more accurately the width, height, and length of the defect. The sequence can be understood and explained based on *skin-effect* phenomenon. The depth is most influenced by the *skin-effect* and the features chosen are sufficient enough to give accurate results. This is not in the case with length, width, and height. Only when accurate and the most influencing depth information is given as input, length and width can be determined accurately.

5.3.3 Chain classification on test data

The performance of chain classification has been compared with independent classification using experimentally obtained images. Table 5.7 and Table 5.8 show the performance of the chain classification on experimental images using RBF neural network and SVM respectively. The predicted value for depth and height are the midpoint of the class ranges (refer Table 3.3). As can be observed from Table 5.7 and Table 5.8, the results of the chain classification closely agree with the actual value for all the four defect characteristics, measured using optical method with an accuracy of 0.05 mm.

Table 5.7 Performance of the RBF neural network in chain classification.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-1	25.0	25.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-2	25.0	25.0	2.0	2.0	3.0	3.0	2.0	2.8
Defect-3	25.0	25.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-4	30.0	30.0	2.0	2.0	3.0	3.0	2.0	2.8
Defect-5	30.0	30.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-6	30.0	30.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-7	35.0	35.0	2.0	2.0	3.0	3.0	2.0	1.0
Defect-8	30.0	30.0	2.0	2.0	1.0	0.8	4.0	4.3
Defect-9	35.0	35.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-10	25.0	25.0	2.0	2.0	0.0	0.0	2.0	1.0

Table 5.8 Performance of the SVM in chain classification.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-1	25.0	25.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-2	25.0	25.0	2.0	2.0	3.0	3.0	2.0	2.8
Defect-3	25.0	25.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-4	30.0	30.0	2.0	2.0	3.0	3.0	2.0	2.8
Defect-5	30.0	30.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-6	30.0	30.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-7	35.0	35.0	2.0	2.0	3.0	3.0	2.0	1.0
Defect-8	30.0	30.0	2.0	2.0	1.0	0.8	4.0	4.3
Defect-9	35.0	35.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-10	25.0	25.0	2.0	2.0	0.0	0.0	2.0	1.0

Table 5.9 and Table 5.10 report the performance of the independent classifiers on the experimental images using the RBF neural network and SVM respectively. Independent classification using RBF neural network has misclassified height of **Defect-1** and **Defect-4**, and width of **Defect-7**, while SVM has misclassified height of **Defect-1**, and width of **Defect-4** and **Defect-7**. It is noteworthy that Defect-1, Defect-4, and Defect-7 are deeper (located 3.0 mm below the surface) among all the defects considered. This success of chain classification is attributed to the influence of dependency and its effective incorporation by the chain classification.

Table 5.9 Performance of RBF neural network using independent classification.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-1	25.0	25.0	2.0	2.0	2.0	2.0	3.0	1.0
Defect-2	25.0	25.0	2.0	2.0	3.0	3.0	2.0	1.8
Defect-3	25.0	25.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-4	30.0	30.0	2.0	3.0	3.0	3.0	2.0	1.0
Defect-5	30.0	30.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-6	30.0	30.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-7	35.0	35.0	2.0	1.0	3.0	3.0	2.0	2.8
Defect-8	30.0	30.0	2.0	2.0	1.0	0.8	4.0	4.3
Defect-9	35.0	35.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-10	25.0	25.0	2.0	2.0	0.0	0.0	2.0	1.0

Table 5.10 Performance of SVM using independent classification.

DEFECT	LENGTH, mm		WIDTH, mm		DEPTH, mm		HEIGHT, mm	
	Actual	Predicted	Actual	Predicted	Actual	Predicted	Actual	Predicted
Defect-1	25.0	25.0	2.0	2.0	2.0	2.0	3.0	1.0
Defect-2	25.0	25.0	2.0	2.0	3.0	3.0	2.0	2.8
Defect-3	25.0	25.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-4	30.0	30.0	2.0	1.0	3.0	3.0	2.0	2.8
Defect-5	30.0	30.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-6	30.0	30.0	3.0	3.0	2.0	2.0	3.0	2.8
Defect-7	35.0	35.0	2.0	1.0	3.0	3.0	2.0	1.0
Defect-8	30.0	30.0	2.0	2.0	1.0	0.8	4.0	4.3
Defect-9	35.0	35.0	2.0	2.0	2.0	2.0	3.0	2.8
Defect-10	25.0	25.0	2.0	2.0	0.0	0.0	2.0	1.0

5.3.4 Testing the robustness of the chain classification

Following the *skin effect*, the exponential fall of eddy currents is expected to influence the feature values substantially. Using the trained chain classifiers with subsurface defects, the robustness of the chain classification is tested using EC images of 3 surface defects (depth: 0.0 mm) and 3 near-surface defects (depth: 0.2 mm). The heights of the defects have been taken as 1.0, 2.0, and 3.0 mm, while length and width of the defects have been kept constant at 30.0 mm and 2.0 mm, respectively. The EC images of a typical surface and the near surface defects at 5 kHz are shown in Figure 5.15. The extracted features for these defects are given in Table 5.11.

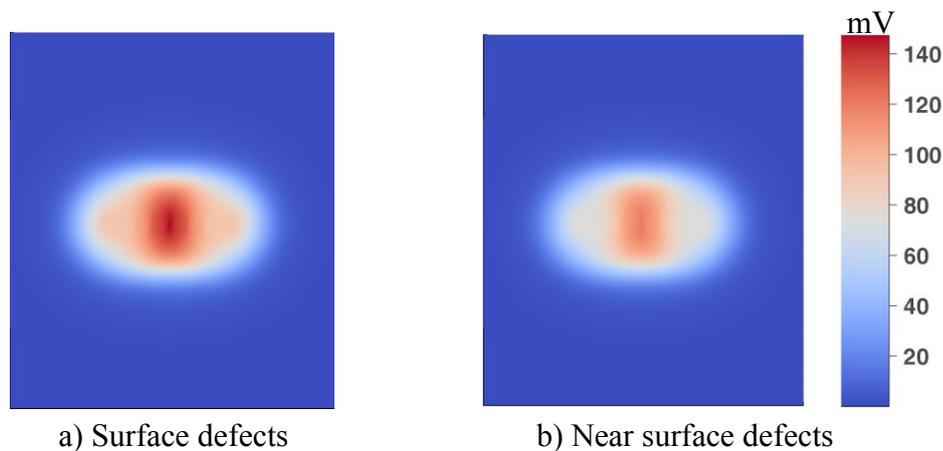


Figure 5.15 EC images of a) surface and b) near-surface defect (length: 30.0 mm, width: 2.0 mm and height: 1.0 mm) at 5 kHz.

Table 5.11 Extracted features for a surface defect (depth: 0.0 mm) and near surface defect (depth: 0.2 mm).

FEATURE	5 kHz		FEATURE	10 kHz	
	DEPTH: 0.0 mm	DEPTH: 0.2 mm		DEPTH: 0.0 mm	DEPTH: 0.2 mm
F1	23.0	23.0	F10	22.0	22.0
F2	12.0	12.0	F11	10.0	10.0
F3	34.0	34.0	F12	34.0	34.0
F4	1104.0	1104.0	F13	968.0	1012.0
F5	432.0	480.0	F14	432.0	432.0
F6	0.0	0.0	F15	0.0	0.0
F7	147.0	119.0	F16	401.0	321.0
F8	-0.8	-0.8	F17	2.7	2.7
F9	7.1	7.1			

Chain classification with RBF neural networks and SVM has successfully classified depth and height as class D1 and H1, respectively. This step has been repeated for other 10 defects and correct classification has been observed for all the 10 defects. Thus, the chain classification can be used to size both surface and subsurface defects.

5.4 SUMMARY

Chain classification methodology has been proposed to incorporate dependency among the class variables (length, width, depth, and height); this otherwise assumed to be independent for efficient defect sizing from eddy current images. For optimization of the best sequence to be used in the chain, a new algorithm called greedy breadth-first-search (GBFS) has been proposed. GBFS with RBF neural network and SVM have been found to be capable of rapid sequence optimization for chain classification.

Chain classification has successfully resulted in sizing of defects located even 3.0 mm below the surface, from the response obtained through numerical modelling as well as through experimental measurements. $D \rightarrow W \rightarrow H \rightarrow L$ is found to be the optimal sequence, for the first time, by both RBF neural network and SVM. The same sequence determination by two diverse algorithms strongly indicates the dependency structure during sizing. Chain classification is able

to successfully classify, depth and height of the surface as well as near surface defects, confirming its robustness. This sequence can be justified based on *skin-effect*. The depth is the most influenced by the *skin-effect* and hence, once the depth is estimated other characteristics of defects can be sized with better accuracy. Length is usually larger than the probe size, hence least perturbed in the eddy currents and also least influenced by *skin-effect*. The influence of width is dependent on the size of the probe. For a small probe or defects larger than the effective size of the probe, width is expected to behave as length with least influence. In this study, the probe size is larger than the width and hence, it has least influence on length and height.

Studies have been shown that the SVM is better for classification of depth, width, and height while RBF neural network is better for classification of length.

Experimental studies on various fabricated defects confirm the effectiveness and efficiency of the chain classification, indicating that it is a promising method for accurate sizing of defects. Even though, chain classification addresses dependency, it requires explicit optimal sequence determination and also requires several learning algorithms.

6

GENERALIZED WRAPPER FRAMEWORK TO INCORPORATE MULTILABEL ALGORITHMS FOR DEFECT SIZING

As discussed in Chapter 3 and Chapter 5, defect sizing requires multiple class variables and each class variable is a multiclass classification problem (multidimensional learning). Multilabel learning is a recent concept in machine learning which also deals with multiple class variables. However, they are confined to binary classification. In order to utilize the multilabel learning algorithms a generalized wrapper framework is proposed. This chapter introduces the concept of the proposed generalized wrapper framework and provides results of the framework applied to real world benchmark multilabel dataset as well as for eddy current inversion problem.

6.1 MULTILABEL LEARNING

6.1.1 *Multilabel classification*

Multilabel learning studies the learning problem where each input is associated with a set of classes simultaneously. The training dataset is assume to be associated with more than one classes simultaneously and the task is to assign all the associated class from a set for each unseen instance [73]. Multilabel classification can be formally defined as a function $g(\cdot)$ which assigns

a subset of class from a set, e.g. $C = \{1, 2, \dots, d\}$ to each unseen m dimensional vector of feature $\mathbf{x} \in \mathbf{X}$, given a multilabel training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{c}_i) | 1 \leq i \leq n\}$ given with $\mathbf{c} \subset C$:

$$g(\mathbf{x}): \phi_{x_1} \times \phi_{x_2} \times \dots \times \phi_{x_m} \rightarrow 2^C$$

where ϕ_{x_j} denoting the sample space of feature variable x_j for $j=1, 2, \dots, m$. The mapping function $g(\cdot)$ is generally built by considering the dependency between each of class in class space C . Multilabel classification problems mostly solved using multilabel ranking.

6.1.2 **Multilabel ranking**

Multilabel ranking is a problem of learning a function $h(\cdot, \cdot)$ which produces a vector of real numbers with size $d=|C|$ as output to each unseen instance of a feature vector $\mathbf{x} \in \mathbf{X}$ for a training set $\mathbf{D} = \{(\mathbf{x}_i, \mathbf{c}_i) | 1 \leq i \leq n\}$ given by

$$h(\mathbf{x}, \mathbf{l}): \phi_{x_1} \times \phi_{x_2} \times \dots \times \phi_{x_m} \times C \rightarrow \mathbb{R}^d$$

$f_2(\mathbf{x}, \mathbf{l}): \phi_{x_1} \times \phi_{x_2} \times \dots \times \phi_{x_m} \times C \rightarrow \mathbb{R}^d$ where \mathbb{R} is the set of real numbers. The function $h(\cdot, \cdot)$ is expected to have the property of ordering the set of labels C , so that the topmost labels are more related to the unknown input. Most of the multilabel classification problems deal with post processing of multilabel ranking using a threshold function. With multilabel ranking, labels which are above the threshold for a given unknown instance are considered as the classes of that instance, such that

$$g(\mathbf{x}) = \{c' | h(\mathbf{x}, c') > t(\mathbf{x}), c' \in C\} \quad (6.1)$$

where $t(\cdot)$ is threshold function which can be a constant function [74,75] or stacking method [76] or algorithm specific threshold [77]. However, multidimensional learning problem cannot be solved using conventional multilabel classification because threshold based multilabel classification has two undesired possibilities [78]:

- None of the class in a given dimension exceeds threshold.
- More than one class in a given dimension exceeds threshold.

The brief characteristics of multilabel learning are the following:

- Size of classes has to be classified is unknown *a priori* and vary with each input instance of data.
- Dependency exists within input features, between features and classes as well as within the classes.

Since the size of the predicted classes is unknown *a priori* in multilabel learning problems, there are possibilities that output for some of dimensions are not predicted, if they are considered for multidimensional learning. Numerous multilabel learning algorithms were proposed in the literature which takes care of the above mentioned properties including dependency [73,79]. Since, the size of the predicted classes in multilabel learning vary with each input instance of data, multilabel learning can predict more than one class in a dimension for multidimensional problems as described in Section 3.1. Hence, multilabel learning cannot be directly used for multidimensional learning. In order to utilise the multilabel classification for multidimensional classification which can be further used for defect sizing a framework is propose in this thesis.

6.2 WRAPPER FRAMEWORK FOR MULTIDIMENSIONAL SUPERVISED LEARNING

The proposed framework for multidimensional learning (MDLearn) is a wrapper method. Figure 6.1 gives the schematic of the wrapper framework. Consider a d dimension domain of class space \mathbf{C} containing all possible classes in each dimension, such that $\mathbf{C} = \{C^1 = \{c_1^1, c_2^1, \dots, c_{k_1}^1\}, C^1 = \{c_1^2, c_2^2, \dots, c_{k_2}^2\}, \dots, C^D = \{c_1^D, c_2^D, \dots, c_{k_D}^D\}\}$ and multilabel representation can be achieved by treating each possible class in each dimension as an individual class in a single dimension such that $\psi = \bigcup_{i=1}^D C^i$ that is $\psi = \{c_1^1, c_2^1, \dots, c_{k_1}^1, c_1^2, c_2^2, \dots, c_{k_2}^2, \dots, c_1^d, c_2^d, \dots, c_{k_d}^d\}$ and $K = |\psi| =$

$\sum_{d=1}^D k_d K = |\psi| = \sum_{i=1}^d k_i$. Now, multilabel learning can be performed on ψ as a class space instead of C . A vector $\mathbf{k} = \{k_1, k_2, \dots, k_d\}$ retains the cardinality of each dimension.

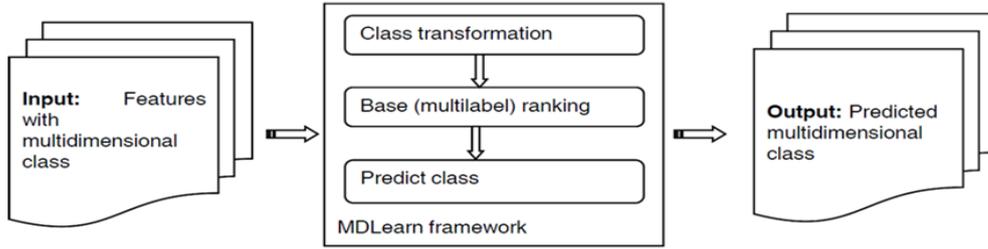


Figure 6.1 A wrapper framework for multidimensional supervised learning.

Algorithm 6.1 gives a pseudocode for the class transformation phase of the MDLearn wrapper. This algorithm codes as +1 for true class and -1 for all other classes in a given dimension, hence there will be d true classes.

Algorithm 6.1 MDLearn wrapper - class transformation.

$\psi = \text{Transform_Class}(\mathbf{c}, \mathbf{k})$

Input:

$\mathbf{c} = \{c_1, c_2, \dots, c_d\}$, an instance of d dimensional index of true class at each dimension

$\mathbf{k} = \{k_1, k_2, \dots, k_d\}$, $k_i \leftarrow \begin{matrix} \dots \\ \dots \\ \dots \end{matrix}$ for $1 \leq i \leq d$, an integer vector comprises cardinality of each dimension

Output:

ψ , multilabel class representation

Process:

$\psi \leftarrow \text{NULL}$

for $i \leftarrow 1$ to d **do** // each dimension of C

for $j \leftarrow 1$ to k_i **do** // each class in ϕ_i

if $c_i = j$ **then** $c' \leftarrow +1$

else $c' \leftarrow -1$

$\psi \leftarrow \{\psi, c'\}$

end

end

As a second stage of wrapper, any multilabel ranking algorithm [73] which considers the class dependency property can be applied as a base ranking method. In this thesis, evaluation of this

proposed framework is carried out using multilabel radial basis function (ML-RBF) and multilabel k-nearest neighbours (ML-KNN) for comparison purpose [74,75].

For a successfully learned system, it is claimed to exhibit the property $R(c_j^a) > R(c_k^a)$, for every $k \neq j$ in a given dimension a , where $R(\cdot)$ is the real valued rank of a class obtained through $h(\cdot, \cdot)$. Hence, a multidimensional function $f(\mathbf{x}_i)$ can be derived from $h(\mathbf{x}_i, \boldsymbol{\psi}_i)$, a multilabel ranking algorithm which produces real valued output for all the classes such that :

$$f(\mathbf{x}_i) = \{\hat{\mathbf{c}} = \{\hat{c}^1, \hat{c}^2, \dots, \hat{c}^D\}\}$$

$$\hat{c}^a = c_j^a, \text{ where } R(c_j^a) > R(c_k^a), k \neq j, \{c_j^a, c_k^a\} \in C^a \text{ for } 1 \leq a \leq d \quad (6.2)$$

A pseudocode for multidimensional representation and prediction of classes based on this is given in Algorithm 6.2.

Algorithm 6.2 MDLearn wrapper - class prediction.

$\mathbf{c}' = \text{Predict_Class}(\boldsymbol{\psi}', \mathbf{k})$

Input:

$\boldsymbol{\psi}'$, a vector of predicted rank of class obtained through multilabel ranking

$\mathbf{k} = \{k_1, k_2, \dots, k_d\}, k_i \leftarrow \dots$ for $i = \{1, 2, \dots, d\}$, an integer vector

comprises cardinality of each dimension

Output:

$\mathbf{c}' = \{c'_1, c'_2, \dots, c'_d\}$ an instance of d dimensional predicted class vector

Process:

$j \leftarrow 1$

foreach i in k_i **do**

$h \leftarrow j + k_i - 1$

$c'_i \leftarrow \text{class}(\text{index}(\max(\boldsymbol{\psi}'_{j..to.h})))$ //max is the regular maximum function

$j \leftarrow h + 1$

end

The algorithm takes two vectors. The first vector is the predicted rank for each class in $\boldsymbol{\psi}$ obtained through multilabel ranking algorithm. The second vector is \mathbf{k} which comprises

cardinality of each dimension. For each dimension, a class with maximum rank among other classes is declared as the predicted class for that particular dimension.

6.3 DATASET DESCRIPTION

Performance evaluation has been carried out using two multilabel ranking algorithms, i) ML-RBF neural network and ii) ML-KNN, on the proposed framework and they are termed as $MDLearn_{ML-RBF}$ and $MDLearn_{ML-KNN}$ respectively. There are no benchmark multidimensional datasets available. Hence, before applying the proposed framework to eddy current data, for validation of assumptions made in building the wrapper framework, studies have been carried out using real world benchmark multilabel dataset (by casting it into multidimensional model). Results have been compared against the same base ranking algorithm (ML-RBF neural network and ML-KNN) with appropriate threshold function for assigning the classes, as commonly used in multilabel learning. Subsequently, studies on the proposed framework have been carried out using multidimensional dataset of defect sizing. EC images of 300 subsurface defects as described in Section 3.3.1.4 have been used for defect sizing using the MDLearn framework.

The real world datasets available at <http://mulan.sourceforge.net/datasets.html> has been used for validation of the assumptions. The description of the datasets is the following:

1. YEAST: The yeast data consists of genes described by concatenation of micro-array expression data and phylogenetic profile of the Yeast *saccharomyces cerevisiae*, which is one of the best studied organisms. Each gene is organized into 103 features. Elisseeff and Weston [80] preprocessed the dataset where only the known structure of the functional classes belonging to the top level of hierarchy were retained. This dataset consists of 2417 genes and 14 possible functional classes for each gene. The task is to predict the set of functional classes associated with each gene among the 14 functional classes.

2. **EMOTIONS:** The dataset for emotion classification was collected by Trohidis *et al* [81]. It consists of 100 songs from each of the following 7 different genres: classical, reggae, rock, pop, hip-hop, techno and jazz. From each song a period of 30 seconds after the initial 30 seconds was extracted and consists of 593 sound clips among which 72 features were extracted. The Tellegen-Watson-Clark model was employed for labelling the data with 8 emotions viz. amazed-surprised, happy-pleased, relaxing-calm, quit-still, sad-lonely and angry-aggressive. The task is to predict the set of classes for each sound clip among the 8 classes.
3. **SCENE:** In natural scene classification, each natural scene image may belong to several image types (classes) simultaneously, for example the image shown in Figure 6.2a can be classified as a mountain as well as a foliage, while the image shown in Figure 6.2b can be classified as a sea as well as a mountain. Through analyzing images with known label sets, a multilabel learning system will automatically predict the sets of labels for unseen images. The above process of semantic scene classification can be applied to many areas, including content-based indexing, organization, and image enhancement. The dataset of images used for scene classification was preprocessed by Boutell *et al* [82] containing 2407 pictures and each picture is associated with a set of semantic classes among six classes (beach, sunset, foliage, field, mountain and urban) and each picture is represented with 294 attributes based on colour information of images.



a) Class labels: mountain, foliage



b) Class labels: mountain, sea

Figure 6.2 Examples of multilabel images.

6.4 RESULTS AND DISCUSSION

6.4.1 *Real world benchmark multilabel data*

Table 6.1 summarizes the results of the evaluation metrics viz. mean accuracy, global accuracy, and entropy of accuracy. Results are interpreted as mean \pm std. deviation of 10 independent runs as conventionally carried out to compare the machine learning algorithms. It has been assumed that the independent runs are Gaussian in nature. Performance of all evaluation metrics (refer Section 3.3.3) based on pair wise t -test at 5% significant level has been carried out and the results are given in Table 6.2. $A > B$ represents performance of A is statistically better than B and $A \approx B$ represents performance of A is statistically similar to B. From the results obtained from Table 6.1 and further analysis of two-tailed paired t -test in Table 6.2 on the benchmark multilabel dataset, it is evident that the proposed wrapper framework is able to produce identical results with that of the threshold based multilabel ranking algorithm. Hence, it can be stated that all the assumptions made in design of the framework hold good.

Table 6.1 performance evaluation (mean \pm std. deviation) for ten-fold cross validation on benchmark multilabel dataset.

DATASET	MDLEARN _{ML-RBF}	ML-RBF	MDLEARN _{ML-KNN}	ML-KNN
MEAN ACCURACY				
Yeast	0.7946 \pm 0.0021	0.8050 \pm 0.0013	0.7762 \pm 0.0015	0.8047 \pm 0.0004
Scene	0.9218 \pm 0.0014	0.9133 \pm 0.0009	0.8888 \pm 0.0011	0.8885 \pm 0.0010
Emotion	0.7817 \pm 0.0036	0.7640 \pm 0.0020	0.7181 \pm 0.0024	0.7418 \pm 0.0047
ENTROPY ACCURACY				
Yeast	0.7926 \pm 0.0022	0.8032 \pm 0.0013	0.7739 \pm 0.0015	0.8029 \pm 0.0004
Scene	0.9212 \pm 0.0014	0.9128 \pm 0.0009	0.8879 \pm 0.0012	0.8875 \pm 0.0011
Emotion	0.7808 \pm 0.0036	0.7633 \pm 0.0020	0.7173 \pm 0.0024	0.7414 \pm 0.0047
GLOBAL ACCURACY				
Yeast	0.1664 \pm 0.0041	0.1820 \pm 0.0022	0.1118 \pm 0.0070	0.1799 \pm 0.0027
Scene	0.6369 \pm 0.0063	0.5988 \pm 0.0055	0.4398 \pm 0.0025	0.6292 \pm 0.0034
Emotion	0.2511 \pm 0.0069	0.2077 \pm 0.0092	0.2029 \pm 0.0030	0.1333 \pm 0.0123
HAMMING LOSS				
Yeast	0.2053 \pm 0.0021	0.1949 \pm 0.0013	0.2819 \pm 0.0024	0.1952 \pm 0.0004
Scene	0.0782 \pm 0.0014	0.0866 \pm 0.0009	0.2010 \pm 0.0014	0.0867 \pm 0.0010
Emotion	0.2182 \pm 0.0036	0.2360 \pm 0.0020	0.1111 \pm 0.0011	0.2581 \pm 0.0047

Table 6.2 Overall performance of all evaluation metrics based on two-tailed paired *t*-test at 5% significant level (MDLearn_{ML-RBF} \rightarrow A, ML-RBF \rightarrow B, MDLearn_{ML-KNN} \rightarrow C, ML-KNN \rightarrow D)

DATASET	MEAN ACCURACY	ENTROPY ACCURACY	GLOBAL ACCURACY	HAMMING LOSS
Yeast	A \asymp B, C \asymp D	A \asymp B, C \asymp D	A \asymp B, C $<$ D	A \asymp B, C $<$ D
Scene	A \asymp B, C \asymp D	A \asymp B, C \asymp D	A \asymp B, C $<$ D	A \asymp B, C $<$ D
Emotion	A \asymp B, C \asymp D	A \asymp B, C \asymp D	A $>$ B, C $<$ D	A \asymp B, C $>$ D

Further, the results obtained for real world dataset has been compared with the results published previously in [74,75,67] using ML-RBF neural network and ML-KNN and the results are given in Table 6.3. It is observed that the results obtained were consistent or slightly better than the previously published results.

Table 6.3 Comparison of results for hamming loss reported in [67,74,75] and proposed framework.

DATASET	ML-RBF NEURAL NETWORK		ML-KNN	
	RESULTS REPORTED IN [74]	RESULTS OBTAINED	RESULTS REPORTED IN [67,75]	RESULTS OBTAINED
Yeast	0.1950 \pm 0.0110	0.1949 \pm 0.0013	0.1950 \pm 0.0100	0.1952 \pm 0.0004
Scene	-	0.0866 \pm 0.0009	0.1804 \pm 0.0092	0.0867 \pm 0.0010
Emotion	-	0.2360 \pm 0.0020	0.3867 \pm 0.0169	0.2581 \pm 0.0047

6.4.2 Eddy current testing data

The proposed framework has been applied to eddy current data of subsurface defects as discussed in Section 3.3.1.4. Table 6.4 gives the performance of MDLearn framework as well as conventional uni-dimensional variant of the same learning algorithm for 10-fold cross validation of individual dimensions (defect characteristics).

Table 6.4 Accuracy of MDLearn and conventional uni-dimensional algorithm for individual dimension of EC data.

ALGORITHMS	PERFORMANCE EVALUATION - ACCURACY			
	LENGTH	WIDTH	DEPTH	HEIGHT
MDLearn _{ML-RBF}	0.9807 ± 0.010	0.9118 ± 0.013	0.9207 ± 0.010	0.9072 ± 0.011
RBF	0.9793 ± 0.080	0.9020 ± 0.080	0.9207 ± 0.060	0.8580 ± 0.083
MDLearn _{ML-KNN}	0.8774 ± 0.008	0.7158 ± 0.011	0.8239 ± 0.006	0.8353 ± 0.013
KNN	0.8737 ± 0.007	0.6745 ± 0.013	0.8180 ± 0.005	0.8175 ± 0.080

It can be observed from Table 6.4 that all the characteristics of defects considered such as length, width, depth, and height could be classified with considerable accuracy. Further, it can be noted that the depth and length are relatively easier to classify than that of the height and width. The results have been evaluated using multidimensional evaluation metrics such as mean accuracy, global accuracy, and entropy of accuracy and given in Table 6.5.

Table 6.5 Performance of MDLearn and conventional uni-dimensional algorithm for EC data.

ALGORITHMS	PERFORMANCE EVALUATION		
	MEAN ACCURACY	GLOBAL ACCURACY	ENTROPY OF ACCURACY
MDLearn _{ML-RBF}	0.9300 ± 0.004	0.7947 ± 0.012	0.9288 ± 0.004
RBF	0.9144 ± 0.004	0.7265 ± 0.006	0.9112 ± 0.004
MDLearn _{ML-KNN}	0.8120 ± 0.004	0.5433 ± 0.008	0.8102 ± 0.005
KNN	0.7925 ± 0.004	0.3859 ± 0.009	0.7912 ± 0.005

As can be observed from Table 6.5, the MDLearn framework under ML-RBF neural network for four defect characteristics viz, length, width, depth, and height has produced 93% mean accuracy, 79% global accuracy and 92% entropy of accuracy, when compared to that of the conventional RBF neural networks, which produced 91% mean accuracy, 72% global and 91%

entropy of accuracy. Being a naïve algorithm, KNN has produced a modest 79% mean accuracy. However, this has been improved to 81% by MDLearn framework using ML-KNN.

It can be inferred from Table 6.4 and Table 6.5 that the MDLearn using ML-RBF neural network and ML-KNN has given better results when compared to their uni-dimensional counterparts, RBF and KNN respectively. This is directly attributed to the fact that statistical dependency exists between each defect characteristic. ML-RBF neural network as base ranking for MDLearn framework has given the maximum performance, as compared to the other three algorithms. The entropy of accuracy in Table 6.5 remains equal with a small margin to the mean accuracy in all the algorithms, with which it can be interpreted that the accuracy is not biased towards any particular defect characteristic. It can be noted that there is a significant improvement in terms of global accuracy with MDLearn_{ML-RBF} as compared to the conventional RBF neural network. Hence, with MDLearn_{ML-RBF} the probability of getting accurate classification across all the dimensions is high. It is evident from Table 6.4 and Table 6.5 that proposed MDLearn framework is capable of producing efficient results for true multidimensional dataset (≥ 3 class in each dimension) and superior than uni-dimensional learning. Thus, the MDLearn framework enables simultaneous prediction of length, width, depth, and height of a defect from its EC images from two frequencies. This framework can be used to other NDE techniques such as radiography techniques for defect sizing and ultrasonic testing for defect classification.

6.5 SUMMARY

In order to utilize the multilabel learning algorithms, a generalized wrapper framework is proposed, for the first time. In this framework, input multidimensional data are represented in the form of multilabel data. Later, class for each dimension is predicted based on rank of class present in each dimension assuming true class gives maximum rank with multilabel ranking. This framework has been applied to real world benchmark multilabel dataset viz. yeast, scene,

and emotion for validation of assumptions made in building the wrapper framework. The results produced by the proposed framework have been found to be consistent or better than the previously published results using real world benchmark multilabel dataset (by casting it into multidimensional model). This has validated the assumptions of the framework.

Proposed framework has been applied for defect sizing using EC images. It has been observed that the framework using ML-RBF neural network has achieved a significant improvement in global accuracy to 79% for four dimensions (4 defect characteristics viz, length, width, depth, and height). In contrast, the conventional uni-dimensional RBF neural network has been able to produce 72% global accuracy. It can also be stated that the MDLearn framework enables simultaneous prediction of length, width, depth, and height of a defect from its EC images. This framework can be applied to other NDE techniques for defect classification and sizing.

CONCLUSION

The thesis has focused on development of novel machine learning algorithms for defect sizing from eddy current (EC) images of metallic materials. The major conclusions of the thesis are the following:

A new multidimensional radial basis function (MD-RBF) neural network has been proposed. It is capable of sizing simultaneously length, width, depth, and height when all the input features are fed together, through a single training. It overcomes the limitation of conventional machine learning algorithms including ANN and RBF neural networks that require the selective input for each characteristic of a defect and a separate neural network for each defect characteristic. MD-RBF also incorporates dependency implicitly among the defect characteristics.

EC images have been segmented by identifying a ROI around the defect response. Features are extracted from EC images and studied for their relevance to defect characteristics. It has been established that the extracted features show good correlation with defect characteristics. The features have been found to be noise tolerant up to 15% additive white Gaussian noise in EC images. It has been also found that these extracted features are rotation invariant. MD-RBF neural network has been trained with EC images obtained from numerical modeling of rectangular defects while evaluated using EC experiments.

With MD-RBF neural networks, there is a significant improvement in the sizing performance. Global accuracy increased to 83% with a relatively less number of computations as compared to the global accuracy of 76% achievable by the conventional RBF neural networks. Significant

improvement in sizing height has been observed for MD-RBF neural network (94%) than RBF neural networks (90%). The MD-RBF neural network has also been evaluated to defects which are non-rectangular in shape and found that the results are consistent. The MD-RBF neural network results in approximated length, width, depth, and height when input is features of real crack or other shape.

The performance of MD-RBF neural network with change in scan pitch, for the chosen probe, has been studied, for the first time. From this study, it has been identified that the proposed MD-RBF neural network is robust for defect sizing for EC images obtained even at certain higher scan pitches (lower resolutions) than that of the training images. It has been found, for the first time, that the limit scan pitch λ , for the chosen probe varies for different defect characteristics ($\lambda_l = 6.0$ mm, $\lambda_w = 6.0$ mm, and $\lambda_d = 30.0$ mm). Sizing length and width of defects has more influence on the scan pitch and sizing depth of defects has found to be the least influence on the scan pitch. Although the effect of scan pitch is studied for MD-RBF neural network in this thesis, this study is generic in nature and applicable to any inversion algorithm.

The explicit dependency structure of the defect characteristics has been systematically studied in this thesis using a novel chain classification methodology. For optimization of the best sequence to be used in the chain, a new algorithm called greedy breadth-first-search (GBFS) has been proposed. GBFS with RBF neural network and support vector machine is capable of rapid sequence optimization for chain classification, thus avoiding the search of poor sequences, if any, that degrade the classification performance. It has been observed, for the first time, that depth→width→height→length is the optimal sequence for eddy current. The sequence has been identified by two diverse algorithms (RBF neural network and SVM) which strongly indicate the dependency structure during empirical eddy current inversion. Unlike MD-RBF neural network, chain classification cannot simultaneously size all the defect characteristics. It requires

identification of optimal sequence for better performance. It is computationally intensive than the MD-RBF neural network.

In order to expand the scope of empirical inversion by way of incorporating other robust learning algorithms, a novel MDLearn wrapper framework has been proposed. This framework has been applied to real world benchmark multilabel dataset viz. yeast, scene, and emotion for validation of assumptions made in building the wrapper framework. The results from the proposed framework have established that the assumptions made in the design of the framework hold good. Proposed framework using ML-RBF neural network has produced 79% global accuracy for four dimensions (4 defect characteristics) as compared to 72% achievable by the conventional uni-dimensional RBF neural networks. From the studies using benchmark real world multilabel dataset, and EC images of defects, it has been established that the MDLearn framework can be effectively used for empirical inversion. This framework can also be applied for defect classification and sizing using any other NDE techniques.

The approaches proposed in this thesis can easily be generalized to other diffusion based NDE techniques. One such potential technique is lock-in thermography. The propagation of heat energy in a material is diffusive like the electromagnetic fields in EC testing. In such a situation, lock-in thermography images obtained from components can be subjected to the multidimensional learning algorithms proposed in this thesis for efficient inversion. The features proposed in this thesis such as maximum magnitude, phase, and ratio of magnitude can be extracted from lock-in thermography images and they are directly related to the defect depth and height. Lock-in thermography can quickly detect defects in components with relatively larger surface area and the proposed multidimensional learning algorithms for subsurface defect sizing using can be effectively applied to lock-in thermography.

7.1 TECHNICAL AND SCIENTIFIC CONTRIBUTIONS

The major contributions of this thesis are the following:

- For simultaneous sizing of four important defect characteristics viz. length, width, depth, and height incorporating dependency, a novel MD-RBF neural network has been proposed. The MD-RBF neural network has been found to be more effective than the individual RBF neural networks. It can detect and size surface as well as subsurface defects.
- Influence of scan pitch of EC images on performance of defect sizing by the proposed MD-RBF neural network has been studied for the first time. From this study, it has been found that the limit scan pitch λ , exists and it is different for defect characteristics ($\lambda_l = 6.0$ mm, $\lambda_w = 6.0$ mm, and $\lambda_d = 30.0$ mm). Sizing length and width of defects has more influence on the scan pitch and sizing depth of defects has found to be the least influence on the scan pitch.
- A methodology using chain classification and novel algorithm called greedy breadth-first-search (GBFS) have been proposed to study the influence of dependency among the considered four defect characteristics for sizing. It has been observed, for the first time, that depth→width→height→length is the optimal sequence for empirical eddy current inversion.
- In order to expand the scope of the empirical inversion to incorporate other robust multilabel machine learning algorithms for NDE inversion, a novel MDLearn wrapper framework has been proposed. From the studies using benchmark real world multilabel dataset, it has been established that the MDLearn framework can be effectively used for inversion.

8

FUTURE WORKS

The work presented in this thesis has significantly improved capabilities of empirical inversion for accurate and automated sizing of defects using eddy current images. The thesis proposed new and efficient algorithms to size more than one defect characteristics and their dependency of defect characteristics for sizing. It also provided better insights into the existence of dependencies among the defect characteristics and highlighting the role of dependency on sizing accuracy during inversion. The work in this thesis can be extended in the following ways:

The eddy current inversion studies carried throughout this thesis are concentrated on sizing of 4 important characteristics of a defect i.e. length, width, depth, and height. However, a defect cannot be always restrained to 4 dimensions and it cannot be applicable to defects varying in shape and type, such as pitting corrosion, voids, and material loss. A defect can be a branching crack or sometimes it can be a crack which varying width at start of the crack and increasing gradually or irregularly. These defects are not studied in this thesis for sizing. However, all the methods and inversion algorithms reported in this thesis can be applied for sizing these types of defects in the metallic material. Hence, it may be worth exploring other characteristics of defects such as orientation to XYZ planes and other regular and irregular geometries using the methods proposed in this thesis. Dataset with realistic defects will capture the real world statistical distribution and will perform accurate sizing than artificial defect. It is also worth exploring the proposed methods using dataset with complete realistic defects for training and testing.

In the present study, input features from only two excitation frequencies have been extracted. However, more information can be obtained from multi-frequency EC data and it will be useful, especially when disturbing variables such as surface roughness, probe tilt, and variation in lift off are present during the measurements replicating the actual test conditions. Use of multi-frequency inputs is expected to increase the sizing accuracy in the presence of these disturbing variables, however, at the cost of computation time.

In the present study, features are extracted directly from the EC signals and images. However, it is also beneficial to study the use of other derived features such as PCA, wavelet coefficients, etc, and their combinations as input features for MD-RBF neural network.

The inversion methodologies studied in this thesis are applicable to only those EC images which are acquired with EC probe having identical dimensions and spectral characteristics. However, all the proposed machine learning algorithms in this thesis cannot be applied for inversion using any probe. In this direction, it is beneficial to study the performance of the proposed inversion methodologies using multiple probes by varying diameters and configurations towards generalization of the inversion algorithms.

MD-RBF proposed in this thesis can be improved to address multidimensional regression problems, by training with the actual value of the defect characteristics rather than categorization as performed in this thesis. This can also be achieved by categorizing and considering more than one output and interpolating relatively. Extensive comparative study may be required in order to find the better among them.

It was identified from the study that there is variation in the influence of the scan pitch limit on performance for defect sizing. The limit scan pitch found in this thesis is limited to the chosen probe. However, this work can be extended by studying the limit scan pitch for various

diameters and an empirical relationship between the limit scan pitch for an inversion algorithm and the probe diameter, can be obtained for generalization purpose.

The optimal sequence depth→width→height→length following the dependency has been obtained for the probe chosen and in stainless steels. It may be required to study the sequence for different materials and probes.

In principle, lock-in thermography works similar to eddy current testing as the underlying physical phenomenon is same i.e. diffusion. However, the disturbing variables and source of noise are different namely, reflectivity, emissivity, etc. Electrical conductivity and magnetic permeability have no effect on lock-in thermography. Thus, it may be worth studying inversion using fused redundant and complimentary information from eddy current testing and lock-in thermography.

REFERENCES

- [1] S.K. Gupta, D.A. Bourne, K.H. Kim, and S.S. Krishnan, "Automated process planning for sheet metal bending operations," *Journal of Manufacturing Systems*, vol. 17, no. 5, pp. 338-360, 1998.
- [2] El-Sayed A. El-Dahshan, Heba M. Mohsen, Kenneth Revett, and Abdel-Badeeh M. Salem, "Computer-aided diagnosis of human brain tumor through MRI: A survey and a new algorithm," *Expert Systems with Applications*, vol. 41, no. 11, pp. 5526-5545, 2014.
- [3] Mukta Paliwal and Usha A. Kumar, "Neural networks and statistical techniques: A review of applications," *Expert Systems with Applications*, vol. 36, no. 1, pp. 2-17, 2009.
- [4] Fu-Kuo Chang, "Structural health monitoring 2003: from diagnostics & prognostics to structural health management," in *Proceedings of the 4th International Workshop on Structural Health Monitoring*, 2003.
- [5] Baldev Raj, T. Jayakumar, and M. Thavasimuthu, *Practical Non-Destructive Testing*, 3rd ed.: Alpha Science International Ltd, 2007.
- [6] *Nondestructive testing handbook - Nondestructive testing overview*, 2nd ed.: Columbus, OH: American Society for Nondestructive Testing, 1996, vol. 10.
- [7] Peter J. Shull, *Nondestructive Evaluation: Theory, Techniques, and Applications*.: CRC Press, 2002.
- [8] S. S. Udpa and Patrick O. Moore, *Nondestructive Testing Handbook, Third Edition: Volume 5, Electromagnetic Testing*, 3rd ed.: ASNT, 2004.
- [9] K V. Rajkumar et al., "Characterisation of ageing behaviour in M250 grade maraging steel using eddy current non-destructive methodology," *Materials Science and Engineering-A*, vol. 464, no. 1, pp. 233-240, 2007.
- [10] S. Thirunavukkarasu, B P C. Rao, T. Jayakumar, N. Narayanankutty, and P V. Prabhakaran, "Eddy current based assessment of thickness of silicon carbide coating on carbon-carbon composites," *Journal of Aerospace Sciences and Technology*, vol. 63, no. 3, pp. 223-229, 2011.
- [11] B. Sasi, B P C. Rao, T. Jayakumar, and Baldev Raj, "Characterisation of solution annealed VT-14 titanium alloy using eddy current based electrical resistivity measurements," *Transactions of Indian Institute of Metals*, vol. 63, no. 5, pp. 773-777, 2010.
- [12] B.P.C. Rao, "Eddy Current Testing: Basics," *Journal of Non Destructive Testing & Evaluation*, vol. 10, no. 3, pp. 7-16, 2011.
- [13] B.P.C. Rao, *Practical Eddy Current Testing*.: Alpha Science Int'l Ltd, 2006.
- [14] H L. Libby, *Introduction to electromagnetic NDT methods*. New York: Wiley-

Interscience, 1971.

- [15] L Udpa, "Imaging of electromagnetic NDT phenomena," Colorado State University, Ph.D. dissertation 1986.
- [16] S. Thirunavukkarasu, B.P.C. Rao, T. Jayakumar, and Baldev Raj, "Techniques for processing remote field eddy current signals from bend regions of steam generator tubes of prototype fast breeder reactor," *Annals of Nuclear Energy*, vol. 38, no. 4, pp. 817-824, 2011.
- [17] B Sasi, Cacciola Matteo, Babu Rao C, Jayakumar T, and Baldev Raj, "Hybrid Signal Processing Approach for Enhanced Detection of Defects in EC-NDE," *Research in Nondestructive Evaluation*, vol. 24, pp. 51–61, 2013.
- [18] David C. Copley, "Eddy-Current Imaging for Defect Characterization," in *Library of Congress Cataloging in Publication Data*, Donald O. Thompson and Dale E. Chimenti, Eds.: Springer US, 1983, vol. 2A, pp. 1527-1540.
- [19] R. McCary, D. Oliver, K. Silverstein, and J. Young, "Eddy current imaging," *Magnetics, IEEE Transactions on*, vol. 20, no. 5, pp. 1986-1988, 1984.
- [20] R. Joynson, R. McCary, D. Oliver, K. Silverstein-Hedengren, and L. Thumhart, "Eddy current imaging of surface-breaking structures," *Magnetics, IEEE Transactions on*, vol. 22, no. 5, pp. 1260-1262, 1986.
- [21] Baldev Raj, T. Jayakumar, and B.P.C. Rao, "Review of NDT techniques for structural integrity," *Sadhana*, vol. 20, pp. 5-38, 1995.
- [22] S I Kabanikhin, "Definitions and examples of inverse and ill-posed problems," *Journal of Inverse and Ill-posed Problems*, vol. 16, pp. 317-357, 2008.
- [23] Jari P. Kaipio and Erkki Somersalo, *Statistical and Computational Inverse Problems*.: Springer New York, 2005.
- [24] Fatih Yaman, Valery G. Yakhno, and Roland Potthast, "A Survey on Inverse Problems for Applied Sciences," *Mathematical Problems in Engineering*, vol. 2013, pp. 1-19, 2013.
- [25] Richard C. Aster, Brian Borchers, and Clifford H. Thurber, *Parameter Estimation and Inverse Problems*, Renata Dmowska, James R. Holton, and H. Thomas Rossby, Eds.: Elsevier academic press, 2005.
- [26] N. Gagunashvili, "Machine learning approach to inverse problem and unfolding procedures," 2011. [Online]. <http://arxiv.org/abs/1004.2006>
- [27] Stephen J. Norton and John R. Bowler, "Theory of eddy current inversion," *Journal of Applied Physics*, vol. 73, no. 2, pp. 501-512, 1993.
- [28] Noritaka Yusa, "Development of computational inversion techniques to size cracks from eddy current signals," *Nondestructive Testing and Evaluation*, vol. 24, no. 1, pp. 39-52, 2009.
- [29] Franco Scarselli and Ah Chung Tsoi, "Universal Approximation Using Feedforward Neural Networks: A Survey of Some Existing Methods, and Some New Results," *Neural Networks*, vol. 11, no. 1, pp. 15-37, 1998.

- [30] Maria Wrzuszcak and Janusz Wrzuszcak, "Eddy current flaw detection with neural network applications," *Measurement*, vol. 38, no. 2, pp. 132-136, 2005.
- [31] Shejuan Xie, Zhenmao Chen, Li Wang, Toshiyuki Takagi, and Tetsuya Uchimoto, "An inversion scheme for sizing of wall thinning defects from pulsed eddy current testing signals," *International Journal of Applied Electromagnetics and Mechanics*, vol. 39, no. 1, pp. 203-211, jan 2012.
- [32] Shuxia Tian, Zhenmao Chen, Masashi Ueda, and Takuya Yamashita, "Signal processing schemes for Eddy Current Testing of steam generator tubes of nuclear power plants," *Nuclear Engineering and Design*, vol. 245, no. 0, pp. 78-88, 2012.
- [33] L.S. Rosado, F.M. Janeiro, P.M. Ramos, and M. Piedade, "Defect Characterization With Eddy Current Testing Using Nonlinear-Regression Feature Extraction and Artificial Neural Networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 5, pp. 1207-1214, 2013.
- [34] Pradeep Ramuhalli, "Neural network based iterative algorithms for solving electromagnetic NDE inverse problems," Iowa State University, Ames, Iowa, Ph.D. dissertation 2002.
- [35] Yuan Chen, Hong-Wei Ma, and Guang-Ming Zhang, "A support vector machine approach for classification of welding defects from ultrasonic signals," *Nondestructive Testing and Evaluation*, vol. 29, no. 3, pp. 243-254, #apr# 2014.
- [36] Hernán D. Benítez et al., "Defect characterization in infrared non-destructive testing with learning machines," *NDT & E International*, vol. 42, no. 7, pp. 630-643, 2009.
- [37] S. Saha, S. Mukhopadhyay, U. Mahapatra, S. Bhattacharya, and G.P. Srivastava, "Empirical structure for characterizing metal loss defects from radial magnetic flux leakage signal," *NDT & E International*, vol. 43, no. 6, pp. 507-512, 2010.
- [38] Gang Wang and T. Warren Liao, "Automatic identification of different types of welding defects in radiographic images," *NDT & E International*, vol. 35, no. 8, pp. 519-528, 2002.
- [39] Jiawei Han and Micheline Kamber, *Data Mining: Concepts and Techniques*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000.
- [40] Tom M Mitchell, *Machine learning*, 1st ed.: McGraw-Hill, 1997.
- [41] Robert Wall, Pdraig Cunningham, Paul Walsh, and Stephen Byrne, "Explaining the output of ensembles in medical decision support on a case by case basis," *Artificial Intelligence in Medicine*, vol. 28, no. 2, pp. 191-206, 2003.
- [42] Dengsheng Zhang, Md. Monirul Islam, and Guojun Lu, "A review on automatic image annotation techniques," *Pattern Recognition*, vol. 45, no. 1, pp. 346-362, 2012.
- [43] Moataz El Ayadi, Mohamed S. Kamel, and Fakhri Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition*, vol. 44, no. 3, pp. 572-587, 2011.
- [44] Trevor Hastie, Robert Tibshirani, and Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed.: Springer, 2009.

- [45] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, *Introduction to Data Mining*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2005.
- [46] Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin, *Learning From Data*.: AMLBook, 2012.
- [47] Christopher M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.
- [48] Donald F. Specht, "Probabilistic Neural Networks," *Neural Networks*, vol. 3, no. 1, pp. 109-118, 1990.
- [49] Vladimir N. Vapnik, *Statistical Learning Theory*, 1st ed.: Wiley-Interscience, 1998.
- [50] L. Udpa and S. S. Udpa, "Eddy Current Defect Characterisation Using Neural Networks," *Materials Evaluation*, vol. 48, no. 9, pp. 342-347, 353, 1990.
- [51] K.H. Hedengren, D.E. Ritscher, and B.R. Groshong, "Probe Footprint Estimation in Eddy-Current Imaging," in *Review of Progress in Quantitative Nondestructive Evaluation*, Donald O. Thompson and Dale E. Chimenti, Eds.: Springer US, 1990, pp. 845-852--.
- [52] J. M. Mann, L. W. Schmerr, and J. C. Moulder, "Neural network inversion of uniform-field eddy current data," *Materials Evaluation*, vol. 49, no. 1, pp. 34-39, 1991.
- [53] J. R. Bowler, S. J. Norton, and D. J. Harrison, "Eddy current interaction with an ideal crack, Part II: The inverse problem," *Journal of Applied Physics*, vol. 75, no. 12, pp. 8138-8144, 1994.
- [54] S. K. Burke, "Eddy-current inversion in the thin-skin limit: Determination of depth and opening for a long crack," *Journal of Applied Physics*, vol. 76, pp. 3072-3080, 1994.
- [55] S. K. Burke, "Crack Depth Measurement using Eddy-Current NDE," , 2001.
- [56] M.T. Shyamsunder et al., "Pattern recognition approaches for detection and characterisation of defects by eddy current testing," *Materials Evaluation*, vol. 58, no. 1, pp. 93-101, 2000.
- [57] Sung-Jin Song and Young-Kil Shin, "Eddy current flaw characterization in tubes by neural networks and finite element modeling," *NDT & E International*, vol. 33, no. 4, pp. 233-243, 2000.
- [58] B.P.C. Rao, "An intelligent imaging scheme for automated eddy current testing of austenitic stainless steel plates and welds," Saarland University, Saarbruecken, Ph.D. dissertation 2000.
- [59] B.P.C. Rao, Baldev Raj, T. Jayakumar, P. Kalyanasundaram, and W. Arnold, "A New Approach for Restoration of Eddy Current Images," vol. 20, no. 2, pp. 61-72--, 2001.
- [60] B.P.C. Rao, B Raj, T Jayakumar, and P Kalyanasundaram, "An artificial neural network for eddy current testing of austenitic stainless steel welds," *NDT & E International*, vol. 35, no. 6, pp. 393-398, 2002.
- [61] S. Thirunavukkarasu, B.P.C. Rao, T. Jayakumar, P. Kalyanasundaram, and B. Raj, "Quantitative Eddy Current Testing Using Radial Basis Function Neural Networks," *Materials evaluation*, vol. 62, no. 12, pp. 1213-1217, 2004.

- [62] Noritaka Yusa, Weiying Cheng, Zhenmao Chen, and Kenzo Miya, "Generalized neural network approach to eddy current inversion for real cracks," *NDT & E International*, vol. 35, no. 8, pp. 609-614, 2002.
- [63] T. Chady and P. Lopato, "Flaws Identification Using an Approximation Function and Artificial Neural Networks," *IEEE Transactions on Magnetics*, vol. 43, no. 4, pp. 1769-1772, 2007.
- [64] Helena G. Ramos, Tiago Rocha, Jakub Král, Dário Pasadas, and Artur L. Ribeiro, "An SVM Approach with Electromagnetic Methods to Assess Metal Plate Thickness," *Measurement*, vol. 54, pp. 201-206, 2014.
- [65] A Bernieri, Li Ferrigno, M Laracca, and M Molinara, "Crack Shape Reconstruction in Eddy Current Testing Using Machine Learning Systems for Regression," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 9, pp. 1958-1968, 2008.
- [66] A. Bernieri, G. Betta, L. Ferrigno, M. Laracca, and S. Mastrostefano, "Multifrequency Excitation and Support Vector Machine Regressor for ECT Defect Characterization," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 5, pp. 1272-1280, 2014.
- [67] C. Bielza, G. Li, and P. Larrañaga, "Multi-dimensional classification with Bayesian networks," *International Journal of Approximate Reasoning*, vol. 52, no. 6, pp. 705-727, 2011.
- [68] C. Reboud, G. Pichenot, D. Prémel, and R. Raillon, "Benchmark Results: Modeling with CIVA of 3D Flaws Responses in Planar and Cylindrical Work Pieces," , 2009.
- [69] Salvatore Ventre, Flavio Calvano, Grégoire Pichenot, Pierre Calmon, and Antonello Tamburrino, "ECT benchmark results for 3D defect signatures in industrial applications," *NDT & E International*, vol. 44, no. 4, pp. 376-386, 2011.
- [70] Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank, "Classifier Chains for Multi-label Classification," , Berlin, Heidelberg, 2009, pp. 254-269.
- [71] L. Enrique Sucar et al., "Multi-label Classification with Bayesian Network-based Chain Classifiers," *Pattern Recogn. Lett.*, vol. 41, pp. 14-22, 2014.
- [72] Stuart J. Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed.: Pearson Education, 2003.
- [73] G Tsoumakas and I Katakis, "Multi-Label Classification: An Overview," *International Journal of Data Warehousing and Mining*, vol. 3, no. 3, pp. 1-13, 2007.
- [74] Min-Ling Zhang, "Ml-rbf: RBF Neural Networks for Multi-Label Learning," *Neural Processing Letters*, vol. 29, no. 2, pp. 61-74, 2009.
- [75] Min-Ling Zhang and Zhi-Hua Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognition*, vol. 40, no. 7, pp. 2038-2048, 2007.
- [76] José Ramón Quevedo, Oscar Luaces, and Antonio Bahamonde, "Multilabel Classifiers with a Probabilistic Thresholding Strategy," *Pattern Recogn.*, vol. 45, no. 2, pp. 876-883, 2012.
- [77] Johannes Fürnkranz, Eyke Hüllermeier, Eneldo Loza Mencía, and Klaus Brinker, "Multilabel Classification via Calibrated Label Ranking," *Mach. Learn.*, vol. 73, no. 2,

pp. 133-153, 2008.

- [78] Jonathan Ortigosa-Hernández et al., "Approaching Sentiment Analysis by Using Semi-supervised Learning of Multi-dimensional Classifiers," *Neurocomputing*, vol. 92, pp. 98-115, 2012.
- [79] Min-Ling Zhang and Zhi-Hua Zhou, "A Review on Multi-Label Learning Algorithms," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 26, no. 8, pp. 1819-1837, 2014.
- [80] André Elisseeff and Jason Weston, "A Kernel Method for Multi-Labelled Classification," , vol. 14, 2001, pp. 681-687.
- [81] Konstantinos Trohidis, Grigorios Tsoumakas, George Kalliris, and Ioannis Vlahavas, "Multi-label classification of music by emotion," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2011, no. 1, pp. 1-9, 2011.
- [82] Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757-1771, 2004.

APPENDIX

1 Matlab code for training of MD-RBF neural network

```
function [Centroids,Sigma,Weights, classes]=
MD_RBFTrain(train_data,train_target,ratio,mu)

%train_data is a matrix a×b
%train_data is a matrix a×c
%ratio (clustering parameter  $\alpha$ ) is a real number the range 0-1
%mu is a real number in the range 0-1

[instance,labels]=size(train_target);
for i=1:labels
    numClass(i)=size(unique(train_target(:,i)),1);
end
fLabel=-1*ones(instance,sum(numClass));n=0;
class=1;
for i=1:labels
    mc=unique(train_target(:,i));
    cls{i}=unique(train_target(:,i));
    class=[class,mc'];
    for j=1:numClass(i)
        n=n+1;
        fLabel(:,n)=fLabel(:,n)+2*(train_target(:,i)>=mc(j));
    end
end
class(1)=[];
classes.name=class;
classes.num=numClass;
[Centroids,Sigma,Weights]= MD_RBF_train(train_data,fLabel',ratio,mu);

function
[Centroids,Sigma,Weights]=MD_RBF_train(train_data,train_target,ratio,mu)
numClass,num_train]=size(train_target);
Dim=size(train_data,2);
num_cluster=zeros(1,numClass);
for j=1:numClass
    num_cluster(1,j)=ceil(ratio*sum(train_target(j,')==1));
end
num_centroid=sum(num_cluster);
%Initial clustering for each dimensions
Centroids=zeros(num_centroid,Dim);
for j=1:numClass
    temp_index=find(train_target(j,')==1);
    if ~isempty(temp_index) && num_cluster(1,j)>0 &&
(size(temp_index,2)>=num_cluster(1,j))
        temp_train_data=train_data(temp_index,:);

[~,CEN]=kmeans(temp_train_data,num_cluster(1,j),'EmptyAction','singleton');
        low=sum(num_cluster(1:j-1))+1;
        high=sum(num_cluster(1:j));
        Centroids(low:high,:)=CEN;
    else
```

```

        num_cluster(1,j)=0;
    end
end
%%%Computing prototype vectors
distfun='euclidean';
Y1=pdist(Centroids,distfun);
centroid_dist=squareform(Y1);
numerator=sum(sum(triu(centroid_dist,1)));
denominator=num_centroid*(num_centroid-1)/2;
sigma=mu*(numerator/denominator);
Sigma=zeros(1,num_centroid);
counter=0;
for j=1:numClass
    sigma_j=sigma;

    for k=1:num_cluster(j)
        counter=counter+1;
        Sigma(1,counter)=sigma_j;
    end
end

A=zeros(num_train,num_centroid+1);
B=zeros(num_train,numClass);
%Determination of weight

for i1=1:num_train
    temp_vec=zeros(1,num_centroid);

    for k=1:num_centroid
        temp_sigma=Sigma(1,k);
        vec1=train_data(i1,:);
        vec2=Centroids(k,:);
        tmp=sqrt((vec1-vec2)*(vec1-vec2)');
        temp_vec(1,k)=exp((-tmp^2)/(2*temp_sigma^2));
    end
    temp_vec=[1,temp_vec];
    A(i1,:)=temp_vec;

    B(i1,:)=train_target(:,i1)';
end
Weights=A\B;

```

2. Matlab code for testing of MD-RBF neural network

```
function
[Outputs,Predictions]=MDO_RBF_test(test_data,Centroids,Sigma,Weights,classes)

[rows,columns]=size(test_data);
num_centroid=size(Centroids,1);
A=zeros(rows,num_centroid+1);

for i=1:size(test_data,1)

    temp_vec=zeros(1,num_centroid);
    for k=1:num_centroid
        vec1=test_data(i,:);
        vec2=Centroids(k,:);
        tmp=sqrt((vec1-vec2)*(vec1-vec2)');
        temp_sigma=Sigma(1,k);
        temp_vec(1,k)=exp(-tmp^2/(2*temp_sigma^2));
    end
    temp_vec=[1,temp_vec];

    A(i,:)=temp_vec;
end

Outputs=(A*Weights);
Predictions=zeros(length(classes.num),rows);

for i=1:rows
    start=1;
    for j=1:length(classes.num)
        stop=start+classes.num(j)-1;
        tem=find((Outputs(i,start:stop)>0)==1,1,'last');
        Predictions(j,i)=tem(1);
        start=start+classes.num(j);
    end
end
Predictions=Predictions';
count=0;
for i=1:length(classes.num)
    Predictions(:,i)=classes.name(Predictions(:,i)+count);
    count=count+classes.num(i);
end
```

3. Matlab code for chain classification.

```
function
predictions=ChainClassify(features,training,targets,test,predictions,labelC
ount,index,classifierparams1 , classifierparams2)
%clainclassify recursively train for each classifier and a label and in
nextpredictions, previous predictions will be added as training data.

label1=targets(:,index);
label=(full(ind2vec(label1)))';
target=(~label-2)+3*label;
featuredTraining=training(:,[features{index},end:-1:end-index+1]);
featuredTest=test(:,[features{index},end:-1:end-index+1]);
[a,b,c,d]= ... %classifier parameters
  trainClassifier(featuredTraining,target',classifierparams1 ,
classifierparams2); %classifiers -SVMTrain, RBFTrain, MLPTrain etc,
[~,PreLabels,~]= testClassifier(featuredTest',a,b,c,d);
training=[training,targets(:,index)];
PreLabel=(full(ind2vec(PreLabels)))';
PreLabel1=(~PreLabel-2)+3*PreLabel;
test=[test,PreLabels'];
labelCount=labelCount-1;
predictions(:,index)=PreLabels;
index=index+1;
if labelCount~=0
    predictions=
ChainClassify(features,training,targets,test,predictions,la
belCount,index, ratio,mu);
end
end
```

4. Matlab Code for MDLearn wrapper training (class transformation).

```
function
[Centroids, Sigma_value, Weights, tr_time, classes]=MDLearn_train(train_data, tr
ain_target, classifierparams)
%train_data is a matrix a×b
%train_data is a matrix a×c

[instance, labels]=size(train_target);
for i=1:labels
    num_class(i)=size(unique(train_target(:,i)),1);
end
fLabel=-1*ones(instance, sum(num_class)); n=0;
class=1;
for i=1:labels
    mc=unique(train_target(:,i));
    class=[class, mc'];
    for j=1:num_class(i)
        n=n+1;
        fLabel(:,n)=fLabel(:,n)+2*(train_target(:,i)==mc(j));
    end
end
class(1)=[];
classes.name=class;
classes.num=num_class;
[classifierTrainOutputs] = MultilabelClassifierTraining(train_data, fLabel',
classifierparams)
%example
[Centroids, Sigma_value, Weights]=MLRBFTrain(train_data, fLabel', ratio, mu);
```

5. Matlab Code for MDLearn wrapper prediction.

```
function [Predictions]=MDLearnPrediction(classifierTrainOutputs,classes)
%All the input parameters are derived from MDLearn_train

Predictions=zeros(length(classes.num),rows);

for i=1:rows
    start=1;
    for j=1:length(classes.num)
        stop=start+classes.num(j)-1;

        tem=find(classifierTrainOutputs(i,start:stop)==max(classifierTrainOutputs(i
,start:stop)));
        Predictions(j,i)=tem(1);
        start=start+classes.num(j);
    end
end
Predictions=Predictions';
%naming the classes
count=0;
for i=1:length(classes.num)
    Predictions(:,i)=classes.name(Predictions(:,i)+count);
    count=count+classes.num(i);
end
```