Neural networks for direction of arrival estimation in the presence of array sensor failures

Vigneshwaran Subbaraju

2007

Vigneshwaran, S. (2007). Neural networks for direction of arrival estimation in the presence of array sensor failures. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/39065>

<https://doi.org/10.32657/10356/39065>

Neural Networks for Direction of Arrival Estimation in the Presence of Array Sensor Failures

Vigneshwaran Subbaraju

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Master of Engineering

2007

Acknowledgments

I express my gratitude to my supervisor Prof. N. Sundararajan for his relentless effort in guiding and encouraging me during the course of the project. I wish to thank Assoc. Prof. P. Saratchandran, for the invaluable guidance he offered during this period. The illuminating interactions with him and Prof. Sundararajan paved the path for the successful completion of this project. They stood with me through the ups and downs encountered during the course of the project. Their scientific and moral standards are the ones that I aspire to achieve.

I would also like to thank my lab technician Mr. Yock for his support during this period. I would also like to thank my lab colleagues Ms. Wang wen and Ms. Lin ling, and fellow researchers in this area for their help and guidance. Last but not the least I would like to thank my parents for their full support and encouragement during this period.

Contents

Acknowledgments ii

Contents iii

Summary..... vi

List of Figures..... x

List of Tables xv

1 Introduction..... 1

1.1 Background and Motivation..... 1

1.2 Scope and Objectives 4

1.3 Major Contributions of the Thesis 5

1.4 Organization of the Thesis 6

2 Direction of Arrival (DoA) Estimation Problem..... 9

2.1 The Adaptive Antenna Array System and Basic Assumptions..... 9

2.2 Basic Theory and Terminology..... 11

2.2.1 Effect of noise..... 12

2.2.2 Effect of Mutual Coupling..... 14

2.2.3	Effect of Sensor Failure	15
3	Review of existing methods of DoA estimation	17
3.1	A Review of Conventional Methods of DoA estimation	17
3.2	Neural Network Based DoA estimation.....	23
3.3	A Review of Existing Neural Network Based DoA estimation Methods	25
3.4	DoA Estimation under sensor failure	29
4	The Preprocessing Scheme.....	33
4.1	Need for the Pre-Processor.....	33
4.2	Design of the Pre-Processor	35
4.2.1	Pre-Processor Using the Phase Differences.....	35
4.2.2	Pre-Processor Using the Correlation Matrix.....	39
4.2.3	Proposed Preprocessing Scheme	40
4.3	Performance evaluation of Pre-Processing Schemes	45
5	Improved DoA Estimation Using Minimal Resource Allocation Neural Network (MRAN)	49
5.1	A Brief Description of MRAN.....	50
5.2	Performance Study of DoA Estimation with MRAN and the Improved Pre-processing Scheme	53
5.2.1	Performance in an ideal environment:.....	54
5.2.2	Performance with noise:	56
5.2.3	Mutual Coupling:	58

5.2.4	Non-Uniform Array:	61
5.2.5	Unequal Source Power:	63
6	DoA Estimation In the Presence of Sensor Failures.....	67
6.1	Performance with sensor failure but no noise:	68
6.2	Performance with sensor failure and noise:	70
6.3	Varying the element at which failure occurs:.....	73
6.4	Performance with Coherent Sources:	77
7	Conclusions & Future Work.....	82
7.1	Conclusions	82
7.2	Suggestions for Future Work	86
	Author's Publications	89
	References.....	90
	Appendix A – Varying the Element at which Failure Occurs:.....	96

Summary

Direction of Arrival (DoA) estimation is an important step in the implementation of adaptive/smart antennas which are an integral part of mobile communication systems. Conventional array signal processing algorithms have been widely used for DoA estimation. Among the conventional methods eigen-decomposition based methods such as MUSIC, ESPRIT etc have been found to be accurate, and provide higher resolution when compared to other methods. But these methods have been found to be computationally complex and are hence unsuitable for real-time operation. They are also sensitive to sensor failures, noise and other antenna effects like mutual coupling, non-uniformities in the positions of the sensors etc and require calibrated antennas with uniform features. A sensor is said to have failed when it can no longer receive the signals from the sources and its output consists only of the noise produced by the other circuitry involved. Mutual coupling is a phenomenon where the signal at a particular sensor induces a signal at others sensors in its close proximity. In the existing methods, sensor failure is tackled by introducing some corrections to the wrong data acquired after sensor failure and mutual coupling effects are usually overcome by calibrating the array before making the measurements.

DoA estimation can also be viewed as a function-mapping problem from the space of the sensor array outputs to the space of the DoAs. Neural networks are well known for their universal function approximation capability and recently a DoA estimation scheme based on a Radial Basis Function Neural Network (RBFNN) was proposed by

Zooghby et al. This approach avoids the cumbersome eigen-decomposition processes that is usually involved in the conventional signal processing algorithms, and has been found to be far quicker than conventional methods. Apart from being computationally efficient, it is also found to be more immune to noise and yield better performance in the presence of correlated arrivals.

However, in the RBFNN method developed by Zooghby et al, the number of hidden neurons for the RBFNN has been chosen by trial and error studies. This is not desirable since the number of hidden neurons in a RBFNN affects the time taken for training and testing apart from affecting the accuracy. The method is also found to yield a very large error when the sources are around the endfire directions of the uniform linear array (ULA). Further, the effect of various environmental factors and antenna effects have not been studied in the above study. Such factors include noise, sensor failure, mutual coupling, unequal source powers and random errors in the positions of the elements of the ULA. Recently, considerable research has been focused on studying the effect of these factors on the existing DoA estimation algorithms.

In this thesis, an attempt has been made to overcome the deficiencies of the RBFNN based method for the design of DoA estimation scheme that is capable of handling noise, sensor failures and the other antenna effects mentioned above.

A sequential learning scheme for RBFNNs called Minimal Resource Allocation Network (MRAN) has been recently developed which automatically chooses the number of hidden neurons and adjusts its parameters based on the input data during the training process. This process also results in a minimal number of hidden neurons for a given problem and the resultant network consumes less computational resources as the number of hidden neurons is minimal. This property can be exploited for Real-time DoA estimation and a DoA estimation scheme based on MRAN has been developed in this thesis.

The problem of DoA estimation under sensor failure is the main focus of this thesis. Conventional methods like MUSIC and ESPRIT are found to be incapable of handling sensor failures and hence require correction for the wrong data acquired after sensor failure. Such corrective measures, in turn, require the knowledge of time at which the failure occurred and also the exact element which failed. The proposed MRAN based approach handles failure without involving such corrections for sensor failure and hence it attempts to eliminate the need for the knowledge of the element and the time of failure.

The study results indicate superior performance of MRAN in the event of failure, when compared with other existing methods. For example, from the studies presented in the thesis, for a typical SNR of 20dB the RMS testing error of MRAN is 5.39 degrees while the RBFNN gives an error of 18.46 degrees in the presence of sensor failure. The conventional MUSIC algorithm, gives an RMS testing error of 78.86 degrees. DoA estimates from MRAN are found to be more robust to sensor failure since there is only a nominal increase in the estimation error irrespective of whichever element in the sensor array encounters a failure and also irrespective of the position of the sources.

In the presence of noise, both MRAN and RBFNN perform far better than the conventional MUSIC algorithm. Among the neural network based methods MRAN requires a smaller number of hidden neurons than the RBFNN to yield a similar performance. For a typical SNR of 20dB, MRAN gives an RMS testing error of 0.1629 degrees while the RBFNN gives an RMS testing error of 0.1747 degrees. However MRAN requires only 31 hidden neurons while the RBFNN requires 50 neurons. For the same case, the conventional MUSIC algorithm gives an RMS testing error of 47.92 degrees.

MRAN based DoA estimation method is also found to perform better than the existing methods in the presence of uncalibrated mutual coupling between the elements. No additional operations like calibration etc have to be done for this case. However MRAN and RBFNN based methods require a higher number of hidden neurons in the presence

of mutual coupling. MRAN uses 39 hidden neurons while the RBFNN needs 60 hidden neurons.

The studies presented here also indicate that other factors such as inequality in the powers of the sources and errors in the position of the sensors do not have any significant effect on the performance of the MRAN based DoA scheme. A new pre-processing scheme has also been proposed to overcome the problem of high errors when the sources are near the endfire directions of the array. The RBFNN yielded an RMS testing error of 23.13 degrees when used with the existing pre-processing scheme while the same RBFNN produced an RMS testing error of 0.1741 degrees with the new pre-processing scheme.

List of Figures

Figure 2.1 - Illustration of a Typical Adaptive Antenna Array System 10

Figure 3.1 - Basic Structure of a Neural DoA Estimator24

Figure 3.2 - Forward pass of the multi layer perceptron25

Figure 3.3 - Backward pass of the multilayer perceptron26

Figure 4.2 - Schematic for Phase difference pre-processing Scheme.37

Figure 4.3.1 - DoA estimates for Source 1 with the pre-processing scheme in [6]46

Figure 4.3.2 - DoA estimates for Source2 with the pre-processing scheme in [6]47

Figure 4.3.3 - DoA estimates for Source1 with the proposed pre-processing scheme .47

Figure 4.3.4 - DoA estimates for Source2 with the proposed pre-processing scheme .48

Figure 5.2.1 - Neuron history of MRAN during training.....55

Figure 5.2.2 - RMS error for MRAN during training55

Figure 5.2.3 - Neuron history of MRAN during training with mutual coupling.....59

Figure 5.2.4 - RMS error for MRAN during training with mutual coupling60

Figure 5.2.5 - Neuron history of MRAN during training with a Non-uniform array....62

Figure 5.2.6 - RMS error for MRAN during training with a Non-Uniform Array62

Figure 5.2.7 - Neuron history of MRAN when trained with unequal source power64

Figure 5.2.7 - RMS error for MRAN when trained with unequal source power65

Figure 6.1.1 - Performance as a function of source position - No noise, Failure at
150th snapshot.....69

Figure 6.1.2 - Performance as a function of failure instant - No Noise, sources at -9°
and -11°70

Figure 6.2.1 - Performance as a function of source position - SNR=20dB, Failure at
150th snapshot.....71

Figure 6.2.2 - Performance as a function of failure instant - SNR=20dB, sources at -9°
and -11°72

Figure 6.2.3 - Performance as a function of source position - SNR=10dB, Failure at
150th snapshot.....72

Figure 6.2.4 - Performance as a function of failure instant - SNR=10dB, sources at -9°
and -11°73

Figure 6.3.1 - Performance as a function of source position - No Noise, Failure at
150th snapshot, at 1st element.....74

Figure 6.3.2 - Performance as a function of failure instant - No Noise, Sources at -9°
and -11°, failure at 1st element74

Figure 6.3.11- Performance as a function of source position - SNR=20dB, Failure at
150th snapshot, at 1st element.....75

Figure 6.3.12- Performance as a function of failure instant - SNR=20dB, Sources at -9°
and -11°, failure at 1st element75

Figure 6.3.21- Performance as a function of source position - SNR=10dB, Failure at 150 th snapshot, at 1 st element.....	76
Figure 6.3.22- Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 1 st element	76
Figure 6.4.1 - MUSIC spectrum for uncorrelated and coherent sources.....	77
Figure 6.4.2 - Performance as a function of source position - No Noise, Failure at 150 th snapshot, for Coherent Sources.....	78
Figure 6.4.3 - Performance as a function of failure instant - No Noise, Sources at -9° and -11°, for Coherent Sources.....	79
Figure 6.4.4 - Performance as a function of source position - SNR=20dB, Failure at 150 th snapshot, for Coherent Sources.....	79
Figure 6.4.5 - Performance as a function of failure instant - SNR=20dB, Sources at -9° and -11°, for Coherent Sources.....	80
Figure 6.4.6 - Performance as a function of source position - SNR=10dB, Failure at 150 th snapshot, for Coherent Sources.....	80
Figure 6.4.7 - Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, for Coherent Sources.....	81
Figure A1 - Performance as a function of source position - No Noise, Failure at 150 th snapshot, at 2 nd element.....	97
Figure A2 - Performance as a function of failure instant - No Noise, Sources at -9° and -11°, failure at 2 nd element	97
Figure A3 - Performance as a function of source position - No Noise, Failure at 150 th snapshot, at 4 th element	98

Figure A4	- Performance as a function of failure instant - No Noise, Sources at -9° and -11° , failure at 4 th element.....	98
Figure A5	- Performance as a function of source position - No Noise, Failure at 150 th snapshot, at 5 th element	99
Figure A6	- Performance as a function of failure instant - No Noise, Sources at -9° and -11° , failure at 5 th element.....	99
Figure A7	- Performance as a function of source position - No Noise, Failure at 150 th snapshot, at 6 th element	100
Figure A8	- Performance as a function of failure instant - No Noise, Sources at -9° and -11° , failure at 6 th element.....	100
Figure A9	- Performance as a function of source position - SNR=20dB, Failure at 150 th snapshot, at 2 nd element.....	101
Figure A10	- Performance as a function of failure instant - SNR=20dB, Sources at -9° and -11° , failure at 2 nd element	101
Figure A11	- Performance as a function of source position - SNR=20dB, Failure at 150 th snapshot, at 4 th element	102
Figure A12	- Performance as a function of failure instant - SNR=20dB, Sources at -9° and -11° , failure at 4 th element.....	102
Figure A13	- Performance as a function of source position - SNR=20dB, Failure at 150 th snapshot, at 5 th element	103
Figure A14	- Performance as a function of failure instant - SNR=20dB, Sources at -9° and -11° , failure at 5 th element.....	103
Figure A15	- Performance as a function of source position - SNR=20dB, Failure at 150 th snapshot, at 6 th element	104

Figure A16 - Performance as a function of failure instant - SNR=20dB, Sources at -9° and -11°, failure at 6th element..... 104

Figure A17 - Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 2nd element..... 105

Figure A18 - Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 2nd element 105

Figure A19 - Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 4th element 106

Figure A20 - Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 4th element..... 106

Figure A21 - Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 5th element 107

Figure A22 - Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 5th element..... 107

Figure A23 - Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 6th element 108

Figure A24 - Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 6th element..... 108

List of Tables

Table 5.2.1 – Performance comparison in an ideal environment (No failure and No Noise)..... 54

Table 5.2.2 – Performance comparison With Noise and No failure)..... 56

Table 5.2.3 – Performance comparison with CRB for source1 (0 degrees)..... 57

Table 5.2.4 – Performance comparison with CRB for source 2 (2 Degrees)..... 58

Table 5.2.5 – Performance with mutual coupling- No Noise case..... 59

Table 5.2.6 – Performance with Mutual Coupling-With Noise 60

Table 5.2.7 – Performance with a Non-Uniform Array-No Noise case..... 61

Table 5.2.8 – Performance with a Non-Uniform Array-With Noise..... 63

Table 5.2.9 – Performance with Unequal Source Powers -No Noise case 64

Table 5.2.10 - Performance comparison with Unequal Power Sources- with Noise 65

Table 6.2.1 – Performance comparison with failure and Noise 71

Table 7.1 – Performance comparison with Noise (No failures)..... 83

Table 7.2 – Performance comparison with failure and Noise 84

Table 7.3 – Performance with mutual coupling- No Noise case..... 85

Chapter 1

Introduction

1.1 Background and Motivation

Adaptive and smart antennas are seen today as a means to satisfy the increasing demand for bandwidth in mobile communication as well as to reduce the likelihood of co-channel interference. These methods use multiple steered high gain beams at the same carrier frequency to provide service to individual users within the same cell (spatial region). A higher channel capacity is achieved by frequency reuse within the same cell and the use of multiple access techniques like Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA). However even the use of these techniques hasn't fulfilled the need for an increased channel capacity. This issue is addressed in the next generation communication systems using another multiple access scheme called Space Division Multiple Access (SDMA). This scheme works by steering the multiple beams of an antenna array such that a narrow beam is steered towards the mobile user while the other beams are steered such that nulls are directed towards the other interfering sources that use the same frequency. This allows for increased frequency reuse and better suppression of co-channel interference. It has been shown that an antenna array of N elements used in a SDMA scheme can increase the channel capacity N fold. The use of such a scheme is also known to overcome problems due to signal fading and multipath reflection.

There are two main operations involved in using such adaptive or smart antennas, viz.,

1. Determining the direction of the mobile user and the directions of the interfering signals.
2. Directing a narrow beam towards the mobile user while simultaneously directing nulls towards the interfering signals.

The first step is referred to as Direction-of-Arrival (DoA) estimation and the second step is known as Beamforming.

Adaptive array signal processing first involves finding the DoAs of the signals using a super resolution array signal processing algorithm such as MUSIC [9], ESPRIT [10] etc. Once the directions of the users are specified, this information can be used so that the radiation pattern of the array is adapted to allocate the maximum gain towards the mobiles of interest while other interfering sources in the same frequency slot are nulled and the system is able to track these mobiles. Conventional methods like MUSIC are found to be accurate, and are able to provide high resolution. But these methods are largely based on linear algebra and are found to be computationally complex and are hence unsuitable for real-time operation. They also require calibrated antennas with uniform features and are sensitive to sensor failure and other physical uncertainties. In recent years, there has been considerable research that attempts to modify and improve these methods to make them suitable for real-time application and also to make them more robust to noise, sensor failure, uncalibrated mutual coupling and other environmental factors.

Recently, soft-computing techniques have found wide applicability to various signal processing problems. DoA estimation can be viewed as a function-mapping problem from the space of the sensor array outputs to the space of the DoAs. Beam forming is just the inverse problem i.e. a function mapping from the space of DoAs to the space of

array output (weight vector). Soft-Computing methods, particularly the Radial Basis Function Neural Network (RBFNN), are well known for their universal function approximation capability. Hence they become natural candidates to solve the DoA estimation and beamforming problems. A comprehensive survey of various neural network based methods for array signal processing is given in [13]. Since neural methods avoid the cumbersome eigen-decomposition processes used in conventional signal processing algorithms, they are found to be far quicker than conventional methods. Apart from being computationally efficient, neural methods have been observed to be more immune to noise and are found to yield better performance in the presence of correlated arrivals as reported in [1]-[6].

A DoA estimation scheme based on a RBFNN was proposed in [6]. It exploits the universal function approximation capability of RBFNNs to estimate the DoAs. A successful classification of closely separated sources has been reported and the method has been found to perform better than MUSIC when the sources are correlated. In [6] the number of hidden neurons was chosen by trial and error. This is not desirable since the number of hidden neurons in a RBFNN affects the time taken for training and testing apart from affecting the accuracy. It can also be observed from the results presented in [6] that the RBFNN scheme produces a very large error when the sources are around the endfire directions of the uniform linear array (ULA). Further, the sensitivity of various environmental factors and antenna effects has not been studied. Such factors include noise, sensor failure, mutual coupling, unequal source powers and random errors in the positions of the elements of the ULA. Recently, considerable research has been focused on studying the effect of these factors on existing DoA estimation algorithms.

A sequential learning scheme for RBFNNs called MRAN [7] has been recently developed. The algorithm starts with zero hidden neurons and involves adding and pruning of the RBF hidden layer neurons during the training process, so as to achieve a minimal network with better approximation accuracy. When no neurons are added or

removed, the algorithm uses an Extended Kalman Filter (EKF) to update the centers, widths and weights of each of the hidden neurons. The resultant network contains minimal number of hidden neurons and hence consumes less computational resources. This property can be exploited for Real-time DoA estimation.

In this thesis, a DoA estimation method based on MRAN has been developed, which overcomes the deficiencies of the RBFNN based method and also is capable of handling noise, sensor failure and the other anomalies mentioned above.

1.2 Scope and Objectives

The objectives of this thesis are to,

1. Develop a DoA estimation scheme that is more accurate than the existing methods in the event of a sensor failure and operates without the help of any correction made to the data. This eliminates the need for the knowledge of the position and time of occurrence of the failure.
2. Provide a good estimate of the DoAs in the presence of noise, mutual coupling and other antenna effects.
3. Operate with lower computational complexity than conventional eigen-decomposition based methods like MUSIC and ESPRIT and hence more suitable for real-time operation.
4. Overcome the problems in existing neural network based schemes like choosing the architecture of the neural network by trial and error and the observed high errors when the sources are around the endfire directions of the ULA.

In addition, the scope of the research includes studying the performance of the algorithms under different conditions of noise and under the influence of different

anomalies like uncalibrated mutual coupling, inequality in the powers of the sources, errors in the position of the sensors and the presence of coherent sources.

In this work, the scope of the study is limited to the DoA estimation problem with a uniform linear array. This implies that we are only interested in determining either the elevation or the azimuth angle of the sources but not both. The problem of estimating both the elevation and azimuth can be considered as a simple extension of the current research by using a planar array instead of a linear array.

1.3 Major Contributions of the Thesis

Major contributions made in this thesis are:

- The study of neural network based methods for DoA estimation under sensor failures. Conventional methods like MUSIC and ESPRIT are found to be incapable of handling sensor failures and require corrective measures to be taken for the wrong data acquired after sensor failure. Such corrective measures in turn require the knowledge of time at which the failure occurred and the exact element which failed. They also require additional computational time which may not be available. In this thesis a DoA estimation scheme based on MRAN has been proposed which handles failure without involving such corrections for sensor failure and hence it attempts to eliminate the need for the knowledge of the element and the time of failure. DoA estimates from MRAN are found to be more robust to sensor failure since there is only a nominal increase in the estimation error in the event of a sensor failure.
- The proposed MRAN based method is also found to give a smaller RMS testing error than the existing methods in the presence of noise, mutual coupling and errors in the positions of the sensors in the ULA. For the case with mutual coupling the neural network based methods require higher number of hidden neurons to give the same RMS testing error as given by them for the case

without mutual coupling. The MUSIC algorithm on the other hand gives higher error than the RBFNN and MRAN and requires calibrated antennas.

- The problem of choosing the number of hidden neurons by a trial and error approach has been overcome by the proposed MRAN based scheme. MRAN can automatically choose an optimal number of hidden neurons during its training process. Further MRAN is found to need a fewer number of hidden neurons than that required by the conventional RBFNN based approach. This implies a reduction in the computational complexity when using the trained network in the field.
- A new pre-processing scheme is proposed for neural network based DoA estimators. This scheme overcomes the problem of high errors when the sources are near the endfire directions of the array. The scheme is also found to be robust to noise and sensor failure.

This thesis clearly shows that the use of the MRAN based DoA Estimation scheme result in significant performance improvement.

1.4 Organization of the Thesis

The study undertaken in this thesis has been presented in seven chapters.

Chapter 2 gives an overview the DoA estimation problem. The terminology, experimental setup and the fundamental principles involved are explained. The basic concepts of steering vectors, array propagation matrix, source correlation matrix and array correlation matrices are explained with detailed mathematical expressions to calculate them. The mathematical model for the phenomena of sensor failure and mutual coupling are given and also how the expression for the correlation matrix varies under the influence of these phenomena is also mentioned.

Chapter 3 presents a survey of the existing DoA estimation methods. It briefly explains some important conventional methods of DoA estimation such as MUSIC and ESPRIT algorithms and their variations. The advantages and drawbacks of these methods are summarized. This is followed by a brief introduction to neural networks. The Multi Layer Perceptron (MLP) and the Radial Basis Function Neural Network (RBFNN) are briefly explained. How these neural networks can be applied to solve the DoA estimation problem is also dealt with. The basic architecture of a neural network based DoA estimator is explained with the help of a block diagram.

Chapter 4 deals with the design of the preprocessing scheme. The preprocessor is the first block in the architecture of a neural network based DoA estimator. The existing designs for the preprocessor are discussed and their advantages and drawbacks are explained in a structured manner. Then the proposed new preprocessing scheme is introduced. A qualitative explanation on how this new preprocessing scheme will overcome the deficiencies in the existing schemes is substantiated with necessary mathematical equations. Then some simulation results are given to compare the existing preprocessing scheme with the proposed one.

Chapter 5 deals with the proposed method of DoA estimation using MRAN in conjunction with the improved pre-processor designed in chapter 4. A brief introduction to MRAN is first given. This is followed by simulation results that compare the performance of the proposed MRAN based method with the RBFNN based method and MUSIC algorithm. The performance of the three algorithms are compared under different conditions of SNR, mutual coupling, Non-uniformity of the sensor array, inequality in the power of the sources etc. Detailed explanations of the results obtained are given.

Chapter 6 deals with the problem of DoA estimation under sensor failure. The performance of the three methods discussed in chapter three are considered again here

in the presence of sensor failure. Simulation results are given for varying the instant at which failure occurs and for varying the positions of the sources. The studies are done at different values of SNRs. The effect of varying the element at which the failure occurs is also studied. We also study the effect of sensor failure in the presence of coherent sources.

Chapter 7 summarises the conclusions based on the studies performed in this work and gives some recommendations for future work.

Chapter 2

Direction of Arrival (DoA) Estimation Problem

This chapter introduces the fundamental principles and the theory behind the DoA estimation problem. DoA estimation and beamforming are two important steps in the operation of adaptive antenna systems. These systems use an array of antennas/sensors and some signal sources that emit radiation from specific directions with respect to this array of sensors. The directions can be specified by the angle which they make with respect to either the normal to the array or the axis of the array. DoA estimation refers to the problem of estimating these angles from a few samples of the array output. The following section presents an analysis of this problem with the help of a typical adaptive antenna array system.

2.1 The Adaptive Antenna Array System and Basic Assumptions

A Uniform-Linear-Array (ULA) of 'M' omni-directional antennas with an inter-element spacing 'd' is considered throughout. The number of sinusoidal point sources in the far-field of the antenna array is denoted by 'K'. Since the sources are assumed to be in the

far-field we can safely assume that the wave-fronts that are incident on the array due to these sources are plane wavefronts.

In this thesis, Direction-Of-Arrival (DoA) of a source refers to the angle between the normal to the array (broad side direction) and the line joining the point source to the centre of the array as shown in the figure 2.1.

The number of sources (K), is assumed to be known and our concern is to develop a neural network based method obtain an estimate of the DoAs of all the signals, from the array output.

Figure 2.1 Illustration of a Typical Adaptive Antenna Array System

2.2 Basic Theory and Terminology

Consider the l^{th} source, whose DoA is θ_l . The time taken by the wave-front due to this source, to travel between two adjacent antennas is given by,

$$\tau_l = \frac{d \sin \theta_l}{C} \quad (2.2.1)$$

Where 'C' is the velocity of the electromagnetic wave.

Assuming the antenna A_1 as the spatial and temporal reference element, the Signal at the antenna A_p , due to the source at θ_l is a delayed version of the signal at A_1 . The exact time delay is $(p-1) \tau_l$. Let $s_1(t)$ denote the signal induced on antenna A_1 due to the source at θ_l . Then the signal induced due to the same source at antenna A_p can be written as $s_1(t) e^{j\omega(p-1)\tau_l}$.

Thus the array output due to the source at θ_l can be written as $s(t) \mathbf{a}_{\theta_l}$,

$$\text{Where } \mathbf{a}_{\theta_l} = \left(1 \ e^{j\omega\tau_l} \ e^{j2\omega\tau_l} \ \dots e^{jM\omega\tau_l} \right)^T$$

This vector $\mathbf{a}_{\theta_l}$ is called the *direction vector* for angle θ_l .

Since there are 'K' sources at different DoAs the total signal induced on a particular antenna is the superposition of the signals due to the individual sources. Thus the total signal induced on antenna A_p , due to the 'K' different wavefronts incident on it can be written as

$$x_p = \sum_{l=1}^K s_l(t) e^{j(p-1)\omega\tau_l} \quad (2.2.2)$$

And the total array output ' $\mathbf{X}$ ' can be written as

$$\mathbf{X}(t) = \mathbf{A}\mathbf{S}(t) \quad (2.2.3)$$

Where,

$\mathbf{A} = [\mathbf{a}_{\theta_1} \ \mathbf{a}_{\theta_2} \ \dots \ \mathbf{a}_{\theta_K}]$, is an M by K matrix, called the *propagation matrix*, whose columns are the K direction vectors and,

$$\mathbf{S}(t) = [s_1(t) \ s_2(t) \ \dots \ s_K(t)]^T \quad (2.2.4)$$

The array correlation matrix is defined as $\mathbf{R} = E\{\mathbf{X}(t)\mathbf{X}^H(t)\}$, where the operator $E\{.\}$ denotes the average or expected value and the superscript 'H' denotes the complex conjugate transpose.

From previous discussions,

$$\begin{aligned} \mathbf{R} &= E\{\mathbf{X}(t)\mathbf{X}^H(t)\} \\ &= E\{\mathbf{A}\mathbf{S}(t)\mathbf{S}^H(t)\mathbf{A}^H\} \\ &= \mathbf{A}E\{\mathbf{S}(t)\mathbf{S}^H(t)\}\mathbf{A}^H \\ \mathbf{R} &= \mathbf{A}\mathbf{P}\mathbf{A}^H \end{aligned} \quad (2.2.5)$$

Where the matrix $\mathbf{P}$ is the ' $K \times K$ ' source correlation matrix, that gives the correlation between the different sources.

2.2.1 Effect of noise

The discussion so far has not included noise. Now we consider a noisy environment. It is assumed that the noise at any antenna is uncorrelated with the noise at the other antennas and uncorrelated with the signal sources as well and is modelled as a zero-mean Gaussian random process. It is further assumed that the noise power is the same at all antennas. Let $n_l(t)$ denote the noise at the l^{th} antenna and $\mathbf{N}(t) = [n_1(t) \ n_2(t) \ \dots \ n_M(t)]^T$.

Then,

$$\mathbf{X}(t) = \mathbf{A}\mathbf{S}(t) + \mathbf{N}(t);$$

And,

$$\begin{aligned} \mathbf{R} &= E\{\mathbf{X}(t)\mathbf{X}^H(t)\} \\ &= E\{(\mathbf{A}\mathbf{S}(t) + \mathbf{N}(t))(\mathbf{S}^H(t)\mathbf{A}^H + \mathbf{N}^H(t))\} \\ &= \mathbf{A}E\{\mathbf{S}(t)\mathbf{S}^H(t)\}\mathbf{A}^H + \mathbf{A}E\{\mathbf{S}(t)\mathbf{N}^H(t)\} + E\{\mathbf{N}(t)\mathbf{S}^H(t)\}\mathbf{A}^H + E\{\mathbf{N}(t)\mathbf{N}^H(t)\} \\ &= \mathbf{A}\mathbf{P}\mathbf{A}^H + 0 + 0 + \sigma^2 \mathbf{I} \\ \mathbf{R} &= \mathbf{A}\mathbf{P}\mathbf{A}^H + \sigma^2 \mathbf{I} \end{aligned} \quad (2.2.6)$$

The above discussion is based on obtaining the array output in a continuous time basis. In real conditions we sample the array output at discrete time intervals. Each such sample is called a 'snapshot' of the array output. We obtain an estimate of the array correlation matrix from these samples. The sampling interval should be such that the Nyquist criterion for reconstructing the signals should be satisfied.

If we obtain 'N' snapshots of the array output at times $t_1, t_2, \dots, t_N$, then the estimate of the array correlation matrix is obtained from the following expression,

$$\mathbf{R} = \frac{1}{N} \sum_{n=1}^N \{\mathbf{X}(t_n)\mathbf{X}^H(t_n)\} \quad (2.2.7)$$

Where, $\mathbf{X}(t_n) = [x_1(t_n), x_2(t_n), \dots, x_M(t_n)]$ is a vector whose elements $x_1(t_n), x_2(t_n), \dots, x_M(t_n)$ represent the output of the antennas in the ULA along with the additive noise.

The accuracy of the estimated array correlation matrix depends on the number of snapshots and the sampling rate.

In all the simulations in this report, we consider the case when the number of sources, $K=2$ and the number of antennas, $M=6$. For this case $\mathbf{R}$ is a 6×6 matrix and $\mathbf{P}$ is a 2×2

matrix of the form $\begin{pmatrix} P_1 & \gamma^* P_1 \\ \gamma P_1 & |\gamma|^2 P_1 \end{pmatrix}$, Where P_1 is the power of the first source and γ is the correlation-coefficient of the sources given by $E\{s_1(t)s_2^*(t)\}$ and superscript $*$ denotes the complex conjugate.

2.2.2 Effect of Mutual Coupling

Two array elements are mutually coupled when the signal in one of the elements induces a signal in the other element. The presence of unknown mutual coupling degrades the performance of high resolution direction finding and Space-Time adaptive processing methods. Mutual coupling has at least two important effects on an array's performance. One is that the far-field pattern of a single array element is different when it is embedded in the array from what it is in isolation. The other effect is that the radiation impedance of an element is different in the array than it is in isolation. Both effects are due to scattering of radiation from neighbouring elements. Mutual Coupling between array elements is inversely related to the spatial separation between them. Its effect is usually negligible when the separation between them is more than a few wavelengths.

Mutual coupling is usually modelled using the mutual coupling matrix(C). It is usually a sparse square matrix of dimensions $M \times M$ and its diagonal elements will all be unity. The elements of the mutual coupling matrix give us the mutual coupling coefficients between the antennas. For example if we assume that all the antennas in the array are mutually coupled to their immediate neighbours, then the mutual coupling matrix will be a tridiagonal matrix whose diagonal elements are unity and the off-diagonal elements will give us the mutual coupling coefficients between the antennas.

The propagation matrix of the array in the presence of mutual coupling is the product of the mutual coupling matrix and the normal propagation matrix.

i.e $A_m = CA$ and $R = A_m P A_m + \sigma^2 I$ (2.2.8)

2.2.3 Effect of Sensor Failure

Now we consider a situation when an antenna encounters a failure. In such an event the antenna does not pick up the signal from the sources. Hence the output will contain only the noise which arises from the other circuitry involved. Let us assume that a total of 'N' snapshots are needed and that a particular antenna encounters a failure just after the p^{th} snapshot has been taken. Since the entire pack of 'N' snapshots are being taken in a very short time we can assume that the antenna does not recover from the failure for the remaining (N-p) snapshots. Let $\mathbf{X}_f(t_n)$ denote a snapshot with failure, taken at time (t_n) and let us assume that the failure occurs at the second antenna.

Then,

$$\mathbf{X}_f(t_n) = [x_1(t_n), n_2(t_n), \dots, x_M(t_n)]. \quad (2.2.9)$$

We can observe that since the second antenna has encountered a failure we have replaced $x_2(t_n)$, which represents the total signal along with the additive noise, by $n_2(t_n)$, which represents only the noise. Now the Correlation matrix is estimated by the following equation.

$$\mathbf{R} = \frac{1}{N} \left[\sum_{n=1}^p \{ \mathbf{X}(t_n) \mathbf{X}^H(t_n) \} + \sum_{n=p+1}^N \{ \mathbf{X}_f(t_n) \mathbf{X}_f^H(t_n) \} \right] \quad (2.2.10)$$

This chapter has provided an introduction to the DoA estimation problem. The basic theory behind the problem is explained with the help of analytical expressions and the general terminology used in the literature were introduced. The chapter also explained the effects of various factors like noise, mutual coupling, sensor failure and correlation between sources. The analytical expressions that are used to model these effects were

also introduced. In the next chapter, these expressions and terminology will be used to explain the conventional methods that are widely used for DoA estimation.

Chapter 3

Review of existing methods of DoA estimation

The DoA estimation problem is encountered in various fields like mobile communications, RADAR target tracking, Unmanned Aerial Vehicles (UAV), Geo-location, Navigation, Weapons targeting etc, and hence it is one of the most researched areas in array signal processing. Since it has been an active research area for decades and a vast literature is available, this chapter gives a detailed introduction to some of the popular conventional methods for DoA estimation. Recently, a number of neural network based approaches have also been reported to overcome some of the drawbacks of the conventional methods. These neural methods are also explained in detail in this chapter. The problem of DoA estimation under sensor failure is also receiving significant research interest currently. This chapter also gives an overview of the methods that are used to handle this problem.

3.1 A Review of Conventional Methods of DoA estimation

The most popular conventional methods of DoA estimation can be classified into three categories as spectral estimation methods, maximum likelihood methods and subspace based methods. Spectral estimation methods have their roots in time series analysis. They involve formulation and calculation of spatial spectra whose peaks correspond to

the DoAs of the sources. This is similar to mechanically steering the antenna array to different directions and measuring the output power. The resolution powers of these methods depend on the aperture of the array and are found to have some limitations in this regard. Maximum likelihood methods operate by forming a likelihood function that is the joint probability density function of the sampled data given the DoAs. Then a search is performed for the directions that maximise logarithm of this function. The maximisation of the Log-likelihood function is a nonlinear optimization problem and is performed using iterative schemes when a closed form solution is not available. This makes these methods computationally complex. Subspace based methods rely on the orthogonality of the signal and noise subspaces and on the fact that the steering vectors corresponding to the DoAs of the sources lie in the signal subspace. The signal subspace is spanned by the eigenvectors that correspond to the larger eigenvalues of the correlation matrix. These methods are found to be far more accurate than the other classes of methods and are found to have very good resolution capabilities. A good introduction to the commonly used methods of DoA estimation is given in [14]. MUSIC and ESPRIT are by far the most popular among them and are explained below.

MUSIC- Multiple Signal Classification Algorithm:

The MUSIC algorithm [9] is a relatively simple and efficient eigenstructure method of DoA estimation. It has many variations and is perhaps the most studied method in its class. A good introduction to this method and its variations is given in [14]. Some of them are explained below

Spectral MUSIC:

Spectral MUSIC is the standard form of the MUSIC algorithm. This method involves the estimation of the noise subspace from the available samples. This can be done by either eigenvalue decomposition of the estimated array correlation matrix or singular value decomposition of the data matrix, with its N columns being the N snapshots or the array single vectors.

As discussed in the previous chapter, the array correlation matrix is given by the expression,

$$\mathbf{R} = \frac{1}{N} \sum_{n=1}^N \{ \mathbf{X}(t_n) \mathbf{X}^H(t_n) \} \quad (3.1)$$

The noise subspace is estimated to be the space spanned by the $M-K$ eigen vectors corresponding to the $M-K$ smallest eigenvalues of the array correlation matrix and the signal subspace is estimated to be spanned by the eigenvectors corresponding to the remaining K largest eigenvalues of the correlation matrix. The MUSIC algorithm relies on the fact that the signal and noise subspaces are orthogonal to each other and hence the steering vector corresponding to the DoA of a signal must be orthogonal to the noise subspace.

Once the noise subspace has been estimated, a search for K directions is made by looking for steering vectors that are as orthogonal to the noise subspace as possible. This is normally accomplished by searching for peaks in the MUSIC spectrum given by

$$P_{MV}(\theta) = \frac{1}{|\mathbf{s}_\theta^H \mathbf{U}_N|^2}$$

where ' $\mathbf{U}_N$ ' denotes an M by $M-K$ dimensional matrix with its $M-K$ columns being the eigenvectors corresponding to the $M-K$ smallest eigenvalues of the array correlation matrix, and ' $\mathbf{s}_\theta$ ' denotes the steering vector corresponding to the direction .

It should be noted that instead of using the noise subspace and searching for directions with steering vectors orthogonal to this subspace, we may also use the signal subspace and search for directions with steering vectors contained in this space [16]. It is advantageous to use the one with the smaller dimensions.

The Cramer-Rao Lower Bound (CRLB) gives the theoretically lowest value of the covariance of an unbiased estimator. For the case of a single source, the DoA estimate made by the MUSIC method asymptotically approaches the CRLB, that is, when the number of snapshots increases infinitely, the best possible estimate is made. For multiple sources, the same holds for the large SNR cases, that is, when the SNR approaches infinity [17], [18].

Root MUSIC:

This method is based on the view that the DoA estimation problem can also be considered as a problem of finding the roots of a complex polynomial and is hence known as Root-MUSIC [17]. Root-MUSIC is applicable only when a ULA is used. It solves a polynomial rooting problem as contrast to the identification and localization of spectral peaks using spectral MUSIC. Root-MUSIC has a better performance than spectral MUSIC [25] under certain conditions.

SSMUSIC:

A new algorithm called SSMUSIC is proposed in [11], which is based on a linear algebraic relationship between the standard subspace based model of the array correlation matrix and a special signal plus interference model. It has the advantage of estimating the power of the sources along with their DoAs. An approach that uses the wavelet transform of the snapshots instead of the direct snapshots to calculate the array correlation matrix is proposed in [12]. It tries to exploit the duality between the spatial and temporal domain. The method has been found to improve the performance in a low SNR environment and also yield a significant reduction in the computational time.

ESPRIT -Estimation of Signal Parameters via Rotational Invariance Techniques :

ESPRIT [10] is a computationally efficient method of DOA estimation. It uses multiple identical arrays whose corresponding elements have identical displacement with respect

to each other. This condition doesn't however need that we should have separate arrays. For example, consider a 4 element ULA. By considering the first 3 elements as a sub-array and the last 3 elements as another sub-array, this ULA can be viewed as two identical arrays of three elements whose corresponding elements have identical displacement from each other.

Now DoA estimation with ESPRIT algorithm is explained. Let R_1 be array correlation matrix for the first array and R_2 be that of the second array. Let U_1 and U_2 denote the M by K matrices whose columns are the K largest eigen-values of R_1 and R_2 respectively. Both U_1 and U_2 span the same signal subspace. Hence they are related by a non-singular transformation matrix ' ψ '.

Thus,

$$U_1 = \psi U_2$$

The propagation matrices of the 2 ULAs are related by the expression $A_2 = A_1 \phi$, where ϕ is a K by K diagonal matrix whose k^{th} diagonal element is given by

$$\phi_{kk} = e^{j2\pi\delta \cos \theta_k}$$

Where, δ is the displacement between the corresponding elements of the arrays, measured in wavelengths.

Now the matrices U_1 and U_2 are related to their propagation matrices by another unique non-singular matrix T .

Thus,

$$U_1 = A_1 T$$

And

$$U_2 = A_1 \phi T$$

From the above expressions we can write

$$T \psi T^{-1} = \phi$$

From the above equation we observe that the eigen-values of ψ are the diagonal elements of ϕ .

We obtain an estimate of ψ from the array outputs and calculate its eigen-values and equate them to the diagonal elements of ϕ . These equations give the DoAs of the K sources and are obtained as

$$\theta_k = \cos^{-1} \{ \text{Arg}(\lambda_k) / 2\pi\delta \}$$

How one obtains the estimate of ψ from the array outputs has led to many variations of ESPRIT.

One such procedure called TLS-ESPRIT is given below.

1. Estimate the array correlation matrices of the 2 arrays from their outputs and calculate their eigen-values and eigen-vectors and form the matrices U_1 and U_2 .
2. Form a $2K$ by $2K$ matrix $\begin{pmatrix} U_1^H \\ U_2^H \end{pmatrix} (U_1 \ U_2)$ and find its eigen-vectors. Form the matrix V whose columns are these eigen-vectors.
3. Partition V into 4 K by K matrices as

$$V = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix}$$

4. The matrix $-V_{11}V_{22}^{-1}$ is the estimate of ψ . Calculate the eigen-values (λ_k) of this matrix.
5. Obtain the DoA estimates from these eigen-values from the expression

$$\theta_k = \cos^{-1}\{Arg(\lambda_k) / 2\pi\delta\}$$

ESPRIT is observed to a computationally efficient algorithm than MUSIC. But to resolve a given number of sources ESPRIT requires larger number of antennas than MUSIC.

Among the conventional methods MUSIC and ESPRIT are known to be more accurate and yield good performance at low signal-to-noise ratio conditions. However both these methods involve computationally intensive operations like eigen-decomposition and matrix inversion. This makes them unsuitable for real time operation. Further these methods require calibrated antennas with uniform features [13].

3.2 Neural Network Based DoA estimation

In order to counter the difficulties in the conventional methods, neural network based methods have been attempted in recent years and have been found to yield good results. The following are the advantages of the neural networks when compared to the conventional methods.

1. Significant reduction in the computational complexity. This makes neural methods more suitable for real time applications.
2. Better noise immunity

The conceptual form of a neural network based super-resolution system is given below.

Figure 3.1 Basic Structure of a Neural DoA Estimator

The direct output from the sensor array is sent to the pre-processor. The pre-processor converts this data to a form that is suitable to be fed to the neural network. The pre-processor helps in making the learning process of the neural network an efficient one. A detailed description of the need for the pre-processor and its various designs are given in the next chapter.

The neural network first undergoes the training process which is usually done offline. The training vectors must be chosen from all possible scenarios of source positions and separations. Once the training process is complete the frozen neural network can be used for online estimation of the DoAs of the sources. The accuracy and efficiency of the estimation depends on the design of the pre-processor, the choice of the neural network, the choice of the training algorithm for the neural network and the training samples used for collection.

3.3 A Review of Existing Neural Network Based DoA estimation Methods

This section gives a brief overview of some basic types of neural networks and neural network based DoA estimation.

Multilayer Perceptron network (MLP):

MLP is a feed forward neural network with one or more hidden layers of neurons. This has been found to have good classification capabilities and is a universal function approximator. The Back-propagation algorithm is the most popularly used method to train the MLP. Back-propagation algorithm involves 2 passes during a training cycle. In the forward pass the network output is calculated for a particular input. In the backward pass this output is compared with the desired output to produce the error signal that is propagated through the network in the opposite direction. During the backward pass the synaptic weights are adjusted so that the error is minimized.

Figure 3.2 Forward pass of the multi layer perceptron

Figure 3.3 Backward pass of the multilayer perceptron

Radial Basis Function Neural Network (RBFNN):

The RBFNN is a three layer feed forward neural network. It is known for its universal approximation, regularization and optimization capabilities. The hidden layer neurons in this type of network have a radial basis activation function. The Gaussian function is the most commonly used activation function. The chief advantage of RBFNN over MLP is their simplicity in design.

The output of an RBFNN with Gaussian activation function is given by the equation

$$f(x_i) = \alpha_o^i + \sum_{k=1}^h \alpha_k^i \exp\left(-\frac{1}{(\sigma_k^i)^2} \|x_i - \mu_k^i\|^2\right) \quad (3.3.1)$$

where α_k^i is the weight vector ,

μ_k^i is the centre vector ,

and σ_k^i is the spread of the k^{th} neuron at the i^{th} training cycle

A detailed survey of the existing neural network based methods that are used for the DoA estimation problem is given in [13]. The survey has concluded that among the neural network based methods the RBFNN based methods are much faster than the MLP based methods and is receiving more research attention in recent years.

Recently a DoA estimation scheme based on a RBF neural network was proposed in [6] and the following paragraphs explain this method in detail. The method follows the basic structure of the DoA estimator that was shown in Figure 3.1. The data from the sensors is fed to a pre-processor. The pre-processor removes the unwanted information in the direct data and converts it to a form that is suitable for the learning process of the RBFNN. The processed data is then passed to the RBFNN for training. The trained network is then used in the actual setup for real-time DoA estimation. These procedures are described in detail below.

Firstly the sensor array data is passed to the pre-processor. The pre-processor eliminates the unwanted phase information in the direct data by using the array correlation matrix. Since the diagonal elements of the correlation matrix do not contain any information about the DoAs, they are eliminated. The rest of the elements of the correlation matrix are arranged into a single vector in the following fashion,

$$b = [R_{21}, \dots, R_{M1}, R_{12}, \dots, R_{M2}, \dots, R_{1M}, \dots, R_{M(M-1)}]$$

Since the elements of the correlation matrix are complex numbers, the real and imaginary parts of these numbers are arranged next to each other in the vector.

This vector is then normalised by dividing all the elements by the norm of the vector. The normalised vector is then used for training or for DoA estimation in the real system.

Network training is done by acquiring a set of vectors using the above procedure such that each vector corresponds to a unique position of the sources. Since only a ULA is considered in [6] the training vectors are to be selected such that the entire field of view (-90degrees to +90 degrees) is uniformly covered. Once the training set is obtained a suitable procedure can be employed to train the RBF network. Upon completion of the training the frozen network is used for DoA estimation in the actual setup.

While using the trained network in the actual setup for DoA estimation, the sensor array outputs are preprocessed just like they were done for generating a training vector. This vector is fed to the input of the RBF network and its output will give us the DoA estimates of the sources.

Two main scenarios are considered in [6]. One case is when the angular separation between the sources is 2 degrees and the other case is when the angular separation is 5 degrees. For these two cases the DoA estimation performance of the RBF network is compared with the performance of the MUSIC algorithm. The RBF network is found to consume smaller time to perform the DoA estimation than the MUSIC algorithm and is also yield better results when the sources are correlated with each other.

However this method has the following deficiencies,

1. The DoA estimation error is very high when the sources are near the endfire direction of the array.
2. The number of hidden neurons in the neural network has been chosen based on trial and error experiments which can be very time consuming.
3. The problem of DoA estimation under sensor failure has not been considered. The effects of other factors such as mutual coupling, inequality in the powers of the sources and random errors in the positions of the sources have also not been studied.

3.4 DoA Estimation under sensor failure

Sensor Failure is a common phenomenon in array signal processing and DoA Estimation under sensor failure has been under research in recent years. When a sensor is said to have encountered failure, it can no longer receive the signals emitted by the sources and its output contains only the noise that come from the circuitry involved at the receiver. Apart from the physical failure of the sensor there are other scenarios which can be considered to be equivalent to a sensor failure. For example it is possible that some of the data acquired from a particular sensor may be corrupted or may be lost or considered unusable for many reasons. Such a scenario is called the missing data scenario in general literature [32] and is equivalent to the failure of a sensor. It is also possible that some of the sources suddenly disappear from the vicinity or the field of view of the antenna. DoA estimation methods in a scenario involving such suddenly disappearing sources are recently appearing in literature [35]. This scenario can also be considered to be equivalent to the sudden failure of all the sensors in the sensor array. Thus DoA estimation under sensor failure is an important problem that is being actively researched recently and is also the main concern of this thesis.

The subspace based methods of DoA estimation that were discussed in the previous section involve calculation of the correlation matrix, which in turn involves acquiring several snapshots of the sensor array output. Sensor failure can happen during the process of acquiring the snapshots rendering some of the snapshots to be wrong and unusable. The correlation matrix calculated using these wrong data is bound to be erroneous and is found to produce false peaks in the spectrum of the conventional methods.

For handling sensor failures, many modifications to the conventional algorithms have been proposed [32] - [34]. The approach taken in [32] is to arrive at the maximum likelihood estimate of the array correlation matrix under sensor failure and then use this estimated correlation matrix to perform the DoA estimation with the help of standard algorithms such as MUSIC. The situation considered in [32] and the procedure

mentioned to obtain the Maximum Likelihood estimate of the correlation matrix are given below. Consider that there are a total of M_1 sensors in the array and that all these M_1 sensors are working normally while obtaining the first N_1 snapshots. The $M_1 \times M_1$ correlation matrix that can be formed by using these N_1 snapshots be denoted R_1 . After obtaining N_1 snapshots, one or more of the sensors encounter failure and only M_2 sensors are working normally for the next N_2 snapshots and similarly M_k sensors are working normally for the next N_k snapshots and so on. Let R_k denote the $M_k \times M_k$ correlation matrix that can be obtained using the N_k snapshots that are obtained when the M_k sensors are working normally. The procedure mentioned in [32] to arrive at the maximum likelihood estimate of the correlation matrix under the scenario described above is as follows,

1. Compute the R_k values for $k= 1, 2 \dots, p$.
2. Compute the S_{kj} values for $k= 1, 2 \dots, p$ and $j=1, 2 \dots, M_k$ according to the equation

$$S_{kj} = \sum_{i=1}^k \{N_k [0 \ I_j] R_k [0 \ I_j]^T\}$$

3. Compute g_j for $j = M_{k+1}, M_{k+1}+1, \dots, M_k$, according to the equation

$$g_j = \frac{S_{kj}^{-1}u}{u^* S_{kj}^{-1}u}$$

$$\text{Where } u = [1 \ 0 \ 0 \ \dots 0]^T$$

4. Compute $j = 1, 2, \dots, M_1$ for $j = 1, 2, \dots, M_1$, according to the equation

$$d_j^{-1} = \frac{g_j^* S_{kj} g_j}{N_1 + N_2 + \dots + N_k}$$

5. Construct $D = \text{diag}\{d_{m_1}, \dots, 1\}$
6. Construct H using the relations below.

$$g_j = [1 \ h_j]^T \text{ for } j = 1, 2, \dots, M_1$$

and

$$H = \begin{bmatrix} 1 & 0 & 0 & . & . & . & 0 \\ & 1 & . & . & . & & 0 \\ & & 1 & & & & . \\ & & & . & & & . \\ h_{m_1} & & & & . & & . \\ & h_{m_1-1} & & & & . & 0 \\ & & . & . & . & h_2 & 1 \end{bmatrix}.$$

7. Compute the Maximum Likelihood estimate of the Array correlation matrix using the relation $R_{ml} = (HDH^*)^{-1}$

Now the DoA estimation can be performed using a known algorithm like MUSIC with the help of the obtained Maximum Likelihood estimate of the Correlation matrix. In [32] the studies have been performed with different algorithms and the performance of the algorithms is found to improve when using this strategy when there is a sensor failure.

However it must be noted here that this method requires the knowledge of the exact time and position of the sensor failure. It also imposes additional computational burden.

Another method of array failure correction based on genetic algorithm has been proposed in [33]. This method attempts to restore the beam pattern of the failed array to the beam pattern that was observed before failure. This is done by calculating a few weights that are to be multiplied to the array output so that the effective beam pattern is similar to the pattern that was observed before the failure. The calculation of these weights is done using a genetic algorithm. Different mating schemes for such a genetic algorithm are studied [33]. However, the performance of DoA estimation algorithms with this correction has not been studied. Further this method also requires the knowledge of the exact time and position of the sensor failure and also imposes additional computational burden.

The approaches in [32] and [33] require the knowledge of the element and the instant at which the failure has occurred. In [34], a method for the detection of the sensor failure and compensation for it has been proposed. This is done by forming confidence factors for each sensor output after obtaining a snapshot and continuously reviewing the history of confidence factors for all sensors. This imposes additional computational burden and requires knowledge of the maximum tolerable error for forming the confidence factors.

However it is very difficult to ascertain the exact instant at which the failure occurred since the entire process of acquiring the snapshots is done in a very short interval of time. As a result it becomes impossible to determine which of the acquired snapshots are wrong and which of them are right. Hence it is desirable to have a scheme that can perform DoA estimation with the wrong snapshots itself and hence eliminating the need for the knowledge of the instant at which the failure occurs or the location of sensor that has failed. This problem is dealt with in this thesis.

To summarise, this chapter presented an introduction to the existing methods of DoA estimation. The principles behind these methods were explained analytically and a detailed description of the procedures involved is listed. The existing methods of DoA estimation under sensor failure were also described and the drawbacks of these schemes were highlighted. A brief introduction to neural networks was given and the basic structure of a neural network based DoA estimator was explained. A new neural DoA estimation scheme is designed in the coming chapters which aims to overcome the drawbacks of the methods described in this chapter.

Chapter 4

The Preprocessing Scheme

The central idea behind a neural network approach to the direction of arrival problem is that it can be viewed as a nonlinear function mapping problem from the space of the sensor array outputs to the space of the DoAs. The output from the sensor array contains information about several parameters of the environment apart from the directions of the sources. Hence the first issue to be addressed is whether the signals from the sensors can be directly fed to the inputs of the neural network in order to achieve our objectives or do they need some processing before being fed to the neural network.

This chapter deals with the need for the pre-processor and explains the various existing pre-processing methods and their drawbacks. A new pre-processing scheme has been proposed in thesis to overcome the deficiencies in these existing methods. This scheme is explained in detail in this chapter.

4.1 Need for the Pre-Processor

Using the data from the sensors as such without any pre-processing is called as the direct data domain approach. The output at the p^{th} sensor at time t is given by the following expression.

$$x_p = \sum_{l=1}^K s_l(t) e^{j(p-1)\omega(d/C)\sin\theta_l} + n_p(t) \quad (4.1)$$

We can observe that along with the information about the direction of arrival it contains information about the following as well.

1. Time
2. The initial phase of the modulating signal
3. The temporal value of the modulating signal and the carrier.
4. The Amplitude/power of the carrier signal.
5. The temporal value of the additive noise.

In order to perform the function approximation efficiently it is essential to eliminate as much unwanted information as possible. This will reduce the dimensionality of the search space of the training algorithm for the neural network.

Thus using the direct data approach has the following deficiencies

1. High dimension of the search space resulting in an inefficient performance of the neural network.
2. High sensitivity to noise as no steps are taken to alleviate the effects of this noise.
3. In the event of a sensor failure it would directly affect the performance.

So we need to design a pre-processor for the neural network which will help us overcome these deficiencies.

The following are the guidelines that we intend to follow in designing the pre-processor.

1. Eliminate unwanted information and hence reduce the dimensionality of the search space
2. Reduce the effect of the additive noise if possible.
3. The attempt to reduce the number of parameters and the effect of noise should not imply that the resolution and other characteristics that are vital for DoA estimation are affected.

4.2 Design of the Pre-Processor

Since it is clear that we cannot use the direct data to achieve our objectives our first task is to decide on a new set of parameters that follows the above guidelines and will provide the input to the neural network

4.2.1 Pre-Processor Using the Phase Differences

Considering a single source and a uniform linear array, we can observe that the signal received at a sensor is just a delayed version of the signal received at an adjacent sensor. The following equations for the outputs of the p^{th} sensor and the $(p+1)^{\text{th}}$ sensor demonstrate this clearly.

Output of the p^{th} sensor:

$$x_p = s(t) e^{j(p-1)\omega(d/C)\sin\theta} + n_p(t) \quad (4.2)$$

Output of the $(p+1)^{\text{th}}$ sensor:

$$x_{p+1} = s(t) e^{jp\omega(d/C)\sin\theta} + n_{p+1}(t) \quad (4.3)$$

The phase difference between them:

$$\varphi = 2\pi \sin\theta \quad (4.4)$$

We can conclude from the above expressions that the phase differences between the signals at adjacent sensors are a function of the direction of arrival of the source. Further these phase differences do not contain information about the amplitude / power / initial phase of the modulating signal/carrier. They are also not dependent on the temporal values of the modulating signal or the carrier. Hence these phase differences can be used instead of the direct data for calculation of the DoA.

A pre-processor based on the above facts and conclusions has been designed in [3]. It calculates the phase differences between the signals at adjacent sensors and arranges these values of phase differences into a single vector and feeds this vector to the neural network. Since the ensemble of phase differences between the different pairs of adjacent antennas is unique to the position of the source we can train the neural network to establish a function mapping from the space of these phase differences to the actual DoA.

Figure 4.2 gives us a schematic for the pre-processor using the phase differences of the sensors. The phase comparators are used to measure the phase differences between the designated sensors.

Figure 4.2 Schematic for Phase difference pre-processing Scheme.

The signal at the sensor at the far left is considered as the phase reference. Considering a signal source at an angle (θ), the phase differences between the adjacent elements are given by the following expressions.

$$\begin{aligned} \varphi_{1i} &= 2\pi d_{1i} \sin \theta + \delta\varphi_{1i} \\ d_{1i} &= \frac{1}{2} \sum_{j=1}^{i-1} L_j, \quad i = 2, 3, 4 \end{aligned} \quad (4.5)$$

Where,

1. φ_{1i} and $\delta\varphi_{1i}$ are the phase difference and phase error (noise) between the first sensor and the i^{th} sensor respectively.
2. d_{1i} is the normalised physical spacing between the first sensor and the i^{th} sensor. The normalisation is made with respect to the wavelength at the operating frequency.

3. The adjacent sensor spacings (Normalised to half wavelengths at operating frequency) are L_1 between sensor 1 and 2, L_2 between sensors 2 and 3 and L_3 between sensors 3 and 4.
4. θ is the direction of arrival.

Once these phase differences are measured they can be fed to the neural network to perform the mapping of the unknown function mapping from the space of these phase differences to the space of the DoAs. It has been established that the relationship between the measures phase differences and the DoAs is generally a continuous function with small changes in the phase differences yielding small changes in the DoAs.

Goodwin in [36] has established that all the DoA related electrical phase information for a single source can be extracted using four sensors and three phase comparators.

Even though the phase differences as such contain the necessary and sufficient information about the direction of arrival of the signal, each member of the ensemble of phase differences contains discontinuities due to the fact that they can be expressed only as modulo 2π . These discontinuities can complicate the learning task of the neural network and hence need to be smoothened. This is achieved by using the sine and cosines of the phase differences instead of using the phase differences as such.

This pre-processing scheme is summarized below

1. Obtain the phase comparator outputs.
2. Calculate the sines and cosines of these values.
3. Arrange the values obtained in step 2 into a single vector.

However this approach has its deficiencies. Firstly like the data domain approach there has been no effort taken to reduce the effect of noise. Secondly and more importantly this approach cannot be used in a scenario with multiple sources. This is due to the fact

that it is not possible to establish a one to one correspondence between the DoAs of the multiple sources and the phase comparator outputs. Hence we need to look at parameters other than the phase differences if we need to design the pre-processor for multiple sources.

4.2.2 Pre-Processor Using the Correlation Matrix

Among the conventional methods it has been found that subspace based methods are able to achieve higher resolution and are found to be more accurate than the other methods. They are also found to be more immune to noise. Subspace based methods first arrive at an estimate of the signal or noise subspace. The signal and noise subspaces are orthogonal to each other. Subspace based methods exploit this orthogonality of the subspaces to arrive at an estimate of the DoAs. In order to estimate the signal/noise subspace these methods first try to estimate the correlation matrix of the sensor array output.

Drawing inspiration from these conventional methods, there have been attempts to design a suitable pre-processor based on the correlation matrix [6] [27]. It can be seen that the diagonal elements of the correlation matrix contain only the information regarding the power of the sources while all the information regarding the direction of arrival is confined to the non-diagonal elements. The non diagonal elements do not contain unwanted information like the initial phase of the signal, temporal value of the modulation signal or the carrier. Further if the noise is assumed to be a Gaussian and zero mean process and if the estimate of the correlation matrix is made from sufficient number of snapshots of the array output then the non-diagonal elements of the correlation matrix will be free from the effects of noise. Thus it is an attractive idea to feed the non-diagonal elements of the correlation matrix as input to the neural network.

However there are a few problems in using these non-diagonal elements as such. Firstly they contain information about the powers of the signal sources. This can be eliminated

to an extent by normalising these non-diagonal elements. The next problem is that these non-diagonal elements of the correlation matrix are complex valued while most of the neural network architectures available are real-valued. Thus we need to somehow convert these complex valued elements to real valued elements without losing the DoA information present in it. This can be done by presenting the real and imaginary parts of the complex valued elements to separate input nodes of the neural network.

A simple pre-processing scheme following the above discussion is to put all the non-diagonal elements into a single vector, normalise it and feed the real and imaginary parts of these elements to the input of the neural network. This scheme is used in [6] in conjunction with a radial basis function (RBF) neural network. A successful classification of closely separated sources has been reported and the method has been found to perform better than MUSIC when the sources are correlated. This scheme is summarised below.

1. Get 'N' snapshots of the array output vector.
2. Estimate the array correlation matrix.
3. Extract the lower triangular half of this matrix.
4. Form a single vector (b) from these elements by appending all the columns to the first column, in the order of their column number.
5. Form the vector (t1) whose elements are the real parts of the corresponding elements in vector 'b' and the vector (t2) whose elements are the imaginary parts of the corresponding elements in vector 'b'.
6. Form the training vector (t) by appending vector t2 to the vector t1.

4.2.3 Proposed Preprocessing Scheme

Though the scheme used in [6] was sufficient to resolve two closely placed sources an inspection of the results presented in [6] reveals that the error in the estimates of the DoAs is very high when the DoAs of the sources are near ± 90 degrees. The reasons for this high error have been investigated in this thesis and it is found to be due to the

improper design of the Pre-Processor. We will now discuss in detail about these findings and propose a few improvements to the Pre-Processor.

The high error is due to the fact that the obtained values of the real parts of the elements of the array correlation matrix when DoAs of the sources are near +90 degrees are almost the same as the values obtained when the sources are near -90 degrees. The imaginary parts though different, are very small in magnitude when compared to the real parts. Hence the data when the DoAs are near 90 degrees is almost the same as the data when the DoAs are near -90 degrees. Hence the RBF network finds it difficult to differentiate between these data and gives a very high error. This situation can be illustrated by the following example. Consider a four element uniform linear array and consider two unit power sources corresponding to DoAs of -87 and -89 degrees. The estimated correlation matrix for this scenario is given below

$$R = \begin{bmatrix} 2.0000 & -2.0000 + 0.0048i & 2.0000 - 0.0096i & -1.9999 + 0.0144i \\ -2.0000 - 0.0048i & 2.0000 & -2.0000 + 0.0048i & 2.0000 - 0.0096i \\ 2.0000 + 0.0096i & -2.0000 - 0.0048i & 2.0000 & -2.0000 + 0.0048i \\ -1.9999 - 0.0144i & 2.0000 + 0.0096i & -2.0000 - 0.0048i & 2.0000 \end{bmatrix}$$

Similarly the estimated correlation matrix when the sources are at +87 and +89 degrees will be

$$R = \begin{bmatrix} 2.0000 & -2.0000 - 0.0048i & 2.0000 + 0.0096i & -1.9999 - 0.0144i \\ -2.0000 + 0.0048i & 2.0000 & -2.0000 - 0.0048i & 2.0000 + 0.0096i \\ 2.0000 - 0.0096i & -2.0000 + 0.0048i & 2.0000 & -2.0000 - 0.0048i \\ -1.9999 + 0.0144i & 2.0000 - 0.0096i & -2.0000 + 0.0048i & 2.0000 \end{bmatrix}$$

As discussed earlier, only the non-diagonal elements of the correlation matrix are fed to the neural network. A comparison of these non diagonal elements of the above matrices

will reveal that they are almost the same except for the sign of their imaginary parts. Further the imaginary parts are very small when compared to the corresponding real parts. Hence the input to the neural network when the sources are at -89 and -87 degrees is almost the same when the sources are at +87 and +89 degrees. This makes it difficult for the neural network to learn the function and hence gives and abnormally large error in these cases.

This problem can be overcome by considering the magnitude and phase angle of the elements of the array correlation matrix instead of the real and imaginary parts.

The total signal at $(p+1)^{\text{th}}$ antenna is given by,

$$x_{p+1} = \sum_{l=1}^K s_l(t) e^{j\omega\tau_l} + n_{p+1}(t) \quad (4.6)$$

Now let us consider the case when $K=2$ and $M=6$.

Consider the $(p+1, q+1)^{\text{th}}$ element of the correlation matrix. It is given by the expression,

$$\begin{aligned} R_{p+1, q+1} &= E\{x_{p+1}x_{q+1}^*\} = E\{[s_1(t)e^{j\omega p\tau_1} + s_2(t)e^{j\omega p\tau_2} + n_{p+1}(t)][s_1(t)e^{j\omega q\tau_1} + s_2(t)e^{j\omega q\tau_2} + n_{q+1}(t)]^*\} \\ &= E\{s_1(t)s_1^*(t)e^{j\omega(p-q)\tau_1}\} + E\{s_1(t)e^{j\omega p\tau_1}s_2^*(t)e^{-j\omega p\tau_2}\} + E\{s_1(t)e^{j\omega p\tau_1}n_{q+1}^*(t)\} \\ &\quad + E\{s_2(t)e^{j\omega p\tau_2}s_1^*(t)e^{-j\omega q\tau_1}\} + E\{s_2(t)s_2^*(t)e^{j\omega(p-q)\tau_2}\} + E\{s_2(t)e^{j\omega p\tau_2}n_{q+1}^*(t)\} \\ &\quad + E\{n_{p+1}(t)s_1(t)e^{-j\omega q\tau_1}\} + E\{n_{p+1}(t)s_2^*(t)e^{-j\omega q\tau_2}\} + E\{n_{p+1}(t)n_{q+1}^*(t)\} \\ &= P_1e^{j\omega(p-q)\tau_1} + \gamma e^{j\omega(p\tau_1 - q\tau_2)} + 0 + \gamma^* e^{j\omega(p\tau_2 - q\tau_1)} + P_2e^{j\omega(p-q)\tau_2} + 0 + 0 + 0 + 0 \\ E\{x_{p+1}x_{q+1}^*\} &= P_1e^{j\omega(p-q)\tau_1} + \gamma e^{j\omega(p\tau_1 - q\tau_2)} + \gamma^* e^{j\omega(p\tau_2 - q\tau_1)} + P_2e^{j\omega(p-q)\tau_2} \end{aligned} \quad (4.7)$$

That gives us the expression for any element in the array correlation matrix. Evidently it is a complex number. Now we need to calculate the magnitude and angle of this

particular complex number. For simplicity of calculations let us assume that $P_1 = P_2 = 1$ and $\gamma = 0$. This means the sources are perfectly uncorrelated.

$$\begin{aligned}
 \text{angle}[E\{x_{p+1}x_{q+1}^*\}] &= \text{angle}[P_1 e^{j\omega(p-q)\tau_1} + \gamma e^{j\omega(p\tau_1 - q\tau_2)} + \gamma^* e^{j\omega(p\tau_2 - q\tau_1)} + P_2 e^{j\omega(p-q)\tau_2}] \\
 &= \text{angle}[e^{j\omega(p-q)\tau_1} + e^{j\omega(p-q)\tau_2}] \\
 &= \omega(p-q) \left[\frac{\tau_1 + \tau_2}{2} \right] \\
 \text{angle}[E\{x_{p+1}x_{q+1}^*\}] &= \frac{\omega(p-q)d(\sin \theta_1 + \sin \theta_2)}{2C} \quad (4.8)
 \end{aligned}$$

$$\begin{aligned}
 \text{mag}[E\{x_{p+1}x_{q+1}^*\}] &= \text{mag}[e^{j\omega(p-q)\tau_1} + e^{j\omega(p-q)\tau_2}] \\
 &= 2 \cos \left\{ \omega(p-q) \left[\frac{\tau_1 - \tau_2}{2} \right] \right\} \\
 \text{mag}[E\{x_{p+1}x_{q+1}^*\}] &= 2 \cos \left\{ \frac{\omega(p-q)d(\sin \theta_1 - \sin \theta_2)}{2C} \right\} \quad (4.9)
 \end{aligned}$$

Now let us consider the case when $P_1 = P_2 = 1$ and $\gamma = 1$. This means that the sources are perfectly correlated (i.e) both the signals are identical.

$$\begin{aligned}
 \text{angle}[E\{x_{p+1}x_{q+1}^*\}] &= \text{angle}[P_1 e^{j\omega(p-q)\tau_1} + \gamma e^{j\omega(p\tau_1 - q\tau_2)} + \gamma^* e^{j\omega(p\tau_2 - q\tau_1)} + P_2 e^{j\omega(p-q)\tau_2}] \\
 &= \text{angle}[e^{j\omega(p-q)\tau_1} + e^{j\omega(p\tau_1 - q\tau_2)} + e^{j\omega(p\tau_2 - q\tau_1)} + e^{j\omega(p-q)\tau_2}] \\
 &= \omega(p-q) \left[\frac{\tau_1 + \tau_2}{2} \right] \\
 \text{angle}[E\{x_{p+1}x_{q+1}^*\}] &= \frac{\omega(p-q)d(\sin \theta_1 + \sin \theta_2)}{2C} \quad (4.10)
 \end{aligned}$$

$$\begin{aligned}
 \text{mag}[E\{x_{p+1}x_{q+1}^*\}] &= \text{mag}[e^{j\omega(p-q)\tau_1} + e^{j\omega(p\tau_1 - q\tau_2)} + e^{j\omega(p\tau_2 - q\tau_1)} + e^{j\omega(p-q)\tau_2}] \\
 &= 4 \cos \left\{ \frac{\omega(p-q)d(\sin \theta_1 - \sin \theta_2)}{2C} \right\} \quad (4.11)
 \end{aligned}$$

From the above expressions we can see that, the angles and magnitudes of the elements of the array correlation matrix are functions of either the sum or difference of the sines of the DoAs. Given this information the neural network can perform the inverse operation of extracting the DoAs from the sum of their sines. This approach can solve the problem of high errors when the DoAs are near ± 90 degrees.

Since the array correlation matrix is hermitian in nature, the elements in the upper triangular half of the matrix are the complex conjugates of the elements in the lower triangular half. Hence it is sufficient that we use either half of the correlation matrix as both halves contain the same information about the DoAs of the sources.

The revised pre-processing scheme used to generate the training data based on the above discussions is summarised below:

1. Get 'N' snapshots of the array output vector
2. Estimate the array correlation matrix
3. Extract the lower triangular half of this matrix
4. Form a single column vector (**b**) from these elements by appending the columns to the first column, in the order of their column number.
5. Form the vector (t1) whose elements are the magnitudes of the corresponding elements in vector 'b' and the vector (t2) whose elements are the phase angles of the corresponding elements in vector 'b'.
6. Form the training vector (t) by appending vector t2 to the vector t1.

The training set is formed by generating such training vectors corresponding to different directions of arrival. However, during the training process, the vectors in the training set are presented in random order. The testing data is also obtained using the same procedure above.

When the sources are very close to each other i.e when $\theta_1 \approx \theta_2$, the magnitude of the elements of the correlation matrix is always a constant. This makes it difficult to extract the DoA information embedded in it. Hence in such cases magnitude information can be discarded and only the information in the angles of the complex elements of the correlation matrix can be used. It must be noted however that including the magnitude information will help when the sources are sufficiently separated. Thus when the sources are sufficiently close to each other we can obtain the training data using only the angles.

4.3 Performance evaluation of Pre-Processing Schemes

A study of pre-processing schemes for high resolution neural network based DoA estimation is given in [27]. The study compares the performance of the correlation matrix based pre-processing scheme mentioned in section 4.4.2 with a few other schemes. One of the other pre-processing schemes studied in [27] involves computing the orthonormal eigenvectors spanning the noise subspace and feeding the elements of these vectors to the neural network. Another one involves using an adaptive filter that uses a superresolution algorithm such that the entire system becomes a recursive superresolution system. The study indicates that in order to obtain high resolution capabilities it is imperative to use the elements of the correlation matrix and concludes that the pre-processing scheme mentioned in section 4.2.2 i.e the one used in [6] provides the best performance and is the easiest to implement. Hence we choose to compare the performance of our new pre-processing scheme with the pre-processing scheme used in [6].

In order to evaluate the performance of the pre-processing scheme, we compare the DoA estimates obtained from the same neural network but with different pre-processors. All other training and testing conditions are also to be maintained the same. We use the 'newrb' function of MATLAB to generate a RBFNN with 50 hidden neurons and compare the DoA estimates obtained when using the proposed pre-processing scheme and the scheme mentioned in [6]. Figure 4.3.1 & 4.3.2 present the

estimates of the DoAs obtained from the RBFNN, using the pre-processing scheme in [6] while Figure 4.3.3 & 4.3.4 give the results obtained using the new scheme, for the same RBFNN. The angular separation between the sources was assumed to be 2° . Two hundred training vectors and hundred testing vectors were used. It can be seen that the new pre-processing scheme yields better results for the DoA estimation especially when both the sources are near ± 90 degrees. Hence, in all our subsequent studies, the new pre-processing scheme has been used for the neural methods.

Figure 4.3.1 DoA estimates for Source1 with the pre-processing scheme in [6]

Figure 4.3.2 DoA estimates for Source2 with the pre-processing scheme in [6]

Figure 4.3.3 DoA estimates for Source1 with the proposed pre-processing scheme

Figure 4.3.4 DoA estimates for Source2 with the proposed pre-processing scheme
This chapter discussed the need for the pre-processor and provided an overview of the various existing pre-processing schemes. The new pre-processing scheme proposed to overcome deficiencies of the existing scheme was explained in detail and its performance was evaluated and compared with the existing scheme. The next chapter gives a study of the performance of a DoA estimation scheme based on the recently published Minimal Resource Allocation Neural Network (MRAN) and use the new pre-processing scheme developed in this chapter.

Chapter 5

Improved DoA Estimation Using Minimal Resource Allocation Neural Network (MRAN)

One of the main drawbacks of the RBFNN based method in [6] is that the optimal number of hidden neurons has been chosen based on trial and error studies. This implies that a significant amount of time should be spent for training networks of different sizes before zeroing on an optimal architecture. In order to overcome this problem the use of MRAN is proposed in this thesis.

A brief introduction to the MRAN algorithm is given in this chapter and its application to the DoA estimation problem is investigated. The detailed studies that have been made to evaluate the DoA estimation performance of MRAN under different environments have also been explained in this chapter.

5.1 A Brief Description of MRAN

MRAN [7] is a sequential learning algorithm for designing radial basis function neural networks. The algorithm starts with zero neurons and gradually adds neurons until the required error criterion is satisfied and the parameters of the network are updated using an Extended Kalman filter. Additionally, the algorithm has a pruning strategy to delete the neurons that have been continuously inactive for a long period of time. This algorithm results in parsimonious network architecture and has been observed to have fast convergence properties.

In the classical approach to radial basis functions (RBF) network implementation, the basis functions are usually chosen as Gaussian and the number of hidden units (i.e., the centres and widths of the Gaussian functions) is fixed a priori based on the properties of the input data. The weights connecting the hidden and output units are estimated by a linear least squares method.

The MRAN learning algorithm succeeds the Resource Allocation Network (RAN) [38] and the Resource Allocation Network via Extended Kalman filter (RANEKF) [39] research initiatives. Both are algorithms that exhibit sequential learning and start with no computational units. The number of hidden units need not be selected *a priori*. When each input is received, a new hidden unit is allocated, based on the ‘novelty’ of the input. A new pattern is considered novel if that input pattern is far away from the existing centres and if the error between the network output and the desired output is large. If no additional units are added, the parameters of the existing neurons, such as centres, widths and weights, are updated. RAN and RANEKF differ in the manner in which the parameters are updated. RAN uses a Least Means Square (LMS) algorithm, while the RANEKF uses an Extended Kalman Filter. It has been shown that using the EKF improves the rate of convergence and further reduces the network complexity.

The primary disadvantage of using RAN and RANEKF is that the appended neurons are not removable. Neurons that were previously active could be redundant to the network performance during the mature stages in the life of a neural network. However, these neurons still persist in the RAN and RANEKF networks. This drawback has been aptly addressed in the MRAN. The MRAN learning algorithm allocates a new hidden neuron based on the novelty of observation and when the need arises, prunes those hidden units, which have insignificant contributions to the outputs of the network. Therefore the architecture of MRAN will be adjusted automatically to fit closely with the observation data.

The MRAN adopts the basic idea of RANEKF and augments it with a pruning strategy to remove the hidden neurons that have insignificant contribution to the total output of the network. This results in a minimal RBF network. The addition of the pruning strategy in the learning algorithm may result in oscillations in the transition of the number of hidden neurons. To overcome the oscillations and make the transition smooth, an additional growth criterion based on the Root Mean Square (RMS) error of the network output over a sliding window was introduced in the MRAN. This reduces the effect of measurement noise in the data. The addition of a third growth criteria and a pruning strategy resulted in an algorithm, which obtained a smaller network size, without compromising the accuracy of the results.

The pruning is done by observing the output of each of the hidden nodes for a period of time, and then removing the node that has not been contributing a significant output for that period and hence the output of that node falls below a threshold value during the period. The detection and removal of such hidden units during the learning process ensures that a more parsimonious network topology is realized.

Thus, MRAN involves the allocation of new hidden units as well as adjusting the network parameters and pruning of unwanted hidden neurons. This algorithm has been summarised below.

For each observation (x_i, y_i)

compute overall network output

$$f(x_i) = \alpha_o^i + \sum_{k=1}^h \alpha_k^i \exp\left(-\frac{1}{(\sigma_k^i)^2} \|x_i - \mu_k^i\|^2\right)$$

Where, α_k^i is the weight vector connecting the k^{th} hidden neuron to the output

μ_k^i is the centre vector of the k^{th} hidden neuron

and σ_k^i is the centre vector of the k^{th} hidden neuron

calculate the parameters required in the growth criterion:

$$\varepsilon_i = \max\{\varepsilon_{\max} \gamma^i, \varepsilon_{\min}\}, (0 < \gamma < 1),$$

$$\|e_i\| = \|y_i - f(x_i)\|,$$

$$e_{rms}^i = \sqrt{\frac{\sum_{j=i-(n_w-1)}^i e_j T e_j}{n_w}}$$

apply the criterion for adding hidden units:

$$\text{if } (\|e_i\| > \varepsilon_{\min}) \text{ and } (\|x_i - \mu_{nr}^i\| > \varepsilon_i) \text{ and } (e_{rms}^i > e_{\min}^i)$$

then allocate a new hidden unit $h+1$ with:

$$\alpha_{h+1}^i = e_i$$

$$\mu_{h+1}^i = x_i$$

$$\sigma_{h+1}^i = \kappa \|x_i - \mu_{nr}^i\|$$

else adjust the network parameters α_o^i and $\mu_k^i, \sigma_k^i, \alpha_k^i$

$(k = 1, \dots, h)$ using EKF:

$$w_i = w_{i-1} + K_i e_i,$$

$$K_i = P_{i-1} B_i \left[R_i + B_i^T P_{i-1} B_i \right]^{-1}$$

$$P_{i-1} = \left[I - K_i B_i^T \right] P_{i-1} + q_0 I$$

end if;

check the criterion for pruning hidden units:

compute the output vectors of all hidden units o_k^i ,

$(k = 1, \dots, h)$ according to the eqn.

$$O_{ik}^i = \alpha_{ik}^i \exp\left(-\frac{1}{(\sigma_k^i)^2} \|x_i - \mu_k^i\|^2\right)$$

find the largest absolute hidden unit output vector

$$\|o_{\max}^i\|;$$

compute the normalized output vectors $r_k^i, (k = 1, \dots, h)$
according to the eqn. ,

$$r_{lk}^i = \frac{\|o_{lk}^i\|}{\|o_{l,\max}^i\|}$$

if all elements in r_k^i are less than δ for n_w consecutive
observations,

then remove the k^{th} hidden unit,

reduce the dimensionality of EKF,

end if;

5.2 Performance Study of DoA Estimation with MRAN and the Improved Pre-processing Scheme

In this section, the performance of DoA estimation for MUSIC, RBFNN and MRAN are evaluated in different environments.

Studies have been performed with 2 sinusoidal sources ($K=2$) of unit power and with 6 elements ($M=6$) in the ULA. The sources are assumed to be uncorrelated. In all the studies, the angular separation between the two sources was kept at 2 degrees. The number of snapshots for this study is taken as 200. Half-wavelength inter-element spacing is assumed in the ULA, similar to [6]. RBFNN was simulated using the MATLAB's 'newrb' function. For RBFNN, the number of hidden neurons was fixed at 50 as in [6] and the width of the neurons was fixed as 1. The parameters of MRAN were chosen as $e_{\min}=0.001$, $e_{\min1}=0.001$, $e_{\max}=1$, $\epsilon_{\min}=0.001$, $\gamma=0.9980$, $\delta=0.0001$, $\kappa=0.95$. For both the neural schemes, the same training and testing data sets were used. 200 training vectors were used such that the target DoAs is uniformly distributed

in the range (-90° to $+90^{\circ}$). 100 testing vectors obtained in a similar fashion were used for testing.

5.2.1 Performance in an ideal environment:

Table 5.2.1 presents the performance comparison of the three algorithms in an ideal environment that is free from noise, sensor failure or mutual coupling. 200 training samples and 100 testing samples were used in the simulations and each sample corresponds to different angular positions (DoAs) of the sources. The RMS error for the 100 testing samples is called the RMS testing error and the RMS error for the 200 training samples is called the RMS Training error. The values given in the table are average values of the different parameters obtained over 50 independent trials. It can be seen from the table, that MUSIC outperforms both the neural schemes in terms of accuracy. Among the neural schemes, MRAN produces lower error than RBFNN. However, it can be noted that MUSIC algorithm takes 32.7 seconds for DoA estimation of the 100 testing samples, which is three orders of magnitude greater than that of the neural schemes. Then, among the neural schemes MRAN takes lesser time for training and testing and produces a more compact network (31 Vs 50 of RBFNN). Figure 5.2.1 and 5.2.2 show the neuron history and the logarithm of the RMS training error for MRAN, respectively.

	RMS Testing error	RMS training error	No of hidden neurons	CPU time for training (seconds)	CPU time for testing (seconds)
MRAN	0.1626	0.1627	31	154.5	0.058
RBFNN	0.1741	0.1954	50	6.3	0.082
MUSIC	0	NA	NA	NA	32.7

Table 5.2.1 – Performance comparison in an ideal environment (No failure and No Noise)

Figure 5.2.1 Neuron history of MRAN during training

Figure 5.2.2 RMS error for MRAN during training

5.2.2 Performance with noise:

In this case, the signals are corrupted with noise. The additive noise at each antenna is modelled as a zero mean, white, Gaussian random process and is assumed to be uncorrelated with the signal and uncorrelated with the noise at other antennas as well. Table 5.2.2 gives the mean value of the testing error obtained over 50 trials when the network was trained with no noise case and tested with noisy data at different SNRs. From Table 5.2.2, it can be seen that MUSIC can not handle noisy data whereas neural schemes are able to handle with little deviations from no noise case for SNR of 20 db. For SNR of 10 db, the errors for both neural schemes are higher. However in all these cases MRAN produces less error than RBFNN.

	RMS Testing error		
	No noise	SNR 20dB	SNR 10dB
MRAN	0.1626	0.1629	1.08
RBFNN	0.1741	0.1747	1.23
MUSIC	0	47.92	53.61

Table 5.2.2 – Performance comparison With Noise and No failure)

Performance comparison with Cramer-Rao Bound in the Presence of noise:

Here the performance the three algorithms in the presence of noise are compared with the Cramer-Rao bound (CRB). A set of simple algebraic expressions for calculating the CRB is given in [37]. In order to perform this study we fix the positions of the sources and calculate the variance of the estimation error over 1000 independent trials.

The positions of the sources are fixed at 0 and 2 degrees. All the other aspects of the experimental setup are the same as for section 5.2.1. Table 5.2.3 gives the variance of the estimation error for the source at 0 degrees while Table 5.2.4 corresponds to the source at 2 degrees. Three different SNRs are considered and the corresponding values of the CRB are also given. It can be observed that for High SNRs, MUSIC algorithm nearly attains the Cramer Rao Bound and its variance is far lower than of the neural network based methods. However as the SNR decreases the variance in the error for MUSIC becomes very high. This is due to the fact that the two sources become irresolvable in the MUSIC spectrum at lower SNR and a spurious peak is picked up as a signal source. The neural network based methods perform far better than MUSIC algorithm at lower SNRs since they don't encounter the problem of resolvability. Further it can be observed that among the neural network based methods MRAN has a lower variance than the RBFNN method.

It must be mentioned here that comparisons with the Cramer Rao bound are usually done for studying the performance of unbiased estimators only. The neural network based methods cannot be considered as unbiased estimators since they give a finite error even when there is no noise. However this small error can be reduced to an arbitrarily small level by choosing a smaller error criterion while training.

	RMS Testing error		
	SNR 50dB	SNR 20dB	SNR 10dB
MRAN	0.00046	0.01461	0.0462
RBFNN	0.00075	0.2389	0.8044
MUSIC	1.8x10-8	16.9	17.07
CRB	1.99x10-8	4.37x10-6	0.00011

Table 5.2.3 – Performance comparison with CRB for source1 (0 degrees)

	RMS Testing error		
	SNR 50dB	SNR 20dB	SNR 10dB
MRAN	0.00046	0.01461	0.0462
RBFNN	0.00075	0.2396	0.8066
MUSIC	1.8x10-8	17.07	21.62
CRB	1.99x10-8	4.37x10-6	0.00011

Table 5.2.4 – Performance comparison with CRB for source 2 (2 Degrees)

5.2.3 Mutual Coupling:

The effect of mutual coupling between the antennas is studied in this experiment. We consider the same experimental setup as considered in section 5.2.1. The mutual coupling model is chosen to be the one considered in [31]. It is assumed that each antenna is affected by mutual coupling from its adjacent antennas only. The effect of mutual coupling due to the other far away antennas is assumed to be negligible. The mutual coupling coefficients are modelled as a random variable with zero mean and a variance of 0.01. Table 5.2.5 gives a comparison of the errors and time consumed by the various algorithms. The results presented are the average of 50 independent trials. We can observe that the neural network based methods produce the same error as given by them for Section 5.2.1 but require a higher number of neurons. This is due to the fact that the training of the neural networks was done with the data that included the mutual coupling effects and the networks had to learn some additional parameters (The mutual Coupling Co-efficients). It can also be observed that MUSIC algorithm gives a very high error with mutual coupling, as it is unable to resolve the two closely spaced sources in many trials. Figure 5.2.3 and 5.2.4 gives the training characteristics of MRAN. The neuron history and RMS error history during the training process are plotted.

	RMS Testing error	RMS training error	No of hidden neurons	CPU time for training (seconds)	CPU time for testing (seconds)
MRAN	0.1670	0.1672	39	206.7	0.069
RBFNN	0.1781	0.1946	60	7.1	0.098
MUSIC	25.101	NA	NA	NA	32.7

Table 5.2.5 – Performance with mutual coupling- No Noise case

Figure 5.2.3 Neuron history of MRAN during training with mutual coupling

Figure 5.2.4 RMS error for MRAN during training with mutual coupling

Mutual Coupling–With noise:

Here we study the effects of mutual coupling in the presence of noise. Table 5.2.6 give a comparison of the errors given by the various algorithms under different SNRs. Here again we can observe that the errors of the neural network based methods with mutual coupling are comparable to their errors in section 5.2.2.

	RMS Testing error		
	No noise	SNR 20dB	SNR 10dB
MRAN	0.1670	0.1834	1.21
RBFNN	0.1781	0.1971	1.31
MUSIC	25.101	69.31	73.41

Table 5.2.6 – Performance with Mutual Coupling-With Noise

5.2.4 Non-Uniform Array:

The study so far has been concerned with a uniform linear array. In practice however it may not be possible to realise such an array. To study the effect of non-uniformity we consider the same setup considered in section 5.2.1. Now we introduce a random error in the position of each element. The position error is assumed to be a Gaussian random variable with zero mean and a standard deviation of 0.03(in wavelengths). The position errors of the different elements of the ULA are assumed to be independent of each other. Table 5.2.7 gives the values of the training characteristics of the different algorithms. As in the previous experiments training is done with the noise free data. The results given are the average values obtained from 50 independent trials. It can be observed that neural network based methods give almost the same error as for the uniform linear array case whereas MUSIC algorithm gives a deteriorated performance. The high error of MUSIC is due to the fact that it was unable to resolve the closely spaced sources in some of the trials. Further it can also be observed that the number of hidden neurons for this case is the same as for the Uniform Linear array case. Figures 5.2.5 and 5.2.6 give the neuron and training error history of MRAN.

	RMS Testing error	RMS training error	No of hidden neurons	CPU time for training (seconds)	CPU time for testing (seconds)
MRAN	0.1629	0.1631	31	206.7	0.058
RBFNN	0.1746	0.1957	50	7.1	0.082
MUSIC	32.14	NA	NA	NA	32.7

Table 5.2.7 – Performance with a Non-Uniform Array-No Noise case

Figure 5.2.5 Neuron history of MRAN during training with a Non-uniform array

Figure 5.2.6 RMS error for MRAN during training with a Non-Uniform Array

Non-Uniform array-Performance with noise

Here we study the performance of the three algorithms with a non-uniform linear array in the presence of noise. For MRAN and RBFNN we use the frozen network that was obtained after training with noise free data. Table 5.2.8 gives the RMS testing error under different SNRs for the three algorithms. For the neural network based methods the error obtained in this case is found to be very close to the results with a Uniform linear array. However MUSIC algorithm gives a higher error for this case than the Uniform linear array.

	RMS Testing error		
	No noise	SNR 20dB	SNR 10dB
MRAN	0.1631	0.1635	1.12
RBFNN	0.1746	0.1752	1.25
MUSIC	32.14	69.31	73.41

Table 5.2.8 – Performance with a Non-Uniform Array-With Noise

5.2.5 Unequal Source Power:

Here we study the performance of the algorithms when the power of one of the sources is significantly higher than the other source. We assume the same experimental setup as in section 5.2.1, with the only difference that the power of the source1 is 100 times the power of source2. Table 5.2.9 gives the training characteristics of the three algorithms for this scenario. Training is done with the noise-free data. We find that all the three algorithms give the same performance as given by them for the equi-power case.

	RMS Testing error	RMS training error	No of hidden neurons	CPU time for training (seconds)	CPU time for testing (seconds)
MRAN	0.1638	0.1638	31	153.7	0.058
RBFNN	0.1744	0.1924	50	6.3	0.082
MUSIC	0	NA	NA	NA	32.7

Table 5.2.9 – Performance with Unequal Source Powers -No Noise case

Figure 5.2.7 Neuron history of MRAN when trained with unequal source powers

Figure 5.2.7 RMS error for MRAN when trained with unequal source powers

Unequal Source Power - Performance Comparison With noise

Here we study the effect of unequal power sources in the presence of noise. Table 5.2.10 gives the RMS testing error for the various algorithms under different SNRs. We can observe that the neural network based methods give almost the same error given by them for equi-power sources whereas MUSIC algorithm gives a higher error.

	RMS Testing error		
	No noise	SNR 20dB	SNR 10dB
MRAN	0.1638	0.1641	1.14
RBFINN	0.1744	0.1753	1.27
MUSIC	0	62.26	68.14

Table 5.2.10 - Performance comparison with Unequal Power Sources- with Noise

In summary, this chapter provided a brief introduction to the recently published MRAN and studied its application to the DoA estimation problem. The performance of MRAN based DoA estimation method was evaluated and compared with the conventional MUSIC algorithm and the existing RBFNN based scheme. The performance study was done under different conditions of noise and in the presence of different anomalies like mutual coupling, unequal powers of the sources and non uniformities in the array. The performance evaluation of these algorithms in the presence of sensor failures which is the main topic of this work is described in the next chapter.

Chapter 6

DoA Estimation In the Presence of Sensor Failures

This chapter deals with the problem of DoA estimation under sensor failures using the MRAN based DoA estimation scheme studied in the previous chapter. When a sensor in the Uniform Linear Array (ULA) encounters failure it fails to pick up signals from the environment and its output is assumed to consist only of the noise generated from the other circuitry involved. When the n^{th} sensor in the ULA encounters failure, the n^{th} row and n^{th} column of the correlation matrix are affected.

Conventional methods like MUSIC and ESPRIT are unable to cope with sensor failure since they are heavily dependent on the accuracy of the correlation matrix. To handle sensor failures, many modifications to the conventional algorithms have been proposed [32] - [34]. In [32], a method to form the maximum likelihood estimate of the correlation matrix in the presence of sensor failure has been proposed and is shown to improve the performance of DoA estimation algorithms like MUSIC under failure. A method of array failure correction based on genetic algorithm has been proposed in [33]. This method restores the beam pattern of the failed array to the beam pattern that was observed before failure. However, the performance of DoA estimation algorithms with this correction has not been studied. The approaches in [32] and [34] require the

knowledge of the element and the instant at which the failure has occurred. In [35], a method for the detection of the sensor failure and compensation for it has been proposed. This is done by forming confidence factors for each sensor output after obtaining a snapshot and continuously reviewing the history of confidence factors for all sensors. This imposes additional computational burden and requires knowledge of the maximum tolerable error for forming the confidence factors.

Hence we consider the problem of DoA estimation under sensor failures without prior knowledge of the element or time of failure and without performing any additional operation for the compensation of failure.

6.1 Performance with sensor failure but no noise:

To study the performance under the sensor failure case, we vary the instant at which the failure occurs and also the position of the sources. For this study, we first present the result when the 3rd sensor in the ULA has failed just after obtaining the 150th snapshot. For the failure cases, the neural networks that are used for testing are the same obtained under no failure cases as given table 5.2.1 (Training is done under no failure, no noise cases). Figure 6.1.1 shows the DoA error in degrees for the source S_1 for different source angles for the three algorithms in the presence of the failure at the 150th snapshot. For example, when the sources are at -9 (S_1) and -11 (S_2) degrees, MUSIC gives an error of 34.7 degrees, while RBFNN gives an error of 7.4 degrees and MRAN gives 3.6 degrees. Also, it can be noted these errors become significant for MUSIC when the source angles are at their extremes while for MRAN they are still small. Next, we study the case where the position of the sources is fixed and the instant at which failure occurs is varied from 0 to 200th snapshot. Figure 6.1.2 shows the total DoA error versus the failure instant for the sources at -9 and -11 deg. From Figure 6.2.2, it is clear that the neural network based methods are more robust to sensor failure than the MUSIC algorithm. Further the performance of MRAN is better than that of the RBFNN.

During the training process MRAN had higher number of tuneable parameters as it tunes the centres, weights and spread of the neurons whereas the RBFNN tunes just the weights. This gives MRAN higher generalisation capability and is hence able to give lower error in the presence of sensor failure. However this comes at a cost of higher training time.

Figure 6.1.1 Performance as a function of source position - No noise, Failure at 150th snapshot

Figure 6.1.2 Performance as a function of failure instant - No Noise, sources at -9° and -11°

6.2 Performance with sensor failure and noise:

Here the performance of the three algorithms is considered in an environment that is similar to the one considered in section 6.1, except that the signals are corrupted with noise. Figures 6.2.1 and 6.2.3 present the total error against the source position for two values of SNRs, namely 20 db and 10 db. The neural schemes produce much less error than MUSIC and among the neural schemes MRAN performs better. As one expects, with a higher noise level the error are more. Figures 6.2.2 and 6.2.4 show the total error against the instant at which the failure occurs for different values of SNR. Here also MRAN does better than other schemes. Table 6.2.1 presents these points in a quantitative manner.

	RMS Testing error		
	No noise	SNR 20dB	SNR 10dB
MRAN	3.1	5.39	13.70
RBFNN	16.47	18.46	24.46
MUSIC	76.58	78.86	79.01

Table 6.2.1 – Performance comparison with failure and Noise

Figure 6.2.1 Performance as a function of source position - SNR=20dB, Failure at 150th snapshot

Figure 6.2.2 Performance as a function of failure instant - SNR=20dB, sources at - 9° and -11°

Figure 6.2.3 Performance as a function of source position - SNR=10dB, Failure at 150th snapshot

Figure 6.2.4 Performance as a function of failure instant - SNR=10dB, sources at -9° and -11°

6.3 Varying the element at which failure occurs:

The previous experiments assumed that the failure occurs at the third element of the ULA. Now the effect of varying the element that encounters failure will be studied. The performance under scenarios with failure and noise as in the previous cases will be considered while the sensor at which failure occurs is varied. As a typical case the studies are performed when the 1st element is assumed to encounter failure. Figures 6.3.1 and 6.3.2 show the DoA estimation error when varying the position of the sources and the failure instant in the no noise case. They correspond to figures 6.1.1 and figures 6.1.2. It is found that the errors follow a similar trend as in section 6.1. Figures 6.3.3 and 6.3.4 correspond to an SNR of 20 dB and figures 6.3.5 to 6.3.6 correspond to an SNR of 10 dB. Here again the errors are observed to follow a trend that is similar to the cases with the failure occurring at the 3rd element. The effects of failure at the other sensors are given in appendix A and are observed to follow the same trends. It can be concluded from these observations that the element at which the failure occurs has very little effect on the error for the neural network based methods.

Figure 6.3.1 Performance as a function of source position - No Noise, Failure at 150th snapshot, at 1st element

Figure 6.3.2 Performance as a function of failure instant - No Noise, Sources at -9° and -11°, failure at 1st element

Figure 6.3.11 Performance as a function of source position - SNR=20dB, Failure at 150th snapshot, at 1st element

Figure 6.3.12 Performance as a function of failure instant - SNR=20dB, Sources at -9° and -11°, failure at 1st element

Figure 6.3.21 Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 1st element

Figure 6.3.22 Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 1st element

6.4 Performance with Coherent Sources:

In this section, the performance of the three algorithms is considered in a scenario similar to the one in section 6.1, but in the presence of coherent sources. MUSIC algorithm cannot handle such coherent signals. Figure 6.4.1 gives the plot of the MUSIC spectrum when the sources are uncorrelated and coherent for the scenario in section 5.2.1 (No Noise and No failure) with the sources assumed to be at -9° and -11° . We can observe that there are 2 distinct peaks at -9° and -11° for the uncorrelated case. For the coherent case we can observe that the spectrum is degenerated and contains spurious peaks.

Figure 6.4.1 MUSIC spectrum for uncorrelated and coherent sources.

Since MUSIC algorithm cannot handle coherent signals, only the neural network based algorithms are compared in this section. Figures 6.4.2 and 6.4.3 correspond to the no noise case. For comparison results of uncorrelated sources are also given in these figures. Like in the previous cases we first fix the failure instant and vary the positions

of the sources. Figure 6.4.2 gives the DoA estimation error for this case while figure 6.4.3 gives the DoA estimation error when we fix the position of the sources and vary the failure instant. It is found that MRAN gives a smaller error than the RBFNN in both cases. It can also be observed that MRAN gives lower error than MRAN both in the presence of coherent sources and in the presence of uncorrelated sources. The effect of noise is also studied for the similar cases. Figures 6.4.4 and 6.4.5 correspond to an SNR of 20 dB and figures 6.4.6 to 6.4.7 correspond to an SNR of 10 dB. We can observe the same trends in these figures also and hence we can conclude that MRAN gives a better performance than the RBFNN in the presence of coherent sources and sensor failure.

Figure 6.4.2 Performance as a function of source position - No Noise, Failure at 150th snapshot, for Coherent Sources

Figure 6.4.3 Performance as a function of failure instant - No Noise, Sources at -9° and -11° , for Coherent Sources

Figure 6.4.4 Performance as a function of source position - SNR=20dB, Failure at 150^{th} snapshot, for Coherent Sources

Figure 6.4.5 Performance as a function of failure instant - SNR=20dB, Sources at -9° and -11°, for Coherent Sources

Figure 6.4.6 Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, for Coherent Sources

Figure 6.4.7 Performance as a function of failure instant - SNR=10dB, Sources at - 9° and -11° ,for Coherent Sources

In summary, DoA estimation under sensor failures, without having any knowledge of the element that has encountered failure or the time of failure is the main result of this chapter. The performance of MRAN for DoA estimation under such conditions was studied and compared with the performance of the conventional MUSIC algorithm and the existing RBFNN based scheme. The major conclusions from these studies are summarised in the next chapter.

Chapter 7

Conclusions & Future Work

This chapter summarises the conclusions based on the studies performed in the earlier chapters and provides a few recommendations for further research as a sequel to this work.

7.1 Conclusions

A new neural network based DoA estimation scheme has been developed using the recent MRAN algorithm and an improved pre-processing scheme. The performance of the MRAN based method has been compared with the existing conventional and neural methods in the presence of sensor failures, noise, and other antenna effects like mutual coupling and random errors in the sensor positions. The main conclusions are:

MRAN based DoA Estimation:

MRAN solves the problem of choosing the number of hidden neurons in a RBFNN. In the RBFNN based scheme proposed by Zooghy et al, the number of hidden neurons was chosen based on trial and error experiments. MRAN automatically chooses an optimal number of hidden neurons during the training process. While the previous

RBFNN based method had chosen 50 hidden neurons, it has been observed that in an ideal environment MRAN gives a better performance with just 31 hidden neurons.

The compact network obtained by using MRAN consumes lesser training time than consumed by the RBFNN. Also, the MUSIC algorithm takes longer time and hence MRAN based method is more suitable for real-time operation.

Performance in the presence of Noise:

The performance of the three methods viz. MUSIC, RBFNN and MRAN have been compared (Section 5.2.2) under different noise levels (No noise, SNR of 20 dB and 10 dB). MUSIC algorithm performs better than all other methods in the no noise case but it gives a larger error in the presence of noise since it fails to resolve the closely spaced sources (angular separation of just 2 degrees) . MRAN and RBFNN on the other hand give a reasonable performance in the no noise case and perform better in the presence of noise. They benefit from the pre-processor design which minimizes the effect of noise by operating only on the non diagonal elements of the correlation matrix. Among the neural network based methods, MRAN gives a slightly better performance than the RBFNN with a fewer number of hidden neurons. The RMS testing errors obtained using the three algorithms MUSIC, RBFNN and MRAN are summarized in Table 7.1.

	RMS Testing error		
	No noise	SNR 20dB	SNR 10dB
MRAN (neurons-31)	0.1626	0.1629	1.08
RBFNN (neurons-50)(	0.1741	0.1747	1.23
MUSIC	0	47.92	53.61

Table 7.1 – Performance comparison with Noise (No failures)

Performance under Antenna Sensor Failure:

The problem of DoA estimation under sensor failure has been studied in detail (Chapter 6) and MRAN is found to perform better than the conventional MUSIC algorithm and also the RBFNN method of Zooghy et al. MUSIC algorithm is unable to resolve two closely spaced sources in the event of a sensor failure. On the other hand, neural network based methods are found to be more tolerant to failure and produce only a slight increase in the RMS testing error. Among the neural network based methods MRAN is found to perform better than the RBFNN method.

Studies were conducted by varying both the time of the failure and the position of the sources. In both cases, MRAN shows a superior performance when compared to other algorithms. Studies were also done by varying the element at which the failure occurs and it was found that the results show the same trend. Thus, MRAN does not need the knowledge of the element at which failure occurs and knowledge of the time of failure. Both these details are needed for the existing methods to perform DoA estimation under sensor failures.

The RMS testing errors obtained for the case when the third sensor in the ULA encounters failure after obtaining the 150th snapshot. And also under different conditions of noise were compared in table 7.2,

	RMS Testing error		
	No noise	SNR 20dB	SNR 10dB
MRAN (Neurons 31)	3.1	5.39	13.70
RBFNN(Neurons 50)	16.47	18.46	24.46
MUSIC	76.58	78.86	79.01

Table 7.2 – Performance comparison with failure and Noise

Effects of Mutual Coupling

In the presence of mutual coupling, .MRAN and RBFNN need a higher number of hidden neurons (39 and 60 neurons respectively) to give testing error that is comparable to the case without mutual coupling This change in the number of hidden neurons can be attributed to the fact that the training data in the presence of mutual coupling contains information about some additional parameters (Mutual coupling coefficients). However from the studies we can observe that the introduction of mutual coupling produces only a very negligible change in the RMS testing error of MRAN as given by table 7.3

	RMS Testing error	RMS training error	No of hidden neurons	CPU time for training (seconds)	CPU time for testing (seconds)
MRAN	0.1670	0.1672	39	206.7	0.069
RBFNN	0.1781	0.1946	60	7.1	0.098
MUSIC	25.101	NA	NA	NA	32.7

Table 7.3 – Performance with mutual coupling- No Noise case

Effects of Non Uniform Array

The performances of the three algorithms with random errors in the positions of the sensors were studied (Section 5.2.6). The positions errors were assumed to be random Gaussian variables with zero mean and a standard deviation of 0.03 (in wavelengths). MUSIC algorithm gives a higher error in such a scenario as it fails to resolve the two closely spaced sources in some of the trials. MRAN and RBFNN on the other hand produced only a negligible increase in the RMS testing error. MRAN achieves a slightly better accuracy than the RBFNN with a fewer number of hidden neurons (RBFNN =50 neurons, MRAN =31 neurons). The trends are similar when noise is also introduced into the setup.

Effects of Improved Pre-processing Scheme:

A new pre-processor has been designed and its performance has been compared with the existing pre-processors. The existing pre-processors produce similar data when the sources are at the two endfire directions of the linear array. Neural networks find it difficult to learn from this data and produce a very high error when the sources are near the endfire directions of the array.

This problem has been overcome by using the magnitude and angles of the complex numbers instead of the real and imaginary parts used in [6]. The mathematical justification behind using this type of pre-processing scheme has been explained.

A comparison of the proposed pre-processing scheme with the scheme used in [6] was done by training and testing the same RBFNN with 50 neurons under identical conditions with the only change being the pre-processor. The RBFNN yielded an RMS testing error of 23.13 degrees when used with the pre-processing scheme mentioned in [6], and the same RBFNN produced an RMS testing error of 0.1741 degrees with the proposed pre-processing scheme.

7.2 Suggestions for Future Work

The main conclusions based on the studies indicate that MRAN based DoA scheme gives a better performance under antenna failures and noise effects. Based on the studies, a few recommendations for further work are summarized below,

Complex Neural Networks:

In the MRAN based approach proposed in this thesis and in the existing neural network based approaches it is necessary for the pre-processor to convert the complex valued output of the ULA into real valued training vectors. Since the existing neural networks are real valued, this becomes a mandatory step and it is usually done by arranging the real parts and imaginary parts of these complex numbers in a definite order before

feeding it to the neural network. This process disconnects the real and imaginary parts of the complex number which is against the very notion of a complex number. Hence it is necessary to study the use of complex valued neural networks so that the connections between the real and imaginary parts of the complex numbers are not lost. Recently, extension to MRAN for Complex valued inputs has been developed in [40] and its application has been studied for the communication channel equalisation problem. It will be useful to study the application of this algorithm to the DoA estimation problem.

Reduction of Training Time:

The training time of MRAN is far higher than that taken by the RBFNN. Since the training is done offline and only the frozen network is used to perform the DoA estimation this training time does not have a serious impact for the scenarios considered. However, this time will be prohibitive when the number of antennas in the linear array is high. Recently a faster version of MRAN, Extended Minimal Resource Allocation Network (EMRAN) has been developed in [30]. EMRAN uses the winner neuron strategy in addition to the growing and pruning strategy of MRAN and tunes the parameters of only the neuron that gives the highest response to the current input. This makes it faster than MRAN while compromising a little on accuracy. Hence it might be interesting to study the use of EMRAN for the DoA estimation problem.

Application of MRAN to other Array Signal Processing Problems:

Beamforming is the process of determining the array output from the signal sources of interest whose DoAs are known and is considered the inverse of the DoA estimation problem. Since MRAN has been applied successfully to the DoA estimation problem it might be interesting to study its application to the Beamforming problem. This study can be also be extended to examine the applicability of MRAN to other array signal processing problems like moving target tracking.

Recent Improvements to MRAN:

Recently a number of improvements have been made to the MRAN algorithm such as the Growing and Pruning RBF (GAP_RBF) algorithm [41], the Generalised Growing and Pruning RBF (GGAP-RBF) algorithm [42] and the Extreme Learning Machine (ELM) [43]. It would be interesting to study the application of these methods to the DoA estimation problem.

Author's Publications

S.Vigneshwaran, N.Sundararajan and P.Saratchandran, "Direction of Arrival Estimation under Sensor Failures Using Minimal Resource Allocation Neural Networks", Submitted to IEEE transactions on Antennas and Propagation., First Submission: 27th December 2004, Revised Manuscript Submitted on 22nd August 2005.

References

- [1] T.Maruyama, Y.Kuwahara, Estimation of propagation structure by means of Hopfield neural network, Antennas and Propagation Society, 2001 *IEEE International Sym*, Volume: 1, 8-13 July 2001.
- [2] L.Badidi, L.Radouane, A neural network approach for DOA estimation and tracking, *Proceedings of the Tenth IEEE Workshop on Statistical Signal and Array Processing*, 2000., 14-16 Aug. 2000, pp. 434-438.
- [3] Ching-Sung Shieh, Chin-Teng Lin, Direction of arrival estimation based on phase differences using neural fuzzy network, *IEEE Transactions on Antennas and Propagation*, Volume: 48 Issue: 7, July 2000, pp.1115 –1124.
- [4] A.H.El Zooghby, C.G.Christodoulou, M.Georgiopoulos, A neural network-based smart antenna for multiple source tracking, *IEEE Transactions on Antennas and Propagation*, Volume: 48 Issue: 5 , May 2000, pp. 768 -776.
- [5] A.H.El Zooghby, C.G.Christodoulou, M.Georgiopoulos, Experimental validation of a neural network direction finder, Antennas and Propagation Society, 1999. *IEEE International Symposium 1999*, Volume: 3, 11-16 July 1999, pp. 1592 - 1595.

- [6] A.H.El Zooghby, C.G.Christodoulou, M.Georgiopoulos, Performance of radial-basis function networks for direction of arrival estimation with antenna arrays, *IEEE Transactions on Antennas and Propagation*, Volume: 45, Issue: 11, Nov. 1997, pp.1611 –1617.
- [7] Lu Yingwei , N.Sundararajan and P.Saratchandran, A sequential learning scheme for function approximation using minimal radial basis function neural networks , *Neural Computation*, Vol. 9, pp. 461 – 478, Feb 1997.
- [8] Lu Yingwei , N.Sundararajan and P.Saratchandran, Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm, *IEEE Transactions on Neural Networks*, Vol. 9, No. 2, pp 308 – 318, March 1998.
- [9] R.O.Schmidt, Multiple emitter location and signal parameter estimation, *IEEE Transactions on Antennas and Propagation*, Vol: 34 Issue: 3, Mar 1986, pp. 276 –280.
- [10] R.Roy, A.Paulraj, T Kailath , Direction-of-arrival estimation by subspace rotation methods – ESPRIT, *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '86.* , Volume: 11, Apr 1986, pp. 2495 –2498.
- [11] M.L. McCloud,L.L. Scharf, A new subspace identification algorithm for high-resolution DOA estimation , *IEEE Transactions on Antennas and Propagation* , Volume: 50 Issue: 10, Oct. 2002, pp. 1382 –1390.
- [12] Wang Buhong, Wang Yongliang, Chen Hui, Spatial wavelet transform preprocessing for direction of arrival estimation, *Antennas and propagation Society International Symposium*, 2002. IEEE, Volume:4, 16-21 June 2002 , pp. 672 -675 vol.4.

- [13] K. L. Du, A. K. Y. Lai, K. K. M. Cheng and M. N. S. Swamy, Neural methods for antenna array signal processing: a review, *Signal Processing (Elsevier)*, Volume 82, Issue 4, April 2002, pp. 547-561.
- [14] L.C.Godara, Application of antenna arrays to mobile communications. II. Beam-forming and direction-of-arrival considerations, *Proceedings of the IEEE*, Volume: 85 Issue: 8, Aug. 1997 pp. 1195 –1245.
- [15] R.D.DeGroat, L.E. Dowling, D.A. Linebarger, The constrained MUSIC problem, *IEEE Transactions on Signal Processing* , Volume: 41 Issue: 3 , March 1993, pp. 1445 –1449.
- [16] A.Barabell, Improving the resolution performance of eigenstructure based direction finding algorithms, *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'83*, Volume: 8, Apr 1983, pp. 336 –339.
- [17] B.Friedlander, A sensitivity analysis of the MUSIC algorithm, *International Conference on Acoustics, Speech, and Signal Processing, ICASSP-89*, May 1989 vol.4, pp. 2811 -2814.
- [18] B.Porat, B. Friedlander, On the asymptotic relative efficiency of the MUSIC algorithm, *International Conference on Acoustics, Speech, and Signal Processing, ICASSP-88.*, April 1988, vol.4, pp. 2376 -2379 .
- [19] H.B.Lee, M.S.Wengrovitz, Resolution threshold of beamspace MUSIC for two closely spaced emitters, *IEEE Transactions on Acoustics, Speech, and Signal Processing* , Volume: 38 Issue: 9 , Sept. 1990, pp.1545 –1559.
- [20] Xiao-Liang Xu, K. Buckley, An Analysis of Beam-Space Source Localization, *IEEE Transactions on Signal Processing*, Volume: 41 Issue: 1 , January 1993, pp. 501

- [21] M.D.Zoltowski, S.D.Silverstein, C.P. Mathews, Beam-space Root-MUSIC for minimum redundancy linear arrays, *IEEE Transactions on Signal Processing*, Volume: 41 Issue: 7, July 1993, pp. 2502 –2507.
- [22] M.D.Zoltowski, G.M.Kautz, S.D.Silverstein, Beam-space Root-MUSIC, *IEEE Transactions on Signal Processing*, Volume: 41 Issue: 1, January 1993, pp: 344
- [23] H.L. Southall, J.A. Simmers, T.H. O'Donnell, Direction finding in phased arrays with a neural network beamformer, *IEEE Transactions on Antennas and Propagation*, Volume: 43 Issue: 12, Dec. 1995, pp. 1369 –1374.
- [24] B.D. Rao, K.V.S Hari, Performance analysis of ESPRIT and TAM in determining the direction of arrival of plane waves in noise, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Volume: 37 Issue: 1989, pp.1990-1995.
- [25] J. Mayhan, L. Niro, Spatial spectral estimation using multiple beam antennas, *IEEE Transactions on Antennas and Propagation*, Volume: 35 Issue: 8, Aug 1987, pp. 897 –906.
- [26] Z.Q. Wu, and C.J. Harris, Adaptive Neurofuzzy Kalman Filter. *Proceedings of FUZZ-IEEE '96 - Proceedings of the fifth IEEE International Conference on Fuzzy Systems*, Vol.2, pages pp. 1344-1350.
- [27] D.Torrieri, K. Bakhru, Neural-network superresolution, *Proceedings of MILCOM 97*, Volume: 3, 2-5 Nov. 1997 vol.3, pp. 1594 -1598.
- [28] R.H Roy, T.Kailath, ESPRIT-estimation of signal parameters via rotational invariance techniques, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Volume: 37 Issue: 7, July 1989, pp: 984 –995.

- [29] Guanghan Xu, S.D. Silverstein, R.H Roy, T.Kailath, Beamspace ESPRIT, *IEEE Transactions on Signal Processing*, Volume: 42 Issue: 2 , Feb. 1994, pp. 349 – 356.
- [30] Li Yan, N.Sundararajan, P. Saratchandran, Analysis of minimal radial basis function network algorithm for real-time identification of nonlinear dynamic systems, *IEE Proceedings- Control Theory and Applications*, Volume: 147 Issue: 4, July, pp. 476 –484.
- [31] A. Lee Swindlehurst, T. Kailath, A performance analysis of subspace based methods in the presence of model errors, Part1 : The MUSIC algorithm, *IEEE Transactions on Signal Processing*, Volume: 40, No 7, July 1992, pp. 1758 -1774.
- [32] E.G.Larsson, P. Stoica, High-resolution direction finding: the missing data case, *IEEE Transactions on Signal Processing*, Volume: 49 , Issue: 5 , May 2001, pp.950 – 958.
- [33] Beng-Kiong Yeo; Yilong Lu, Array failure correction with a genetic algorithm, *IEEE Transactions on Antennas and Propagation*, Volume: 47, Issue: 5, May 1999, pp. 823 – 828.
- [34] T.W.Pirinen, J.Yli-Hietanen, P. Pertila, A.Visa, Detection and compensation of sensor malfunction in time delay based direction of arrival estimation, *Proceedings of the 2004 International Symposium on Circuits and Systems*, 2004. ISCAS '04, Volume 4, 23-26 May 2004, pp: IV - 872-875.
- [35] J.F.Yang, J.H.Hung, Adaptive eigen-subspace algorithms for tracking suddenly disappearing sources, *Radar, Sonar and Navigation*, *IEE Proceedings* -Volume 142, Issue 6, Dec. 1995, pp. 277 – 280.
- [36] M.M.Goodwin, Frequency-independent beamforming, *Final Program and Paper Summaries of IEEE Workshop on Applications of Signal Processing to Audio and*

Acoustics, 1993. 17-20 Oct. 1993, pp. 60-63, Digital Object Identifier 10.1109/ASPAA.1993.379996

- [37] D.N.Swinger; Simple Approximations to the Cramer Rao Lower bound on Directions of Arrival for closely separated sources, *IEEE Transactions on signal Processing*, Volume: 41, No 4, April 1993, pp. 1668 -1672.
- [38] J.Platt, A Resource Allocating Network for Function Interpolation, *Neural Computation*, Vol. 3, Issue 2, 1991, pp.213-225.
- [39] V.Kadirkamanathan and M.Niranjan, A Function Estimation Approach to Sequential Learning with Neural Networks, *Neural Computation*, Vol 5, 1993, pp. 954-975.
- [40] Deng Jianping, N.Sundararajan, P.Saratchandran, Communication channel equalization using complex-valued minimal radial basis function neural networks, *IEEE Transactions on Neural Networks*, Volume 13, Issue 3, May 2002, pp. 687 – 696.
- [41] G.B. Huang, P.Saratchandran, N.Sundararajan, An efficient sequential learning algorithm for growing and Pruning RBF (GAP_RBF) networks, *IEEE Transactions on Systems, Man and Cybernetics*, Volume 34, Issue 6, Nov 2004 pp. 2284-2292
- [42] G.B. Huang, P.Saratchandran, N.Sundararajan, An generalised growing and Pruning RBF (GGAP_RBF) neural network for function approximation, *IEEE Transactions on Neural Networks*, Volume 16, Issue 1, Jan 2005, pp. 57-67.
- [43] G.B. Huang, Q.Y.Zhu, C.K.Siew , Extreme Learning Machine: RBF network case, *Proceedings of the 8th International conference on Control, Automation, Robotics and Vision (ICARV-2004)*, Volume 2, Dec 2004 , pp. 1029-1036.

Appendix A – Varying the Element at which Failure Occurs:

This appendix supplements the conclusions from section 6.3 with additional results. It was observed from section 6.3 the performance of MRAN under scenarios with failure and noise does not depend on the element that encounters failure. Sections 6.1 and 6.2 considered the failure at the 3rd element while section 6.3 considered the failure at the first element only. Here we consider the failure at the 2nd, 4th, 5th and 6th elements of the uniform linear array.

Like in the studies done in section 6.3 we plot the DoA estimation error for two different cases. First we fix the instant at which failure occurs and vary the position of the sources. Next we fix the position of the sources and vary the instant at which failure occurs. Three different SNRs are considered just like in section 6.3.

Figs A1 to A8 correspond to the no noise case while figures A9 to A16 correspond to an SNR of 20 dB and figures A17 to A24 correspond to an SNR of 10 dB. It can be observed from these figures that the element at which the failure occurs has very little effect on the error for the neural network based methods.

Figure A1 Performance as a function of source position - No Noise, Failure at 150th snapshot, at 2nd element

Figure A2 Performance as a function of failure instant - No Noise, Sources at -9° and -11°, failure at 2nd element

Figure A3 Performance as a function of source position - No Noise, Failure at 150th snapshot, at 4th element

Figure A4 Performance as a function of failure instant - No Noise, Sources at -9° and -11°, failure at 4th element

Figure A5 Performance as a function of source position - No Noise, Failure at 150th snapshot, at 5th element

Figure A6 Performance as a function of failure instant - No Noise, Sources at -9° and -11°, failure at 5th element

Figure A7 Performance as a function of source position - No Noise, Failure at 150th snapshot, at 6th element

Figure A8 Performance as a function of failure instant - No Noise, Sources at -9° and -11°, failure at 6th element

Figure A9 Performance as a function of source position - SNR=20dB, Failure at 150th snapshot, at 2nd element

Figure A10 Performance as a function of failure instant - SNR=20dB, Sources at -9° and -11°, failure at 2nd element

Figure A11 Performance as a function of source position - SNR=20dB, Failure at 150th snapshot, at 4th element

Figure A12 Performance as a function of failure instant - SNR=20dB, Sources at - 9° and -11°, failure at 4th element

Figure A13 Performance as a function of source position - SNR=20dB, Failure at 150th snapshot, at 5th element

Figure A14 Performance as a function of failure instant - SNR=20dB, Sources at - 9° and -11°, failure at 5th element

Figure A15 Performance as a function of source position - SNR=20dB, Failure at 150th snapshot, at 6th element

Figure A16 Performance as a function of failure instant - SNR=20dB, Sources at - 9° and -11°, failure at 6th element

Figure A17 Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 2nd element

Figure A18 Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 2nd element

Figure A19 Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 4th element

Figure A20 Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 4th element

Figure A21 Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 5th element

Figure A22 Performance as a function of failure instant - SNR=10dB, Sources at -9° and -11°, failure at 5th element

Figure A23 Performance as a function of source position - SNR=10dB, Failure at 150th snapshot, at 6th element

Figure A24 Performance as a function of failure instant - SNR=10dB, Sources at - 9° and -11°, failure at 6th element