

**DEVELOPMENT OF A  
COMPUTATIONALLY EFFICIENT  
ANTENNA BEAMFORMING  
ALGORITHM AND ITS  
IMPLEMENTATION USING FPGA**

*A THESIS*

Submitted by

**Mini P. R.**

*for the award of the degree*

*of*

**DOCTOR OF PHILOSOPHY**



**DIVISION OF ELECTRONICS ENGINEERING  
SCHOOL OF ENGINEERING  
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY  
KOCHI - 682 022**

**JULY 2017**

## THESIS CERTIFICATE

This is to certify that the thesis entitled **DEVELOPMENT OF A COMPUTATIONALLY EFFICIENT ANTENNA BEAMFORMING ALGORITHM AND ITS IMPLEMENTATION USING FPGA** submitted by **Mini P. R.** to the Cochin University of Science and Technology, Kochi for the award of the degree of Doctor of Philosophy is a bonafide record of research work carried out by her under my supervision and guidance at the Division of Electronics, School of Engineering, Cochin University of Science and Technology. The contents of this thesis, in full or in parts, have not been submitted to any other University or Institute for the award of any degree or diploma.

I further certify that the corrections and modifications suggested by the audience during the pre-synopsis seminar and recommended by the Doctoral Committee of **Mini P. R.** are incorporated in the thesis.

Kochi - 682 022  
Date : 27.07.2017

**Prof. (Dr.) S. Mridula**  
Research Guide  
Division of Electronics Engineering  
School of Engineering  
CUSAT, Kochi - 682 022

## DECLARATION

I hereby declare that the work presented in the thesis entitled **DEVELOPMENT OF A COMPUTATIONALLY EFFICIENT ANTENNA BEAMFORMING ALGORITHM AND ITS IMPLEMENTATION USING FPGA** is based on the original research work carried out by me under the supervision and guidance of **Dr. S. Mridula**, Professor, Division of Electronics, School of Engineering, Cochin University of Science and Technology for the award of degree of Doctor of Philosophy with Cochin University of Science and Technology. I further declare that the contents of this thesis in full or in parts have not been submitted to any other University or Institute for the award any degree or diploma.

Kochi - 682 022

Date : 27.07.2017

Mini P. R.

(Reg. No. : 3828)

## ACKNOWLEDGMENTS

First and foremost, I would like to thank God Almighty for all the blessings showered on me.

With deepest regard and gratitude, I express my sincere thanks to Dr. S. Mridula for guiding me through my research. This work is a testimony of her relentless support and encouragement. Her commitment to excellence in all activities, uncompromising stance on quality and high degree of professionalism has left an indelible mark on me.

I am deeply indebted to Dr. P. Mohanan, UGC BSR Professor, Department of Electronics, CUSAT for his valuable suggestions and for the encouragement given to me during the course of my research. I gratefully acknowledge the contributions of Dr. Binu Paul, Associate Professor, Division of Electronics, School of Engineering, CUSAT in evaluating my work and giving me valuable advice. Heartfelt thanks to Dr. Anju Pradeep, Associate Professor, Division of Electronics, School of Engineering, CUSAT for all the support extended to me.

I earnestly thank Dr. R. Gopika Kumari, Professor, Division of Electronics, School of Engineering, CUSAT for her invaluable suggestions. I also acknowledge with gratitude, the encouragement and good wishes offered by Dr. Rajesh Cherian Roy, Professor, Adi Sankara Institute of Engineering and Technology, Kalady.

Thankful appreciation is extended to Dr. P. S. Sreejith, Dean, Faculty of Engineering, CUSAT for the suggestions in my pre-synopsis seminar and thesis. I also thank Dr. M. R. Radhakrishna Panicker, Principal, School of Engineering, CUSAT and Dr. G. Madhu, former Dean, School of Engineering, CUSAT for their support.

I also thank my fellow researchers Rajesh Mohan R, Sumi M, Dr. Bindu C. J., Basil J Paul, Anjit T. A. and Reshma Lakshmanan for all the help extended to me. I would like to place on record my sincere gratitude to Dr. George Isaac, Principal, FISAT, Dr. K. S. M. Panicker, Director (Academics), FISAT and the Management of FISAT, without whose support this research could not have been completed. I also thank all my colleagues in the Department of Electronics and Communication, FISAT for the immense encouragement extended to me.

Technical support offered by Mr. Venkatesha K., Senior Application Engineer, Entuple Technologies is gratefully acknowledged. My wholehearted appreciation to all my teachers who have moulded me to what I am today.

I could not have achieved my goal without the unstinting support of my family. I thank my parents and siblings for their love and encouragement. I have no words to express my thanks to my husband Jith T. S. and my daughter Ananya T Jith for their relentless support and unbounded love and for the immense sacrifices they made to help me realize my goal.

## ABSTRACT

**KEYWORDS:** Beamforming; R-1D-MRT; Computational Complexity; FPGA implementation; Xilinx System Generator model.

Beamforming is a versatile form of spatial filtering. It is a computationally intensive process which combines the signals from an array of sensors/antennas to ensure maximum strength in the desired direction and nulls in the undesired directions. This thesis presents the development and implementation of “Reduced One Dimensional Mapped Real Transform (R-1D-MRT)”, a computationally efficient algorithm for beamforming applications. The signals at the array input are transformed by applying R-1D-MRT, reducing the size of the data for beamforming to half the original data size, resulting in reduced computational load and computational complexity of the beamformer. The efficiency of the algorithm was validated on various classes of beamformers. The reduction in computational complexity and computation time was also analyzed for each beamformer. The performance of the R-1D-MRT based beamformers in AWGN, Rayleigh and Rician channels were evaluated. Xilinx System Generator models of the R-1D-MRT based beamformers were created and simulated. These models were synthesized and built on a Spartan 3AN FPGA using the Altium NB3000 NanoBoard.

# TABLE OF CONTENTS

Contents	Page
ACKNOWLEDGEMENTS .....	i
ABSTRACT .....	iii
LIST OF TABLES .....	ix
LIST OF FIGURES .....	xiii
ABBREVIATIONS .....	xx
NOTATION .....	xxiii
<b>CHAPTER 1 INTRODUCTION</b>	
1.1 Beamforming .....	2
1.2 Classes of beamformers.....	4
1.3 Motivation for the work.....	7
1.4 Literature survey.....	8
1.5 Objectives .....	20
1.6 Methodology.....	20
1.7 Thesis organization .....	24
1.8 Chapter summary .....	27
<b>CHAPTER 2 DEVELOPMENT OF REDUCED ONE DIMENSIONAL MAPPED REAL TRANSFORM</b>	
2.1 One Dimensional Mapped Real Transform.....	28
2.2 The Proposed Reduced One Dimensional Mapped Real Transform ...	31
2.3 R-1D-MRT conventional beamformer .....	33
2.4 Beam pattern and beamformer output of R-1D-MRT CBF.....	35
2.5 Computational complexity of R-1D-MRT CBF .....	39
2.6 Computation time of R-1D-MRT CBF .....	41

<b>Table of Contents (continued)</b>		<b>Page</b>
2.7	Performance of R-1D-MRT CBF in various communication channels .....	42
2.8	Interference rejection of R-1D-MRT CBF .....	47
2.9	FPGA implementation of R-1D-MRT CBF .....	48
2.9.1	Xilinx System Generator model of R-1D-MRT CDBF .....	52
2.9.2	Beam pattern and beamformer output of XSG model of R-1D-MRT CDBF .....	55
2.9.3	FPGA resource utilization of R-1D-MRT CDBF .....	57
2.10	Chapter Summary .....	58
<b>CHAPTER 3 R-1D-MRT MVDR AND LCMV BEAMFORMERS</b>		
3.1	R-1D-MRT MVDR beamformer .....	60
3.1.1	Beam pattern and beamformer output of R-1D-MRT MVDR beamformer .....	62
3.1.2	Computational complexity of R-1D-MRT MVDR beamformer .....	65
3.1.3	Computation time of R-1D-MRT MVDR beamformer .....	67
3.1.4	Performance of R-1D-MRT MVDR beamformer in various communication channels .....	69
3.1.5	FPGA implementation of R-1D-MRT MVDR beamformer .....	72
3.1.5.1	Xilinx System Generator model of R-1D-MRT MVDR beamformer ...	72
3.1.5.2	Beam pattern and beamformer output of XSG model of R-1D-MRT MVDR beamformer .....	82
3.1.5.3	FPGA resource utilization of R-1D-MRT MVDR beamformer .....	84
3.2	R-1D-MRT LCMV beamformer .....	85
3.2.1	Beam pattern and beamformer output of R-1D-MRT LCMV beamformer .....	87
3.2.2	Computational complexity of R-1D-MRT LCMV beamformer .....	90
3.2.3	Computation time of R-1D-MRT LCMV beamformer .....	92
3.2.4	Performance of R-1D-MRT LCMV beamformer in various communication channels .....	93

<b>Table of Contents (continued)</b>	<b>Page</b>
3.2.5 FPGA implementation of R-1D-MRT LCMV beamformer . . . . .	96
3.2.5.1 Xilinx System Generator model of R-1D-MRT LCMV beamformer . . .	96
3.2.5.2 Beam pattern and beamformer output of XSG model of R-1D-MRT LCMV beamformer . . . . .	106
3.2.5.3 FPGA resource utilization of R-1D-MRT LCMV beamformer . . . . .	108
3.3 Chapter summary . . . . .	109

## **CHAPTER 4 R-1D-MRT LMS AND NLMS BEAMFORMERS**

4.1 R-1D-MRT LMS beamformer . . . . .	110
4.1.1 Beam pattern and beamformer output of R-1D-MRT LMS beamformer . . . . .	113
4.1.2 Computational complexity of R-1D-MRT LMS beamformer . . . . .	116
4.1.3 Computation time of R-1D-MRT LMS beamformer . . . . .	117
4.1.4 Performance of R-1D-MRT LMS beamformer in various communication channels . . . . .	119
4.1.5 FPGA implementation of R-1D-MRT LMS beamformer . . . . .	122
4.1.5.1 Xilinx System Generator model of R-1D-MRT LMS beamformer . . . . .	122
4.1.5.2 Beam pattern and beamformer output of XSG model of R-1D-MRT LMS beamformer . . . . .	125
4.1.5.3 FPGA resource utilization of R-1D-MRT LMS beamformer . . . . .	127
4.2 R-1D-MRT NLMS Beamformer . . . . .	128
4.2.1 Beam pattern and beamformer output of R-1D-MRT NLMS beamformer . . . . .	131
4.2.2 Computational complexity of R-1D-MRT NLMS beamformer . . . . .	134
4.2.3 Computation time of R-1D-MRT NLMS beamformer . . . . .	136
4.2.4 Performance of R-1D-MRT NLMS beamformer in various communication channels . . . . .	137
4.2.5 FPGA Implementation of R-1D-MRT NLMS beamformer . . . . .	141
4.2.5.1 Xilinx System Generator model of R-1D-MRT NLMS beamformer . . . . .	141

<b>Table of Contents (continued)</b>	<b>Page</b>
4.2.5.2 Beam pattern and beamformer output of XSG model of R-1D-MRT NLMS beamformer .....	143
4.2.5.3 FPGA resource utilization of R-1D-MRT NLMS beamformer .....	145
4.3 Chapter summary .....	146

## **CHAPTER 5 R-1D-MRT RLS, SMI AND GSC BEAMFORMERS**

5.1 R-1D-MRT RLS beamformer .....	147
5.1.1 Beam pattern and beamformer output of R-1D-MRT RLS beamformer .....	149
5.1.2 Computational complexity of R-1D-MRT RLS beamformer .....	152
5.1.3 Computation time of R-1D-MRT RLS beamformer .....	154
5.1.4 Performance of R-1D-MRT RLS beamformer in various communication channels .....	155
5.2 R-1D-MRT SMI beamformer .....	158
5.2.1 Beam pattern and beamformer output of R-1D-MRT SMI beamformer .....	161
5.2.2 Computational complexity of R-1D-MRT SMI beamformer .....	165
5.2.3 Computation time of R-1D-MRT SMI beamformer .....	166
5.2.4 Performance of R-1D-MRT SMI beamformer in various communication channels .....	168
5.3 R-1D-MRT GSC beamformer .....	171
5.3.1 Beam pattern and beamformer output of R-1D-MRT GSC beamformer .....	173
5.3.2 Computational complexity of R-1D-MRT GSC beamformer .....	175
5.3.3 Computation time of R-1D-MRT GSC beamformer .....	176
5.3.4 Performance of R-1D-MRT GSC beamformer in various communication channels .....	178
5.4 Chapter summary .....	181

**CHAPTER 6 CONCLUSIONS AND FUTURE SCOPE**

6.1 Thesis highlights ..... 182  
6.2 Performance comparison of R-1D-MRT based beamformers ..... 185  
6.3 Suggestions for future work ..... 186

**APPENDIX A R-1D-MRT BASED DENOISING IN CONVENTIONAL BEAMFORMERS**

A.1 Introduction ..... 188  
A.2 Comparison of R-1D-MRT based denoising and wavelet denoising .... 189  
A.3 Inference ..... 192

**APPENDIX B R-1D-MRT BASED SIDELobe REDUCTION IN CONVENTIONAL BEAMFORMERS**

B.1 Amplitude tapering ..... 194  
B.2 Comparison of R-1D-MRT based sidelobe reduction and wavelet based sidelobe reduction ..... 196  
B.3 Inference ..... 198

**REFERENCES**

**LIST OF PAPERS PUBLISHED ON THE BASIS OF THIS THESIS**

**CURRICULUM VITAE**

## LIST OF TABLES

Table	Title	Page
2.1	One Dimensional Mapped Real Transform .....	30
2.2	Optimum sampling frequency for various data sizes .....	37
2.3	Comparison of computational complexity of CBF and R-1D-MRT CBF .....	40
2.4	Comparison of computation time of CBF and R-1D-MRT CBF .....	41
2.5	Comparison of computation time of CBF and R-1D-MRT CBF .....	42
2.6	Comparison of Euclidean distance parameters of CBF and R-1D-MRT CBF for various communication channels for varying SNR .....	44
2.7	Comparison of FPGA resources of CDBF and R-1D-MRT CDBF .....	57
3.1	Comparison of computational complexity of MVDR BF and R-1D-MRT MVDR BF .....	67
3.2	Comparison of computation time of MVDR BF and R-1D-MRT MVDR BF .....	68
3.3	Comparison of computation time of MVDR BF and R-1D-MRT MVDR BF .....	68
3.4	Comparison of Euclidian distance parameters of MVDR BF and R-1D-MRT MVDR BF for various communication channels for varying SINR .....	69
3.5	Matrix inverse of R-1D-MRT MVDR BF using XSG model .....	80
3.6	Matrix inverse of R-1D-MRT MVDR BF using MATLAB .....	80
3.7	Comparison of FPGA resources of MVDR BF and R-1D-MRT MVDR BF .....	85
3.8	Comparison of computational complexity of LCMV BF and R-1D-MRT LCMV BF .....	91
3.9	Comparison of computation time of LCMV BF and R-1D-MRT LCMV BF .....	92
3.10	Comparison of computation time of LCMV BF and R-1D-MRT LCMV BF .....	93
3.11	Comparison of Euclidian distance parameters of LCMV BF and R-1D-MRT LCMV BF for various communication channels for varying SINR .....	94

## List of Tables (continued)

<b>Table</b>	<b>Title</b>	<b>Page</b>
3.12	Matrix inverse of R-1D-MRT LCMV BF using XSG model . . . . .	104
3.13	Matrix inverse of R-1D-MRT LCMV BF using MATLAB . . . . .	104
3.14	Comparison of FPGA resources of LCMV BF and R-1D-MRT LCMV BF . . . . .	108
4.1	Comparison of minimum number of iterations (I <sub>min</sub> ) for LMS BF and R-1D-MRT LMS BF for convergence . . . . .	115
4.2	Comparison of computational complexity of LMS BF and R-1D-MRT LMS BF . . . . .	117
4.3	Comparison of computation time of LMS BF and R-1D-MRT LMS BF . . . . .	118
4.4	Comparison of computation time of LMS BF and R-1D-MRT LMS BF . . . . .	118
4.5	Comparison of Euclidian distance parameters of LMS BF and R-1D-MRT LMS BF for various communication channels for varying SINR . . . . .	119
4.6	Comparison of FPGA resources of NLMS BF and R-1D-MRT NLMS BF. . . . .	128
4.7	Comparison of minimum number of iterations (I <sub>min</sub> ) for NLMS BF and R-1D-MRT NLMS BF for convergence . . . . .	133
4.8	Comparison of computational complexity of NLMS BF and R-1D-MRT NLMS BF . . . . .	135
4.9	Comparison of computational complexity of NLMS BF and R-1D-MRT NLMS BF . . . . .	135
4.10	Comparison of computation time of NLMS BF and R-1D-MRT NLMS BF. . . . .	136
4.11	Comparison of computation time of NLMS BF and R-1D-MRT NLMS BF. . . . .	137
4.12	Comparison of Euclidian distance parameters of NLMS BF and R-1D-MRT NLMS BF for various communication channels for varying SINR . . . . .	138
4.13	Comparison of FPGA resources of NLMS BF and R-1D-MRT NLMS BF . . . . .	146

## List of Tables (continued)

<b>Table</b>	<b>Title</b>	<b>Page</b>
5.1	Comparison of minimum number of iterations ( $I_{min}$ ) for RLS BF and R-1D-MRT RLS BF for convergence . . . . .	151
5.2	Comparison of computational complexity of RLS BF and R-1D-MRT RLS BF . . . . .	153
5.3	Comparison of computation time of RLS BF and R-1D-MRT RLS BF . . . . .	154
5.4	Comparison of computation time of RLS BF and R-1D-MRT RLS BF . . . . .	155
5.5	Comparison of Euclidian distance parameters of RLS BF and R-1D-MRT RLS BF for various communication channels for varying SINR . . . . .	156
5.6	Comparison of minimum number of iterations ( $I_{min}$ ) of SMI BF and R-1D-MRT SMI BF for convergence for varying block lengths . . . . .	164
5.7	Comparison of minimum number of iterations ( $I_{min}$ ) of SMI BF and R-1D-MRT SMI BF for convergence for varying data sizes . . . . .	164
5.8	Comparison of computational complexity of SMI BF and R-1D-MRT SMI BF . . . . .	166
5.9	Comparison of computation time of SMI BF and R-1D-MRT SMI BF . . . . .	167
5.10	Comparison of computation time of SMI BF and R-1D-MRT SMI BF . . . . .	167
5.11	Comparison of Euclidian distance parameters of SMI BF and R-1D-MRT SMI BF for various communication channels for varying SINR . . . . .	168
5.12	Comparison of computational complexity of GSC BF and R-1D-MRT GSC BF . . . . .	176
5.13	Comparison of computation time of GSC BF and R-1D-MRT GSC BF . . . . .	177
5.14	Comparison of computation time of GSC BF and R-1D-MRT GSC BF . . . . .	177
5.15	Comparison of Euclidian distance parameters of GSC BF and R-1D-MRT GSC BF for various communication channels for varying SINR . . . . .	178

## List of Tables (continued)

<b>Table</b>	<b>Title</b>	<b>Page</b>
6.1	Performance comparison of R-1D-MRT based beamformers and conventional beamformers .....	186
A.1	Comparison of R-1D-MRT denoising and wavelet denoising using Daubechies wavelet with Heursure thresholding in conventional beamformers .....	190
A.2	Comparison of R-1D-MRT denoising and wavelet denoising using Symlet wavelet with Heursure thresholding in conventional beamformers .....	190
A.3	Comparison of R-1D-MRT denoising and wavelet denoising using Daubechies wavelet with Rigrsure thresholding in conventional beamformers .....	190
A.4	Comparison of R-1D-MRT denoising and wavelet denoising using Symlet wavelet with Rigrsure thresholding in conventional beamformers .....	191
A.5	Comparison of R-1D-MRT denoising and wavelet denoising using Daubechies wavelet with Sqtwolog thresholding in conventional beamformers .....	191
A.6	Comparison of R-1D-MRT denoising and wavelet denoising using Symlet wavelet with Sqtwolog thresholding in conventional beamformers .....	191
A.7	Comparison of R-1D-MRT denoising and wavelet denoising using Daubechies wavelet with Minimaxi thresholding in conventional beamformers .....	192
A.8	Comparison of R-1D-MRT denoising and wavelet denoising using Symlet wavelet with Minimaxi thresholding in conventional beamformers .....	192
B.1	Comparison of sidelobe reduction .....	198

## LIST OF FIGURES

Figure	Title	Page
1.1	Uniform linear array of $N_s$ elements .....	3
1.2	Classes of beamformers .....	5
1.3	Schematic diagram of non - adaptive beamformer .....	6
1.4	Schematic diagram of adaptive beamformer .....	7
1.5	Methodology .....	21
1.6	FPGA Schematic .....	21
1.7	Altium NB3000 NanoBoard .....	23
1.8	Experimental setup with Altium NB3000 NanoBoard .....	24
1.9	Organization of the thesis .....	27
2.1(a)	Schematic diagram of conventional beamformer .....	33
2.1(b)	Schematic diagram of R-1D-MRT conventional beamformer .....	34
2.2	Normalized beam pattern of R-1D-MRT CBF for varying number of antennas/sensors .....	35
2.3	Normalized beam pattern of R-1D-MRT CBF for varying source locations .....	36
2.4	Normalized beamformer output of R-1D-MRT conventional beamformer .....	37
2.5	Beamformer output of R-1D-MRT CBF for varying look angles .....	38
2.6	Scatterplots of R-1D-MRT CBF in AWGN channel for varying SNR .....	46
2.7	Scatterplots of R-1D-MRT CBF in Rayleigh channel for varying SNR .....	46
2.8	Scatterplots of R-1D-MRT CBF in Rician channel for varying SNR ...	47
2.9	Normalized beam pattern of R-1D-MRT CBF for 2 sources .....	48
2.10	Schematic diagram of R-1D-MRT CDBF .....	49
2.11	Schematic diagram of Digital Down Converter .....	50
2.12	Schematic diagram of CIC decimating filter .....	52

**List of Figures (continued)**

<b>Figure</b>	<b>Title</b>	<b>Page</b>
2.13(a)	XSG model of R-1D-MRT CDBF .....	53
2.13(b)	CIC decimation filter subsystem of R-1D-MRT CDBF .....	54
2.13(c)	Weight multiplication subsystem of R-1D-MRT CDBF .....	54
2.14	Beamformer output of XSG model of R-1D-MRT CDBF for various look angles .....	55
2.15	Normalized beam pattern of XSG model of R-1D-MRT CDBF for varying source locations .....	56
2.16	Normalized beam pattern of FPGA implementation of R-1D-MRT CDBF .....	56
3.1(a)	Schematic diagram of conventional MVDR beamformer .....	61
3.1(b)	Schematic diagram of R-1D-MRT MVDR beamformer .....	62
3.2	Normalized beam pattern of R-1D-MRT MVDR BF for varying number of antennas/sensors .....	63
3.3	Normalized beam pattern of R-1D-MRT MVDR BF for varying source and interference locations .....	63
3.4	Beamformer output of R-1D-MRT MVDR BF for varying look angles .	64
3.5	Beamformer output of R-1D-MRT MVDR BF for varying number of sensors .....	65
3.6	Scatterplots of R-1D-MRT MVDR BF in AWGN channel for varying SINR .....	70
3.7	Scatterplots of R-1D-MRT MVDR BF in Rayleigh channel for varying SINR .....	71
3.8	Scatterplots of R-1D-MRT MVDR BF in Rician channel for varying SINR .....	71
3.9(a)	XSG model for generating desired and interfering signals of R-1D-MRT MVDR BF .....	73
3.9(b)	XSG model for phase shifting signals of R-1D-MRT MVDR BF .....	73
3.10	XSG model of correlation matrix of R-1D-MRT MVDR BF .....	74
3.11	Subsystem of correlation matrix model of R-1D-MRT MVDR BF .....	74
3.12	Model for row transformation of R-1D-MRT MVDR BF .....	76

**List of Figures (continued)**

<b>Figure</b>	<b>Title</b>	<b>Page</b>
3.13	Model for first transformation of 2 <sup>nd</sup> , 3 <sup>rd</sup> and 4 <sup>th</sup> rows of R-1D-MRT MVDR BF .....	77
3.14	Model for second transformation of 3 <sup>rd</sup> and 4 <sup>th</sup> rows of R-1D-MRT MVDR BF .....	78
3.15	Model for third transformation of 4 <sup>th</sup> row of R-1D-MRT MVDR BF .....	78
3.16	Back substitution model of R-1D-MRT MVDR BF .....	79
3.17	Model for multiplication of $R^{-1}$ and Q matrices of R-1D-MRT MVDR BF .....	80
3.18	Model for calculating the weights of R-1D-MRT MVDR BF .....	81
3.19	Subsystem of model for calculating weights of R-1D-MRT MVDR BF .....	81
3.20	Model for obtaining the beamformer output of R-1D-MRT MVDR BF .....	82
3.21	Normalized beam pattern of XSG model of R-1D-MRT MVDR BF .....	83
3.22	Beamformer output of XSG model of R-1D-MRT MVDR BF for varying look angles .....	83
3.23	Normalized beam pattern of FPGA implementation of R-1D-MRT MVDR BF .....	84
3.24(a)	Schematic diagram of conventional LCMV beamformer .....	86
3.24(b)	Schematic diagram of R-1D-MRT LCMV beamformer .....	87
3.25	Normalized beam pattern of R-1D-MRT LCMV BF for varying number of sensors .....	87
3.26	Normalized beam pattern of R-1D-MRT LCMV beamformer for varying source and interference angles .....	88
3.27	Beamformer output of R-1D-MRT LCMV beamformer for varying look angles .....	89
3.28	Beamformer output of R-1D-MRT LCMV beamformer for look angle of 30° for varying gain constraints .....	90

## List of Figures (continued)

Figure	Title	Page
3.29	Scatterplots of R-1D-MRT LCMV BF for AWGN channel for varying SINR .....	94
3.30	Scatterplots of R-1D-MRT LCMV BF for Rayleigh channel for varying SINR .....	95
3.31	Scatterplots of R-1D-MRT LCMV BF for Rician channel for varying SINR .....	95
3.32(a)	XSG model for generating desired and interfering signals of R-1D-MRT LCMV BF .....	97
3.32(b)	XSG model for phase shifting signals of R-1D-MRT LCMV BF .....	97
3.33	Correlation matrix model of R-1D-MRT LCMV BF .....	98
3.34	Subsystem of correlation matrix model of R-1D-MRT LCMV BF .....	98
3.35	Model for row transformation of R-1D-MRT LCMV BF .....	100
3.36	Model for first transformation of 2 <sup>nd</sup> , 3 <sup>rd</sup> and 4 <sup>th</sup> rows of R-1D-MRT LCMV BF .....	101
3.37	Model for second transformation of 3 <sup>rd</sup> and 4 <sup>th</sup> rows of R-1D-MRT LCMV BF .....	102
3.38	Model for third transformation of 4 <sup>th</sup> row of R-1D-MRT LCMV BF .....	102
3.39	Back substitution model of R-1D-MRT LCMV BF .....	103
3.40	Model for multiplying $R^{-1}$ and Q matrices of R-1D-MRT LCMV BF ..	104
3.41	Model for obtaining the weights of R-1D-MRT LCMV BF .....	105
3.42	Weight calculation subsystem of R-1D-MRT LCMV BF .....	105
3.43	Model for obtaining beamformer output of R-1D-MRT LCMV BF .....	106
3.44	Beamformer output of XSG model of R-1D-MRT LCMV BF for varying look angles .....	106
3.45	Normalized beam pattern of XSG model of R-1D-MRT LCMV BF .....	107
3.46	Normalized beam pattern of FPGA implementation of R-1D-MRT LCMV BF .....	107
4.1(a)	Schematic diagram of conventional LMS beamformer .....	112

## List of Figures (continued)

Figure	Title	Page
4.1(b)	Schematic diagram of R-1D-MRT LMS beamformer .....	113
4.2	R-1D-MRT LMS beamformer output .....	114
4.3	R-1D-MRT LMS beamformer output .....	114
4.4	Scatterplots of R-1D-MRT LMS BF for AWGN channel for varying SINR .....	120
4.5	Scatterplots of R-1D-MRT LMS BF for Rayleigh channel for varying SINR .....	121
4.6	Scatterplots of R-1D-MRT LMS BF for Rician channel for varying SINR .....	121
4.7	XSG model for generating desired and interfering signals of R-1D-MRT LMS BF .....	123
4.8	Training phase XSG model of R-1D-MRT LMS BF .....	123
4.9	Weight adaptation subsystem of R-1D-MRT LMS BF training phase .....	124
4.10	XSG model for obtaining beamformer output of R-1D-MRT LMS BF .....	125
4.11	R-1D-MRT LMS BF output with XSG modeling .....	126
4.12	Normalized beam pattern of XSG simulation and FPGA implementation of R-1D-MRT LMS BF .....	127
4.13(a)	Schematic diagram of conventional NLMS beamformer .....	130
4.13(b)	Schematic diagram of R-1D-MRT NLMS beamformer .....	131
4.14	R-1D-MRT NLMS beamformer output .....	132
4.15	R-1D-MRT NLMS beamformer output .....	132
4.16	Scatterplots of R-1D-MRT NLMS BF for AWGN channel for varying SINR .....	139
4.17	Scatterplots of R-1D-MRT NLMS BF for Rayleigh channel for varying SINR .....	139
4.18	Scatterplots of R-1D-MRT NLMS BF for Rician channel for varying SINR .....	140

**List of Figures (continued)**

<b>Figure</b>	<b>Title</b>	<b>Page</b>
4.19	XSG Model of source and interfering signal generation of R-1D-MRT NLMS BF .....	141
4.20	Training model for R-1D-MRT NLMS BF .....	142
4.21	Weight adaptation subsystem for training model of R-1D-MRT NLMS BF .....	142
4.22	XSG model for obtaining output of R-1D-MRT NLMS BF .....	143
4.23	R-1D-MRT NLMS BF output with XSG modelling .....	144
4.24	Normalized beam patterns of XSG simulation and FPGA implementation of R-1D-MRT NLMS BF .....	145
5.1(a)	Schematic diagram of conventional RLS beamformer .....	148
5.1(b)	Schematic diagram of R-1D-MRT RLS beamformer .....	149
5.2	R-1D-MRT RLS beamformer output .....	150
5.3	R-1D-MRT RLS beamformer output. ....	150
5.4	Scatterplots of R-1D-MRT RLS BF for AWGN channel for varying SINR .....	157
5.5	Scatterplots of R-1D-MRT RLS BF for Rayleigh channel for varying SINR .....	157
5.6	Scatterplots of R-1D-MRT RLS BF for Rician channel for varying SINR .....	158
5.7(a)	Schematic diagram of conventional SMI beamformer .....	160
5.7(b)	Schematic diagram of R-1D-MRT SMI beamformer .....	161
5.8	R-1D-MRT SMI beamformer output .....	162
5.9	R-1D-MRT SMI beamformer output .....	162
5.10	MSE of SMI BF and R-1D-MRT SMI BF for varying block lengths .....	163
5.11	Scatterplots of R-1D-MRT SMI BF for AWGN channel for varying SINR .....	169
5.12	Scatterplots of R-1D-MRT SMI BF for Rayleigh channel for varying SINR .....	170

**List of Figures (continued)**

<b>Figure</b>	<b>Title</b>	<b>Page</b>
5.13	Scatterplots of R-1D-MRT SMI BF for Rician channel for varying SINR .....	170
5.14(a)	Schematic diagram of conventional GSC beamformer .....	172
5.14(b)	Schematic diagram of R-1D-MRT GSC beamformer .....	173
5.15	Normalized beam pattern of R-1D-MRT GSC BF for varying constraints .....	173
5.16	Normalized beam pattern of R-1D-MRT GSC BF for varying constraints .....	174
5.17	Normalized beam pattern of R-1D-MRT GSC BF for varying constraints .....	174
5.18	Beamformer output of R-1D-MRT GSC BF for varying gain constraints. ....	175
5.19	Scatterplots of R-1D-MRT GSC BF for AWGN channel for varying SINR .....	179
5.20	Scatterplots of R-1D-MRT GSC BF for Rayleigh channel for varying SINR .....	180
5.21	Scatterplots of R-1D-MRT GSC BF for Rician channel for varying SINR .....	180
B.1	Comparison of normalized beam patterns of conventional beamformer with Rigrsure wavelet noising and various taper windows and R-1D-MRT with various taper windows .....	197

## ABBREVIATIONS

1D-MRT	One Dimensional Mapped Real Transform
ASIC	Application Specific Integrated Circuits
ATD-NLMS	Active Tap Detection – Normalized Least Mean Square
AWGN	Additive White Gaussian Noise
CAD	Computer Aided Design
CBF	Conventional Beamformer
CCM	Constrained Constant Modulus
CDBF	Conventional Digital Beamformer
CDMA	Code Division Multiple Access
CIC	Cascade Integrated Comb
CLB	Configurable Logic Blocks
CM	Constant Modulus
CMA	Constant Modulus Algorithm
DBF	Digital Beamformer
DDC	Digital Down Converter
DDS	Digital Direct Synthesizer
DFT	Discrete Fourier Transform
DMI	Direct Matrix Inversion
DOA	Direction of Arrival
DSP	Digital Signal Processing
DUC	Digital Up Converter
EDA	Engineering Design Automation
FIR	Finite Impulse Response
FPGA	Field Programmable Gate Array
GSC	Generalized Sidelobe Canceller

## Abbreviations (continued)

HDL	Hardware Descriptive Language
IF	Intermediate Frequency
IIR	Infinite Impulse Response
INR	Interference to Noise Ratio
ISE	Integrated Synthesis Environment
LCMV	Linearly Constrained Minimum Variance
LLMS	Least Mean Square –Least Mean Square
LMMSE	Linear Minimum Mean Square Error
LMS	Least Mean Square
MIMO	Multiple Input Multiple Output
MMSE	Minimum Mean Square Error
MRT	Mapped Real Transform
MSE	Mean Square Error
MSINR	Maximum Signal to Interference plus Noise Ratio
MVDR	Minimum Variance Distortionless Response
MWT	Multiwavelet Transform
NCO	Numerically Controlled Oscillator
NLMS	Normalized Least Mean Square
OQPSK	Offset Quadrature Phase Shift Keying
PCB	Printed Circuit Board
R-1D-MRT	Reduced One Dimensional Mapped Real Transform
RADAR	Radio Detection And Ranging
RLS	Recursive Least Square
RLS-SW	Recursive Least Square Sliding Window
RTL	Register Transfer Level
SINR	Signal to Interference plus Noise Ratio

### **Abbreviations (continued)**

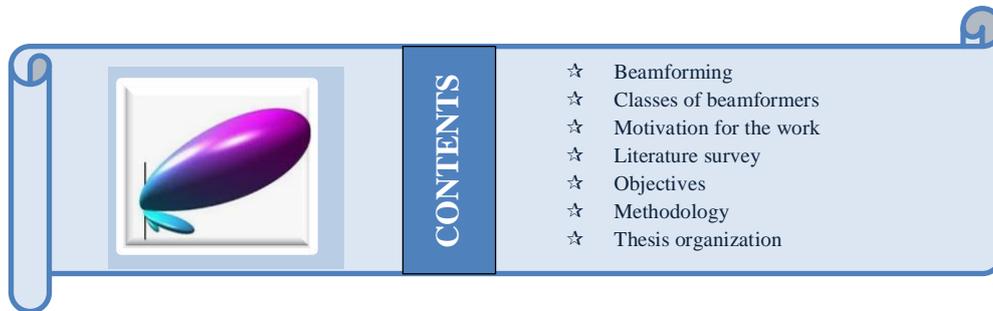
SIR	Signal to Interference Ratio
SL-CLMS	Shrinkage Linear Complex valued Least Mean Square
SNR	Signal to Noise Ratio
S-SMI	Subtraction Based Sample Matrix Inversion
SWD	Spatial Wavelet Denoising
SWL-CLMS	Shrinkage Widely Linear Complex valued Least Mean Square
SWPT	Spatial Wavelet Packet Transform
TWD	Temporal Wavelet Denoising
UMTS	Universal Mobile Telecommunications System
VRLS	Vector Recursive Least Square
W-CDMA	Wide Band Code Division Multiple Access
XSG	Xilinx System Generator

## NOTATION

$ x $	absolute value of $x$
$\mu$	step size
$\mu_1$	normalized step size
$A$	array response matrix
$a(\theta)$	array steering vector
$C$	correlation matrix
$C$	correlation matrix
$D$	constraint matrix
$d$	inter element spacing
$D_a$	signal blocking matrix
$E(.)$	expectation operator
$g$	gain
$I$	identity matrix
$K$	number of sources
$L$	number of linear constraints
$N$	data size
$\mathbf{N}$	noise vector
$n$	sample
$N_s$	no of antennas/sensor elements
$R$	upper triangular matrix
$\mathbf{W}$	weight vector
$\mathbf{W}_a$	adjustable weight vector
$\mathbf{W}^H$	Hermitian transpose of weight vector
$\mathbf{W}_N$	Twiddle factor
$\mathbf{W}_q$	quiescent weight vector
$\mathbf{X}$	data vector
$Y$	output of beamformer
$Y_k^{(p)}$	1D-MRT
$\zeta$	forgetting factor
$\alpha$	adjustable parameter
$\delta$	regularization parameter
$\varepsilon$	stabilizing factor
$\theta$	direction of arrival
$\lambda$	wavelength of signal
$\sigma$	standard deviation
$\psi$	phase difference

# CHAPTER 1

## INTRODUCTION



The soaring demand for high performance wireless communication systems necessitates enhanced data rates and higher spectral efficiency with superior coverage for a wide variety of users. For reducing the overall network costs and making the services affordable to subscribers, it is necessary to accommodate more users per base station. Since the available broadcast spectrum is limited, efforts to intensify the traffic within a fixed bandwidth, generates more interference in the system and degrades the signal quality. Use of omnidirectional antennas at the base station also causes interference during transmission and reception. To improve the quality of wireless communication, greater power in the desired direction and interference suppression are also indispensable. These requirements can be met using Smart Antennas which focuses the maximum power in the direction of the desired user while nulling the signal from undesired directions. Smart antennas use arrays of antennas along with signal processing techniques to emphasize the signal of interest and minimize the interfering signals by adapting the beam pattern. This is done by varying the relative phase of the respective signals feeding the antennas in such a way that the effective radiation pattern of the array is reinforced in the desired direction

and suppressed in undesired directions to model any desired radiation pattern. Smart antenna techniques find application in radar, radio astronomy and cellular systems like W-CDMA and UMTS. The most imperative process in smart antenna system is beamforming (Girma, 2012; Khumane et al., 2011). An overview of beamforming is presented in section 1.1.

## 1.1 BEAMFORMING

Beamforming or spatial filtering is the fundamental technology in smart antenna systems. It is a signal processing technique used in antenna/sensor arrays for directional signal transmission or reception. The signals from the array are combined in such a way that signals from desired angles experience constructive interference while signals from undesired angles experience destructive interference. Beamforming can be used at both the transmitting and receiving ends in order to achieve spatial selectivity. It has found numerous applications in radar, sonar, seismology, wireless communications, radio astronomy, acoustics and biomedicine. The signal model for beamforming is as follows:

Fig 1.1 illustrates a uniform linear array of  $N_s$  antennas/sensors, spaced at  $d = \lambda/2$ , receiving signals from  $K$  sources represented as  $s_k(n)$ ,  $k = 1, 2 \dots K$  with  $n$  denoting the samples and  $\lambda$  the wavelength. One of these  $K$  sources represents the desired signal while the rest denote undesired interfering signals. The  $K$  sources have directions of arrival (DOA) given by  $\{\theta_k, k = 1, 2 \dots K\}$  (VanVeen and Buckley, 1988; Lin et al., 2010).

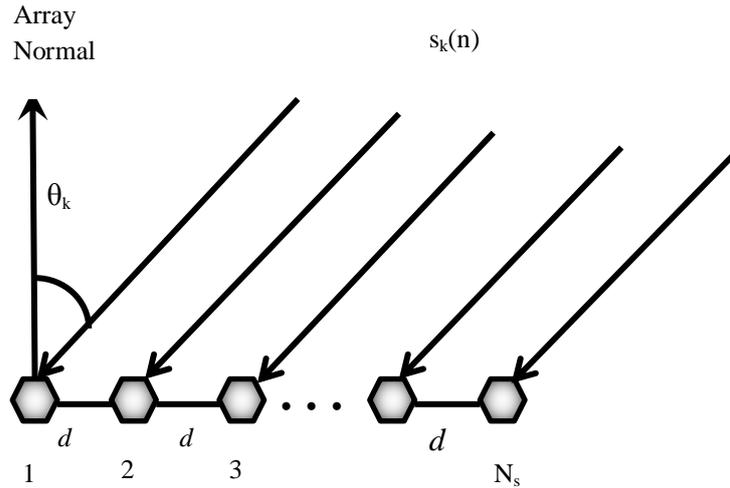


Fig 1.1 Uniform linear array of  $N_s$  elements

The signal from a source  $s_k$  arriving from a direction  $\theta_k$  will be received by each array element at different time instants in lieu of their separation. The phase difference between the successive elements of the array is given by

$$\psi_k = (2\pi d \sin\theta_k)/\lambda \quad (1.1)$$

where,  $d$  is the inter-element spacing and  $\lambda$  is the wavelength of the signal. The array steering vector for a signal arriving from a direction  $\theta_k$  is given by

$$\mathbf{a}(\theta_k) = [1 \ e^{j\psi_k} \ e^{j2\psi_k} \ \dots \ \dots \ \dots \ e^{j(N_s-1)\psi_k}]^T \quad (1.2)$$

The signal received by the array can be represented as

$$\mathbf{X}(n) = [\mathbf{a}(\theta_1) \ \mathbf{a}(\theta_2) \ \dots \ \dots \ \mathbf{a}(\theta_K)] \begin{bmatrix} s_1(n) \\ s_2(n) \\ \vdots \\ s_K(n) \end{bmatrix} + \mathbf{N}(n) \quad (1.3)$$

$$\mathbf{X}(n) = \mathbf{A}\mathbf{s}(n) + \mathbf{N}(n) \quad (1.4)$$

where  $\mathbf{X}(n) = [x_1(n) \ x_2(n) \ \dots \ x_{N_s}(n)]^T$  is the column vector of data received by the array,  $\mathbf{A} = [\mathbf{a}(\theta_1) \ \mathbf{a}(\theta_2) \ \dots \ \mathbf{a}(\theta_K)]$  represents array response matrix or steering matrix,

$\mathbf{s}(n) = [s_1(n) \ s_2(n) \ \dots \ s_K(n)]^T$  is the signal column vector generated by the sources,  $\mathbf{N}(n) = [n_1(n) \ n_2(n) \ \dots \ n_{N_s}(n)]^T$  is zero mean spatially uncorrelated Additive Gaussian Noises and  $(\cdot)^T$  represents the transpose. The array response matrix  $\mathbf{A}$  is represented as:

$$\mathbf{A}(\theta) = \begin{bmatrix} 1 & 1 & \dots & 1 \\ e^{j\frac{2\pi}{\lambda}d \sin\theta_1} & e^{j\frac{2\pi}{\lambda}d \sin\theta_2} & \dots & e^{j\frac{2\pi}{\lambda}d \sin\theta_K} \\ \vdots & \vdots & \vdots & \vdots \\ e^{j\frac{2\pi}{\lambda}(N_s-1)d \sin\theta_1} & e^{j\frac{2\pi}{\lambda}(N_s-1)d \sin\theta_2} & \dots & e^{j\frac{2\pi}{\lambda}(N_s-1)d \sin\theta_K} \end{bmatrix} \quad (1.5)$$

The received signal is multiplied with the coefficients of the weight vector  $\mathbf{W}$ , where the weights are chosen according to the type of beamformer (Torres Rossario, 2005).

$$\mathbf{W} = [w_1 \ w_2 \ \dots \ w_{N_s}]^T \quad (1.6)$$

The weighted signal is summed up to give the beamformer output  $Y$ .

$$Y(n) = \mathbf{W}^H \mathbf{X}(n) \quad (1.7)$$

$\mathbf{W}^H$  is the Hermitian transpose of the weight vector.

Beamformers use various approaches for computing the weights to achieve a desired radiation pattern. They are broadly classified based on the method of weight adaptation chosen. The following section discusses the various classes of beamformers.

## 1.2 CLASSES OF BEAMFORMERS

Beamformers can be classified as either non-adaptive or adaptive, depending on the method of choosing the weights (Hema Singh and Rakesh Mohan Jha, 2012). The classification of the beamformers evaluated in this thesis is illustrated in Fig 1.2.

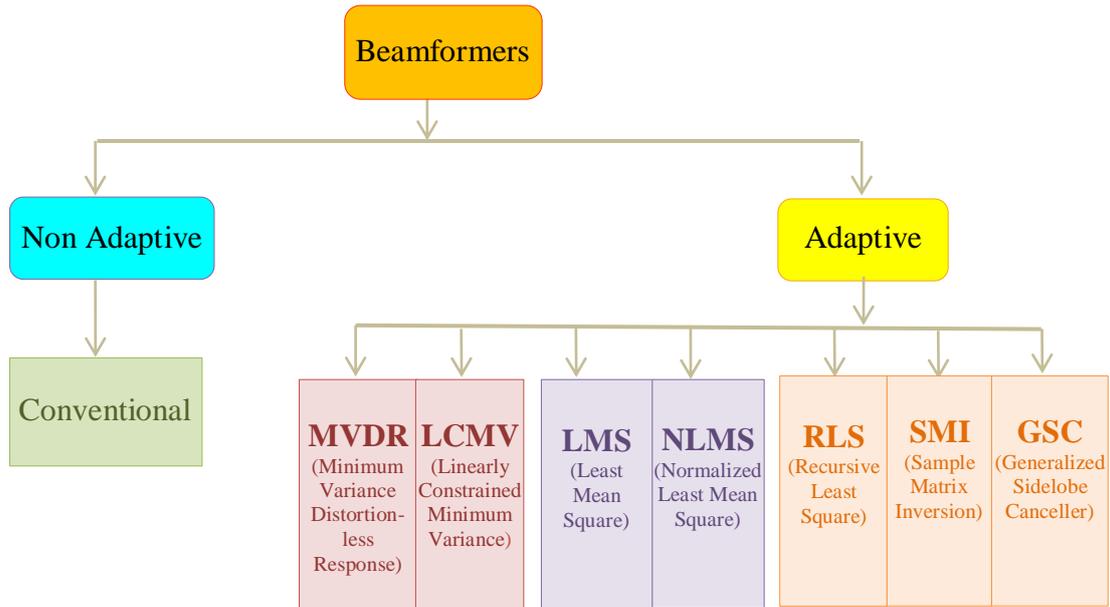


Fig 1.2 Classes of beamformers

Non - adaptive beamformers have weights which are not adjustable and hence cannot adapt to a rapidly changing environment. The weights are independent of the array data and are chosen to present a specified response for all signal/interference scenarios. The most common type of non-adaptive beamformers is Conventional beamformers. They use a fixed set of amplitude weights and time delays or phase shifts to combine the signal from the elements in the array, using only information about the location of the array elements in space and the direction of the desired signal. In single target environments, they exhibit robust performance but their interference rejection capability is poor.

The schematic diagram of the non - adaptive beamformer is shown in Fig 1.3.

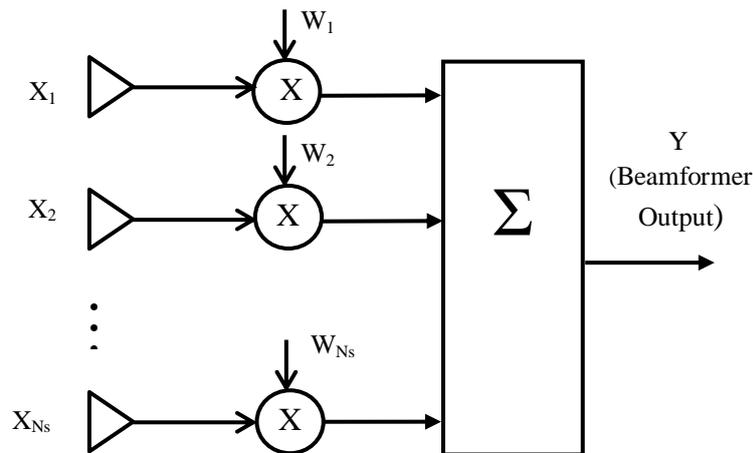


Fig 1.3 Schematic diagram of non - adaptive beamformer

Adaptive beamformers perform adaptive spatial signal processing. They are used to detect and estimate the signal-of-interest at the output of a sensor array by means of optimal spatial filtering and interference rejection. The weights in a statistically optimum beamformer are chosen based on the statistics of the array data to optimize the array response. In general, the adaptive beamformer places nulls in the directions of interfering sources in an attempt to maximize the signal-to-noise ratio at the beamformer output. Adaptive beamformers employ various algorithms for adapting the weights of the array elements to achieve the desired radiation pattern, resulting in different types of adaptive beamformers.

The schematic diagram of the adaptive beamformer is shown in Fig 1.4.

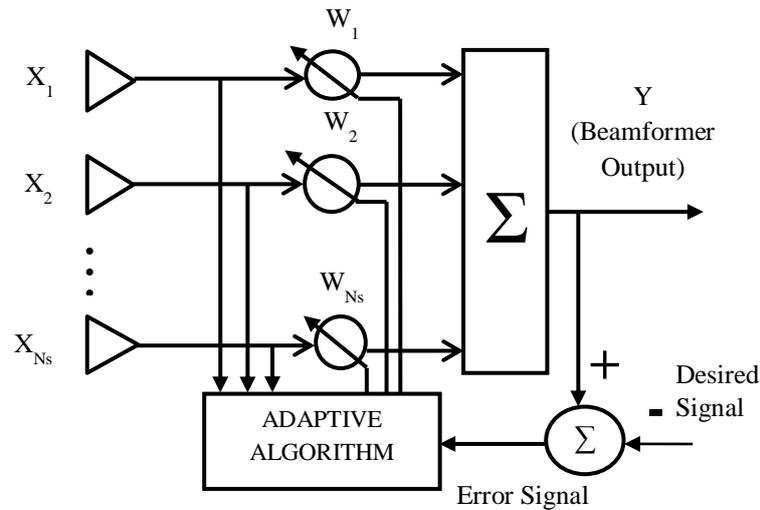


Fig 1.4 Schematic diagram of Adaptive beamformer

### 1.3 MOTIVATION FOR THE WORK

As discussed in the previous section, beamforming is a crucial component in smart antennas. Beamformers are required to steer the radiation pattern in the direction of the desired user while placing nulls at the interferer locations. Increasing the number of elements in the array improves the angular resolution of the beamformer and gives higher degrees of freedom for placing the nulls but requires higher computational complexity. The computational load also escalates with increase in size of the data received at the array elements. Any reduction in computational load translates to reduction in computation time and also reduction in requirement of system resources. In this context, developing a method of minimizing the computational burden of the beamformer would be of interest.

Beamformers employing stochastic gradient methods for weight adaptation like the Least Mean Square (LMS), Normalized Least Mean Square (NLMS), Recursive Least Square (RLS) etc. have a trade-off between convergence time and step size.

Increasing step size ensures faster convergence with compromise on accuracy and stability while smaller step size has slower convergence but provides more accurate output. It would be beneficial for these beamformers if faster convergence can be achieved with smaller step size.

#### **1.4 LITERATURE SURVEY**

A detailed literature survey conducted on the works reported in the area of beamforming is presented in this section.

Array processors with parameter estimation using subspace based methods for beamforming show good performance (Krim and Viberg, 1996). Beamforming operation can be performed using phased arrays, Butler Matrix, Blass Matrix etc. Digital beamforming, adaptive beamforming with phased arrays and MIMO antennas are used in smart antennas (Mathias WeiB, 2009). Projection of the estimated covariance matrix into the subspace of the ideal signal plus interference covariance matrix, results in an effective reduction in covariance matrix estimation error. Better SINR and faster convergence are also obtained but with higher computational complexity (Rahmani and Bastani, 2014).

The various classes of smart antennas include adaptive arrays, multibeam multiple -input - multiple - output antennas and diversity - antenna systems. Usage of smart antennas instead of traditional antennas results in improved performance (Chang and Hu, 2012). Adaptive beamforming algorithms find application in smart antenna systems for wireless communications. The convergence behavior of adaptive algorithms like Least Mean Square (LMS) and Recursive Least Square (RLS) varies with change in parameters including number of interference signals, number of

antenna elements, LMS step size, number of iterations and RLS weighting factor (Shubair and Merri, 2005).

For reducing system complexity and energy consumption, a reference signal based adaptive beamformer for vector sensor arrays consisting of crossed dipoles is used with focus on reducing the number of sensors involved in the adaptation. Effective beamforming is obtained with reweighted zero-attracting quaternion-valued least-mean-square algorithm (Jiang et al., 2016). The interference-plus-noise covariance matrix and the desired signal covariance matrix are reconstructed by matched spectrum processing and the weight vector is directly obtained using the general-rank minimum variance distortionless response method resulting in a robust beamforming algorithm. This method requires only little prior information and efficient beamforming is obtained (Chen et al., 2015).

The array steering vector is treated as a vector lying within the intersection of two subspaces and estimated using a closed-form formula. The average of the noise Eigen values replaces the desired signal Eigen value in the covariance matrix and only requires knowledge of the antenna array geometry and angular sector. Robust performance is achieved even in the presence of large look direction error in the array as long as the input signal-to-noise ratio (SNR) is not close to the interference-to-noise ratio (INR) and the desired signal and interference signals are well separated (Shen et al., 2015).

The MVDR and LCMV beamformers can also be used when the interference and ambient noise coexist with the target source. The MVDR is decomposed into the LCMV and a matched filter. Both components are properly weighted to achieve maximum interference-plus-noise reduction. Closed-form expressions for output

signal-to-interference ratio (SIR) and output signal-to-noise ratio (SNR) are obtained, with trade-off between noise reduction and interference rejection (Souden et al., 2010). A spatiotemporal framework for spectral estimation based on the linearly constrained minimum variance (LCMV) beamforming method offers robustness to the effects of noise and reverberation. The desired signal is modeled by an autoregressive process, and the resulting autoregressive coefficients are embedded in the linear constraints, resulting in significant reduction in anomalous estimates and lesser contribution from unwanted noise (Dmochowski et al., 2008).

Two robust methods for coherent focused wideband Minimum Variance Distortionless Response (MVDR) beamforming, aim at reducing the sensitivity of the coherent MVDR to focusing errors. The first method is based on modifying the beamformer optimization problem and generalizing it to bring into account the focusing transformations and the second is based on modifying the focusing scheme itself. Significant performance improvement of the proposed robust schemes is demonstrated when the Wavefield Interpolated Narrowband Generated Subspace focusing transformation is applied (Bucris et al., 2010). An adaptive beamformer with near instantaneous convergence for highly non stationary scenarios is obtained by frequently updating the beamformer weight vector. This caters to the dynamic scenario but requires higher computational requirement due to matrix inverse computation. A computationally efficient implementation is devised, implemented and mapped on to TigerSHARC Digital Signal Processor (Sajith Mohan et al., 2012).

The beamformer estimation problem is projected onto the blind source extraction algorithm by parameterizing the mixing system and analyzed using three methods. First, with one procedure LCMV and MVDR beamformers are obtained. Second, the

desired source is obtained using only two appropriate linear combinations of correlation matrices. Third, selection beamformers are designed for a subset of sensors while the final beamformer exploits data from all sensors (Bloemendal et al., 2012). The theoretical solutions of the optimal weighted vector and maximum SINR are derived by using variable substitution method for the optimal multiple linearly constrained minimum variance (LCMV) beamformer under multiple desired signals case. Superior performance in comparison with conventional LCMV is obtained in SINR and computational load (Zhen-Hai Xu et al., 2013). Linearly constrained minimum variance (LCMV) beamformer can be applied to sensor networks. Suboptimal LCMV beamformers utilizing only a subset of the available sensors for scenarios with multiple desired and interfering sources in multipath environments result in signal enhancement. Procedures are derived for adding or removing either an active sensor or a constraint from an existing LCMV beamformer in closed-form, as well as generalized sidelobe canceller (GSC)-form implementations. Computational burden is reduced by using the previous coefficients of the beamformer in the updation process (Markovich, 2012).

A non-recursive computationally efficient implementation of the LCMV beamformer with fixed constraints is obtained by using the rich structure of the LCMV expression (Jakobsson and Alty, 2006). The implementation is extended by means of its time varying displacement structure to derive an efficient time-updating algorithm of the spatial spectral estimate, increasing the computational gain, especially for large arrays and fine frequency grids.

The least mean square – least mean square (LLMS) algorithm, which employs an array image factor , sandwiched in between two least mean square (LMS) algorithm

sections is applied for array beamforming. The convergence of LLMS algorithm is analyzed for two different operation modes; namely with external reference or self-referencing. Its overall error signal is derived by feeding back the error signal from the second LMS algorithm stage to combine with that of the first LMS algorithm section. Superior convergence performance, reduced sensitivity to variations in input SNR, higher stability in AWGN environments, and robust performance in the presence of Rayleigh fading are obtained. The fidelity of the beamformer output signal is indicated using scatterplots and error vector magnitude (Srar et al., 2010).

The Bessel beamformer with least mean square (LMS) beamforming algorithm using offset quadrature phase shift keying (OQPSK) minimizes MSE and gives good performance in terms of signal recovery, directive gain, saving in transmitted power and rate of convergence (Yasin and Akhtar, 2012). MMSE law is used for evaluating an adaptive array structure using a widely-linear weight. Widely-linear model of the received signal estimates the desired signal and provides a unified framework for beamforming in circular and non-circular sources with reduced MMSE between the beamforming output and the desired signal (JinYou et al., 2007).

Active Tap Detection-Normalized Least Mean Square (ATD-NLMS) algorithm for robust smart antenna system used in place of conventional Least Mean Square (LMS) and Normalized Least Mean Square (NLMS) algorithms give a high convergence rate as it estimates only the active taps for the beamformer. Narrow beam width with high gain and reduced side lobes lead to a better system efficiency of smart antenna system (Razia et al., 2012). A robust constrained LMS beamformer searches the optimal weight vector with constrained LMS update and jointly searches the actual array

steering vector of the desired signal based on steering vector-expanded algorithm to lower computational complexity and improve performance (Xin et al., 2006).

An FPGA-based methodology is used for the analysis, modelling and prediction of power dissipation in embedded array signal processing systems containing adaptive beamforming components. The adaptive beamforming design space is explored in terms of power, timing, overhead, arithmetic precision and computational resources. Design-space exploration is enabled in real-time and on actual received waveforms. A hardware prototype based on Xilinx's Virtex 7 FPGA is implemented for a four channel Least-Mean-Squares (LMS) beamformer (Waheed, 2015).

The shrinkage linear complex-valued least mean squares (SL-CLMS) and shrinkage widely linear complex-valued least mean squares (SWL-CLMS) algorithms are suited for adaptive beamforming. By exploiting the relationship between the noise-free a posteriori and a priori error signals, the SL-CLMS method provides a variable step size to update the weight vector for the adaptive beamformer, significantly enhancing the convergence speed and decreasing the steady-state misadjustment. Along with adopting a variable step size determined by minimizing the square of the augmented noise-free a posteriori errors, the SWL-CLMS approach exploits the noncircular properties of the signal of interest, which considerably improves the steady-state performance. Improved performance with reduced computational complexity results but the convergence is slower (Shi et al., 2015). Two online implementations of the sample matrix inversion minimum variance distortionless response (SMI-MVDR) antenna array beamformer, based on recursive updating of the diagonal loading triangular matrix decomposition are discussed. The first beamformer uses the Cholesky factorization recursive updating of the matrix while the second beamformer

uses the Householder transform (HT) recursive updating of the modified input data matrix. The implementations consist primarily of vector operations and are suitable for parallel implementations with FPGAs or DSPs (Zaharov and Teixeira, 2008).

The subtraction-based SMI (S-SMI) beamformer imparts robustness by blocking the desired signal from the received data before calculating the beamformer weight vector. The effect of both finite sample size and arrival angle mismatch are considered for forming closed-form approximations of the expected value of the signal - to - interference - plus - noise ratio (SINR). True values of the SINR are obtained when the sample size is small and the arrival direction mismatch exists (Yu et al., 2010).

Corrections of the sample methods in Sample Matrix Inversion resulting in reduced computational cost are defined that counteract their performance degradation in the small sample size regime and keep their optimality in large sample size situations. The twofold approach considers shrinkage estimators which shrink the sample LMMSE or sample MVDR filters towards a variously called matched filter or conventional beamformer in array processing for small filters and random matrix theory for obtaining the optimal shrinkage factors for large filters (Serra and Nájjar, 2014).

Modifications are made to the conventional recursive least square (RLS) beamformer for reducing the computational complexity by applying vector operations instead of matrix operations and triangular decomposition. This ensures reduction in computational complexity as well as increased robustness against computational errors (Zaharov and Teixeira, 2005).

For realistic environments which uses blind adaptive beamforming algorithms for smart antennas, a recursive least squares (RLS) type optimization algorithm using a constrained constant modulus design criterion is derived. Results indicate significantly faster convergence rate, better robustness to changeable environments and better tracking capability (Wang and de Lamare, 2007). The maximum signal-to-interference plus noise ratio (MSINR) beamforming problem in antenna array CDMA systems is addressed. A modified MSINR criterion is reinterpreted as an unconstrained scalar cost function and recursive least-squares (RLS) algorithm is applied to minimize the cost function along with a rigorous analysis of the convergence properties of the algorithm by using the stochastic approximation theory (Jian Yang, 2005).

A low-complexity variable forgetting factor mechanism is used to enhance the performance of recursive least squares (RLS) algorithms for adaptive blind beamforming. The beamformer is designed according to the constrained constant modulus (CCM) criterion, and the algorithm operates in the generalized sidelobe canceler (GSC) structure for implementation. The variable forgetting factor mechanism employs a new component updated by the time average of the constant modulus (CM) cost function, to adjust the forgetting factor (Qin et al., 2013). The adaptive RLS algorithm is modified to impose linear and quadratic constraints on the adaptive weights to meet certain performance criteria. An adaptive narrowband beamformer with first and second order spatial derivative constraints is considered and simulation studies of its performance are discussed (Tuthill et al., 1995).

The RLS algorithm uses a sliding rectangular window (RLS-SW) to estimate the covariance matrix and invert it recursively, imparting robustness to real-time adaptive

beamforming (Long and Xin-yue, 2010). The fixed point numerical performance of the vector RLS (VRLS) beamformer has linear computational complexity and high convergence rate. The design and development of the core VRLS beamformer block using Xilinx System Generator is presented (Zaharov et al., 2007).

The GSC beamformer uses a systematic scheme for constructing a multiple constraints sparse blocking matrix. The sparsity of the blocking matrix substantially reduces the computational complexity. Signal leakage and the blocking ability of the sparse blocking matrix are analyzed (Markovich-Golan et al., 2013). An affine combination of two adaptive filters is used to enhance the performance of a generalized sidelobe canceller (GSC) in smart antenna array processing. A mixing vector is used to weight the coefficients of each component filter in such a way that the overall performances, such as convergence speed and the steady-state performance, of conventional GSC are improved (Lu, 2012).

A constant beam width discrete Fourier transform (DFT) beamformer based on the generalized sidelobe canceller (GSC) decomposes the broadband signals into frequency bins which are grouped into octaves and tapered individually. The resulting beam pattern possesses constant beam width across the entire operating spectrum, thus ensuring uniform spatial resolution. Further incorporation of the GSC allows adaptive nulling of interference to coincide with uniform resolution, enhancing the beam former's performance. Modification to the constraint equation of the conventional GSC is required to account for the frequency-dependent weighting of sensors (Choo Leng Koh, 2005).

A hardware-reduced digital beamformer is experimentally validated using a coding technique that allows a single analog-to-digital converter to serve all antenna array

elements. The validation test bed consists of two signal path receiver with discrete off-the-shelf components. FPGAs are used for digital post-processing and beamforming (Alwan et al., 2013). A digital beamformer is designed using a uniform linear (or planar) array employing a spatial interpolation process at the digital signal processing section to reduce the number of antenna elements and the corresponding RF modules, analog/digital (A/D) converters, resulting in narrow main beamwidth along with low side lobe levels (Tuan Do-Hong and Russer, 2003). A combination of DSP and FPGA is used to design a digital beamformer for base band signal through band-pass sampling, I / Q digital branch decomposition and digital filtering (Xin et al., 2010).

A compact uniplanar design of 2-D antenna arrays is used for digital beamforming (DBF) and direction-of-arrival (DOA) estimations. The array is based on switchable microstrip slot antennas that are series fed in one dimension and parallel fed in the other dimension. In conjunction with spatial multiplexing of local elements techniques, the array is able to use both the series and parallel-fed antennas for 2-D beamforming and DOA estimations. The combination of series and parallel feeding provides a simple way of constructing 2-D DBF arrays with a simple and uniplanar feed network, ensuring low cost and compatibility with a microwave integrated circuit process for the receiver system (Kim and Wang, 2009).

Cascaded integrated comb (CIC) filter is used in digital up conversion (DUC) and digital down conversion (DDC) for efficient transmission and reception in multirate system. Single stage CIC filter and multistage CIC filters for interpolation and decimation are developed using Xilinx System Generator and their efficiency is analyzed (Elamaram et al., 2013).

In constant modulus algorithm (CMA) and least mean square (LMS), use of the wavelet transform reduces the signal space dimension and hence results in low complexity in addition to data denoising and time saving in signal beamforming processing. Faster convergence is achieved with different wavelet families (Shu and Han, 2009). Wavelet denoising is applied for estimation of the direction of arrival. Prior to the estimation of the DOA, each of the sensor outputs is denoised according to wavelet denoising in order to enhance the input SNR, reflecting a better DOA performance over the conventional methods (El-Khamy and M. Shokry, 2013).

Spatially adaptive multiwavelet (MWT) denoising technique is applied to the GSC in an environment with severe diffused noise. Different approaches for integrating the GSC and the multiwavelet denoiser are presented (Ma et al., 2007). The DOA estimation performance is improved by employing a pre-processor that enhances the SNR, before performing the DOA estimation. Spatial wavelet packet transform (SWPT) followed by a block thresholding scheme based on the norm of SWPT subvectors in different spatial frequency subbands is used for enhancing the SNR. The advantage is that high sampling rates are not required. Spatial wavelet based denoising requires sampling rate just 2-4 times the signal frequency, whereas temporal wavelet denoising (TWD) requires a much higher sampling rate for achieving a comparable SNR gain. SWD shows higher DOA estimation performance indices at lower sampling rates, such as bias, mean square error and resolution, compared to those achieved by TWD or by undenoised data (Sathish and Anand, 2004).

Two algorithms operate in beamspace and provide closed-form arrival angle estimates in conjunction with uniform linear arrays by windowing the array data using either a

cosine shaped window or a Hanning window prior to beamforming. The resulting array patterns have low sidelobe levels and provide enhanced out-of-band source suppression (Mathews and Zoltowski, 2002).

Digital and analog beamformers find application in radar. The implementation of fractional delay filter (FD) using partially serial architecture with FPGA is discussed. Simulation is performed with ISE using devices, SPARTAN-3ADSP and VIRTEX 5. The SPARTAN-3ADSP based XC3SD1800ACS484-4 device is compared with VIRTEX 5 based XC5VLX50TFF1136-3. The fractional delay filter on VIRTEX 5 is observed to be faster than SPARTAN-3ADSP (Khanna et al., 2017).

Appropriate signal model derivation for transmitter and receiver to achieve side lobe reduction in phased array antenna and for determining direction of arrival (DOA) is discussed. Dolph-Chebyshev array structure is used for placing nulls in the antenna radiation pattern. This structure ensures nulls have been placed at proper positions to jam the interferers. For this purpose, interferers can be tracked by using direction of arrival estimation. This work investigates DOA for smart antenna array and thereby computes the error in DOA. This error is then removed by adding appropriate phase offset into input phase value to the array based on look-up table (LUT) (Khedekar and Mukhopadhyay, 2016).

Coprime arrays are observed to be as effective as sparse configurations suitable for radar beamforming and angle-of-arrival estimation applications. An approach for passive beamforming using coprime arrays is discussed with a new detection strategy being proposed, which offers improved performance in terms of peak sidelobe ratio and integrated sidelobe ratio. The advantage of the proposed detector is demonstrated

through the simulation of appropriate array scanned responses and receiver operating characteristic curves (Di Martino and Iodice (2017)).

The next section discusses the objectives of the thesis.

## 1.5 OBJECTIVES

The main objectives of the work presented in this thesis are:

- ☆ To develop a computationally efficient algorithm for beamforming.
- ☆ To validate the efficiency of the algorithm on various beamformers using MATLAB<sup>®</sup> simulations.
- ☆ To implement the beamformers on FPGA using Xilinx System Generator modelling.

## 1.6 METHODOLOGY

The methodology adopted in this thesis is as follows:

The input signal to the antenna/sensor array is transformed using the Reduced One Dimensional Mapped Real Transform (R-1D-MRT) and then used for beamforming. MATLAB<sup>®</sup> simulations of R-1D-MRT based beamformers are carried out to validate the efficiency of the algorithm for beamforming. Xilinx System Generator models of the beamformers are then created and simulated using Simulink<sup>®</sup>. For FPGA implementation, the models are synthesized using the Xilinx ISE<sup>®</sup> and Altium Designer<sup>®</sup> software. The FPGA schematic is created and built in the Altium Designer<sup>®</sup> and the bit file generated is downloaded into the Xilinx Spartan 3AN FPGA in the Altium NB3000 NanoBoard. The output values can be viewed using the NB3000 NanoBoard peripherals and interfaces.

A pictorial representation of the methodology adopted is illustrated in Fig 1.5.

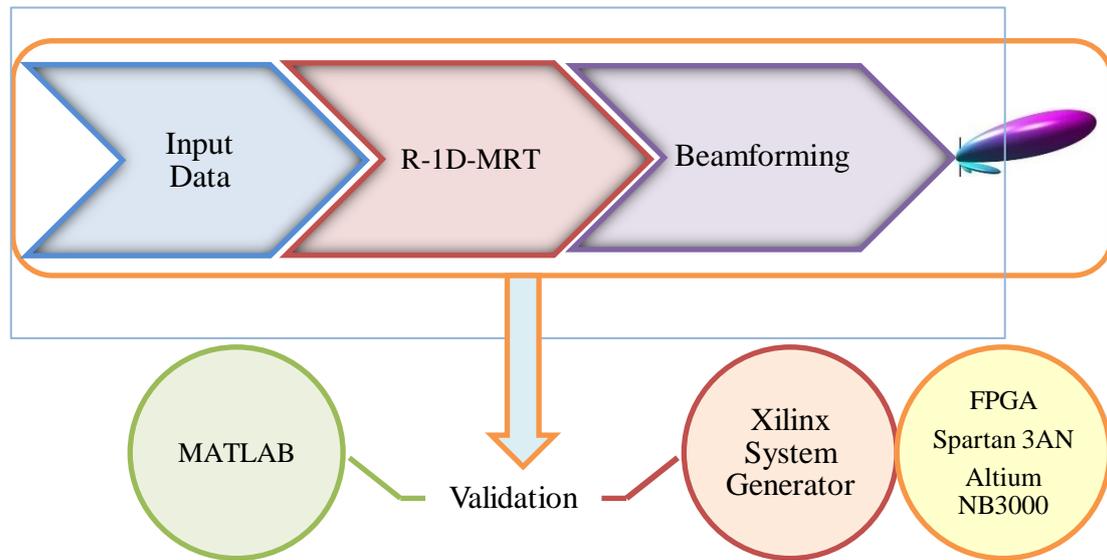


Fig 1.5 Methodology

Field Programmable Gate Arrays (FPGAs) are semiconductor devices that are based around a matrix of configurable logic blocks (CLBs) connected via programmable interconnects. FPGAs can be reprogrammed to desired application or functionality requirements after manufacturing. This feature distinguishes FPGAs from Application Specific Integrated Circuits (ASICs), which are custom manufactured for specific design tasks.

The FPGA schematic is shown in Fig 1.6.

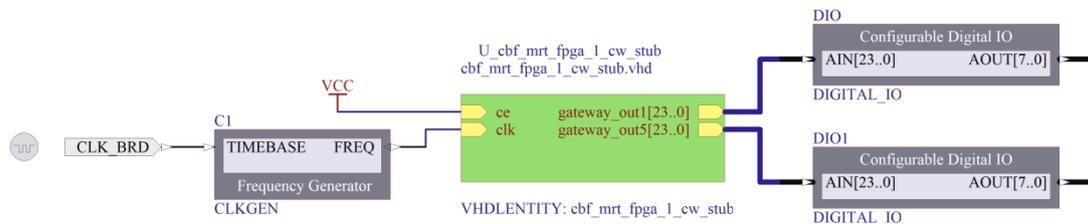


Fig 1.6 FPGA Schematic

The green block is the schematic which contains the Hardware Description Language code corresponding to the Xilinx System Generator model. This block is interfaced with the required input and output peripherals like Vcc, clock and digital IO. The output is observed on the Digital IO.

The Xilinx System Generator for DSP is a plug-in to Simulink<sup>®</sup> that enables designers to develop high-performance DSP systems for Xilinx FPGAs. Designers can design and simulate a system using MATLAB<sup>®</sup>, Simulink<sup>®</sup>, and Xilinx library of bit/cycle-true models. The tool will then automatically generate synthesizable Hardware Description Language (HDL) code mapped to Xilinx pre-optimized algorithms. This HDL design can then be synthesized for implementation on Xilinx FPGAs. As a result, designers can define an abstract representation of a system-level design and easily transform this single source code into a gate-level representation. Additionally, it provides automatic generation of a HDL test bench, which enables design verification upon implementation.

Xilinx ISE (Integrated Synthesis Environment) is a software tool provided by Xilinx for synthesis and analysis of HDL designs, enabling the developer to synthesize their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer. It is a design environment for FPGA products from Xilinx, and is tightly-coupled to the architecture of such chips.

Altium Designer<sup>®</sup> is a premier tool suite for Engineering Design Automation (EDA). All aspects of product development including schematic capture, PCB layout, embedded software development, FPGA design, mechanical CAD, procurement, design for manufacture and formal design data management are performed within a

single unified design environment. It can be used to design and produce high-quality electronics products.

Altium NanoBoards are reconfigurable development platforms that give designers access to the high-capacity, low-cost programmable devices in use today. Working closely with Altium Designer<sup>®</sup>, the NanoBoard allows rapid and interactive implementation and debugging of a digital design, directly on the target FPGA. Development and debugging capabilities include a full embedded tool chain for supported embedded processors and a range of embedded instruments, including a Logic Analyser. The Altium NanoBoard NB3000 is a 3000XN – Xilinx variant, incorporating a Xilinx Spartan-3AN device (XC3S1400AN-4FGG676C) as the user FPGA. A pictorial representation of the Altium NB3000 NanoBoard is shown in Fig 1.7.



Fig 1.7 Altium NB3000 NanoBoard

The experimental setup for FPGA implementation is shown in Fig 1.8.



Fig 1.8 Experimental setup with Altium NB3000 NanoBoard

## 1.7 THESIS ORGANIZATION

The thesis is organized into six chapters and two appendices.

Chapter 1 discusses the concept of beamforming and gives an overview on various classes of beamformers. The motivation for the thesis, literature survey and objectives of the work are discussed. The methodology adopted is presented followed by an introduction to Xilinx System Generator modeling. The features of the Altium NB3000 Nano Board used for FPGA implementation are also described.

Chapter 2 discusses the development of the R-1D-MRT algorithm for beamforming. The proposed algorithm transforms the incoming data vector comprising  $N$  samples into a vector of  $N/2$  samples. The computational load on the beamformer is thus reduced considerably. This chapter also validates the R-1D-MRT algorithm on conventional beamformers (CBF) (Rajesh, 2009).

A comparison of conventional beamforming using raw data and R-1D-MRT transformed data is presented by analyzing their computational complexity and performance in various communication channels. Hardware validation carried out with Xilinx System Generator model of the conventional beamformer and its FPGA implementation is described. A comparison of FPGA resource utilization is also shown (Torres Rossario, 2005).

Conventional beamformers are not suitable for interference prone environments. Chapters 3-6 investigate the efficiency of the proposed algorithm in adaptive beamformers.

The validation of the algorithm on Minimum Variance Distortionless Response (MVDR) beamformer and Linearly Constrained Minimum Variance (LCMV) beamformer is described in Chapter 3. These beamformers minimize the variance of the beamformer output subject to a constraint on the gain of the desired signal and also nullifies the interfering signals. The beamformer output obtained satisfies the constraint specified, besides offering reduced computation time and improved performance in various communication channels. Validation using FPGA implementation of these beamformers is also presented (Pan et al., 2014).

Chapter 4 discusses the validation of the algorithm on Least Mean Square (LMS) and Normalized Least Mean Square (NLMS) beamformers. These beamformers continuously adapt their weights to minimize the mean square error between the beamformer output and the desired signal. Observations also show faster convergence with lower step size. Hardware validation of these beamformers is also carried out on FPGA (Mohammad and Zainol, 2006).

In Chapter 5, the proposed algorithm is validated on Recursive Least Square (RLS), Sample Matrix Inversion (SMI) and Generalized Sidelobe Canceller (GSC) beamformers. The weight adaptation process is a recursive one and ensures faster convergence. Discussions on the characteristics of the RLS beamformer are presented (Simon Haykin, 2012). The SMI beamformer performs weight adaptation using blocks of data to ensure rapid convergence. The GSC beamformer rejects multiple interfering signals. Characteristics of these beamformers are analyzed in this chapter.

Chapter 6 summarises the conclusions drawn from the research work. Suggestions for future studies are also provided.

Appendix A presents a comparison of the noise reduction properties of the R-1D-MRT CBF and wavelet denoised CBF. Various denoising techniques like Rigrsure, Heursure, Minimaxi and Sqtwolog are considered.

Appendix B presents an assessment of the side lobe reduction obtained in conventional beamformers using R-1D-MRT and tapering. Comparison of sidelobe reduction in CBF using wavelet denoising and tapering is also performed. The analysis is carried out with various taper windows like Hamming, Hanning, Kaizer, Bartlett etc.

A pictorial depiction of the organization of the thesis is shown in Fig 1.9.

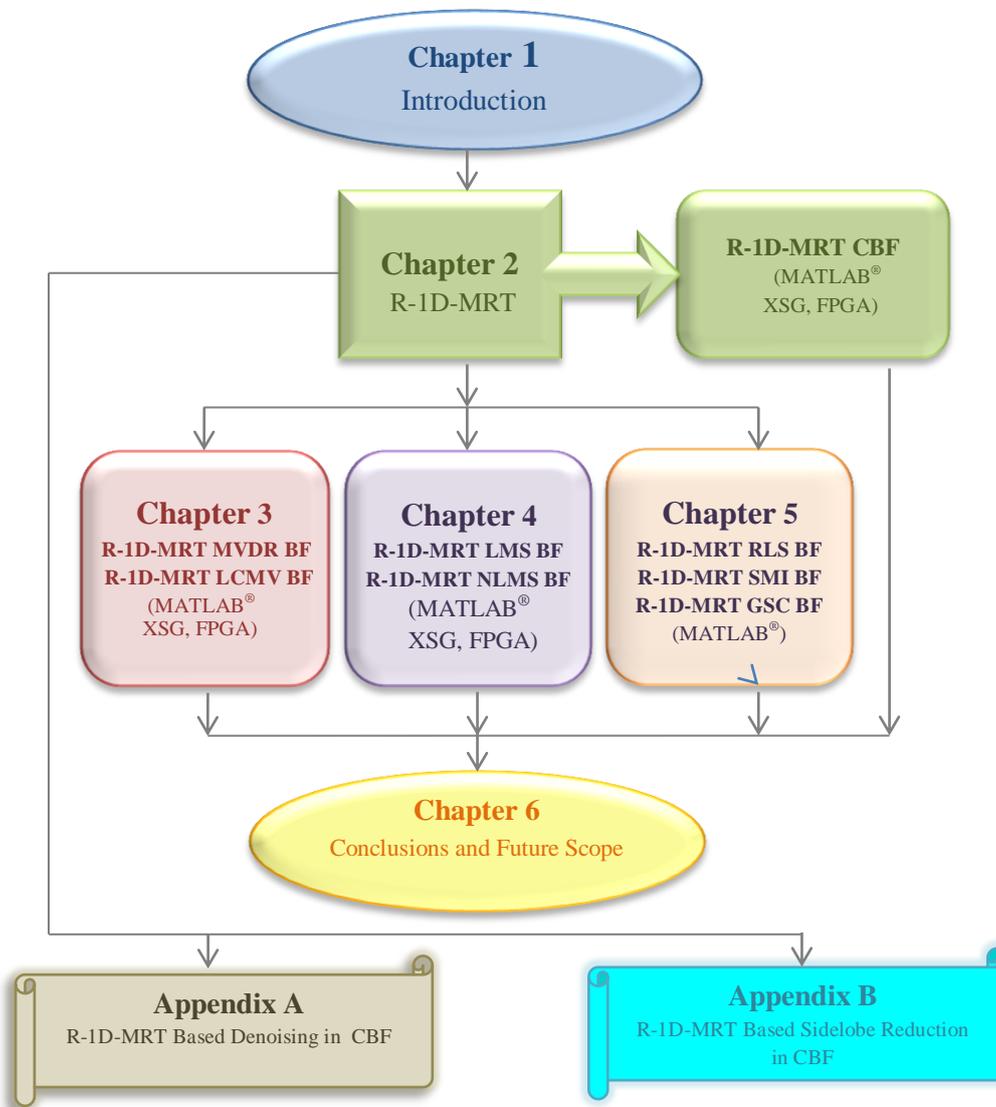


Fig 1.9 Organization of the thesis

## 1.8 CHAPTER SUMMARY

An overview of the concepts of beamforming is presented in this chapter. The various classes of beamformers are also discussed. The motivation for the work, literature survey, objectives of the thesis and methodology adopted are presented. This chapter also outlines the organization of the thesis.

## CHAPTER 2

# DEVELOPMENT OF REDUCED ONE DIMENSIONAL MAPPED REAL TRANSFORM

☆ One Dimensional Mapped Real Transform	CONTENTS	☆ FPGA implementation of R-1D-MRT CBF
☆ The Proposed Reduced One Dimensional Mapped Real Transform		❖ Xilinx System Generator model
☆ R-1D-MRT Conventional Beamformer		❖ Beam pattern and beamformer output
❖ Beam pattern and beamformer output		❖ FPGA resource utilization
❖ Computational complexity		
❖ Computation time		
❖ Performance in various communication channels		
❖ Interference signal rejection		

The development of the Reduced One Dimensional Mapped Real Transform (R-1D-MRT), a computationally efficient algorithm proposed for beamforming applications is presented in this chapter. This algorithm is based on the One Dimensional Mapped Real Transform (1D-MRT), (Rajesh, 2009). The features of 1D-MRT are discussed in section 2.1.

### 2.1 ONE DIMENSIONAL MAPPED REAL TRANSFORM

1D-MRT is an integer to integer transform, formed by grouping data elements based on the phase of the twiddle factor to which it belongs in the framework of the DFT definition. 1D-MRT carries both frequency and phase information and analyses signals from the point of view of phase. Only additions are required for performing the frequency domain analysis of one and two dimensional signals. Data which are related in terms of periodicity and symmetry of exponential terms in the Discrete Fourier Transform are grouped together.

Consider a one dimensional sequence  $x_n$ ,  $0 \leq n \leq N-1$  of size  $[1 \times N]$ .

Its DFT is  $Y_k$ ,  $0 \leq k \leq N-1$  of size  $[1 \times N]$  and is expressed as:

$$Y_k = \sum_{n=0}^{N-1} x_n W_N^{nk}, \quad 0 \leq k \leq N-1 \quad (2.1)$$

where

$$W_N^{nk} = e^{\frac{-j2\pi nk}{N}} \quad (2.2)$$

and  $k$  is the frequency index.

Using the periodicity of the twiddle factor  $W_N$ , equation 2.1 can be expressed as

$$Y_k = \sum_{n=0}^{N-1} x_n W_N^{((nk))_N} \quad (2.3)$$

The exponent  $((nk))_N$  can have a value  $p$ , where  $p$  is the phase index with  $0 \leq p \leq N-1$ .

For a given value of  $k$ , by grouping the data that share the same value of  $p$  for the exponent  $((nk))_N$ , and also using the relation

$$W_N^{p+\frac{N}{2}} = -W_N^p \quad (2.4)$$

$Y_k$  can be expressed as

$$Y_k = \sum_{p=0}^{M-1} Y_k^{(p)} W_N^p \quad (2.5)$$

where  $Y_k^{(p)}$  is the 1D-MRT of  $x_n$ ,  $0 \leq n \leq N-1$ , defined as

$$Y_k^{(p)} = \sum_{\forall n \Rightarrow ((nk))_N = p} x_n \quad \sum_{\forall n \Rightarrow ((nk))_N = p+M} x_n \quad (2.6)$$

where  $M = N/2$ . The size of  $Y_k$  will be  $[M \times N]$ .

The computation of the DFT coefficients requires the use of complex multiplications.

Unlike the DFT, the  $MN$  coefficients of 1D-MRT are computed in terms of additions only. The 1D-MRT maps an array of size  $(1 \times N)$  into a matrix of size  $(M \times N)$ .

As the 1D-MRT exploits the relation

$$W_N^{p+\frac{N}{2}} = -W_N^p \quad (2.7)$$

it is valid for all even values of  $N$ .

The 1D-MRT can be considered as a time frequency representation of a signal. It has localization in time as well as frequency.

Consider an example.

Let  $x = [95 \ 23 \ 61 \ 49 \ 89 \ 76 \ 46 \ 2]$ ,  $N = 8$

Then,  $Y_k^{(p)}$ , the corresponding 1D-MRT of  $x_n$ , is given in Table 2.1.

Table 2.1 One Dimensional Mapped Real Transform

$Y_k^{(p)}$	$k = 0$	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 7$
$p = 0$	441	6	77	6	141	6	77	6
$p = 1$	0	-53	0	47	0	53	0	-47
$p = 2$	0	15	48	15	0	15	-48	-15
$p = 3$	0	47	0	53	0	-47	0	53

From Table 2.1, it can be observed that many of the 1D-MRT coefficients occurs repeatedly with the same value or with negated values. Derivatives of 1D-MRT have been proposed, using selected elements out of the MN coefficients in the 1D-MRT matrix. These have found applications in texture descriptor analysis in image processing, study of chaotic behavior etc. (Meenakshy and Gopikakumari, 2009; Rajesh et al., 2007; Bhadran et al., 2008; Preetha et al., 2012; Bhadran, 2009). Since the computation of MRT involves only additions, the feasibility of reducing the computational burden of beamformers by using selected coefficients of 1D-MRT was

examined during this research work. Encouraging results were obtained when the elements of the 1D-MRT corresponding to a particular frequency index were used for beamforming, subject to sampling the signal at specific frequencies. This led to the development of the proposed Reduced One Dimensional Mapped Real Transform (R-1D-MRT) algorithm for beamforming.

## 2.2 THE PROPOSED REDUCED ONE DIMENSIONAL MAPPED REAL TRANSFORM

In the proposed R-1D-MRT algorithm for beamforming, only the elements of the second column of the 1D-MRT matrix, i.e., elements corresponding to frequency index  $k = 1$ , are chosen for processing. The number of elements in the data vector from each antenna/sensor used for beamforming after this transformation will be reduced from  $N$  elements to  $M$  elements where  $M = N/2$ . The suitability of the second column (corresponding to frequency index  $k = 1$ ) for beamforming is confirmed after exhaustive trials. The coefficients corresponding to  $k = 1$  are obtained by the following operation.

$$\mathbf{Y}_{\text{R-1D-MRT}}^{\substack{N \\ [\frac{N}{2} \times 1]}} = \mathbf{X}_n^{\substack{N \\ [(1:\frac{N}{2}) \times 1]}} - \mathbf{X}_n^{\substack{N \\ [(\frac{N}{2}+1:N) \times 1]}} \quad (2.8)$$

where,  $\mathbf{Y}_{\text{R-1D-MRT}}$  is the R-1D-MRT output vector and  $\mathbf{X}_n$  is the input data vector from the  $n^{\text{th}}$  antenna/sensor. The output data size from each antenna/sensor after applying R-1D-MRT is reduced to half its original size and the coefficients are obtained without involving any complex multiplication, using addition operations only. The result is a significant reduction in the size of the data required to be processed by the beamformer.

For  $N = 8$ , the relationship between the data elements and the MRT coefficients for  $k = 1$  is:

$$Y_1^{(0)} = x_0 - x_4; Y_1^{(1)} = x_1 - x_5; Y_1^{(2)} = x_2 - x_6; Y_1^{(3)} = x_3 - x_7;$$

Given that  $\mathbf{X} = [95 \ 23 \ 61 \ 49 \ 89 \ 76 \ 46 \ 2]$

The coefficients for  $k = 1$  are obtained as [Table 2.1]:

$$[(95 \ 23 \ 61 \ 49) - (89 \ 76 \ 46 \ 2)] = [(95-89) \ (23-76) \ (61-46) \ (49-2)] = [6 \ -53 \ 15 \ 47]$$

In beamforming applications, the input data is complex in nature, because of the phase shift introduced at the array. Hence the R-1D-MRT coefficients would also be complex.

The Forward R-1D-MRT can be expressed as:

$$\mathbf{Y}_{\text{R-1D-MRT}} = \mathbf{X}_{n_{[1 \times (1:\frac{N}{2})]}} - \mathbf{X}_{n_{[1 \times (\frac{N}{2}+1:N)]}} \quad (2.9)$$

Repeated simulation trials have confirmed that the original data can be obtained from the R-1D-MRT beam former output  $\mathbf{Y}$  using the relation:

$$\mathbf{X} = \left[ \frac{\mathbf{Y}}{2}; -\frac{\mathbf{Y}}{2} \right] \quad (2.10)$$

This transform developed specifically for beamforming applications is proposed as Reduced One Dimensional Mapped Real Transform (R-1D-MRT). (Formerly, it was proposed as One Dimensional Reduced Mapped Real Transform (1D-R-MRT)).

The performance of a conventional beamformer using R-1D-MRT is discussed in the following section.

### 2.3 R-1D-MRT CONVENTIONAL BEAMFORMER

The R-1D-MRT conventional beamformer (R-1D-MRT CBF) is presented here. Conventional beamformers (CBF) are the simplest type of beamformers. The signals from the antennas/sensors in the array are combined, primarily using only information about the location of the antennas/sensors in space and the signal direction of interest. The beamforming weights are set equal to the array response vector of the desired signal [Eqn 1.5]. For any particular direction  $\theta$ , the antenna beam pattern formed using this weight vector  $\mathbf{W}$  has the maximum gain in the direction  $\theta$ . This is accomplished because the weights vector  $\mathbf{W}$  adjust the phases of the incoming signals arriving at each antenna/sensor element from a given direction  $\theta$  for constructive addition. Conventional beamformers also offer robust performance in single target scenarios. They can direct the maximum of the beamformer towards the signal of interest but cannot place nulls in the direction of undesired interferers (VanVeen and Buckley, 1988). The schematic diagram of a conventional beamformer with  $N_s$  antennas/sensors is shown in Fig 2.1(a).

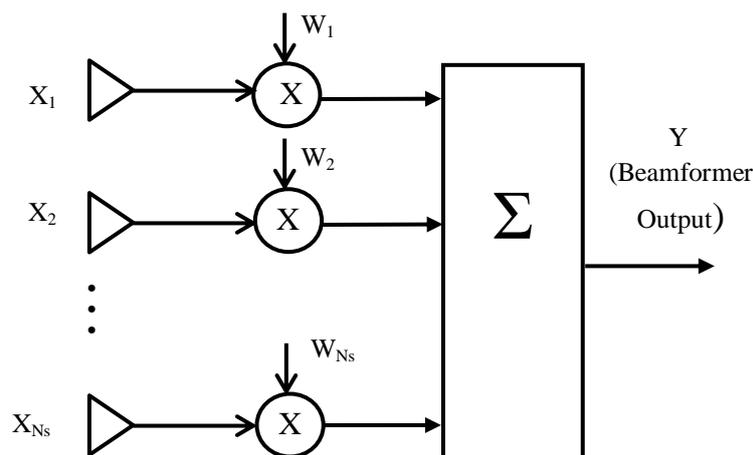


Fig 2.1(a) Schematic diagram of Conventional Beamformer

In the CBF the raw input is used for beamforming whereas, in the R-1D-MRT CBF, the signal received at each array element is transformed by applying the R-1D-MRT algorithm and the transformed signals are used for beamforming. The size of the data at the output of the R-1D-MRT block is half the size of the data at its input. Hence the size of the data required to be processed by the subsequent stages of R-1D-MRT CBF is reduced by half compared to CBF. The schematic diagram of the R-1D-MRT CBF with  $N_s$  antennas/sensors is shown in Fig 2.1(b).

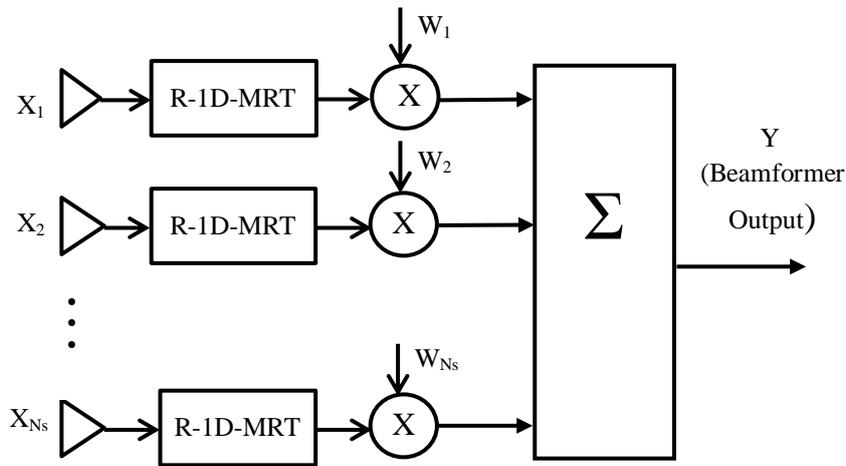


Fig 2.1(b) Schematic diagram of R-1D-MRT Conventional Beamformer

After multiplication by a complex weight, the signals are summed up to form the output  $Y$ .

$$\mathbf{Y} = \mathbf{W}^H \mathbf{X} \quad (2.11)$$

where,  $\mathbf{W}$  is the weight vector,  $\mathbf{X}$  represents the input signals and  $H$  represents Hermitian transpose. The evaluation of the R-1D-MRT CBF with respect to various parameters is presented in the following sections.

## 2.4 BEAM PATTERN AND BEAMFORMER OUTPUT OF R-1D-MRT CBF

This section examines the suitability of the R-1D-MRT CBF for obtaining the desired radiation pattern and beamformer output. An efficient beamformer should direct the main lobe of the radiation pattern towards the desired signal. The normalized beam pattern obtained with the R-1D-MRT CBF for varying number of antenna elements/sensors ( $N_s$ ) for an input data size ( $N$ ) of 64 is illustrated in Fig 2.2.

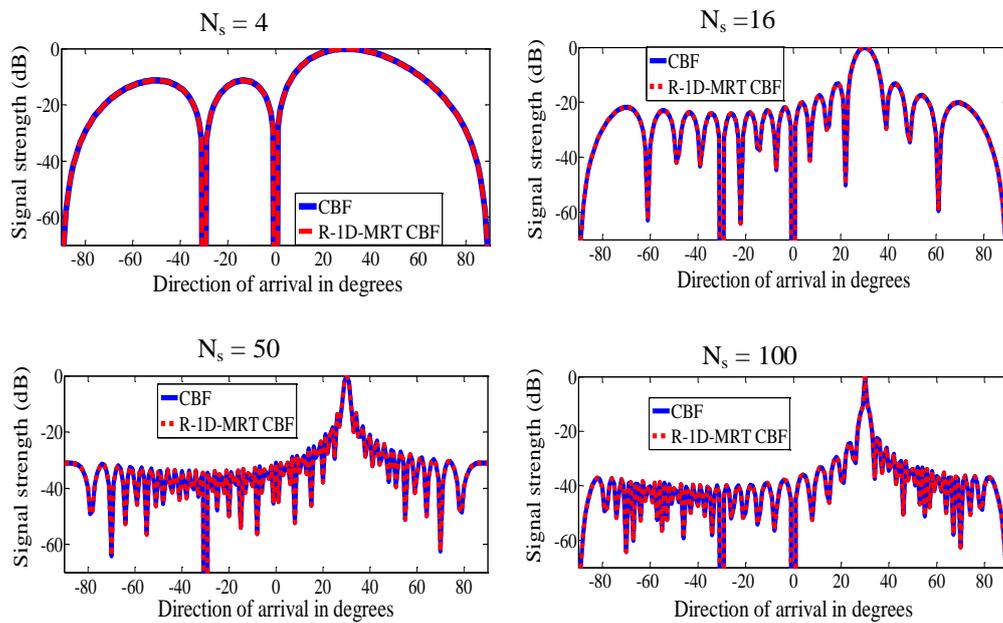


Fig 2.2 Normalized beam pattern of R-1D-MRT CBF for varying number of antennas/sensors [Source angle =  $30^\circ$ ,  $N = 64$ , SNR = 30 dB]

The signal originates from a source located at 30 degrees and signal to noise ratio is 30 dB. The normalized beam patterns show the main lobe at  $30^\circ$  and also exhibit increased directivity with increasing number of antennas/sensors ( $N_s$ ) similar to that of a conventional beamformer.

The simulations are repeated for varying locations of the source for an array of 8 antennas/sensors ( $N_s$ ), input data size ( $N$ ) of 512 and SNR of 30 dB. The normalized beam patterns obtained are plotted in Fig 2.3.

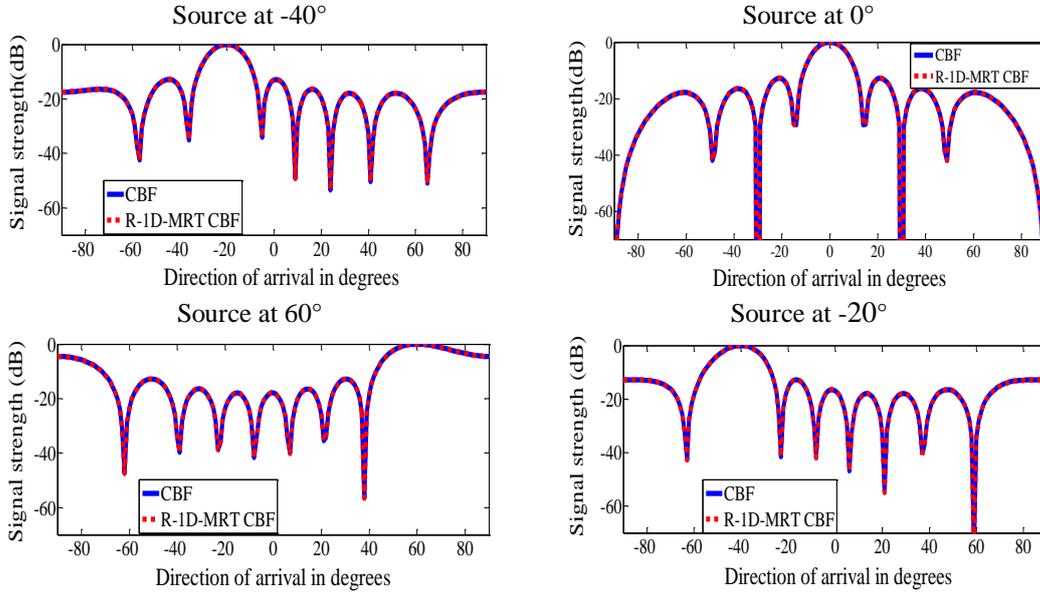


Fig 2.3 Normalized beam pattern of R-1D-MRT CBF for varying source locations [ $N_s = 8$ ,  $N = 512$ , SNR = 30 dB]

It is observed from the above results that the performance of R-1D-MRT CBF is at par with that of CBF using raw data. However, the computational load on the R-1D-MRT CBF is significantly reduced as detailed in section 2.5.

It is indicated by exhaustive simulation studies that faithful reproduction of the input signal is obtained only for specific sampling frequencies, decided by the size of the input data. For beamforming using R-1D-MRT, the signals must be sampled at the optimum sampling frequency ( $f_s$ ) specified for the particular data size ( $N$ ) before applying R-1D-MRT. The optimum sampling frequencies ( $f_s$ ) required for various data sizes ( $N$ ) for an input signal frequency  $f$  is presented in Table 2.2.

Table 2.2 Optimum sampling frequency ( $f_s$ ) for various data sizes ( $N$ )

$N$ (Data size)	Sampling frequency ( $f_s$ )
8192	4.2249 $f$
4096	4.1000 $f$
2048	4.5210 $f$
1024	4.0470 $f$
512	4.0960 $f$
256	4.6550 $f$
128	4.1290 $f$
64	4.9300 $f$
32	4.5700 $f$
16	5.3300 $f$

The normalized beamformer output for optimum sampling frequencies ( $f_s$ ) and for sampling frequencies greater/lesser than the optimum value for varying data sizes ( $N$ ) is presented in Fig 2.4.

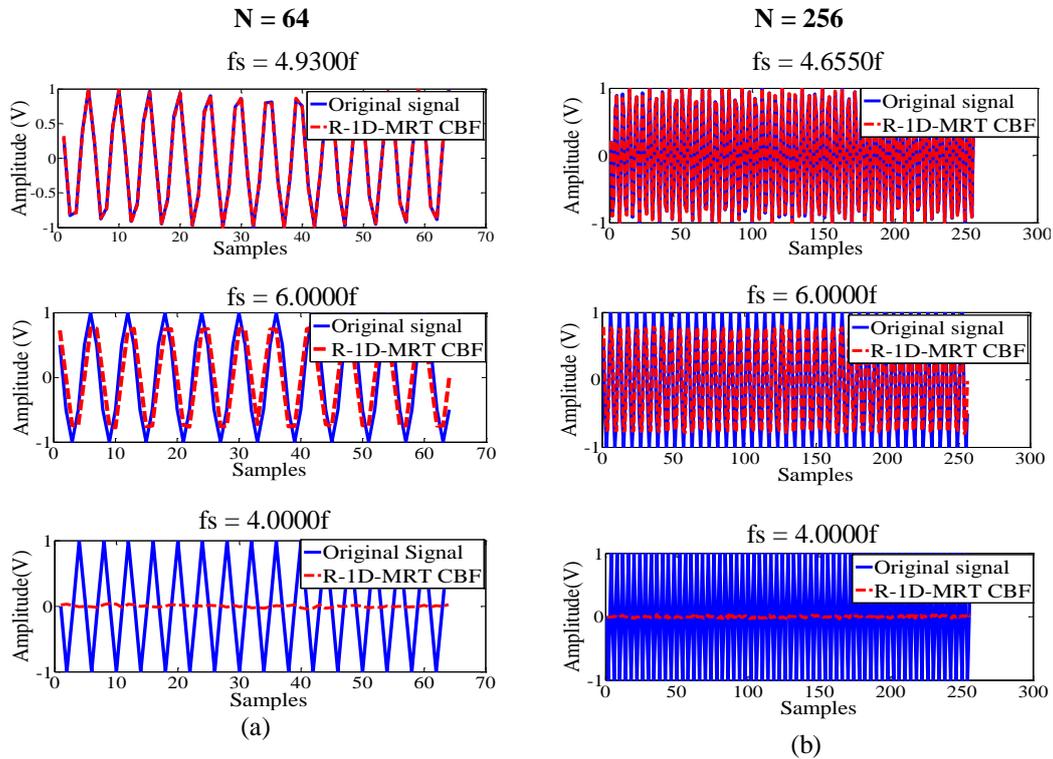


Fig 2.4 Normalized beamformer output of R-1D-MRT CBF  
[ $N_s = 4$ , Source angle =  $30^\circ$ , SNR = 30 dB]

For a data size of 64 samples, the optimum sampling frequency is 4.93 times the input signal frequency. At the optimum sampling frequency, the beamformer output signal matches the original signal as shown in Fig 2.4(a). Similarly for a data size of 256 samples, when the sampling frequency is set at the optimum value of  $4.655f$ , the beamformer output signal is comparable to the input signal as can be observed in Fig 2.4(b). If the sampling frequencies are set at values higher/lower than the optimum values, the beamformer output differs considerably from the input signal. The restriction on the sampling frequency may be attributed to the inherent features of R-1D-MRT, wherein the selected coefficients corresponding to the second column ( $k = 1$ ) of the 1D-MRT matrix, pertain to only the fundamental frequency component of the signal.

The output of R-1D-MRT CBF at varying look angles for a source located at  $30^\circ$  is analyzed in Fig 2.5.

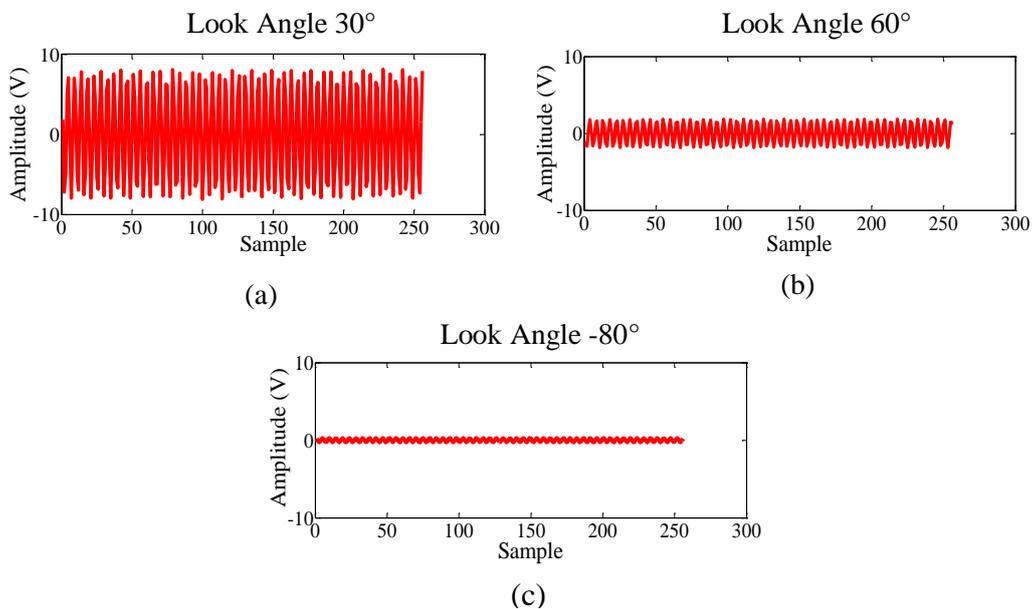


Fig 2.5 Beamformer output of R-1D-MRT CBF for varying look angles  
 $[N_s = 8, \text{Source angle} = 30^\circ, N = 256, \text{SNR} = 30 \text{ dB}]$

The beamformer output is high when the look angle and the source angle match as can be seen in Fig 2.5(a). When the look angle is different from the source angle, the beamformer output reduces as observed in Fig 2.5(b) and Fig 2.5(c). This confirms the fact that the R-1D-MRT CBF has high output in the desired direction.

The suitability of the R-1D-MRT CBF for obtaining the desired beam pattern and beamformer output is thus confirmed. An evaluation of the computational complexity of R-1D-MRT CBF is presented in the next section.

## 2.5 COMPUTATIONAL COMPLEXITY OF R-1D-MRT CBF

The suitability of the R-1D-MRT for beamforming was established in the previous section. To validate the computational efficiency of the R-1D-MRT for beamforming, an evaluation of the number of computations performed by the R-1D-MRT CBF is presented in this section. The input data to the beamformer is complex in nature and must be multiplied with complex weights for beamforming. Multiplication of 2 complex numbers requires 4 real multiplications and 2 real additions. An input of  $N$  complex numbers requires  $2(N-1)$  real additions for converting to R-1D-MRT form, resulting in  $N/2$  complex numbers.

These  $N/2$  complex numbers are multiplied with a complex weight in each sensor path. Hence in each sensor path,  $2N-4$  real multiplications and  $N-2$  real additions are required with the R-1D-MRT transformed data instead of  $4N-4$  real multiplications and  $2N-2$  real additions required with the original data. In the summation stage of the beamformer, the number of addition operations for each antenna input is reduced to  $N-2$  real additions using the R-1D-MRT transformed data instead of  $2N-2$  real additions required with the original data. If the number of antennas/sensors in the

array is  $N_s$ , then the total number of operations for conventional beamforming using the R-1D-MRT data is given as  $N_s \times (2N-4$  real multiplications and  $4N-6$  real additions) compared to  $N_s \times (4N-4$  real multiplications and  $4N-4$  real additions) required with the original data.

A comparison of the number of real multiplication and real addition operations for CBF and R-1D-MRT CBF using arrays of 4 antennas/sensors ( $N_s$ ) and 8 antennas/sensors ( $N_s$ ) for varying data sizes ( $N$ ) is presented in Table 2.3.

Table 2.3 Comparison of computational complexity of CBF and R-1D-MRT CBF

N (Data Size)	$N_s = 4$				$N_s = 8$			
	Number of real multiplications		Number of real additions		Number of real multiplications		Number of real additions	
	CBF	R-1D- MRT CBF	CBF	R-1D- MRT CBF	CBF	R-1D- MRT CBF	CBF	R-1D- MRT CBF
8192	131056	65520	131056	131048	262112	131040	262112	262096
4096	65520	32752	65520	65512	131040	65504	131040	131024
2048	32752	16368	32752	32744	65504	32736	65504	65488
1024	16368	8176	16368	16360	32736	16352	32736	32720
512	8176	4080	8176	8168	16352	8160	16352	16336
256	4080	2032	4080	4072	8160	4064	8160	8144
128	2032	1008	2032	2024	4064	2016	4064	4048
64	1008	496	1008	1000	2016	992	2016	2000
32	496	240	496	488	992	480	992	976
16	240	112	240	232	480	224	480	464

The R-1D-MRT CBF requires lesser number of multiplications and additions compared to CBF. Since the computation time for multiplication is more than for addition, the reduction in the number of multiplications obtained through the use of

R-1D-MRT results in significant saving in computation time, as discussed in the following section.

## 2.6 COMPUTATION TIME OF R-1D-MRT CBF

This section evaluates the effect of the reduction in the computation operations of the R-1D-MRT CBF, on the computation time required for beamforming. The time taken by the R-1D-MRT CBF and CBF for varying number of antenna/sensors and data sizes is compared in Tables 2.4 and 2.5.

Table 2.4 Comparison of computation time of CBF and R-1D-MRT CBF (seconds)  
[Source angle = 30°, SNR = 30 dB]

N (Data size)	N <sub>s</sub> = 4			N <sub>s</sub> = 16		
	CBF	R-1D-MRT CBF	% Reduction	CBF	R-1D- MRT CBF	% Reduction
8192	0.1068	0.0608	43	0.1435	0.0709	50
4096	0.0621	0.0351	43	0.0722	0.0383	47
2048	0.0367	0.0202	45	0.0413	0.0259	37
1024	0.0210	0.0156	26	0.0280	0.0190	32
512	0.0155	0.0131	15	0.0208	0.0132	36
256	0.0131	0.0116	16	0.0139	0.0120	14
128	0.0115	0.0109	5	0.0123	0.0112	9
64	0.0107	0.0102	5	0.0113	0.0109	4
32	0.0102	0.0101	1	0.0107	0.0103	4
16	0.0100	0.0098	2	0.0103	0.0099	4

The simulations were performed for a source at 30 degrees and SNR of 30 dB with a computer using Intel Core i5 CPU working at 2.20 GHz and RAM of 4GB.

Table 2.5 Comparison of computation time of CBF and R-1D-MRT CBF (seconds)  
[Source angle = 30°, SNR = 30 dB]

N (Data size)	$N_s = 50$			$N_s = 100$		
	CBF	R-1D- MRT CBF	% reduction	CBF	R-1D- MRT CBF	% reduction
8192	0.3588	0.1524	57	0.6145	0.3086	50
4096	0.1485	0.0553	62	0.3252	0.1199	63
2048	0.0567	0.0326	43	0.1166	0.0415	64
1024	0.0341	0.0226	33	0.0461	0.0262	43
512	0.0233	0.0186	20	0.0305	0.0198	35
256	0.0196	0.0160	18	0.0219	0.0171	22
128	0.0168	0.0144	13	0.0183	0.0155	15
64	0.0147	0.0110	25	0.0161	0.0146	9
32	0.0124	0.0114	8	0.0150	0.0109	27
16	0.0111	0.0107	4	0.0109	0.0105	4

The R-1D-MRT CBF requires lesser computation time compared to CBF. With large number of antenna/sensors ( $N_s$ ) and high data sizes ( $N$ ), the R-1D-MRT CBF requires only half the computation time taken by the CBF. The reduction in computational complexity and simulation time of the R-1D-MRT CBF validate the computational efficiency of the R-1D-MRT for beamforming. In the next section, the performance of the R-1D-MRT CBF in various communication channels is assessed.

## 2.7 PERFORMANCE OF R-1D-MRT CBF IN VARIOUS COMMUNICATION CHANNELS

The medium of propagation affects the performance of any communication system, namely the characteristics of the channel through which the signal propagates. The received signal may consist of a combination of attenuated, reflected, refracted, diffracted replicas of the transmitted signal and added noise. In this chapter, the performance of the R-1D-MRT CBF is investigated for communication through

Additive White Gaussian Noise (AWGN) channels, Rayleigh and Rician fading channels.

AWGN channel model is a communication channel model which has a linear addition of wideband or white noise with a constant spectral density and a Gaussian distribution of amplitude. It is a simple channel model and does not consider interference, nonlinearity or dispersion.

Rayleigh fading is a form of fading often experienced in environments where large numbers of reflections are present. Rayleigh fading models are used in statistical analysis of the propagation of radio waves for areas such as cellular communications in urban environments which have multiple reflections from buildings. It can also be applied in HF ionospheric radio wave propagation where reflections occur at many points within the ionosphere. The Rayleigh propagation model is most applicable to instances where there are many different signal paths, none of which is dominant. In this way all the signal paths will vary and can have an impact on the overall signal at the receiver. The Rayleigh distribution has a probability density function (pdf) given by:

$$p(r) = \begin{cases} \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right), & 0 \leq r \leq \infty \\ 0, & r < 0 \end{cases} \quad (2.12)$$

where,  $\sigma$  is the rms value of the received voltage signal, and  $\sigma^2$  is the time-average power of the received signal.

In the presence of a dominant stationary (nonfading) signal component, such as a line-of sight propagation path, the small-scale fading envelope distribution is Rician. In such a situation, random multipath components arriving at different angles are

superimposed on a stationary dominant signal. This results in addition of a dc component to the random multipath. The effect of a dominant signal arriving with many weaker multipath signals gives rise to the Rician distribution which is given by

$$p(r) = \begin{cases} \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) I_0\left(\frac{Ar}{\sigma^2}\right), & A \geq 0, r \geq 0 \\ 0, & r < 0 \end{cases} \quad (2.13)$$

The parameter A denotes the peak amplitude of the dominant signal and  $I_0(\bullet)$  is the modified Bessel function of the first kind and zero-order.

The performance of the R-1D-MRT CBF and CBF in various channels are evaluated and presented in terms of the Euclidian distance separation between the corresponding vectors. Euclidian distance metric gives a measure of the separation between 2 points. For 2 points X and Y, the Euclidian distance is computed as

$$d_{X,Y} = \sqrt{\sum_{i=1}^N (X_i - Y_i)^2} \quad (2.14)$$

Higher values of Euclidian distance indicate more separation between the vectors while smaller values indicate less separation between them. The Euclidian distance metric is computed between the original signal and the beamformer output signals for the R-1D-MRT CBF and CBF. The comparison is presented in Table 2.6.

Table 2.6 Comparison of Euclidian distance parameters of CBF and R-1D-MRT CBF for various communication channels for varying SNR [ $N_s = 4$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR	AWGN		Rayleigh		Rician	
	CBF	R-1D-MRT CBF	CBF	R-1D-MRT CBF	CBF	R-1D-MRT CBF
10 dB	1.7891	1.0661	1.5912	1.4718	1.5362	1.5011
20 dB	0.3260	0.2390	1.0257	1.0074	0.8820	0.6596
30 dB	0.1435	0.1198	0.8494	0.8454	0.3597	0.3123

The Euclidian distance metric for the R-1D-MRT CBF is lower than that of the conventional CBF for the communication channels considered, indicating the fidelity of the R-1D-MRT beamformer output signal.

A visual comparison of the performance of R-1D-MRT CBF and CBF using scatterplots is also presented in this chapter. A scatterplot or a constellation diagram uses Cartesian coordinates to plot a mathematical diagram to display values for typically 2 variables for a set of data. It can be used to view signal constellations and observe the impact of channel noise and distortions. A scatterplot is very useful for observing how two comparable data sets agree with each other. A constellation diagram of the original signal is taken as the reference. The more the two data sets agree, the more the scatters tend to concentrate in the vicinity of the reference points. If the two data sets are numerically identical, the scatters fall on the reference points exactly. The effect of noise manifests as a cloud observed around the reference point locations (Eugene Estinto, 2014).

The scatterplots for the original signal, signal with added noise, output of CBF and R-1D-MRT CBF in AWGN, Rayleigh and Rician channels are presented for varying SNR values in Fig 2.6 - Fig 2.8.

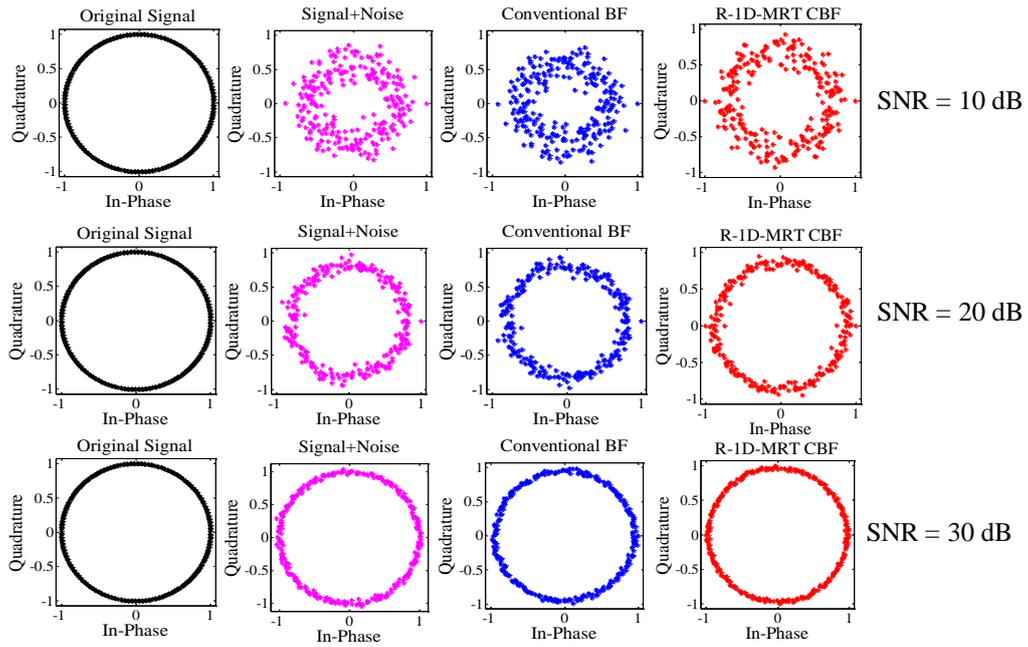


Fig 2.6 Scatterplots of R-1D-MRT CBF in AWGN channel for varying SNR [ $N_s = 4$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

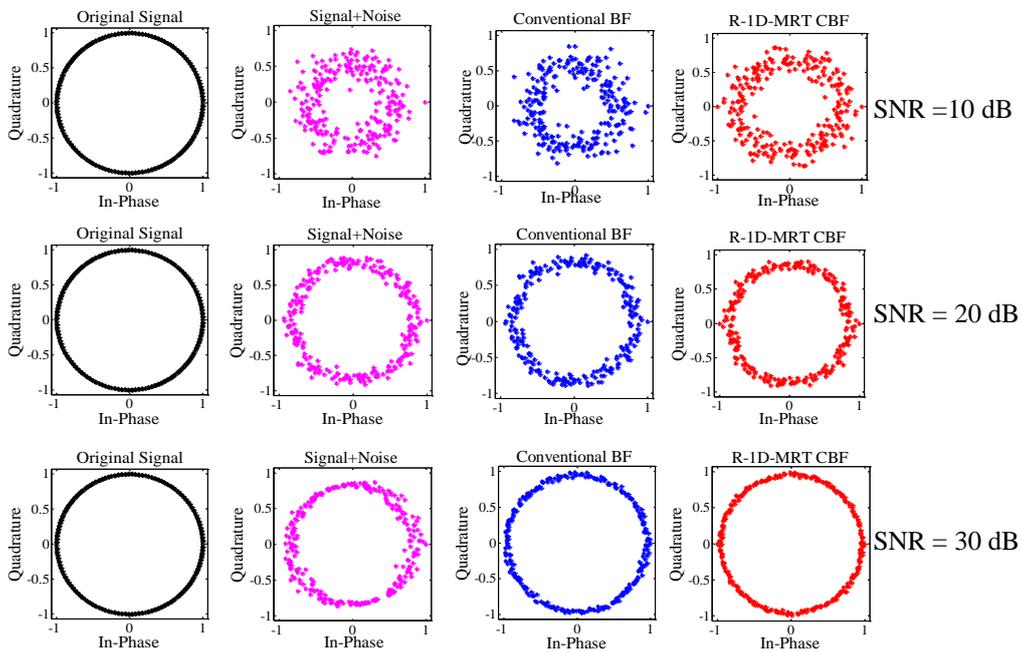


Fig 2.7 Scatterplots of R-1D-MRT CBF in Rayleigh channel for varying SNR [ $N_s = 4$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

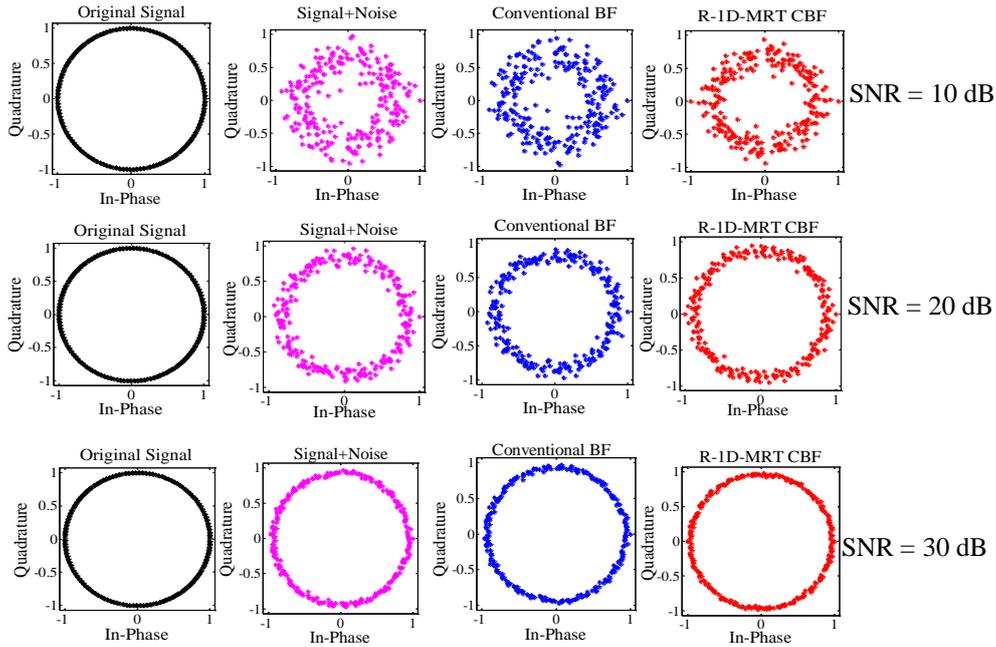


Fig 2.8 Scatterplots of R-1D-MRT CBF in Rician channel for varying SNR [ $N_s = 4$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

Figures 2.6 - 2.8 indicate that in high noise environments, dispersion is less for R-1D-MRT CBF in the AWGN, Rayleigh and Rician channels. As the noise levels reduce, the scatterplots of the R-1D-MRT CBF exhibit more likeness to the original signal constellation diagram. The discussions presented in these sections regarding computational efficiency and performance in various communication channels confirm the validity of R-1D-CBF for beamforming.

## 2.8 INTERFERENCE REJECTION OF R-1D-MRT CBF

Earlier sections have shown that the R-1D-MRT CBF maintains the peak of the beam pattern in the direction of the desired signal. In many situations, signals may also arrive from undesired directions. This section analyses the ability of the R-1D-MRT CBF to reject undesired signals. The beam pattern obtained with a signal

arriving from a desired source at  $30^\circ$  and a signal from an undesired interference at  $-20^\circ$  is illustrated in Fig 2.9.

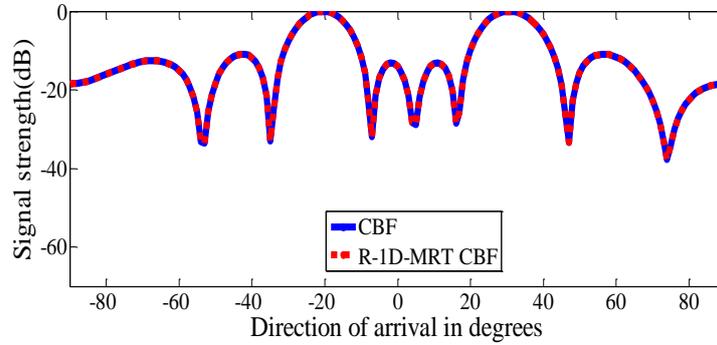


Fig 2.9 Normalized beam pattern of R-1D-MRT CBF for 2 sources [ $N_s = 8$ , Desired source angle  $30^\circ$ , Interference angle  $-20^\circ$ ,  $N = 256$ , SNR = 30 dB]

The normalized beam pattern shows a peak at both the signal directions, indicating the inability of the R-1D-MRT CBF and CBF to reject interfering signals. Hence for rejection of unwanted signals, the conventional method of beamforming is unsuitable. The suitability of adaptive beamformers for interference rejection is studied in the succeeding chapters.

The outcome of the simulation studies of the R-1D-MRT CBF has been presented in sections 2.4 -2.8. In the next section, the hardware implementation of the R-1D-MRT CBF is discussed.

## 2.9 FPGA IMPLEMENTATION OF R-1D-MRT CBF

The efficiency of the R-1D-MRT algorithm for beamforming has been proved in the discussions presented in the sections 2.4-2.8. For further validation, a hardware implementation of the R-1D-MRT CBF using FPGA is presented.

Beamforming can be performed in the analog domain or in the digital domain. In analog beamforming, amplitude/phase variations i.e., weights are directly applied to the analog signal from the various antennas and the weighted signals are then summed up [Fig 1.3]. In digital beamforming, the analog signal at the input of the array elements is first digitized by using analog to digital converters and digital down converters before application of weights and summing of the signals. Digital beamforming offers advantages in terms of power consumption, flexibility and accuracy. Digital systems tend to consume less power in computation operations and have a programmable interface adding versatility to the system. The amplitude scaling, phase shifting and summing up operations are implemented digitally (Kevin McClaning, 2010; Torres-Rossario J A, 2005). The schematic diagram of R-1D-MRT based conventional digital beamformer (R-1D-MRT CDBF) is shown in Fig 2.10.

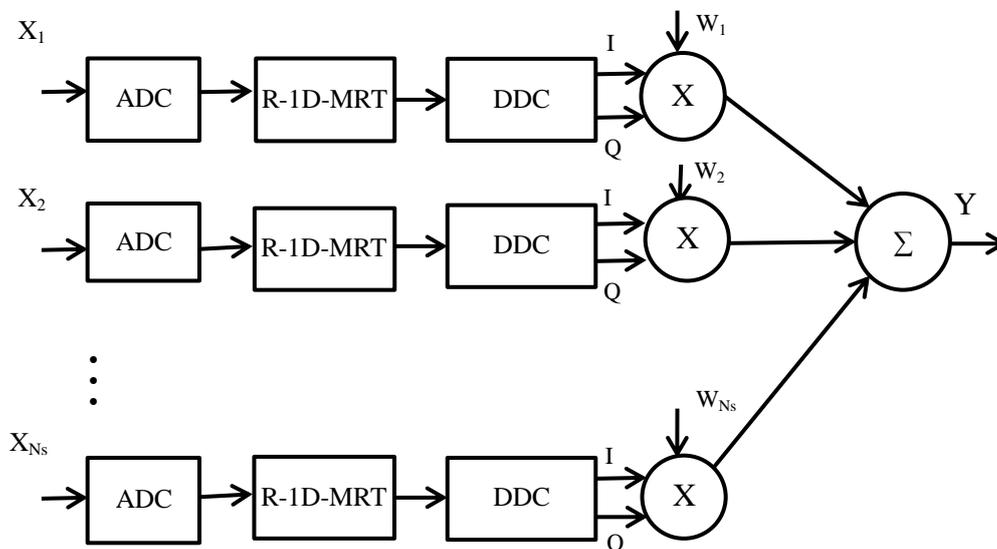


Fig 2.10 Schematic diagram of R-1D-MRT CDBF

The signals arriving at the array are sampled and converted into the digital domain. The digital signals are then transformed using the R-1D-MRT algorithm and then processed by a Digital Down Converter (DDC). The digital down conversion is performed by multiplying the digital signal by a sinusoidal signal and a  $90^\circ$  phase shifted version of the sinusoidal signal, both generated by a numerically controlled oscillator (NCO). DDCs are used to convert a digitized real signal centered at an intermediate frequency to a baseband complex signal centered at a zero frequency. It is also used to reduce the input signal to a lower sampling rate, thus allowing lower-speed processors to process fast signals. It can be implemented with the help of field-programmable gate arrays or application-specific integrated circuits.

The schematic diagram of a DDC is shown in Fig 2.11.

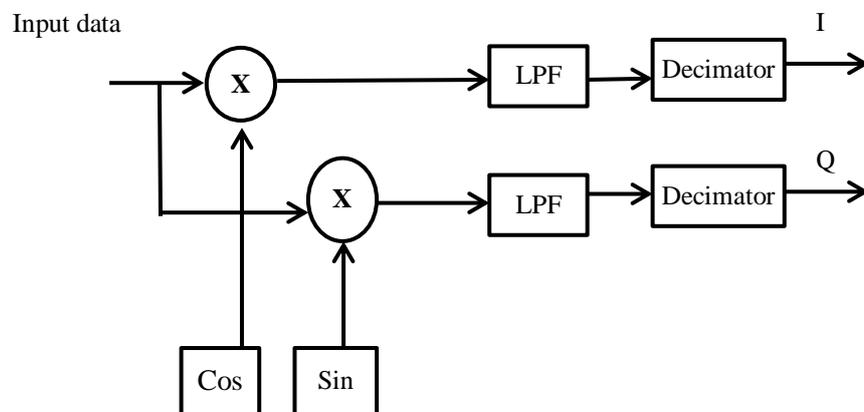


Fig 2.11 Schematic diagram of Digital Down Converter

A DDC consists of three subcomponents: a direct digital synthesizer (DDS), a low-pass filter (LPF), and a downsampler/decimator. The DDS generates a complex sinusoid at the intermediate frequency (IF). Multiplication of the intermediate

frequency with the input signal creates images centered at the sum and difference frequency. A major advantage of a direct digital synthesizer (DDS) is that its output frequency, phase and amplitude can be precisely and rapidly manipulated under digital processor control. Other inherent DDS attributes are ability to tune with extremely fine frequency and phase resolution, and rapidly shift between frequencies. The Xilinx DDS Compiler block is an inbuilt FPGA module in the Xilinx system generator. The block uses a lookup table scheme to generate complex sinusoids. A digital integrator (accumulator) generates a phase that is mapped by the lookup table into the output sinusoidal waveform. The lowpass filters pass the difference (i.e. baseband) frequency while rejecting the sum frequency image, resulting in a complex baseband representation of the original signal. In its new form, it can readily be down sampled and is more convenient for use in many DSP algorithms. Any suitable low-pass filter can be used including FIR, IIR and CIC filters.

Cascade Integrator Comb (CIC) filter is a flexible, multiplier-free filter suitable for hardware implementation and used in multi rate signal processing. It is a class of FIR filter combined with an interpolator or decimator. In a decimating CIC, the input signal is fed through one or more cascaded integrators(I), then a down-sampler or decimator( $\downarrow R$ ), followed by as many comb sections as the number of integrators (C).

A cascade of an integrator and a comb filter is equivalent to an FIR filter with an impulse response of a rectangular window of length M. R is the decimation factor of the filter and M is the differential delay (Loehning et al., 2000; <https://pdfs.semanticscholar.org>). The schematic diagram of CIC decimating filter is shown in 2.12.

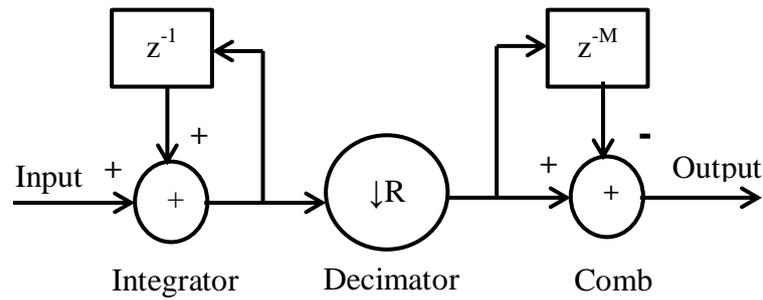


Fig 2.12 Schematic diagram of CIC decimating filter

The output of the DDC consists of 2 components, namely the Inphase (I) and Quadrature (Q) forms of the signal. These are multiplied with a complex weight  $W = w_r + jw_i$  [Fig 2.10].

The beamformer output is obtained as

$$Y = W^H X \quad (2.15)$$

Here X represents the digital down converter output.

$$X = I + jQ \quad (2.16)$$

$$Y = W^H X = (w_r + jw_i)^H (I + jQ) = (w_r I + w_i Q) + j(w_r Q - w_i I) \quad (2.17)$$

### 2.9.1 Xilinx System Generator Model of R-1D-MRT CDBF

Modeling of the R-1D-MRT CDBF is described in this section. For FPGA implementation of the R-1D-MRT CDBF, a model of the beamformer is created using Xilinx System Generator. A model for processes which occur repeatedly can be put into a subsystem, thus saving space and ensuring compactness of the final model. Fig 2.13(a) shows the complete model of the R-1D-MRT CDBF. It includes 2 subsystems - the CIC Decimator subsystem and the weight multiplication subsystem. The signals representing the array inputs are generated, converted to

digital form using ADCs and transformed using R-1D-MRT algorithm. These signals are multiplied with the Cos and Sin outputs of the DDS, passed through the integrator, decimator and comb sections to give the inphase (I) and quadrature (Q) output signals. Since this process has to be repeated with the signals from all the array elements, the blocks for this process are included in the CIC Decimator subsystem shown in Fig 2.13(b). I and Q signals for each array element are then multiplied with the corresponding complex weights using the weight multiplication subsystem. The internal representation of the weight multiplication subsystem is shown in Fig 2.13(c). The real and imaginary components obtained after multiplication with the complex weights are summed up separately. The output signals of various blocks can be viewed with a scope and can also be stored in workspaces for further analysis. The resource estimator block provides details of the FPGA resources used by the model.

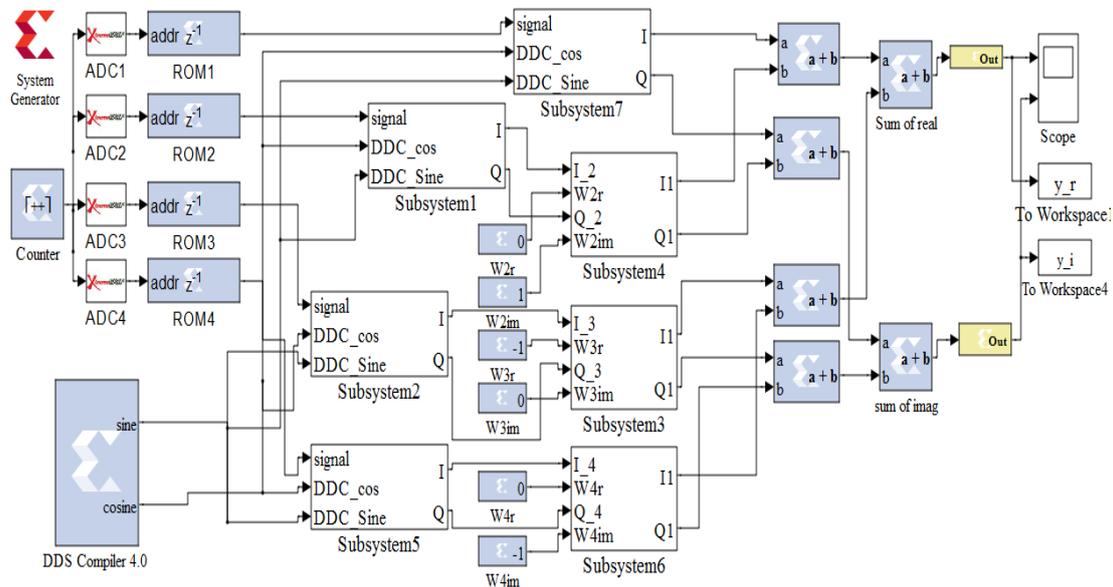


Fig 2.13(a) XSG model of R-1D-MRT CDBF  
 $[N_s = 4, \text{Source angle } 30^\circ, N = 256, \text{SNR} = 30 \text{ dB}]$

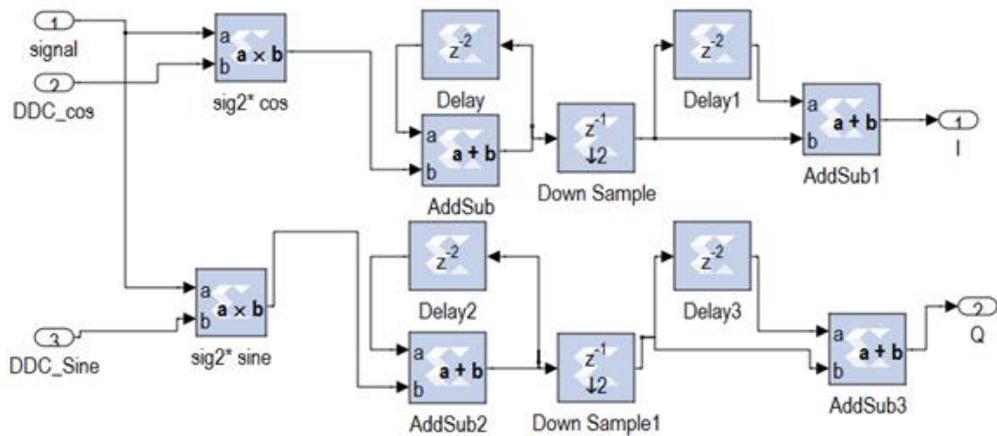


Fig 2.13(b) CIC Decimation filter subsystem of R-1D-MRT CDBF  
 [ $N_s = 4$ , Source angle  $30^\circ$ ,  $N = 256$ , SNR = 30 dB]

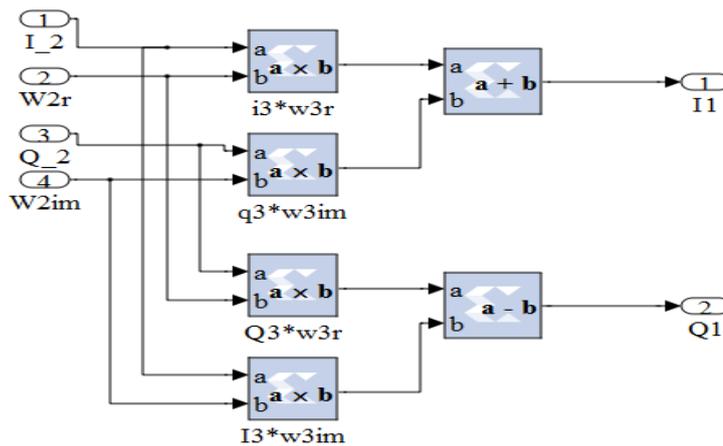


Fig 2.13(c) Weight multiplication subsystem of R-1D-MRT CDBF  
 [ $N_s = 4$ , Source angle  $30^\circ$ ,  $N = 256$ , SNR = 30 dB]

The data stored in the workspaces can be analyzed using MATLAB<sup>®</sup> commands.

The simulation results obtained after execution of the Xilinx System Generator model of R-1D-MRT CDBF is presented in the following section.

### 2.9.2 Beam Pattern and Beamformer Output of XSG Model of R-1D-MRT CDBF

The Xilinx System Generator model of R-1D-MRT CDBF described in the previous section is simulated for various look angles for a source located at 30 degrees. The beamformer output obtained for look angles of 30°, 60° and -80° are shown in Fig 2.14.

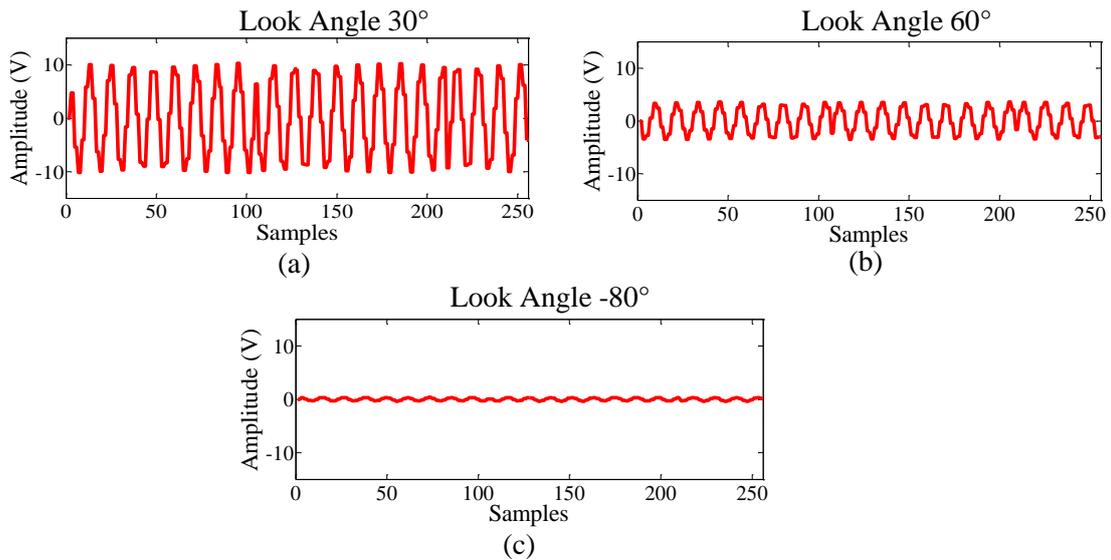


Fig 2.14 Beamformer output of XSG model of R-1D-MRT CDBF for various look angles [ $N_s = 4$ , Source angle 30°,  $N = 256$ , SNR = 30 dB]

For a source at 30 degrees, the beamformer output is high for a look direction of 30° as characterized in Fig 2.14(a). Fig 2.14(b) and Fig 2.14(c) indicate that the beamformer output reduces when the look direction is different from the source direction.

The Xilinx System Generator model of the R-1D-MRT conventional digital beamformer is also simulated for varying source locations. The simulation results show the peak of the normalized beam pattern at the desired angles as illustrated in Fig 2.15.

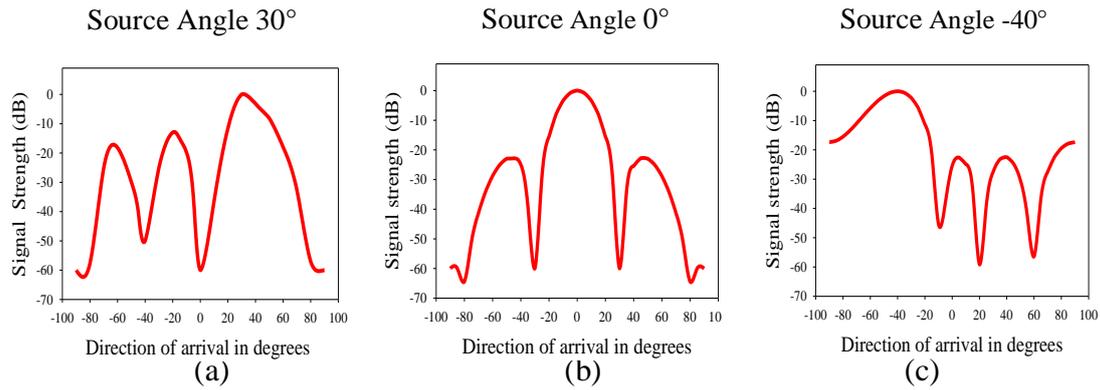


Fig 2.15 Normalized beam pattern of XSG model of R-1D-MRT CDBF for varying source locations [ $N_s = 4$ ,  $N = 256$ ,  $\text{SNR} = 30$  dB]

The normalized beam patterns for XSG simulation exhibit peaks in the radiation pattern at the desired angles of 30°, 0° and -40° as depicted in Fig 2.15(a), 2.15(b) and 2.15(c) respectively.

Following the simulations performed with XSG, the model is then synthesized and built into the Spartan 3AN FPGA on Altium NB3000 using the Altium Designer® software. The normalized beam pattern obtained for a source at 30° is shown in Fig 2.16.

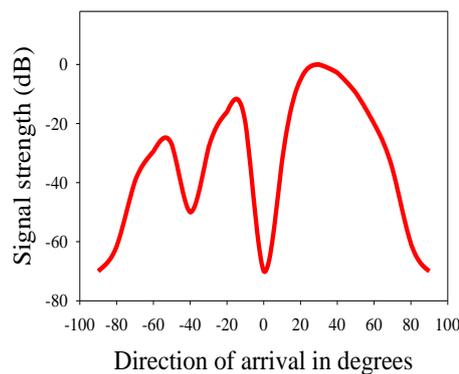


Fig 2.16 Normalized beam pattern of FPGA implementation of R-1D-MRT CDBF [ $N_s = 4$ , Source angle = 30°,  $N = 256$ ,  $\text{SNR} = 30$  dB]

The normalized beam pattern for FPGA implementation of R-1D-MRT CDBF on Altium NB3000 also has peak signal strength in the direction of the source at  $30^\circ$ . The resources utilized by the R-1D-MRT CDBF and CDBF are compared in the next section.

### 2.9.3 FPGA Resource Utilization of R-1D-MRT CDBF

The utilization of FPGA resources by CDBF and R-1D-MRT CDBF is shown in Table 2.7.

Table 2.7 Comparison of FPGA resources of CDBF and R-1D-MRT CDBF [ $N_s = 4$ , Source angle =  $30^\circ$ , SNR = 30 dB]

Resources	N = 256		N = 1024	
	CDBF	R-1D-MRT CDBF	CDBF	R-1D-MRT CDBF
Slices	842	841	843	843
Flip Flops	392	391	394	393
Block RAM	4	4	14	7
Look Up Tables	1383	1381	1385	1384
Input Output blocks	68	68	68	68
Multipliers	20	20	20	20
Tristate Buffers	0	0	0	0

The resources consumed by R-1D-MRT CDBF are less compared to CBF. The efficiency of the FPGA implementation of R-1D-MRT CBF is thus validated. The hardware models of other R-1D-MRT based adaptive beamformers show a marked reduction in FPGA resource utilization, as detailed in the forthcoming chapters. A summary of the discussions of this chapter is presented in the next section.

## 2.10 CHAPTER SUMMARY

The development of the proposed computationally efficient R-1D-MRT algorithm for beamforming is presented in this chapter. The application of the R-1D-MRT algorithm to the array input reduces the size of the data required to be processed by the subsequent sections of the beamformer. The resultant reduction in computational complexity is shown in the initial sections. In the subsequent sections, the effectiveness of the algorithm for beamforming is validated on R-1D-MRT CBF. The effective reduction in computation time and complexity while retaining the desired beam patterns is presented. A performance analysis of the R-1D-MRT CBF in various communications channels using distance metrics and scatterplots is introduced to further confirm the competence of the algorithm. The hardware validation by creating Xilinx System Generator model of the R-1D-MRT CDBF and its FPGA implementation is demonstrated in the final section. As the R-1D-MRT CBF lacks the ability to null interfering signals, the subsequent chapters validate the R-1D-MRT on adaptive beamformers which are more suitable for interference prone environments. The next chapter evaluates the R-1D-MRT Minimum Variance Distortionless Response beamformer and R-1D-MRT Linearly Constrained Minimum Variance beamformer.

## CHAPTER 3

### R-1D-MRT MVDR AND LCMV BEAMFORMERS

<b>CONTENTS</b>	
☆ <b>R-1D-MRT MVDR beamformer</b>	☆ <b>R-1D-MRT LCMV beamformer</b>
◆ Beam pattern and beamformer output	◆ Beam pattern and beamformer output
◆ Computational complexity	◆ Computational complexity
◆ Computation time	◆ Computation time
◆ Performance in various communication channels	◆ Performance in various communication channels
☆ <b>FPGA implementation of R-1D-MRT MVDR beamformer</b>	☆ <b>FPGA implementation of R-1D-MRT LCMV beamformer</b>
◆ Xilinx System Generator model	◆ Xilinx System Generator model
◆ Beam pattern and beamformer output	◆ Beam pattern and beamformer output
◆ FPGA resource utilization	◆ FPGA resource utilization

The previous chapter discussed the R-1D-MRT conventional beamformers. Conventional beamformers enhance the gain of the beamformer in the desired direction proportional to the number of antennas/sensors ( $N_s$ ) but their interference rejection capability is very poor. In environments where interfering signals are present, the beamformer has to protect the target signal, while simultaneously cancelling the interference so that the output signal to noise ratio is maximized. Unlike the conventional beamformers, adaptive beamformers are capable of shaping their beam pattern to maximize radiation in the desired direction while nulling the interfering signals. One method of adaptive beamforming is to minimize the variance of the beamformer output subject to a constraint on the gain of the desired signal. Adaptive beamformers using this method are Minimum Variance Distortionless Response (MVDR) beamformer and Linearly Constrained Minimum Variance (LCMV) beamformer. MVDR beamformer passes the desired signal with unity gain constraint while the LCMV beamformer imposes a non-unity gain constraint on the

desired signal. In this chapter, the validation and FPGA implementation of R-1D-MRT MVDR and R-1D-MRT LCMV beamformers are presented.

### 3.1 R-1D-MRT MVDR BEAMFORMER

In the MVDR beamformer the weights are chosen to maintain a distortionless response with unity gain for the signals in the desired direction  $\theta_d$  while minimizing the output power of the interfering signals and noise (Raghunath and Reddy, 1992; Liu et al., 2011).

The output power of the MVDR beamformer is expressed as:

$$P = [E(\mathbf{Y}^2)] = E[\mathbf{W}^H \mathbf{X} \mathbf{X}^H \mathbf{W}] = \mathbf{W}^H E[\mathbf{X} \mathbf{X}^H] \mathbf{W} = \mathbf{W}^H \mathbf{C} \mathbf{W} \quad (3.1)$$

where  $\mathbf{X}$  is the array input signal,  $\mathbf{W}$  is the weight vector and  $H$  is Hermitian transpose,  $\mathbf{C} = E[\mathbf{X} \mathbf{X}^H]$  is the correlation matrix of the received signal and  $E(\cdot)$  is the expectation operator.

The MVDR adaptive algorithm is represented as:

$$\min_{\mathbf{W}} \{\mathbf{W}^H \mathbf{C} \mathbf{W}\} \text{ subject to } \mathbf{W}^H \mathbf{a}(\theta_d) = 1 \quad (3.2)$$

The minimization of the total output noise, while maintaining the output signal constant, is equivalent to maximizing the output SINR direction (Kiong et al., 2014).

The steering vector for the desired signal is given by

$$\mathbf{a}(\theta_d) = \left[ 1 \quad e^{j\frac{2\pi}{\lambda}d(\sin \theta_d)} \quad \dots \quad \dots \quad e^{j\frac{2\pi}{\lambda}(N_s-1)d(\sin \theta_d)} \right]^T \quad (3.3)$$

where  $\lambda$  is the wavelength of the signal,  $d$  is the spacing between the array elements and  $N_s$  is the number of antennas/sensors. The optimized weight vector of the MVDR beamformer is expressed as

$$W_{\text{MVDR}} = [w_1 \quad w_2 \quad \cdots \quad \cdots \quad w_{N_s}]^T \quad (3.4)$$

$$W_{\text{MVDR}} = \frac{C^{-1}a(\theta_d)}{a^H(\theta_d)C^{-1}a(\theta_d)} \quad (3.5)$$

In MVDR beamformer, the signals received at the array are multiplied with the weights computed using the MVDR algorithm and the weighted signals are then summed up. The schematic diagram of a conventional MVDR beamformer is shown in Fig 3.1(a).

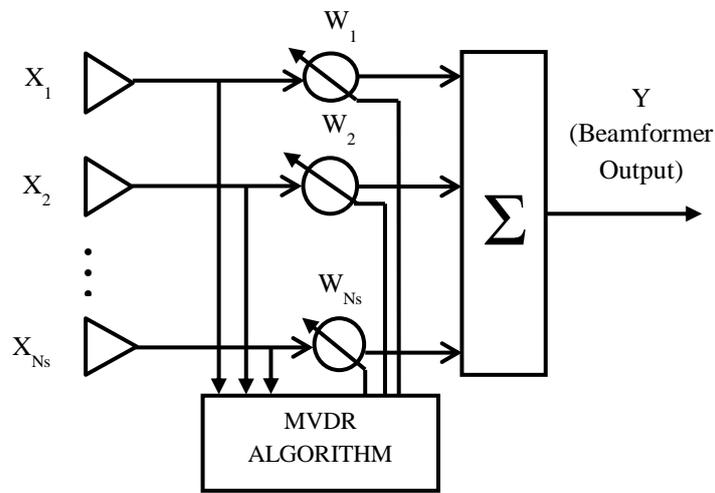


Fig 3.1(a) Schematic diagram of conventional MVDR beamformer

In the R-1D-MRT MVDR beamformer, the input data at the array is transformed by applying R-1D-MRT. The transformed signals and the weights, adapted using the MVDR algorithm, are multiplied to give the weighted signal. These are finally summed up to give the beamformer output. The schematic diagram of the R-1D-MRT MVDR beamformer is shown in Fig 3.1(b).

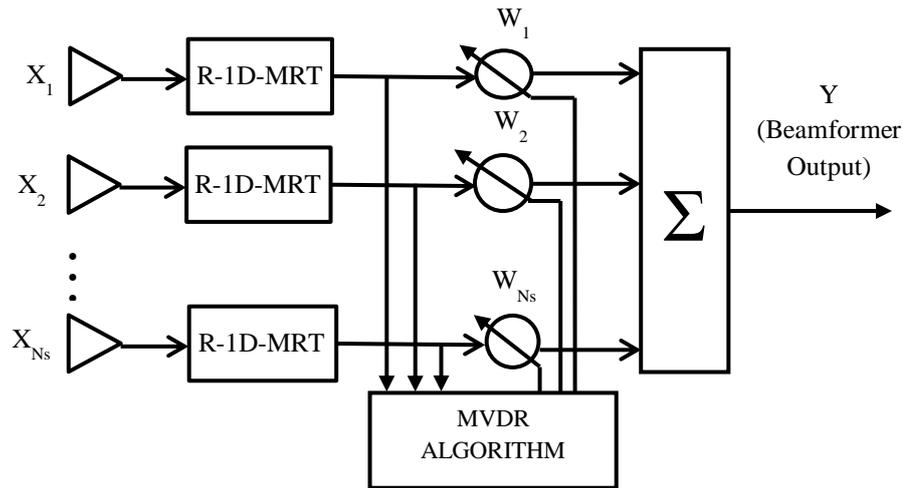


Fig 3.1(b) Schematic diagram of R-1D-MRT MVDR beamformer

The validation of R-1D-MRT MVDR using MATLAB<sup>®</sup> simulation is presented in the following sections.

### 3.1.1 Beam Pattern and Beamformer Output of R-1D-MRT MVDR Beamformer

The beam pattern of the R-1D-MRT MVDR beamformer must have the main lobe directed towards the direction of the desired signal and nulls directed towards interfering signals. The normalized beam patterns of R-1D-MRT MVDR beamformer for varying number of antennas/sensors ( $N_s$ ), illustrated in Fig 3.2, show peak power in the desired direction and nulls in the direction of interfering signals. The results are at par with results obtained with MVDR beamformer using raw data.

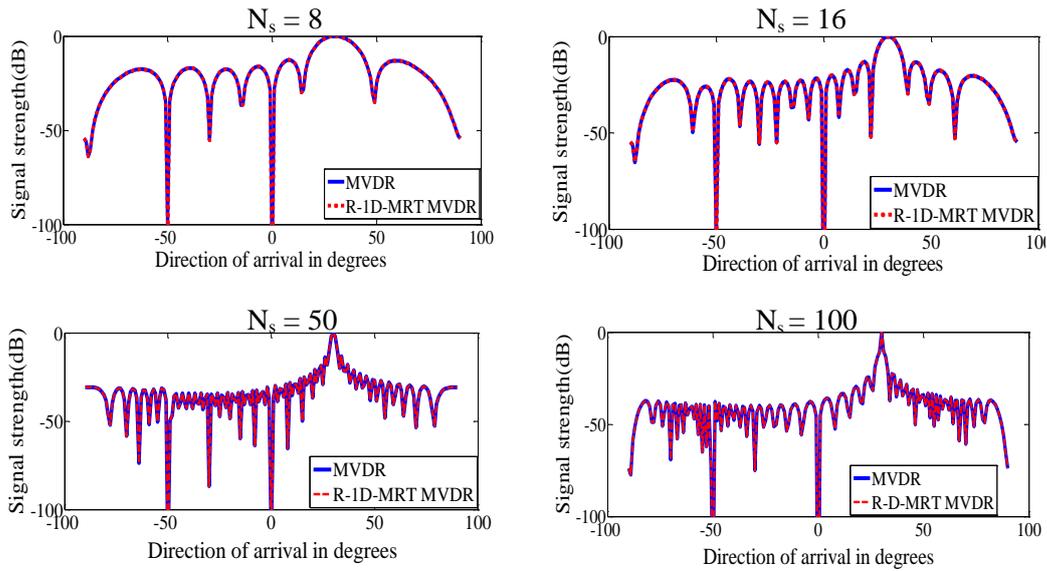


Fig 3.2 Normalized beam pattern of R-1D-MRT MVDR BF for varying number of antennas/sensors ( $N_s$ ) [Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB]

The normalized beam patterns simulated for varying directions of desired and interfering signals are illustrated in Fig 3.3. Here also peaks are obtained at the desired directions along with strong nulls in the interfering directions.

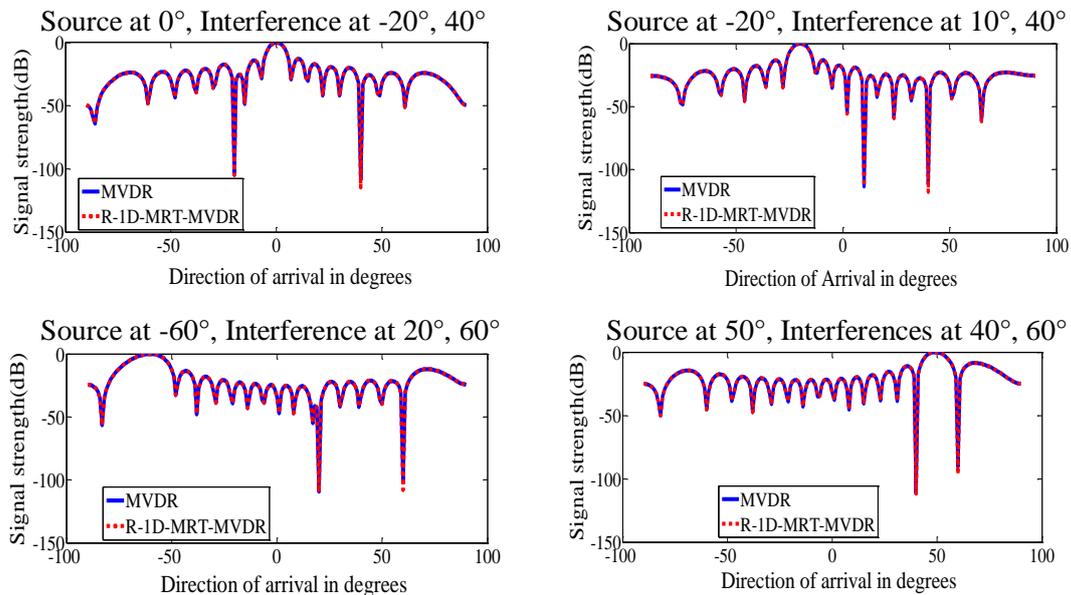


Fig 3.3 Normalized beam pattern of R-1D-MRT MVDR BF for varying source and interference locations [ $N_s = 16$ ,  $N = 512$ , SINR = 30 dB]

The output of the R-1D-MRT MVDR beamformer for varying look angles, with source at  $30^\circ$  and interferences at  $0^\circ$  and  $-50^\circ$  is shown in Fig 3.4.

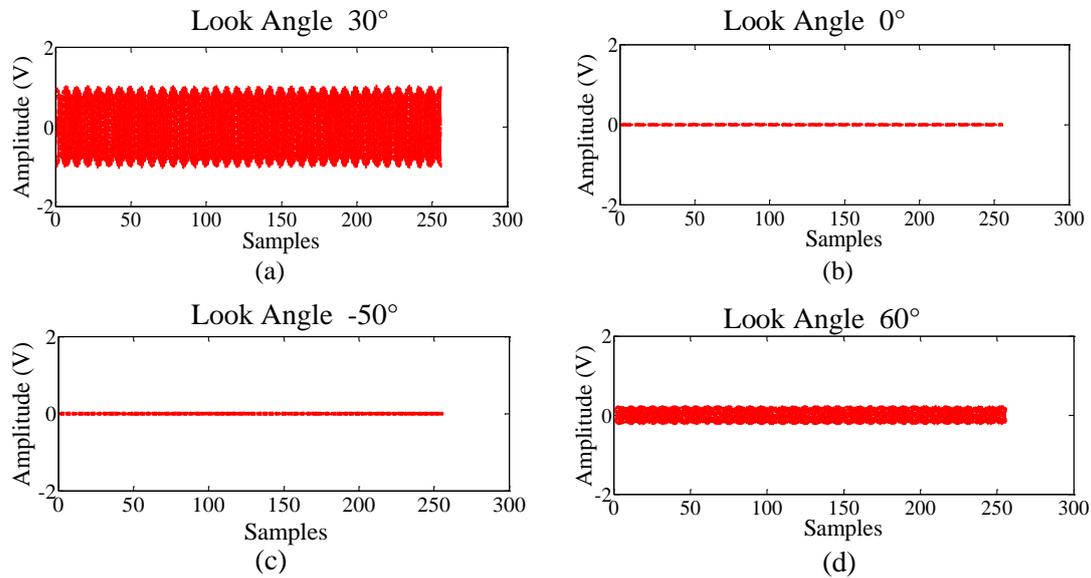


Fig 3.4 Beamformer output of R-1D-MRT MVDR BF for varying look angles [ $N_s = 8$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB]

The beamformer output of R-1D-MRT MVDR beamformer has unit amplitude when the look angle is same as the source angle of  $30^\circ$  as shown in Fig 3.4(a) and zero amplitude when look angle is equal to the interfering angles of  $0^\circ$  and  $-50^\circ$  as shown in Fig 3.4(b) and (c) respectively. The beamformer output shows a value greater than zero for a look angle of  $60^\circ$  as observed in Fig 3.4(d). The constraint of unity gain in the desired direction and nulls in interfering directions is hence satisfied. On the other hand, the output of conventional beamformer in the desired direction is proportional to the number of antennas/sensors ( $N_s$ ) (Fig 2.5(a)). The output of the R-1D-MRT MVDR beamformer for varying number of antennas/sensors is illustrated in Fig 3.5.

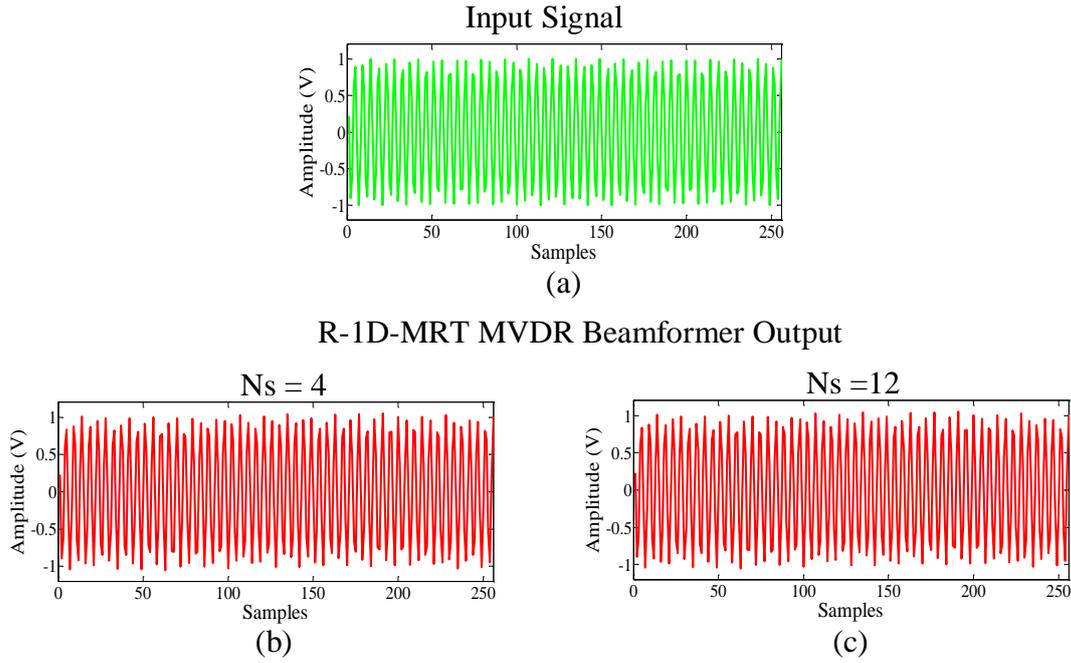


Fig 3.5 Beamformer output of R-1D-MRT MVDR BF for varying number of sensors ( $N_s$ ) [Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB]

The signal fed in at the beamformer input is shown in Fig 3.5(a). The output of R-1D-MRT MVDR beamformer in the desired direction for 4 and 12 sensors, illustrated in Fig 3.5(b) and Fig 3.5(c) respectively, prove that the unity gain constraint is satisfied irrespective of the number of antennas/sensors ( $N_s$ ) in the array.

### 3.1.2 Computational Complexity of R-1D-MRT MVDR Beamformer

This section presents a comparison of the computations required by the MVDR and R-1D-MRT MVDR beamformers. An input of  $N$  complex numbers requires  $2(N-1)$  real additions for transforming to R-1D-MRT form, resulting in  $N/2$  complex numbers [Section 2.3]. These  $N/2$  complex numbers are multiplied with a complex weight in each antenna/sensor path [Fig 3.1(b)]. Hence in each antenna/sensor path,  $2N-4$  real multiplications and  $N-2$  real additions are required with the R-1D-MRT transformed

data instead of  $4N-4$  real multiplications and  $2N-2$  real additions required with the original data. In the summation stage of the beamformer, the number of addition operations for each antenna/sensor input is reduced to  $N-2$  real additions using the R-1D-MRT transformed data instead of  $2N-2$  real additions required with the original data. For calculating the weights, the covariance matrix  $C = E(XX^H)$  has to be computed. This requires  $N_s^2 N$  complex multiplications and  $N_s^2 (N-1)$  complex additions while the R-1D-MRT MVDR beamformer requires  $N_s^2 N/2$  complex multiplications and  $N_s^2 \times ((N/2)-1)$  complex additions. Each complex multiplication requires 4 real multiplications and 2 real additions while each complex addition requires 2 real addition operations.

Thus for calculation of the covariance, the MVDR beamformer requires  $4N_s^2 (N-1)$  real multiplications and  $2N_s^2(4N - 6)$  real addition operations while R-1D-MRT MVDR beamformer requires  $2N_s^2(N-2)$  real multiplication and  $2N_s^2 (N - 2)$  real addition operations. After computing the covariance matrix, the weight calculation process requires the same number of computations for both MVDR and R-1D-MRT MVDR beamformers and therefore it is not considered in the comparison study.

Thus the total number of operations required by R-1D-MRT MVDR beamformer is  $(2N_s^2(N-2) + 2N_s(N-2))$  real multiplications and  $(2N_s^2(N-2) + 2N_s(N-3))$  real additions while the MVDR beamformer requires  $(4N_s^2(N-1) + 4N_s(N-1))$  real multiplications and  $2N_s^2(2N-3) + 4N_s(N-1)$  real additions. A comparison of the number of multiplication and addition operations for MVDR and R-1D-MRT MVDR beamformers using arrays of 4 antennas/sensors ( $N_s$ ) and 8 antennas/sensors ( $N_s$ ) for varying data sizes ( $N$ ) is presented in Table 3.1.

Table 3.1 Comparison of computational complexity of MVDR BF and R-1D-MRT MVDR BF

N (Data size)	$N_s = 4$				$N_s = 8$			
	Number of real multiplications		Number of real additions		Number of real multiplications		Number of real additions	
	MVDR BF	R-1D-MRT MVDR BF	MVDR BF	R-1D-MRT MVDR BF	MVDR BF	R-1D-MRT MVDR BF	MVDR BF	R-1D-MRT MVDR BF
8192	655280	327600	655248	327592	2359008	1179360	2358880	1179344
4096	327600	163760	327568	163752	1179360	589536	1179232	589520
2048	163760	81840	163728	81832	589536	294624	589408	294608
1024	81840	40880	81808	40872	294624	147168	294496	147152
512	40880	20400	40848	20392	147168	73440	147040	73424
256	20400	10160	20368	10152	73440	36576	73312	36560
128	10160	5040	10128	5032	36576	18144	36448	18128
64	5040	2480	5008	2472	18144	8928	18016	8912
32	2480	1200	2448	1192	8928	4320	8800	4304
16	1200	560	1168	552	4320	2016	4192	2000

The computational load on the R-1D-MRT MVDR beamformer is significantly lower than that of the MVDR beamformer. The considerable reduction in the number of multiplication operations can also reduce the computation time of the R-1D-MRT MVDR beamformer substantially. This is discussed in the next section.

### 3.1.3 Computation Time of R-1D-MRT MVDR Beamformer

The efficiency of the R-1D-MRT MVDR beamformer in terms of computational load was demonstrated in the previous section. In this section the computation time of R-1D-MRT MVDR beamformer and MVDR beamformer are compared for varying number of antennas/sensors ( $N_s$ ) and data sizes (N). The results obtained are presented in Table 3.2 and 3.3.

Table 3.2 Comparison of computation time of MVDR BF and R-1D-MRT MVDR BF (seconds) [Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB]

N (Data size)	$N_s = 4$			$N_s = 16$		
	MVDR BF	R-1D-MRT MVDR BF	% Reduction	MVDR BF	R-1D-MRT MVDR BF	% Reduction
8192	0.0040	0.0026	35	0.0099	0.0066	33
4096	0.0023	0.0019	17	0.0051	0.0038	25
2048	0.0017	0.0012	29	0.0034	0.0023	32
1024	0.0012	0.0011	9	0.0022	0.0017	22
512	0.0008	0.0005	37	0.0016	0.0011	25
256	0.0007	0.0007	0	0.0012	0.0010	16
128	0.0007	0.0007	0	0.0009	0.0009	0
64	0.0006	0.0007	-16	0.0009	0.0009	0
32	0.0005	0.0006	-20	0.0010	0.0008	20
16	0.0005	0.0006	-20	0.0008	0.0008	0

Table 3.3 Comparison of computation time of MVDR BF and R-1D-MRT MVDR BF (seconds) [Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB]

N (Data size)	$N_s = 50$			$N_s = 100$		
	MVDR BF	R-1D-MRT MVDR BF	% Reduction	MVDR BF	R-1D-MRT MVDR BF	% Reduction
8192	0.0352	0.0196	44	0.0927	0.0482	48
4096	0.0183	0.0120	34	0.0494	0.0258	47
2048	0.0098	0.0074	24	0.0244	0.0157	35
1024	0.0059	0.0043	27	0.0149	0.0105	30
512	0.0039	0.0032	17	0.0084	0.0079	7
256	0.0028	0.0025	8	0.0059	0.0058	2
128	0.0025	0.0021	16	0.0049	0.0046	6
64	0.0019	0.0018	5	0.0042	0.0041	2
32	0.0017	0.0016	5	0.0038	0.0035	7
16	0.0016	0.0014	12	0.0037	0.0034	8

The simulations were performed for a source at 30 degrees, interference angles at 0 and -50 degrees and SINR of 30 dB with a computer using Intel Core i5 CPU working at 2.20 GHz and RAM of 4GB. Lesser computation time is required by the R-1D-MRT MVDR beamformer compared to MVDR beamformer. The reduced computational load and computation time of the R-1D-MRT MVDR beamformer in comparison to MVDR beamformer, validates the computational efficiency of the R-1D-MRT algorithm for MVDR beamforming.

### 3.1.4 Performance of R-1D-MRT MVDR Beamformer in Various Communication Channels

This section investigates the performance of the R-1D-MRT MVDR beamformer in the AWGN, Rayleigh and Rician channels for varying noise plus interference levels. The Euclidian distance metric is computed between the original signal and the beamformer output of the R-1D-MRT MVDR and MVDR beamformers. The comparison is presented in Table 3.4.

Table 3.4 Comparison of Euclidian distance parameters of MVDR BF and R-1D-MRT MVDR BF for various communication channels for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ]

SINR	AWGN		Rayleigh		Rician	
	MVDR BF	R-1D-MRT MVDR BF	MVDR BF	R-1D-MRT MVDR BF	MVDR BF	R-1D-MRT MVDR BF
10 dB	1.2416	0.9871	2.8741	2.8275	1.0977	0.7951
20 dB	0.3729	0.3165	0.9640	0.9597	0.7311	0.6331
30 dB	0.0596	0.0302	0.3269	0.3265	0.4960	0.4952

The Euclidian distance metric for the R-1D-MRT MVDR beamformer is lower than that of the MVDR beamformer for the communication channels considered, indicating the closeness of the R-1D-MRT MVDR beamformer output signal to the original signal.

A visual comparison of the performance of R-1D-MRT MVDR and MVDR beamformers in various communication channels is also presented using scatterplots. Fig 3.6 presents the scatterplots for the original signal, signal with noise and interference, beamformer output of MVDR BF and R-1D-MRT MVDR BF in AWGN channel for varying SINR values while Fig 3.7- 3.8 represent the same for Rayleigh and Rician fading channels.

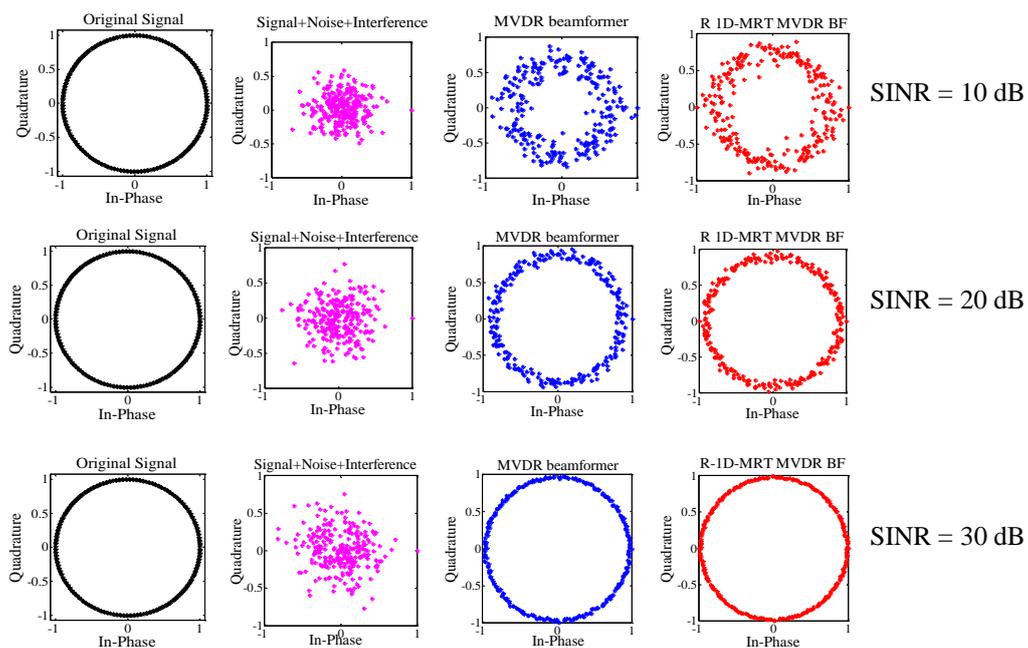


Fig 3.6 Scatterplots of R-1D-MRT MVDR BF in AWGN channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ]

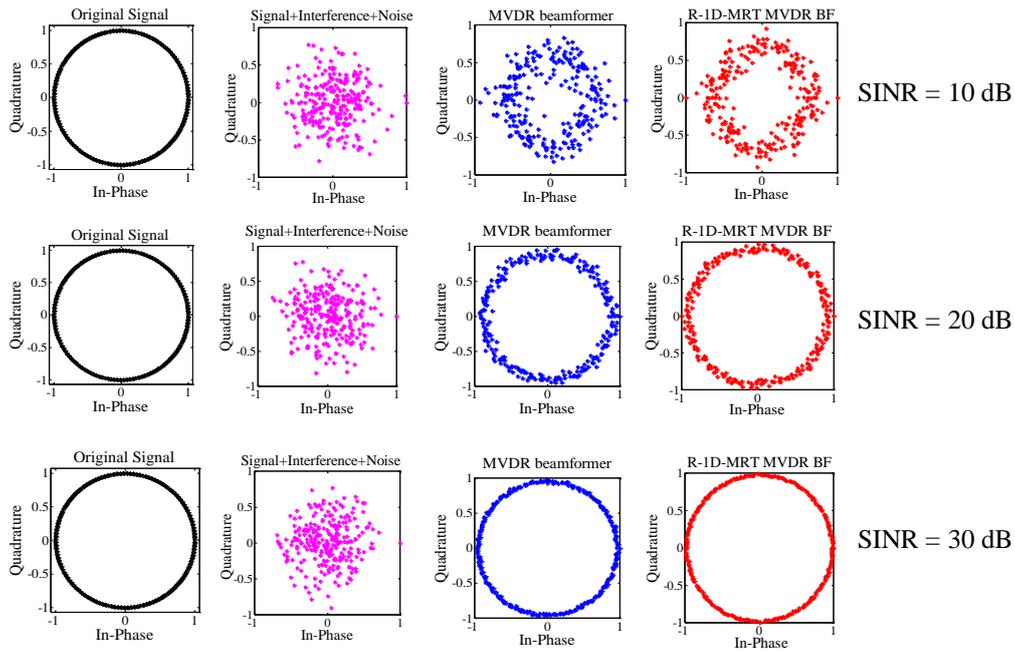


Fig 3.7 Scatterplots of R-1D-MRT MVDR BF in Rayleigh channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ]

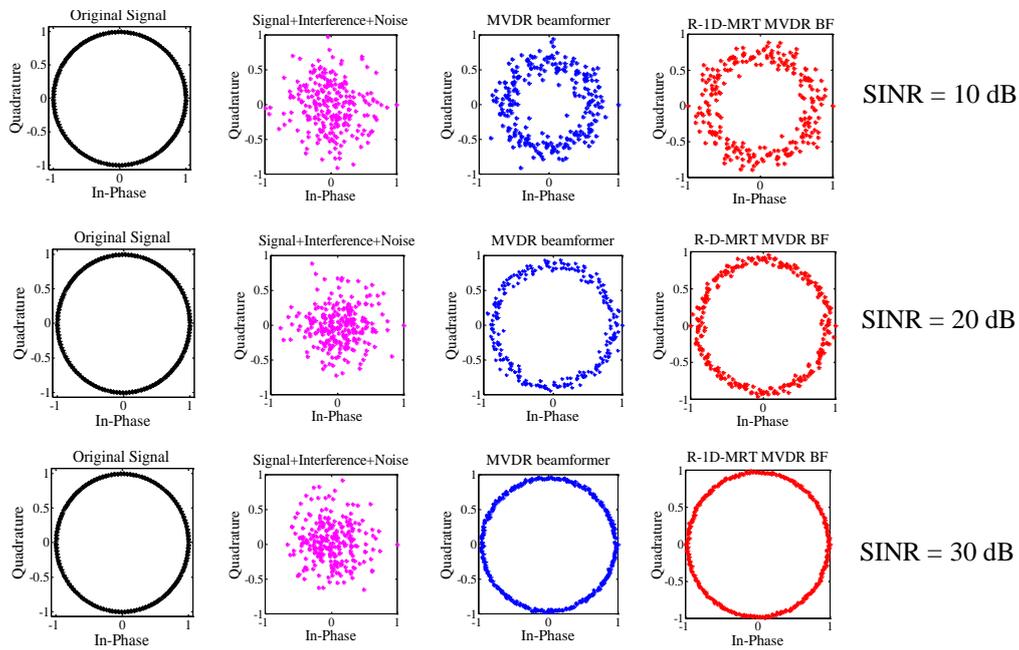


Fig 3.8 Scatterplots of R-1D-MRT MVDR BF in Rician channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ]

The scatterplots of Fig 3.6 – Fig 3.8 show that for AWGN, Rayleigh and Rician channels the similarity between R-1D-MRT MVDR beamformer output and the original signal is more compared to the similarity level of MVDR beamformer output with the original signal. This is more pronounced at low SINR values. The discussions so far validate the efficiency of the R-1D-MRT MVDR beamformer. In the next section, the FPGA implementation of the R-1D-MRT MVDR beamformer is presented.

### **3.1.5 FPGA Implementation of R-1D-MRT MVDR Beamformer**

In the previous sections, competence of the R-1D-MRT MVDR beamformer was examined based on computational complexity, computation time, performance in various communication channels and ability to direct the desired signal to the output with unity gain while nulling the interfering signals. For further validation of these results, the hardware implementation of the R-1D-MRT MVDR beamformer using FPGA is considered.

#### **3.1.5.1 Xilinx System Generator model of R-1D-MRT MVDR beamformer**

The Xilinx System Generator (XSG) models are created for the R-1D-MRT MVDR beamformer with a desired source originating at 30 degrees and an interfering signal at -50 degrees. The model for generating the desired and interfering signals is presented in Fig 3.9(a) and the model for phase shifting the signals is shown in Fig 3.9(b).

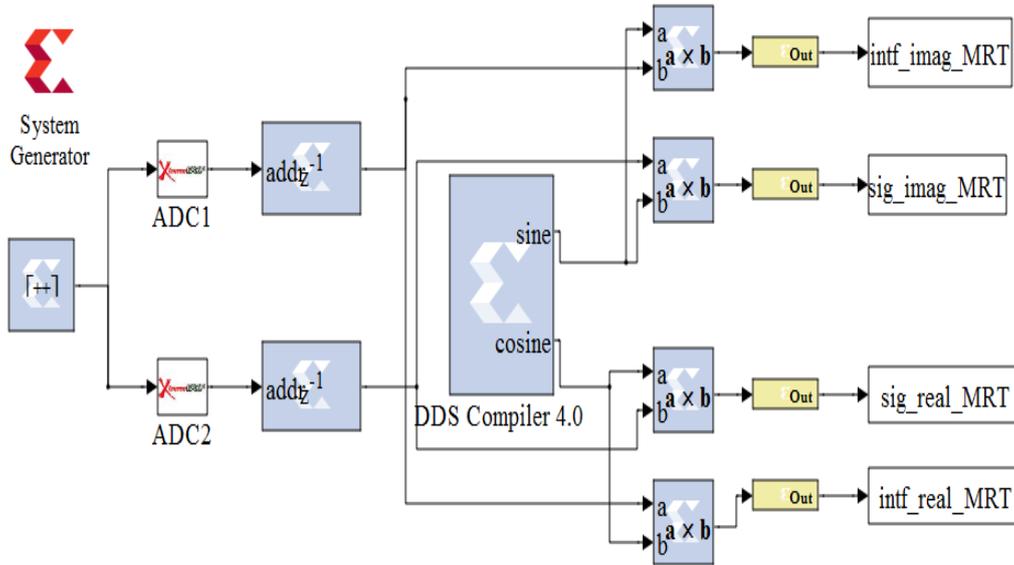


Fig 3.9(a) XSG model for generating desired and interfering signals of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

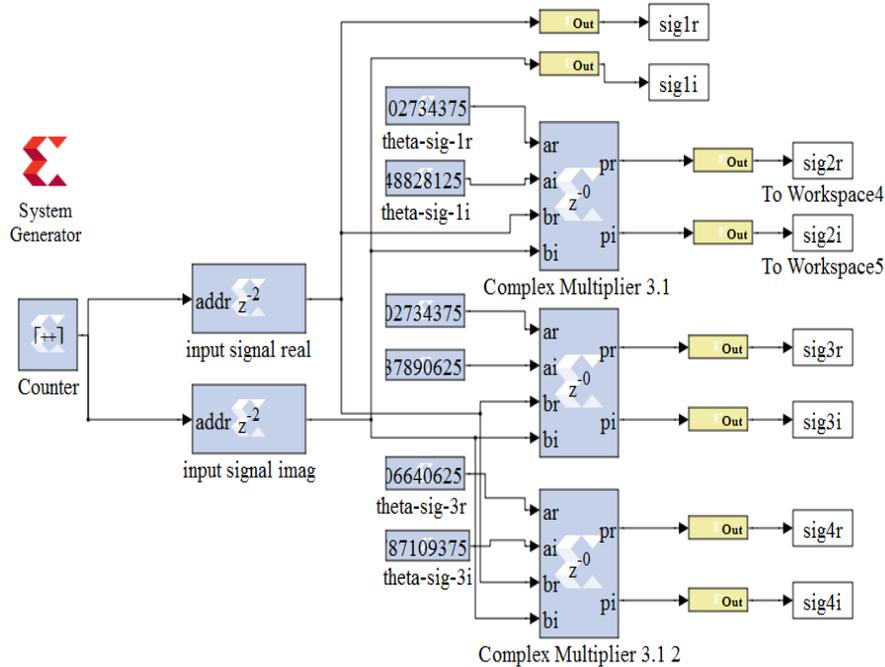


Fig 3.9(b) XSG model for phase shifting signals of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

For estimating the weights, the correlation of the input is required (Eqn 3.1). The XSG model for generating the correlation matrix of the R-1D-MRT transformed input signal is represented in Fig 3.10.

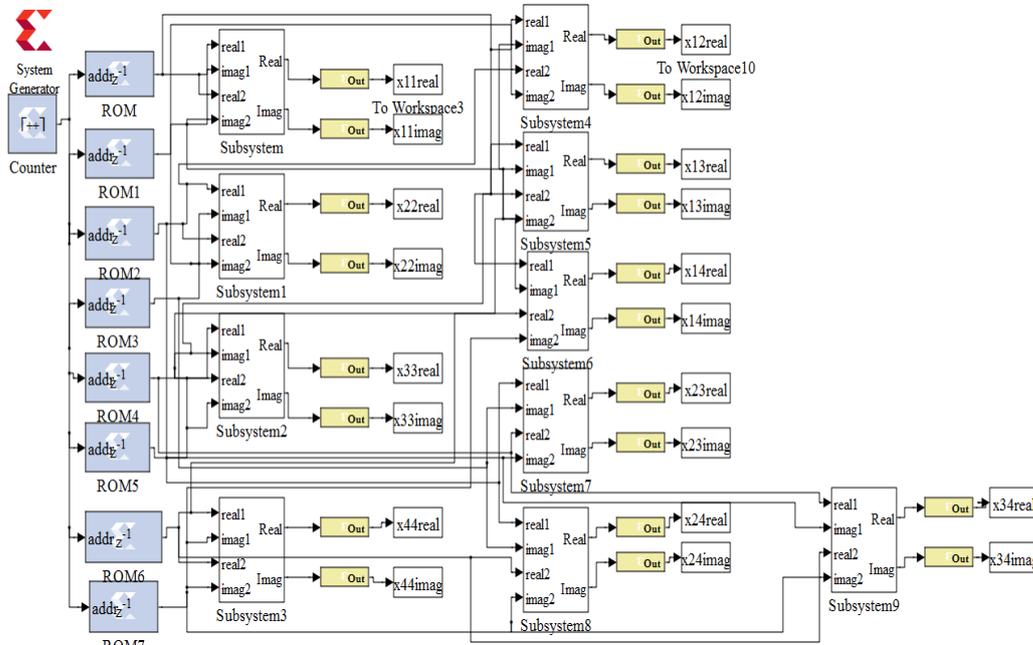


Fig 3.10 XSG model of correlation matrix of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The subsystem block of correlation matrix model is shown in Fig 3.11.

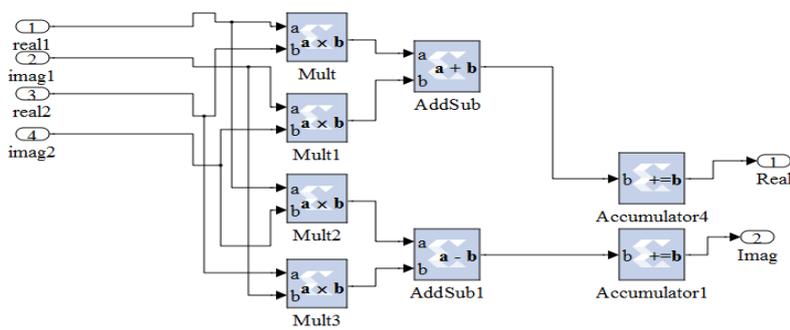


Fig 3.11 Subsystem of correlation matrix model of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The computation of the weights of the MVDR beamformer requires the calculation of the inverse of the correlation matrix. Since the computation of matrix inverse is computationally expensive, QR decomposition is used to compute the inverse of the correlation matrix (Roncero, 2014). The correlation matrix  $C$  of dimension  $[N_s \times N_s]$ , where  $N_s$  is the number of antennas/sensors is given by:

$$C = E[\mathbf{X}^H(n)\mathbf{X}(n)] \quad (3.6)$$

where  $\mathbf{X}$  is the input data vector and  $H$  is the Hermitian transpose. Replacing  $C$  with QR where  $Q$  is an orthogonal matrix and  $R$  is an upper triangular matrix.

$$C^{-1} = R^{-1}Q^H \quad (3.7)$$

Different numerical methods are available for QR decomposition. The Modified Gaussian Elimination method along with back substitution is used in this work for calculating the inverse of the correlation matrix as it is more suitable for implementation in the Altium NB3000 considering the FPGA resources available in it (Roncero, 2014). The Modified Gaussian Elimination method for matrix inversion of the correlation matrix can be performed in the following manner. Row transformations are applied to the correlation matrix, denoted as  $C$ , to get an upper triangular matrix  $R$ . For obtaining  $R$  of a  $4 \times 4$  correlation matrix, the first row of  $C$  is retained as it is and one row transformation is performed on the second row.

$$C2' = C2 - \frac{C2(1)}{C1(1)}C1 \quad (3.8)$$

2 row transformations are applied to the 3<sup>rd</sup> row.

$$C3' = C3 - \frac{C3(2)}{C1(2)}C1 \quad (3.9)$$

$$C3'' = C3' - \frac{C3'(2)}{C1(2)}C2' \quad (3.10)$$

Similarly 3 row transformations are applied on the 4<sup>th</sup> row. The same transformations are applied to the rows of an identity matrix of similar size as the correlation matrix  $C$ . The first row of  $I$  is retained as it is and applying one transformation to the first row of  $I$  of the form

$$I2' = I2 - \frac{C2(1)}{C1(1)} I1 \quad (3.11)$$

Two row transformations have to be applied to the 3<sup>rd</sup> row of  $I$ .

$$I3' = I3 - \frac{C3(2)}{C12} I1 \quad (3.12)$$

$$I3'' = I3' - \frac{C3'(2)}{C1(2)} I2' \quad (3.13)$$

3 row transformations of similar nature are applied to the 4<sup>th</sup> row of  $I$ . The resulting matrix obtained after row transformations on the identity matrix is denoted as  $Q^H$ . The Xilinx System Generator model for transforming a row element is shown in Fig 3.12.

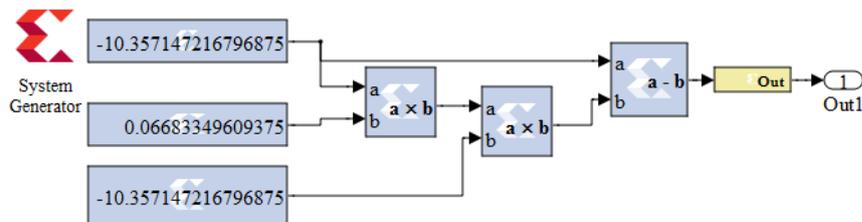


Fig 3.12 Model for row transformation of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The Xilinx System Generator models showing the values of the correlation matrix after applying the first transformation to the 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup> rows are shown in Fig 3.13.

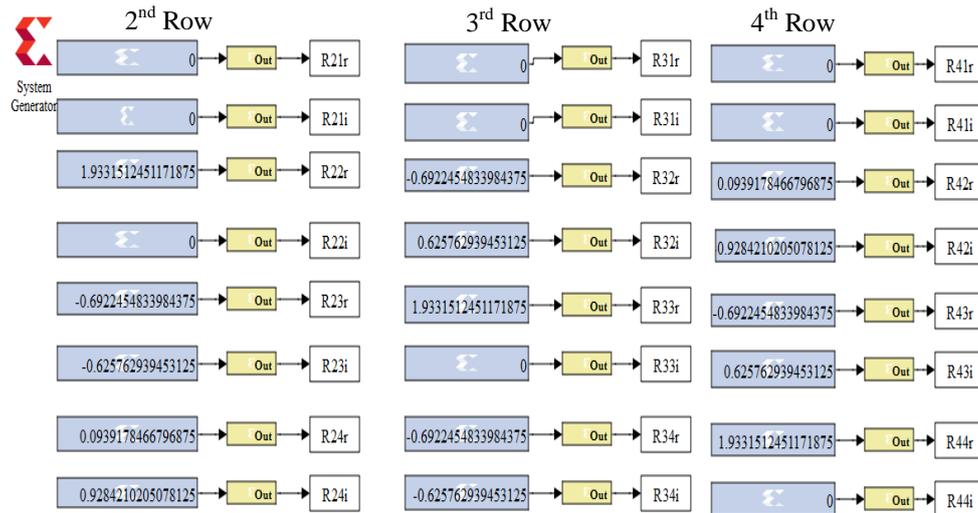


Fig 3.13 Model for first transformation of 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup> rows of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

After the first transformation is applied to any row, the leading element of that row becomes zero. Fig 3.13 shows that after the first transformation the leading elements of the 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup> rows are zero. The real and imaginary values of these elements are represented as R21r, R21i, R31r, R31i, R41r and R41i respectively.

The second row transformations are used to null the second element of the row to which it is applied. Fig 3.14 shows the values after applying the second transformations to the third and fourth rows of the correlation matrix  $C$ .

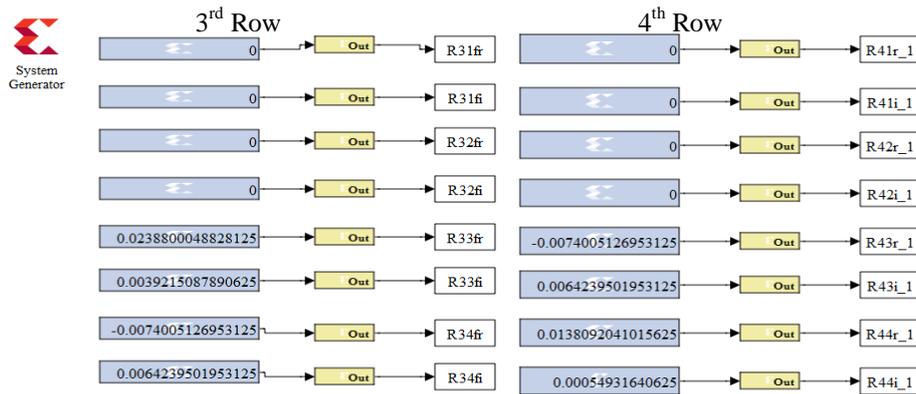


Fig 3.14 Model for second transformation of 3<sup>rd</sup> and 4<sup>th</sup> rows of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

Following the second row transformation, the second elements of the third and fourth row are nullified. It can be verified in Fig 3.14, that after the second transformation, the real and imaginary parts of the leading 2 elements of the 3<sup>rd</sup> and 4<sup>th</sup> rows are zeros. By applying one more row transformation to the fourth row of the correlation matrix, the 3<sup>rd</sup> element of the 4<sup>th</sup> row is nullified. The Xilinx system generator block for the 3<sup>rd</sup> transformation to the 4<sup>th</sup> row is represented in Fig 3.15.

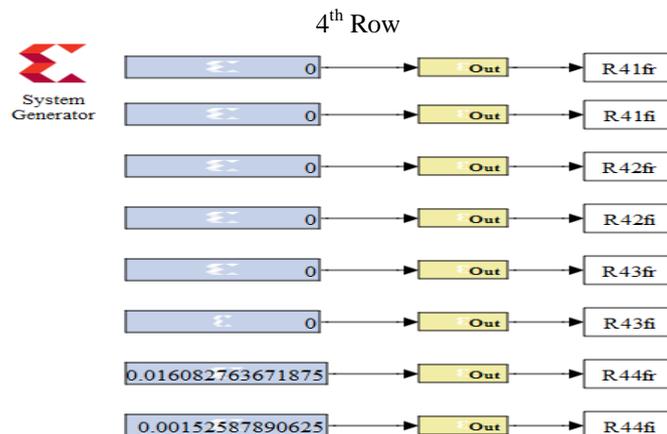


Fig 3.15 Model for third transformation of 4<sup>th</sup> row of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

Fig 3.15 shows the model for last step in converting the 4x4 correlation matrix  $C$  into an upper triangular matrix. It can be observed that after this transformation the first 3 elements of the 4<sup>th</sup> row are zero. The correlation matrix  $C$  has been converted into an upper triangular matrix  $R$ . Similar models are created for transforming an identity matrix  $I$  into the  $Q$  matrix. Applying the same transformation on  $I$  as was done on the correlation matrix  $C$ , a lower triangular matrix  $Q^H$  is obtained.

The upper triangular matrix  $R$  satisfies the relation,

$$R^{-1}R = I \quad (3.14)$$

Using the above relation and back substitution methods, the value of  $R^{-1}$  is computed.

For a 4 x 4 matrix, the back substitution requires 4 similar models, one of which is shown in Fig 3.16.

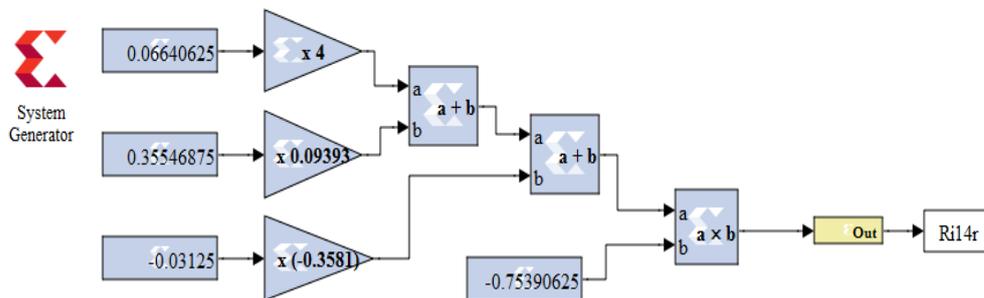


Fig 3.16 Back substitution model of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

Execution of the back substitution models shown in Fig 3.16, generate the  $R^{-1}$  matrix.

Multiplication of this matrix with the  $Q^H$  matrix gives  $C^{-1}$  matrix. Fig 3.17 shows the model created for multiplying these matrices.

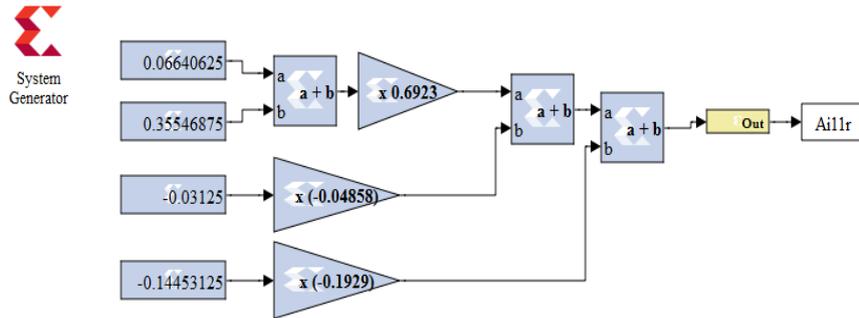


Fig 3.17 Model for multiplication of  $R^{-1}$  and  $Q$  matrices of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

After multiplication of the  $R^{-1}$  and  $Q$  matrices, the inverse of the correlation matrix will be available. The matrix inverse obtained for a 4x4 correlation matrix using the Xilinx System Generator model is shown in Table 3.5 while the matrix inverse for the same correlation matrix calculated using MATLAB<sup>®</sup> is shown in Table 3.6. The results closely agree.

Table 3.5 Matrix inverse of R-1D-MRT MVDR BF using XSG model

0.7500	0.1797+0.1641i	-0.0234-0.2422i	-0.1445+0.1992i
0.1836-0.1641i	0.7500	0.1836+0.1641i	-0.0234-0.2422i
-0.0234+0.2422i	0.1836-0.1641i	0.7578	0.1826+0.1650i
-0.1445-0.1992i	-0.0234+0.2422i	0.1836-0.1641i	0.7539

Table 3.6 Matrix inverse of R-1D-MRT MVDR BF using MATLAB<sup>®</sup>

0.7544	0.1822+0.1647i	-0.0247-0.2444i	-0.1455+0.1978i
0.1836-0.1647i	0.7544	0.1822+0.1647i	-0.0247-0.2444i
-0.0247+0.2444i	0.1822-0.1647i	0.7544	0.1822+0.1647i
-0.1445-0.1978i	-0.0247+0.2444i	0.1822-0.1647i	0.7544

Once the inverse of the correlation matrix is obtained, the weights of the MVDR beamformer are calculated. The Xilinx System generator models for calculating the weights of the R-1D-MRT MVDR beamformer are shown in Fig 3.18 - 3.19.

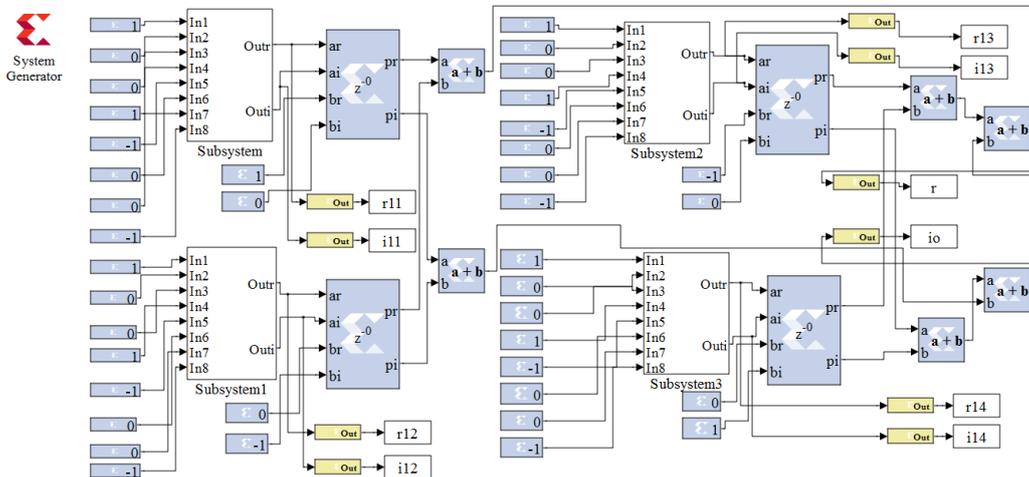


Fig 3.18 Model for calculating the weights of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The subsystem for weight calculation is shown in Fig 3.19.

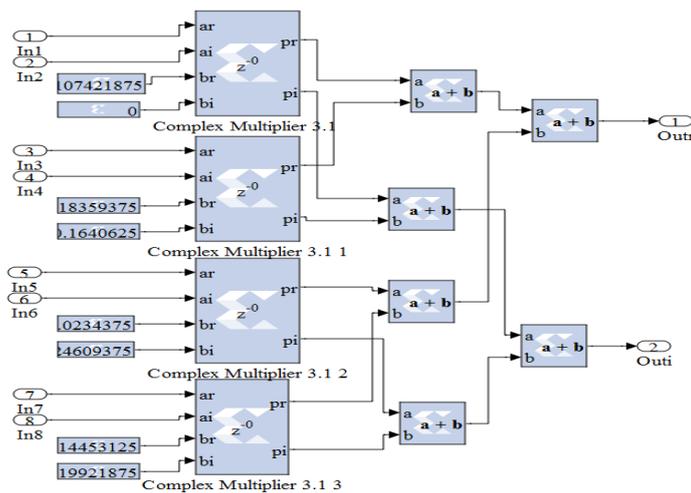


Fig 3.19 Subsystem of model for calculating weights of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The R-1D-MRT transformed signal from the array elements is modified using the weights and summed to give the R-1D-MRT MVDR beamformer output. Fig 3.20 shows the model created for obtaining the beamformer output.

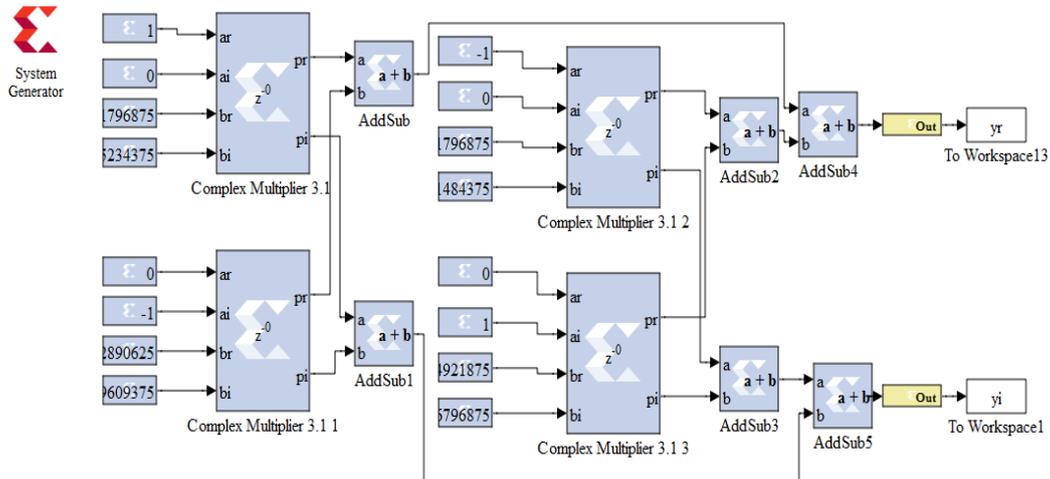


Fig 3.20 Model for obtaining the beamformer output of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The validation of the XSG models of R-1D-MRT MVDR beamformer is discussed in the next section.

### 3.1.5.2 Beam pattern and beamformer output of XSG model of R-1D-MRT MVDR beamformer

In this section, the normalized beam pattern and beamformer output obtained by simulating the XSG model of R-1D-MRT MVDR beamformer is presented. The normalized beam pattern obtained with the Xilinx System Generator for a source at 30 degrees and interference at -50 degrees is shown in Fig 3.21.

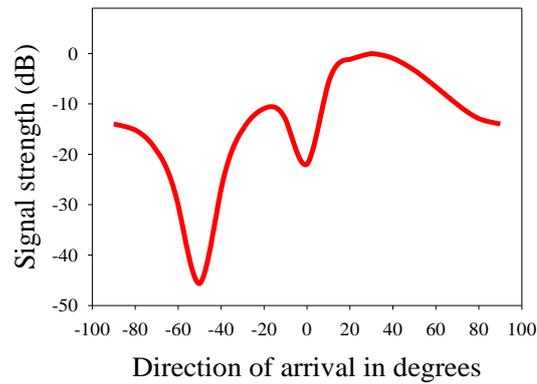


Fig 3.21 Normalized beam pattern of XSG model of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The beamformer output for the desired angle of 30 degrees and interference angle of -50 degrees obtained with the Xilinx System Generator model for varying look angles is illustrated in Fig 3.22(a) and Fig 3.22(b) respectively.

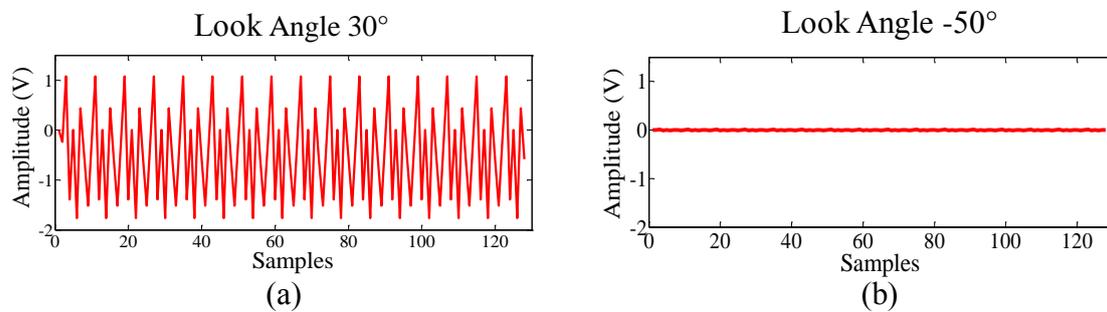


Fig 3.22 Beamformer output of XSG model of R-1D-MRT MVDR BF for varying look angles [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The beamformer output for the desired direction of 30 degrees has unity gain as seen in Fig 3.22(a). The undesired interference is at -50 degrees and the beamformer output is null at the interference direction as can be observed in Fig 3.22(b).

The normalized beam pattern obtained with the FPGA implementation is shown in Fig 3.23.

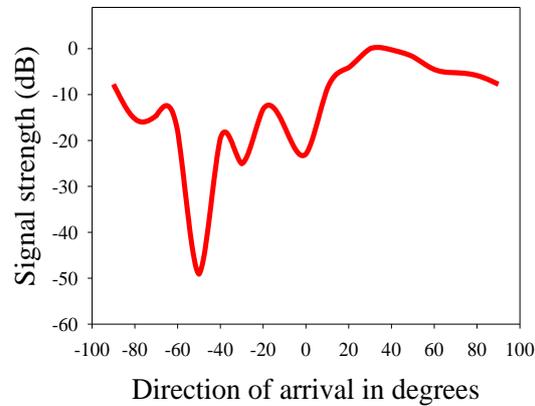


Fig 3.23 Normalized beam pattern of FPGA implementation of R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

The desired beam pattern with peak in the direction of desired signal and null in the direction of interfering signal is obtained in both XSG model simulation and FPGA implementation. The beamformer output shows output signal constrained to unity at the desired directions and null output at the interfering direction. Thus the FPGA implementation of R-1D-MRT MVDR beamformer is validated.

### 3.1.5.3 FPGA resource utilization of R-1D-MRT MVDR beamformer

The MVDR and 1D-R-MRT MVDR beamformers differ only in the sections pertaining to conversion to R-1D-MRT, correlation matrix calculation, weight multiplication and summing. Hence the resource comparison of these 2 beamformers is confined to the models of these sections.

The FPGA resource utilization of the R-1D-MRT MVDR beamformer and the MVDR beamformer are compared in Table 3.7.

Table 3.7 Comparison of FPGA resources of MVDR BF and R-1D-MRT MVDR BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB]

Resources	N = 256		N = 512	
	R-1D-MRT MVDR BF	MVDR BF	R-1D-MRT MVDR BF	MVDR BF
Slices	7782	7786	7786	7790
Flip Flops	6954	6968	6968	7002
Block RAM	14	28	28	50
Look Up Tables	20540	20548	20548	20552
Input Output blocks	1612	1612	1612	1612
Multipliers	168	168	168	168
Tristate Buffers	0	0	0	0

The data in Table 3.7 shows that the resource utilization of the R- 1D-MRT MVDR beamformer is lower than that of MVDR beamformer.

The computational efficiency and performance enhancement in various communication channels of the R-1D-MRT MVDR beamformer has been validated in the previous sections using MATLAB<sup>®</sup> simulations, XSG model simulations and FPGA implementation. The validation of R-1D-MRT is extended to LCMV beamformers in the next section.

### 3.2 R-1D-MRT LCMV BEAMFORMER

The basic idea behind linearly constrained minimum variance (LCMV) beamforming is to constrain the response of the beamformer so that signals from the direction of interest are passed with specified gain and phase. The weights are chosen to minimize output variance or power subject to the response constraint. This has the effect of preserving the desired signal while minimizing contributions to the output due to interfering signals and noise arriving from directions other than the direction of

interest (Van Deen, 2000). While the R-1D-MRT MVDR beamformer rejects interferences and passes the desired signal with unity gain, the Linearly Constrained Minimum Variance (LCMV) beamformer passes the desired signal to the output with a gain constraint  $g$  while nulling interfering signals.

The output power is expressed as

$$P = [E\{Y^2\}] = E[\mathbf{W}^H \mathbf{X} \mathbf{X}^H \mathbf{W}] = \mathbf{W}^H E[\mathbf{X} \mathbf{X}^H] \mathbf{W} = \mathbf{W}^H \mathbf{C} \mathbf{W} \quad (3.15)$$

where  $\mathbf{C} = E[\mathbf{X} \mathbf{X}^H]$  is the covariance matrix.

The optimum set of beamformer weights that minimizes the output power are computed, subject to complex gain constraint  $g$  at the desired direction  $\theta_d$ , i.e.,

$$\min_{\mathbf{W}} \{\mathbf{W}^H \mathbf{C} \mathbf{W}\} \text{ subject to } \mathbf{W}^H \mathbf{a}(\theta_d) = g \quad (3.16)$$

The weights of the LCMV beamformer is given by

$$\mathbf{W}_{\text{LCMV}} = g * \frac{\mathbf{C}^{-1} \mathbf{a}(\theta_d)}{\mathbf{a}^H(\theta_d) \mathbf{C}^{-1} \mathbf{a}(\theta_d)} \quad (3.17)$$

The signals at the array elements are weighted with these weight coefficients and summed to give the beamformer output. The schematic diagram of a conventional LCMV beamformer is shown in Fig 3.24(a).

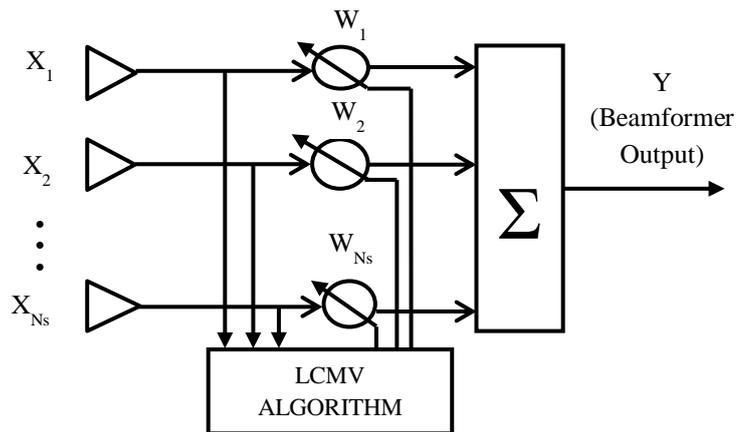


Fig 3.24(a) Schematic diagram of conventional LCMV beamformer

In the R-1D-MRT LCMV beamformer, the input data is transformed using R-1D-MRT and then beam formed using an LCMV beamformer. The schematic diagram of R-1D-MRT LCMV beamformer is shown in Fig 3.24(b).

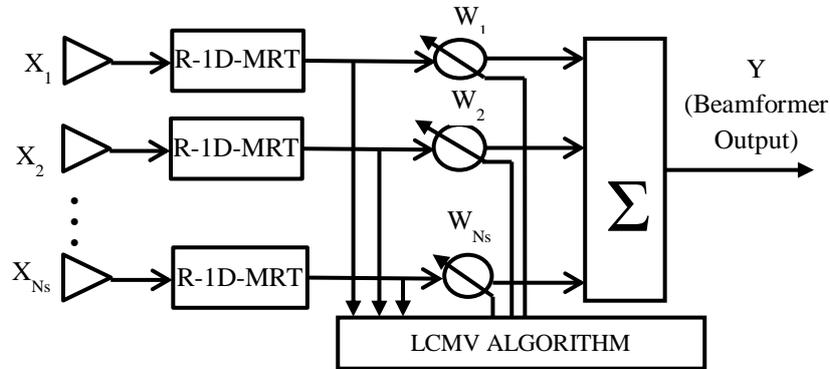


Fig 3.24(b) Schematic diagram of R-1D-MRT LCMV beamformer

### 3.2.1 Beam Pattern and Beamformer Output of R-1D-MRT LCMV Beamformer

The normalized beam pattern of the R-1D-MRT LCMV beamformer for varying number of antennas/sensors ( $N_s$ ) is shown in Fig 3.25.

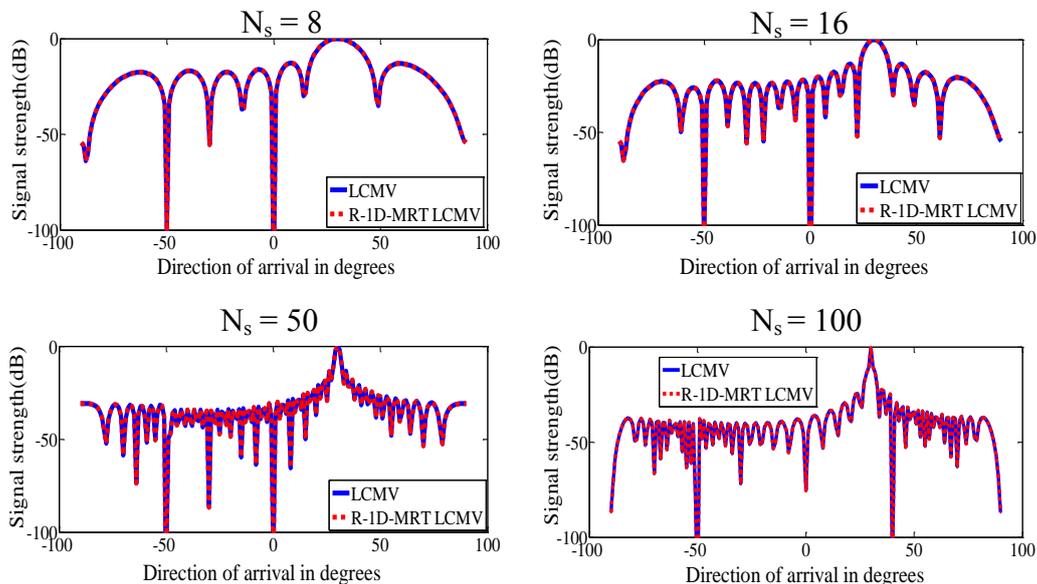


Fig 3.25 Normalized beam pattern of R-1D-MRT LCMV BF for varying number of sensors( $N_s$ ) [Source angle =  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ,  $SINR = 30$  dB,  $g = 20+20i$ ]

The normalized beam pattern of the R-1D-MRT LCMV beamformer for varying angles of desired and interfering signals is shown in Fig 3.26.

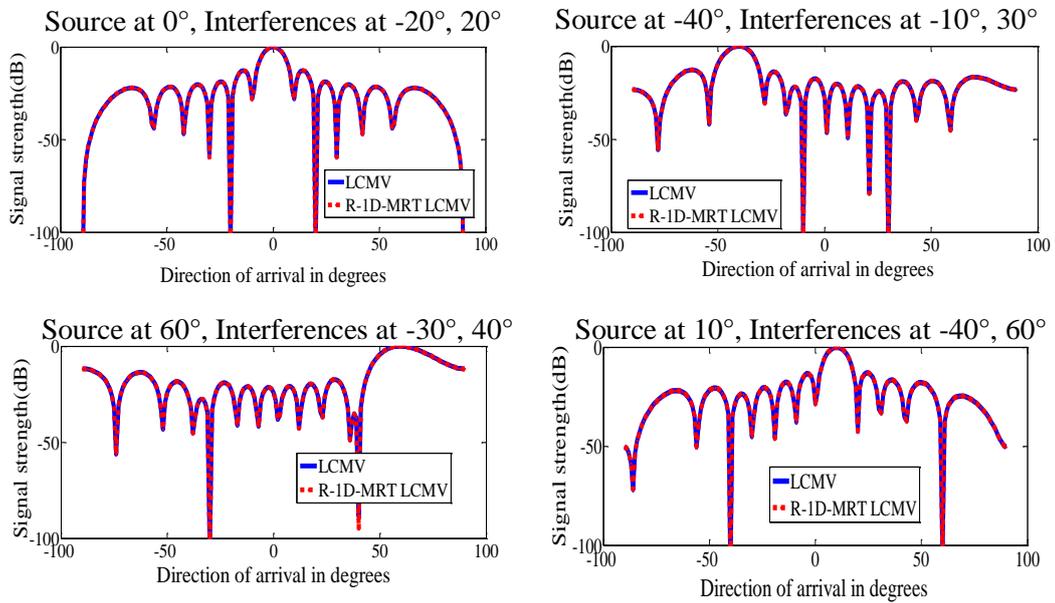


Fig 3.26 Normalized beam pattern of R-1D-MRT LCMV BF for varying source and interference angles [ $N_s = 12$ ,  $N = 512$ ,  $\text{SINR} = 30\text{dB}$ ,  $g = 20 + 20i$ ]

The normalized beam patterns of R-1D-MRT LCMV beamformer show main lobes at the desired directions and nulls at the interfering directions and are akin to that of LCMV beamformer.

A comparison of the beamformer output obtained with the R-1D-MRT LCMV beamformer for various look angles is shown in Fig 3.27.

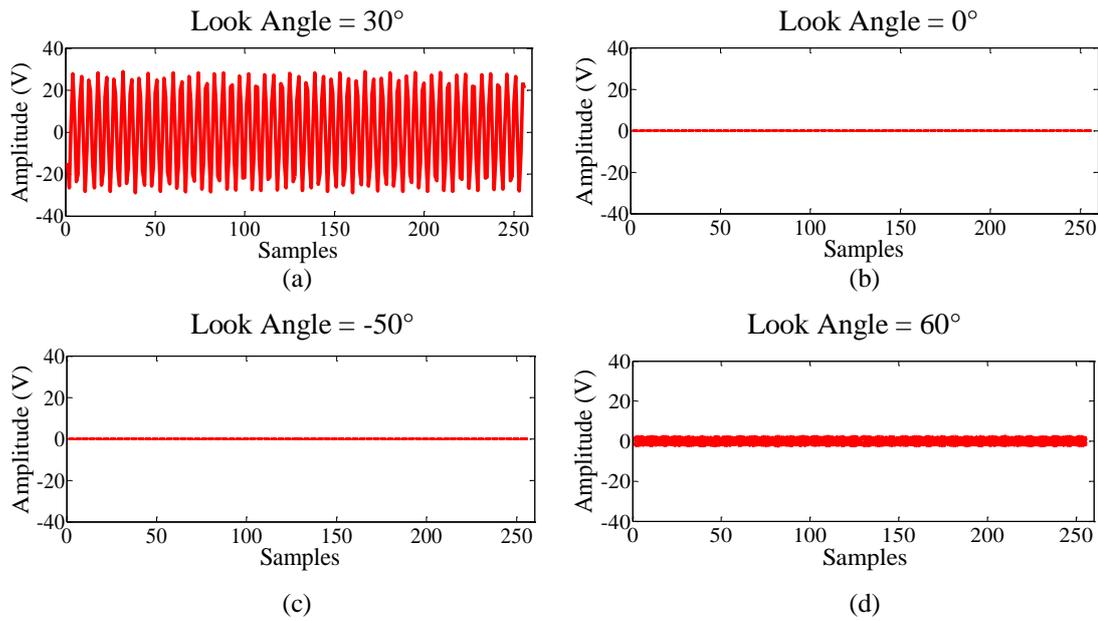


Fig 3.27 Beamformer output of R-1D-MRT LCMV beamformer for varying look angles [ $N_s = 8$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20+20i$ ]

The beamformer output obtained with the R-1D-MRT LCMV beamformer when look angle is same as source angle ( $30^\circ$ ) is seen in Fig 3.27(a). The beamformer output when look angle is in the same direction as the interferences of  $0^\circ$  and  $-50^\circ$  is seen in Fig 3.27(b) and Fig 3.27(c) respectively. Fig 3.27 (d) shows the beamformer output for an arbitrary look angle ( $60^\circ$ ). The beamformer output is maximum in the direction of the source and zero at the interference directions.

The R-1D-MRT LCMV beamformer is designed to pass the desired signal with gain  $g$ . The beamformer output obtained in the desired direction with the R-1D-MRT LCMV beamformer, illustrated in Fig 3.28, shows a gain in accordance with the gain constraint  $g$ .

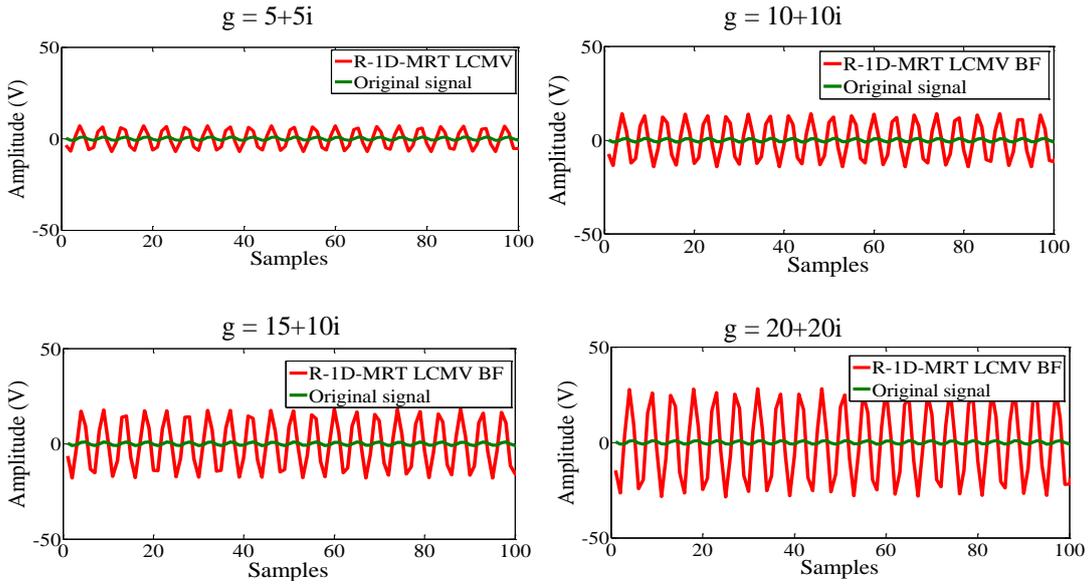


Fig 3.28 Beamformer output of R-1D-MRT LCMV beamformer for look angle of  $30^\circ$  for varying gain constraint ( $g$ ) [ $N_s = 8$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB]

### 3.2.2 Computational Complexity of R-1D-MRT LCMV Beamformer

This section discusses the computational density of the LCMV and R-1D-MRT LCMV beamformers. The computations in R-1D-MRT LCMV beamformer differ from that of R-1D-MRT MVDR beamformer only in the multiplication by a complex gain constraint  $g$  for weight calculation. This weight calculation part is the same for LCMV and R-1D-LCMV beamformers and hence is not considered in this comparison. The computational difference between the LCMV and R-1D-MRT LCMV beamformers occurs in the R-1D-MRT transformation, correlation calculation, multiplication of the signals by weights and final summation. The computational complexities of these sections are the same as that of the corresponding sections in MVDR and R-1D-MRT MVDR beamformers presented in section 3.1.2. Hence the computational load of the LCMV beamformer and R-1D-MRT LCMV beamformer

are same as that of MVDR and R-1D-MVDR beamformers respectively. The comparison of computational complexity of LCMV and R-1D-MRT LCMV beamformers is presented in Table 3.8.

Table 3.8 Comparison of computational complexity of LCMV BF and R-1D-MRT LCMV BF

N (Data size)	$N_s = 4$				$N_s = 8$			
	Number of real multiplications		Number of real additions		Number of real multiplications		Number of real additions	
	LCMV BF	R-1D- MRT LCMV BF	LCMV BF	R-1D- MRT LCMV BF	LCMV BF	R-1D- MRT LCMV BF	LCMV BF	R-1D- MRT LCMV BF
8192	655280	327600	655248	327592	2359008	1179360	2358880	1179344
4096	327600	163760	327568	163752	1179360	589536	1179232	589520
2048	163760	81840	163728	81832	589536	294624	589408	294608
1024	81840	40880	81808	40872	294624	147168	294496	147152
512	40880	20400	40848	20392	147168	73440	147040	73424
256	20400	10160	20368	10152	73440	36576	73312	36560
128	10160	5040	10128	5032	36576	18144	36448	18128
64	5040	2480	5008	2472	18144	8928	18016	8912
32	2480	1200	2448	1192	8928	4320	8800	4304
16	1200	560	1168	552	4320	2016	4192	2000

The R-1D-MRT LCMV beamformer also offers significant reduction in the computational load for beamforming. The considerable reduction in the number of multiplication operations and addition operations can reduce the computation time of the R-1D-MRT LCMV beamformer. The next section presents a comparison of the computation time of the LCMV and R-1D-MRT LCMV beamformer for various data sizes.

### 3.2.3 Computation Time of R-1D-MRT LCMV Beamformer

The effectiveness of the R-1D-MRT LCMV beamformer in terms of computation time is analyzed in this section. The computation time of the LCMV and R-1D-MRT LCMV beamformer for varying number of antennas/sensors ( $N_s$ ) and data sizes ( $N$ ) are compared in Tables 3.9 and 3.10.

Table 3.9 Comparison of computation time of LCMV BF and R-1D-MRT LCMV BF (seconds) [Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $g = 20+20j$ ]

N (Data size)	$N_s = 4$			$N_s = 16$		
	LCMV BF	R-1D-MRT LCMV BF	% Reduction	LCMV BF	R- 1D-MRT LCMV BF	% Reduction
8192	0.0039	0.0028	28	0.0150	0.0067	55
4096	0.0025	0.0021	16	0.0053	0.0032	39
2048	0.0016	0.0013	13	0.0030	0.0021	30
1024	0.0012	0.007	41	0.0021	0.0015	33
512	0.0007	0.0006	14	0.0015	0.0014	6
256	0.0007	0.0006	14	0.0013	0.0013	0
128	0.0006	0.0006	0	0.0010	0.0012	-20
64	0.0006	0.0006	0	0.0009	0.0011	-22
32	0.0005	0.0006	-20	0.0008	0.0009	-12.5
16	0.0005	0.0006	-20	0.0008	0.0009	-12.5

The simulations were performed with a computer using Intel Core i5 CPU working at 2.20 GHz and RAM of 4GB.

Table 3.10 Comparison of computation time of LCMV BF and R-1D-MRT LCMV BF (seconds) [Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $g = 20+20i$ ]

N (Data size)	$N_s = 50$			$N_s = 100$		
	LCMV BF	R-1D-MRT LCMV BF	% Reduction	LCMV BF	R-1D-MRT LCMV BF	% Reduction
8192	0.0382	0.0212	44	0.0970	0.0522	46
4096	0.0172	0.0115	33	0.0498	0.0267	42
2048	0.0091	0.0069	24	0.0245	0.0159	35
1024	0.0060	0.0042	30	0.0143	0.0100	30
512	0.0040	0.0031	22	0.0085	0.0076	10
256	0.0027	0.0020	25	0.0062	0.0055	11
128	0.0022	0.0019	13	0.0050	0.0047	6
64	0.0019	0.0017	10	0.0045	0.0041	8
32	0.0017	0.0017	0	0.0044	0.0039	11
16	0.0016	0.0016	0	0.0038	0.0038	0

The R-1D-MRT LCMV beamformer requires lesser computation time compared to LCMV beamformer.

### 3.2.4 Performance of R-1D-MRT LCMV Beamformer in Various Communication Channels

The R-1D-MRT LCMV beamformer performance is analyzed for the AWGN, Rayleigh and Rician communication channels using scatterplots and by measuring the Euclidian distance between the original signal and R-1D-MRT LCMV beamformer output.

Table 3.11 presents the Euclidian distances of the LCMV and R-1D-MRT LCMV beamformer output for communication through various channels for varying SINR.

Table 3.11 Comparison of Euclidian distance parameters of LCMV BF and R-1D-MRT LCMV BF for various communication channels for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20+20i$ ]

SINR	AWGN		Rayleigh		Rician	
	LCMV BF	R-1D-MRT LCMV BF	LCMV BF	R-1D-MRT LCMV BF	LCMV BF	R-1D-MRT LCMV BF
10 dB	0.9972	0.8763	1.8648	1.6948	1.3328	1.2963
20 dB	0.2879	0.1891	0.6049	0.5525	0.6381	0.5734
30 dB	0.0838	0.0743	0.5010	0.4994	0.3521	0.3477

The Euclidian distance parameters show a reduction in values for the R-1D-MRT LCMV beamformer compared to LCMV beamformer indicating that the R-1D-MRT LCMV beamformer output is more akin to the original signal.

The same is also validated using scatterplots for these channels as illustrated in Figures 3.29 - 3.31.

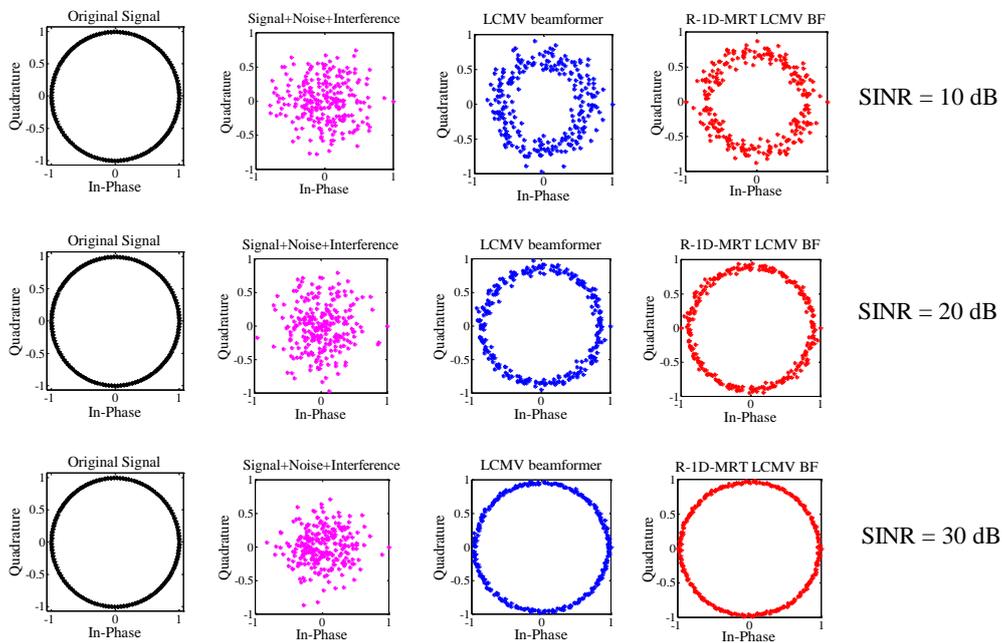


Fig 3.29 Scatterplots of R-1D-MRT LCMV BF for AWGN channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ,  $g = 20 + 20i$ ]

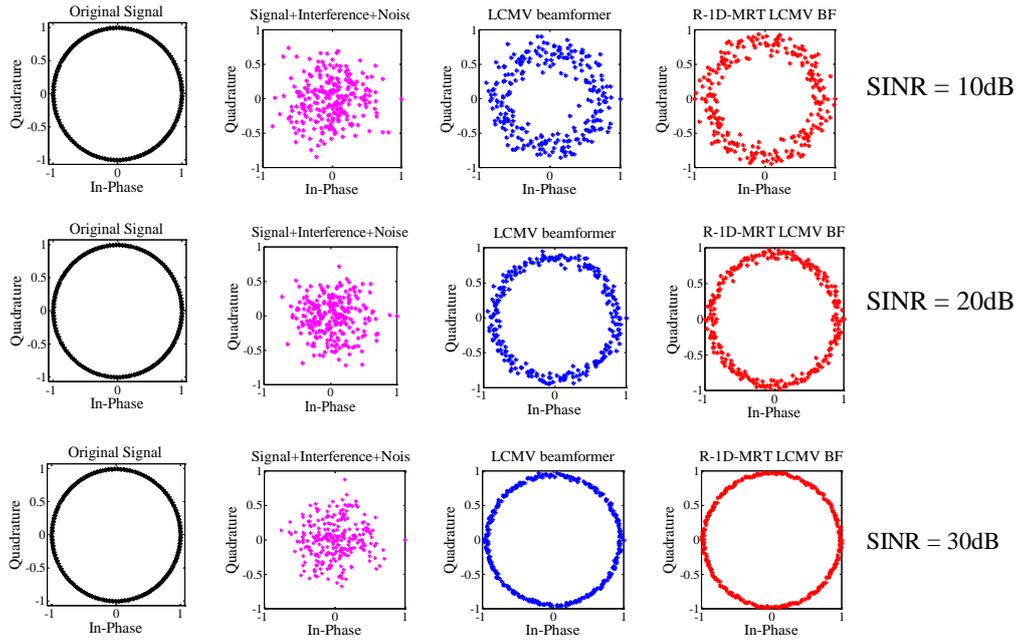


Fig 3.30 Scatterplots of R-1D-MRT LCMV BF for Rayleigh channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ,  $g = 20 + 20i$ ]

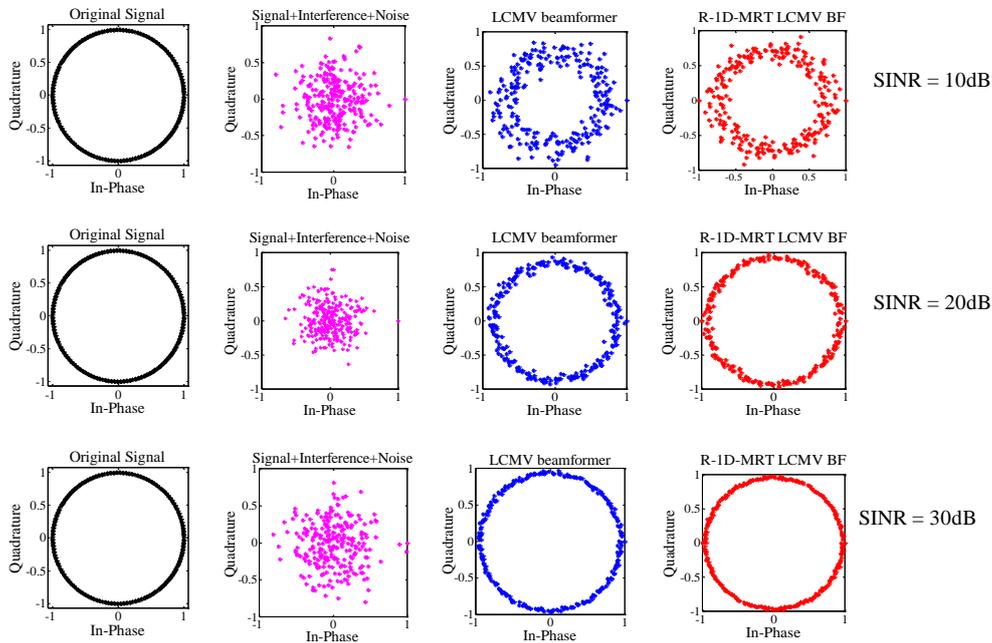


Fig 3.31 Scatterplots of R-1D-MRT LCMV BF for Rician channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ,  $g = 20 + 20i$ ]

The scatterplots of R-1D-MRT LCMV beamformer show more closeness to the original signal scatterplot.

The R-1D-MRT LCMV beamformer is validated using MATLAB<sup>®</sup> simulations as discussed in sections 3.2.1 - 3.2.5. The forthcoming sections are related to its FPGA implementation.

### **3.2.5 FPGA Implementation of R-1D-MRT LCMV Beamformer**

The models for the R-1D-MRT LCMV beamformer are created using Xilinx System Generator. Since the gain constraint alone is different from the MVDR beamformer, only the weight calculation block is different from that of the R-1D-MRT MVDR beamformer. The various models used for the FPGA implementation of the R-1D-MRT LCMV beamformer are presented.

A R-1D-MRT LCMV beamformer with 4 antennas/sensors ( $N_s$ ), desired signal at 30 degrees and an interferer at -50 degrees is considered. The complex gain constraint  $g$  is  $20+20i$ .

#### **3.2.5.1 Xilinx System Generator models of R-1D-MRT LCMV beamformer**

The desired and interfering signals and the R-1D-MRT transformed signals are generated in the input model.

The model for generating the desired and interfering signals is shown in Fig 3.32(a) and for phase shifting the signals the model shown in Fig 3.32(b) is used.

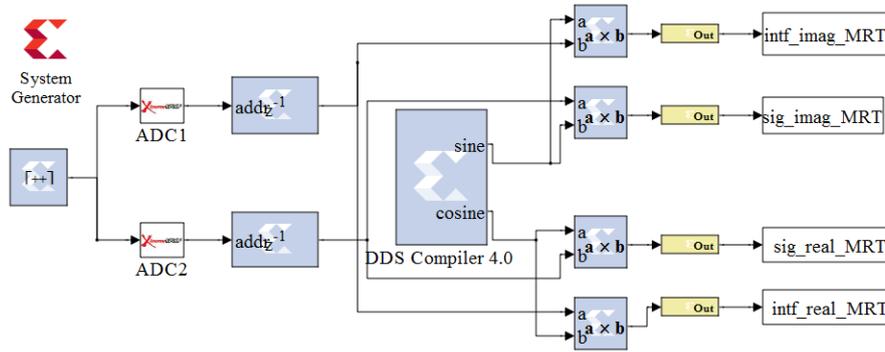


Fig 3.32(a) XSG model for generating desired and interfering signals of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20+20i$ ]

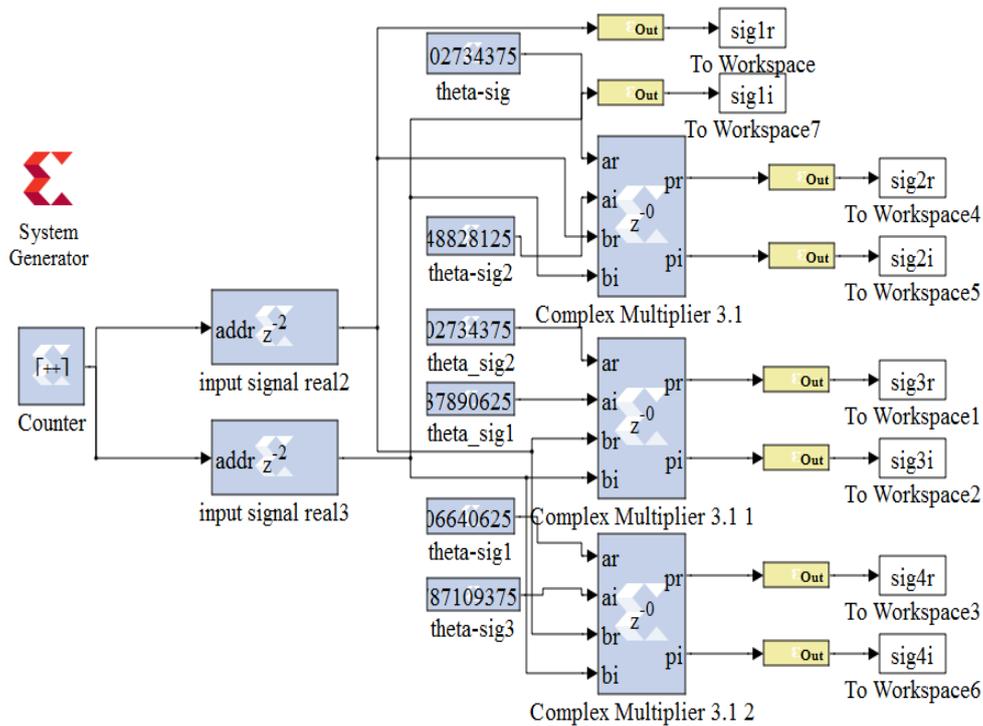


Fig 3.32(b) XSG model for phase shifting signals of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20+20i$ ]

After simulating the input signals, the correlation matrix must be calculated. The model for model for calculation of correlation matrix is shown in Fig 3.33.

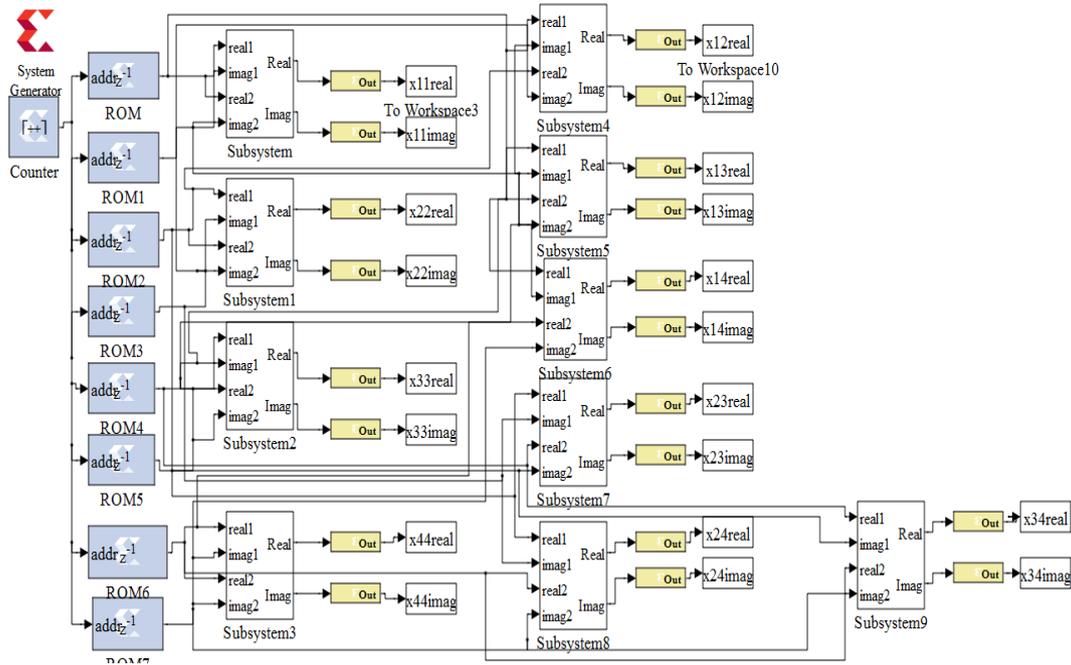


Fig 3.33 Correlation matrix model of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20+20i$ ]

The subsystem for the correlation matrix model is shown in Fig 3.34.

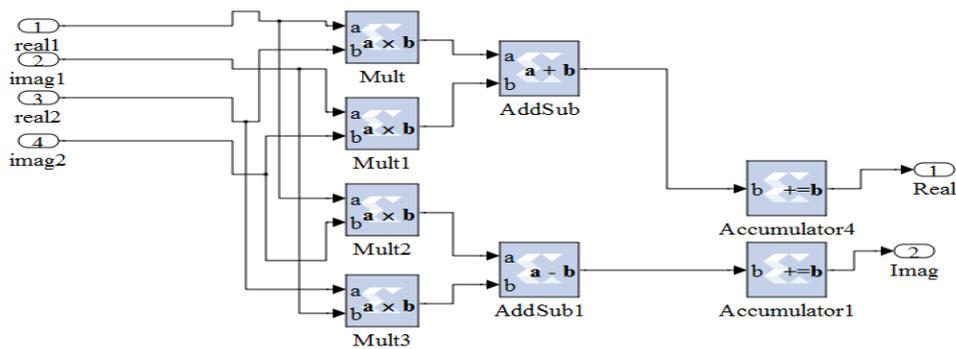


Fig 3.34 Subsystem of correlation matrix model of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20+20i$ ]

The computation of the weights of the LCMV beamformer requires the calculation of the inverse of the correlation matrix. Since the computation of matrix inverse is computationally expensive, QR decomposition can be used to compute the inverse of the correlation matrix (Roncero, 2014). The correlation matrix  $C$  is given by:

$$C = E[X^H(n)X(n)] \quad (3.18)$$

where  $X$  is the input data vector.

Replace  $C$  with  $QR$  where  $Q$  is an orthogonal matrix and  $R$  is an upper triangular matrix.

$$C^{-1} = R^{-1}Q^H \quad (3.19)$$

$R^{-1}$  and  $Q^H$  should be calculated for computing the inverse of the correlation matrix. Different numerical methods are available for QR decomposition. The Modified Gaussian Elimination method along with back substitution is used for calculating the inverse of the correlation matrix. The Modified Gaussian Elimination method for matrix inversion of the correlation matrix is performed in the following manner. Row transformations are applied to the correlation matrix, denoted as  $C$ , to get an upper triangular matrix  $R$ . For obtaining  $R$  of a 4 x 4 correlation matrix, the first row of  $C$  is retained as it is, and one row transformation is applied to the second row.

$$C2' = C2 - \frac{C2(1)}{C1(1)}C1 \quad (3.20)$$

2 transformations are applied to the 3<sup>rd</sup> row.

$$C3' = C3 - \frac{C3(2)}{C1(2)}C1 \quad (3.21)$$

$$C3'' = C3' - \frac{C3'(2)}{C1(2)}C2' \quad (3.22)$$

3 row transformations of similar nature are applied to the 4<sup>th</sup> row of C. The same transformation must be applied to the identity matrix I whose size is identical to that of C. Retain the first row of I as it is and applying one transformation to the first row of I of the form

$$I2' = I2 - \frac{C2(1)}{C1(1)} I1 \quad (3.23)$$

Two row transformations have to be applied to the 3<sup>rd</sup> row of I.

$$I3' = I3 - \frac{C3(2)}{C1(2)} I1 \quad (3.24)$$

$$I3'' = I3' - \frac{C3'(2)}{C1(2)} I2' \quad (3.25)$$

Similarly 3 row transformations are applied to the 4<sup>th</sup> row of I.

This implementation uses ROMS and MATLAB<sup>®</sup> workspaces for transforming the rows of the C and I matrix.

The Xilinx System Generator model for transforming a row element is shown in Fig 3.35.

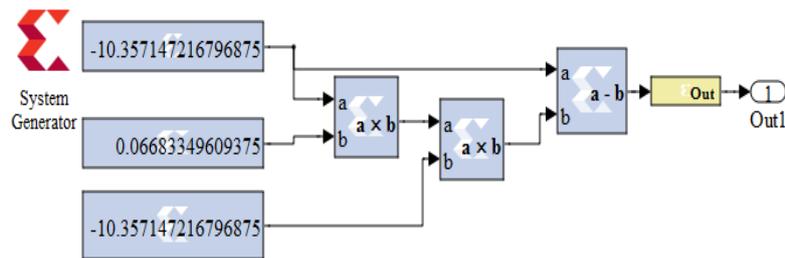


Fig 3.35 Model for row transformation of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

The Xilinx System Generator models showing the values of correlation matrix after applying the first transformation to 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup> rows are shown in Fig 3.36.

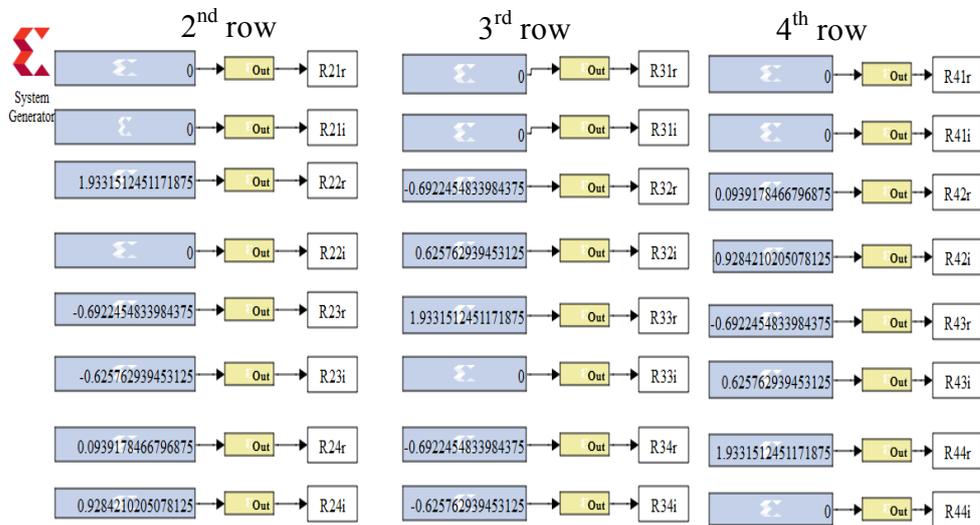


Fig 3.36 Model for first transformation of 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup> rows of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

After the first transformation is applied to any row, the leading element of that row becomes zero. Since beamforming is performed with complex data, the real and imaginary parts are modelled separately. Fig 3.36 shows that after the first transformation the leading elements of the 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup> rows are zero. The real and imaginary values of these elements are represented as R21r, R21i, R31r, R31i, R41r and R41i respectively.

The second row transformations are used to null the second element of the row to which it is applied. The model showing the values after applying the second transformations to the third and fourth rows of the correlation matrix  $C$  is presented in Fig 3.37.

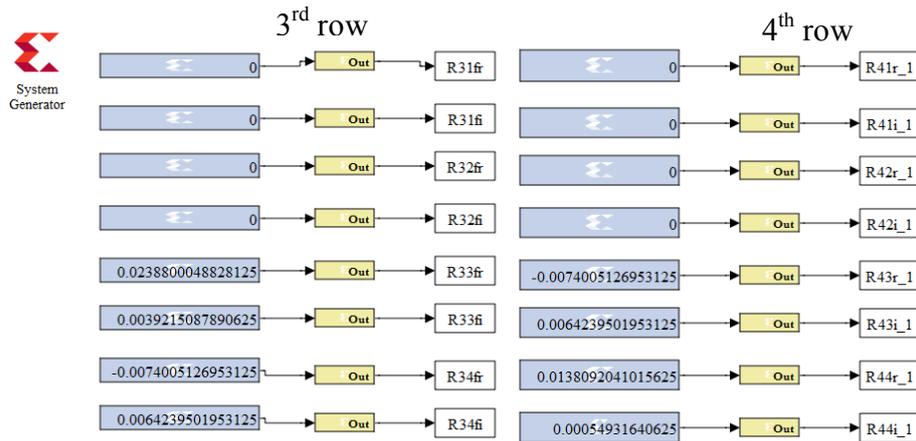


Fig 3.37 Model for second transformation of 3<sup>rd</sup> and 4<sup>th</sup> rows of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

After the second row transformation, the second elements of the third and fourth row must have null values. It can be verified in Fig 3.37, that after the second transformation, the real and imaginary parts of the leading 2 elements of the 3<sup>rd</sup> and 4<sup>th</sup> rows are having zero values. By applying one more row transformation to the fourth row of the correlation matrix, the 3<sup>rd</sup> element of the 4<sup>th</sup> row can be nullified. The Xilinx system generator block for the 3<sup>rd</sup> transformation to the 4<sup>th</sup> row is represented in Fig 3.38.

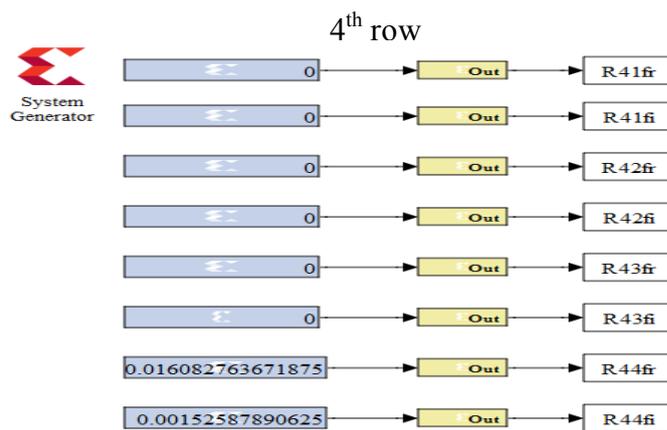


Fig 3.38 Model for third transformation of 4<sup>th</sup> row of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

Fig 3.38 shows the model for last step in converting the 4 x 4 correlation matrix  $C$  to an upper triangular matrix. It can be observed that after this transformation the first 3 elements of the 4<sup>th</sup> row are zero. The correlation matrix  $C$  has been converted to an upper triangular matrix  $R$ .

Similar models are created for transforming an identity matrix  $I$  to the  $Q$  matrix. Applying the same transformation on  $I$  as was done on the correlation matrix  $C$ , a lower triangular matrix  $Q^H$  is obtained.

The upper triangular matrix  $R$  satisfies the relation,

$$R^{-1}R = I \quad (3.26)$$

Using the above relation and back substitution methods, the value of  $R^{-1}$  is computed.

The back substitution model is shown in Fig 3.39.

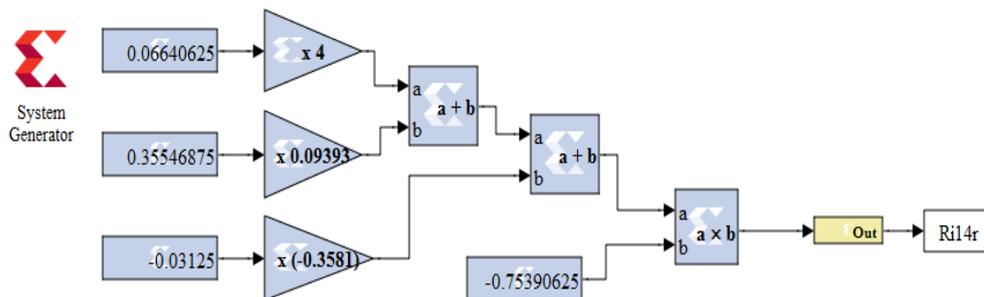


Fig 3.39 Back substitution model of R-ID-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20+20i$ ]

The  $R^{-1}$  matrix will be generated after execution of the back substitution models shown in Fig 3.39. This matrix must be multiplied with the  $Q^H$  matrix. Fig 3.40 shows the model created for multiplying these matrices.

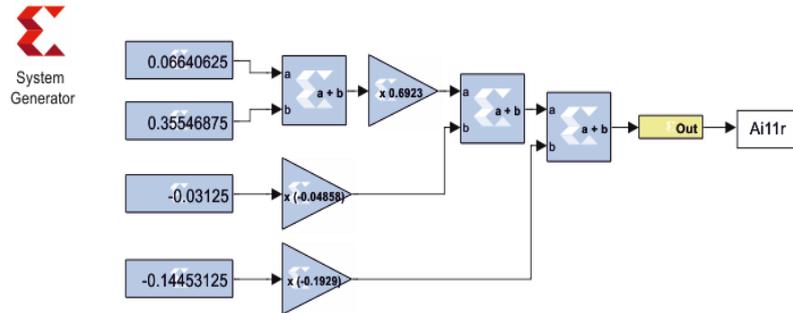


Fig 3.40 Model for multiplying  $R^{-1}$  and  $Q$  matrices of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20+20i$ ]

After multiplication of the  $R^{-1}$  and  $Q$  matrices, the inverse of the correlation matrix will be available. The matrix inverse obtained for a 4 x 4 correlation matrix using the Xilinx System Generator model is shown in Table 3.12 while the matrix inverse for the same correlation matrix obtained using MATLAB<sup>®</sup> is shown in Table 3.13. The results are almost identical.

Table 3.12 Matrix inverse of R-1D-MRT LCMV BF using XSG model

0.7539	$0.1836 + 0.1641i$	$-0.0234 - 0.2461i$	$-0.1445 + 0.1992i$
$0.1797 - 0.1641i$	0.7500	$0.1836 + 0.1641i$	$-0.0234 - 0.2422i$
$-0.0234 + 0.2461i$	$0.1836 - 0.1641i$	0.7578	$0.1836 + 0.1641i$
$-0.1445 - 0.1992i$	$-0.0234 + 0.2422i$	$0.1826 - 0.1650i$	0.7539

Table 3.13 Matrix inverse of R-1D-MRT LCMV BF using MATLAB<sup>®</sup>

0.7544	$0.1822 + 0.1647i$	$-0.0247 - 0.2444i$	$-0.1455 + 0.1979i$
$0.1822 - 0.1647i$	0.7544	$0.1822 + 0.1647i$	$-0.0247 - 0.2444i$
$-0.0247 + 0.2444i$	$0.1822 - 0.1647i$	0.7544	$0.1822 + 0.1647i$
$-0.1455 - 0.1979i$	$-0.0247 + 0.2444i$	$0.1822 - 0.1647i$	0.7544

Once the inverse of the correlation matrix is obtained, the weights of the R-1D-MRT LCMV beamformer can be calculated. The Xilinx System generator model for calculating the weights of the R-1D-MRT LCMV beamformer is shown in Fig 3.41.

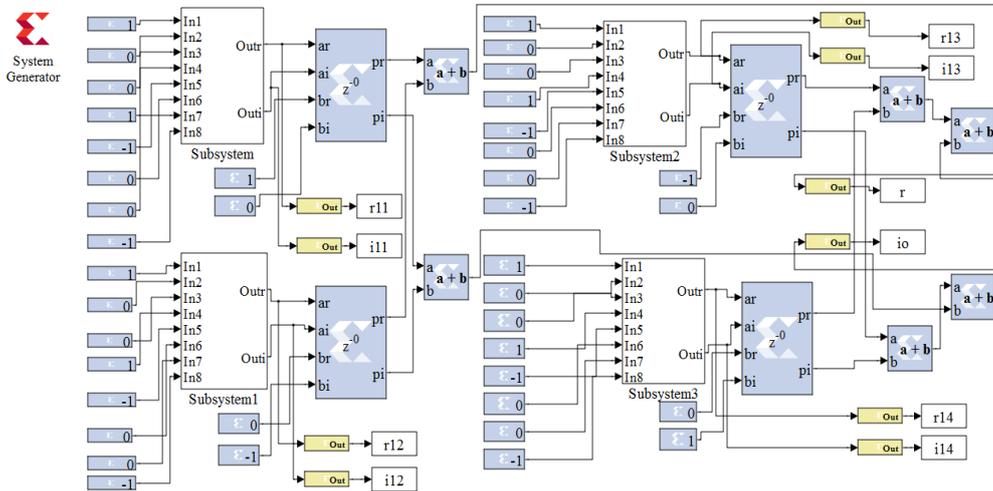


Fig 3.41 Model for obtaining the weights of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

The subsystem for weight calculation is shown in Fig 3.42.

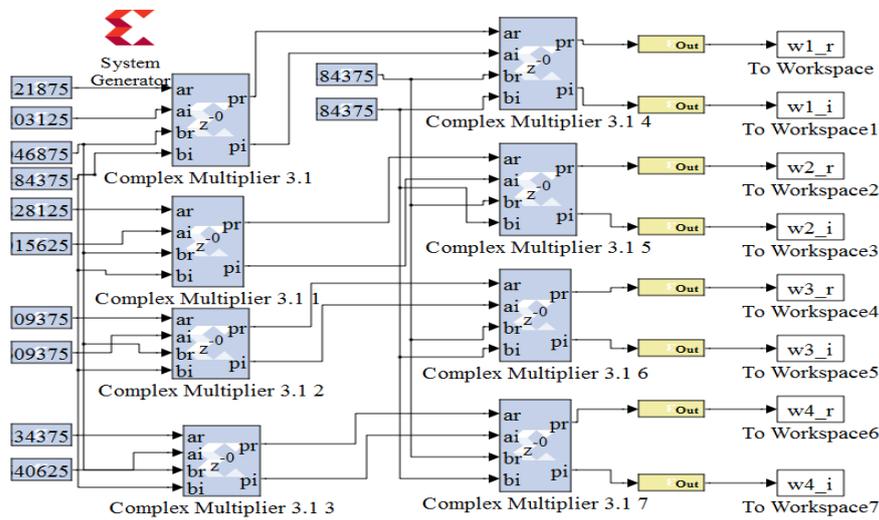


Fig 3.42 Weight calculation subsystem of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

In the weight calculation subsystem block shown in Fig 3.42, the gain constraint is introduced. After obtaining the weight, the input signal must be multiplied with these weights and summed to give the beamformer output. Fig 3.43 shows the model for calculating the beamformer output.

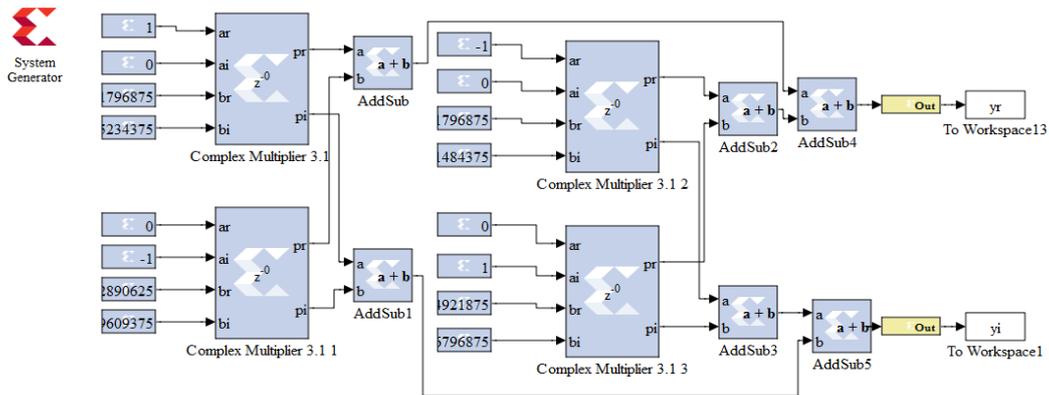


Fig 3.43 Model for obtaining beamformer output of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

### 3.2.5.2 Beam pattern and beamformer output of XSG model of R-1D-MRT LCMV beamformer

The beamformer output of R-1D-MRT LCMV beamformer for varying look angles is shown in Fig 3.44.

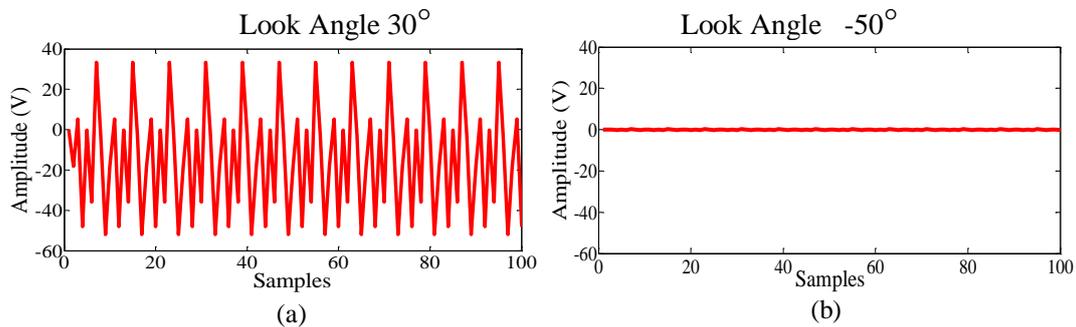


Fig 3.44 Beamformer output of XSG model of R-1D-MRT LCMV BF for varying look angles [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

The beamformer output has a maximum output when the look angle is same as the source angle of  $30^\circ$  (Fig 3.44(a)) and zero output in the interference direction of  $-50^\circ$  (Fig 3.44(b)).

The normalized beam pattern of R-1D-MRT LCMV BF obtained with the XSG model is shown in Fig 3.45.

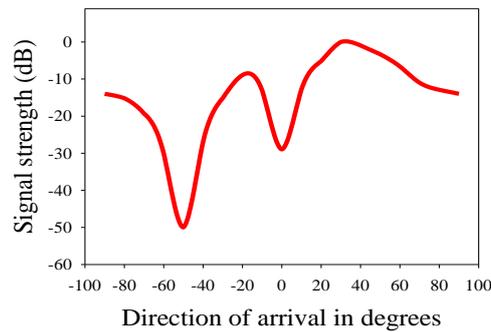


Fig 3.45 Normalized beam pattern of XSG model of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

The above beam pattern shows a peak in the direction of the desired source and null in the direction of the interfering signal. The normalized beam pattern of R-1D-MRT LCMV BF with FPGA implementation is illustrated in Fig 3.46.

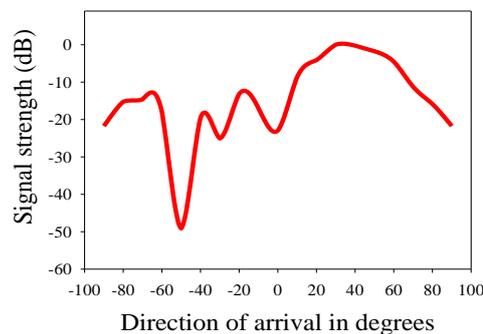


Fig 3.46 Normalized beam pattern of FPGA implementation of R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

The normalized beam pattern obtained with the FPGA implementation is in the desired shape with a main lobe in the desired angle of  $30^\circ$  and very low signal power in the undesired direction of  $-50^\circ$ . The validation is extended to include FPGA resource utilization as presented in the next section.

### 3.2.5.3 FPGA resource utilization of R-1D-MRT LCMV beamformer

The resources for FPGA implementation are compared for the LCMV beamformer and the R-1D-MRT based LCMV beamformer in Table 3.14. Only the sections which are different for the LCMV and R-1D-MRT LCMV beamformers are used in this comparison.

Table 3.14 Comparison of FPGA resources of LCMV BF and R-1D-MRT LCMV BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $g = 20 + 20i$ ]

Resources	N =256		N = 512	
	R-1D-MRT LCMV BF	LCMV BF	R-1D-MRT LCMV BF	LCMV BF
Slices	7782	7786	7786	7790
Flip Flops	6954	6968	6968	7002
Block RAM	14	28	28	50
Look Up Tables	20540	20548	20548	20552
Input Output blocks	1612	1612	1612	1612
Multipliers	168	168	168	168
Tristate Buffers	0	0	0	0

The R-1D-MRT LCMV beamformer utilizes fewer resources for FPGA implementation compared to LCMV beamformer.

### 3.3 CHAPTER SUMMARY

The R-1D-MRT MVDR beamformer and R-1D-MRT LCMV beamformer exhibit a reduction in computation time and computational complexity with respect to conventional MVDR and LCMV beamformers. The R-1D-MRT MVDR beamformer provides unity gain to signals from the desired direction while the R-1D-MRT LCMV beamformer provides a gain corresponding to the gain constraint  $g$  to signals from the desired direction. It also rejects undesired signals. The R-1D-MRT MVDR and R-1D-MRT LCMV beamformers exhibit better performance with respect to conventional MVDR and LCMV beamformers in various communication channels. The FPGA implementation is also carried out and lesser FPGA resources are utilized for implementation of the R-1D-MRT MVDR and R-1D-MRT LCMV beamformers. The effectiveness of R-1D-MRT MVDR and R-1D-MRT LCMV beamformers is hence validated.

In the next chapter, the R-1D-MRT algorithm is validated on Least Mean Square and Normalized Least Mean Square beamformers which employ stochastic gradient methods for weight adaptation.

## CHAPTER 4

### R-1D-MRT LMS AND NLMS BEAMFORMERS

<b>CONTENTS</b>	
☆ <b>R-1D-MRT LMS beamformer</b> <ul style="list-style-type: none"><li>◆ Beam pattern and beamformer output</li><li>◆ Computational complexity</li><li>◆ Computation time</li><li>◆ Performance in various communication channel</li></ul>	☆ <b>R-1D-MRT NLMS beamformer</b> <ul style="list-style-type: none"><li>◆ Beam pattern and beamformer output</li><li>◆ Computational complexity</li><li>◆ Computation time</li><li>◆ Performance in various communication channels</li></ul>
☆ <b>FPGA implementation of R-1D-MRT LMS beamformer</b> <ul style="list-style-type: none"><li>◆ Xilinx System Generator model</li><li>◆ Beam pattern and beamformer output</li><li>◆ FPGA resource utilization</li></ul>	☆ <b>FPGA implementation of R-1D-MRT NLMS beamformer</b> <ul style="list-style-type: none"><li>◆ Xilinx System Generator model</li><li>◆ Beam pattern and beamformer output</li><li>◆ FPGA resource utilization</li></ul>

The R-1D-MRT algorithm was validated on the MVDR and LCMV beamformers in the previous chapter. This chapter presents the validation of the algorithm on Least Mean Square and Normalized Least Mean Square beamformers which use stochastic gradient methods for weight adaptation. The validation using MATLAB<sup>®</sup> simulations and FPGA implementation using Xilinx System Generator models is considered.

#### 4.1 R-1D-MRT LMS BEAMFORMER

The Least Mean Square (LMS) beamformer is an adaptive beamformer employing LMS algorithm for adjusting the weights in response to the environment. LMS algorithm is one of the most popular algorithms for adaptive beamforming. It is an important stochastic gradient algorithm offering the advantages of simplicity and robustness but suffers from the disadvantage of slow convergence (Srar et al., 2010).

It does not require computationally intensive measurements of correlation functions and matrix inversions like the MVDR and LCMV beamformers. These factors have made this algorithm very popular.

The LMS algorithm for beamforming consists of two basic processes:

- 1) an estimation process or training phase in which the beamformer output  $\mathbf{Y}(n)$  is compared with a desired response  $\mathbf{d}(n)$  and an estimation error  $\mathbf{e}(n)$  is generated.
- 2) an adaptation process which automatically adjusts the weights of the beamformer in accordance with the estimation error.

The LMS algorithm changes the weight vector  $\mathbf{W}$  along the direction of the estimated gradient based on the steepest descent method. The weight vector update for LMS algorithm (Imtiaj et al., 2012) is given by

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \mu \mathbf{X}(n) \mathbf{e}^*(n) \quad (4.1)$$

where  $\mu$  is the step size,  $\mathbf{e}(n)$  is the error vector given by

$$\mathbf{e}(n) = \mathbf{Y}(n) - \mathbf{d}(n) \quad (4.2)$$

$\mathbf{e}^*(n)$  is the conjugate of the error vector.

The step size  $\mu$  is a positive real valued constant which determines the convergence rate and performance of the algorithm. The upper bound on  $\mu$  (Srar et al., 2010) is given by

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (4.3)$$

where  $\lambda_{\max}$  is the largest Eigen value of the autocorrelation matrix  $\mathbf{R} = E[\mathbf{X}(n)\mathbf{X}^H(n)]$ .

$\mathbf{X}(n)$  is the input signal and  $E(\cdot)$  is the expectation operator and H is Hermitian transpose. The size of the incremental correction applied to the weight vector between successive iterations is controlled by the step size  $\mu$ . The LMS algorithm

also requires knowledge of the transmitted signal. For this purpose, known pilot sequences are transmitted periodically to the receiver.

The beamformer output is given by

$$\mathbf{Y}(n) = \mathbf{W}^H \mathbf{X}(n) \quad (4.4)$$

where  $\mathbf{W}$  is the weight vector.

The schematic diagram of the conventional LMS beamformer where the signal received at the array is modified using the adaptive weights and summed to give the beam former output is shown in Fig 4.1(a).

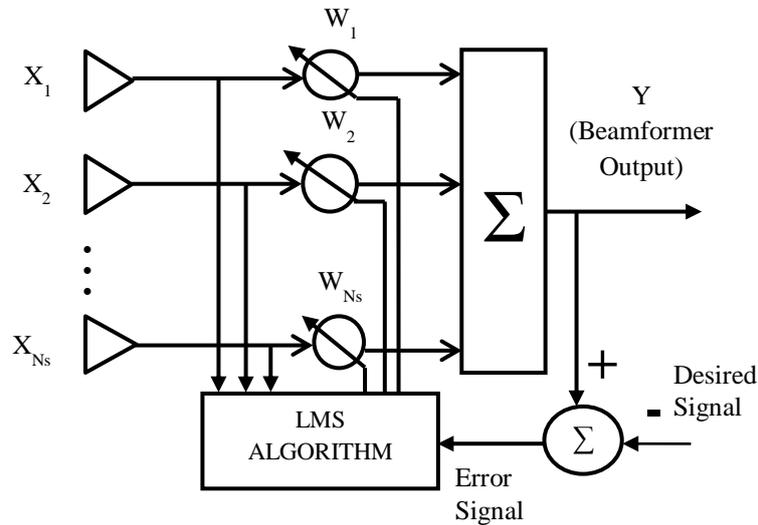


Fig 4.1(a) Schematic diagram of conventional LMS beamformer

In the R-1D-MRT LMS beamformer, the signals received at the antenna/sensor array are transformed using the R-1D-MRT algorithm. These transformed signals are multiplied with the optimum weights obtained using LMS algorithm. The beamformer output is obtained by summing up the weighted signals. The schematic diagram of the R-1D-MRT LMS beamformer is shown in Fig 4.1(b).

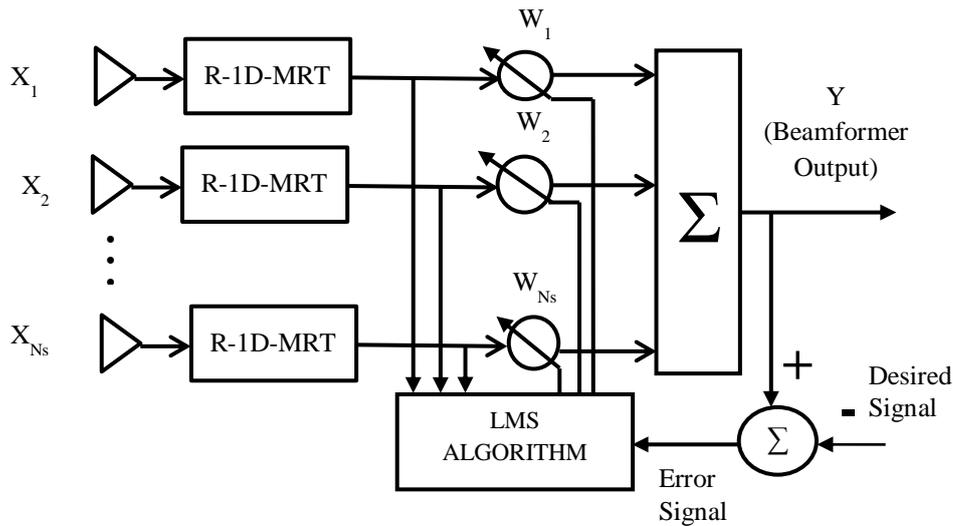


Fig 4.1(b) Schematic diagram of R-1D-MRT LMS beamformer

The validation of R-1D-MRT LMS beamformer through MATLAB<sup>®</sup> simulations is presented in the following sections.

#### 4.1.1 Beam Pattern and Beamformer Output of R-1D-MRT LMS Beamformer

For an interference prone environment, the beam pattern should point the main lobe in the direction of the desired signal and nulls in the direction of interfering signals. The beam pattern, weight adaptation, mean square error and signal tracking capabilities of the R-1D-MRT LMS beamformer are examined in this section. Fig 4.2 and Fig 4.3 illustrate the output characteristics of the R-1D-MRT LMS beamformer for varying conditions.

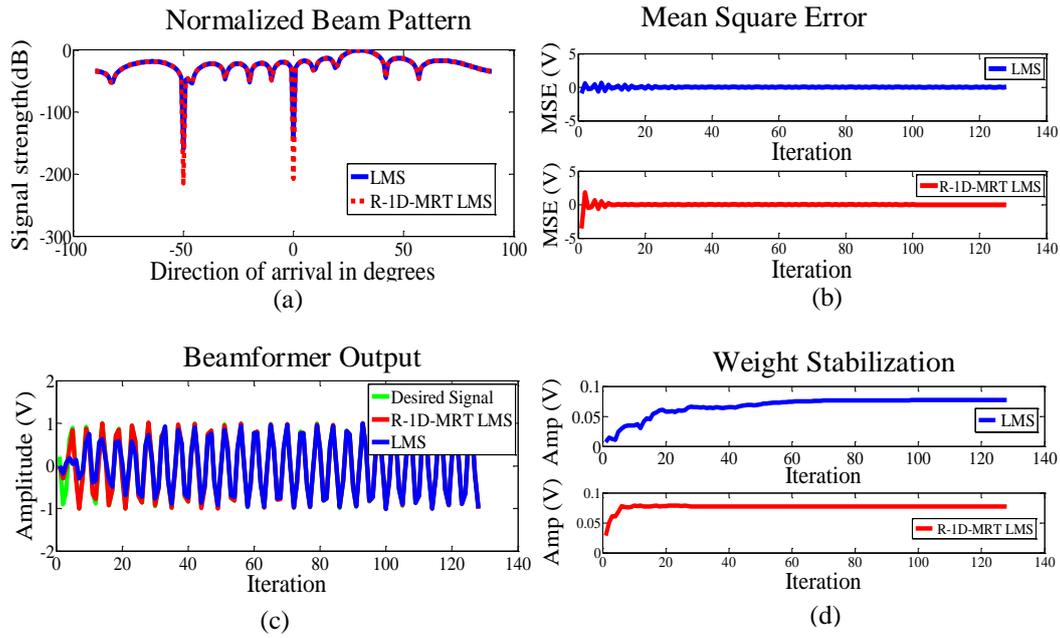


Fig 4.2 R-1D-MRT LMS beamformer output [ $N_s = 12$ , Source angle  $30^\circ$ , Interference angle  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu = 0.005$ ]

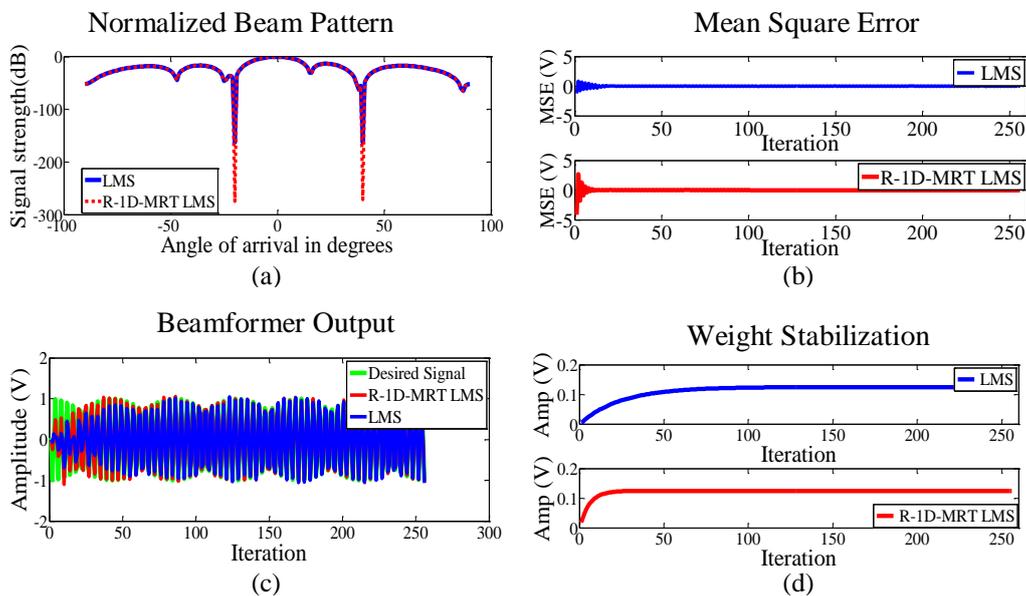


Fig 4.3 R-1D-MRT LMS beamformer output [ $N_s = 8$ , Source angle  $0^\circ$ , Interference angles  $-20^\circ$  and  $40^\circ$ ,  $N = 512$ , SINR = 20 dB,  $\mu = 0.005$ ]

The normalized beam patterns of Fig 4.2(a) and Fig 4.3(a) exhibit peaks at the desired angles and strong nulls in the direction of interfering signals. The mean square error is initially higher for the R-1D-MRT LMS beamformer but falls to zero in lesser number of iterations compared to LMS beamformer as indicated in Fig 4.2(b) and Fig 4.3(b). Fig 4.2(c) and 4.3(c) show that the R-1D-MRT LMS beamformer output signal tracks the original signal much faster than LMS beamformer. The weights of the R-1D-MRT LMS beamformer stabilizes in lesser number of iterations compared to LMS beamformer as demonstrated in Fig 4.2(d) and Fig 4.3(d).

In the R-1D-MRT LMS beamformer, convergence is much faster than the LMS beamformer. A comparison of the minimum number of iterations required by the R-1D-MRT LMS beamformer and the LMS beamformer to converge for varying data sizes ( $N$ ) is presented in Tables 4.1 for 4 and 16 array elements.

Table 4.1 Comparison of minimum number of iterations ( $I_{\min}$ ) for LMS BF and R-1D-MRT LMS BF for convergence [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 20 dB]

N (Data size)	$\mu$	$N_s = 4$		$N_s = 16$	
		LMS BF	R-1D-MRT LMS BF	LMS BF	R-1D-MRT LMS BF
8192	0.0003	3500	1000	2200	600
4096	0.0008	1400	380	400	120
2048	0.003	400	100	150	30
1024	0.004	320	100	87	23
512	0.008	58	24	40	18
256	0.008	52	20	38	16
128	0.01	50	20	30	15
64	0.03	28	13	26	11

The above discussions prove that R-1D-MRT LMS beamformer gives the desired beam pattern with the added advantage of faster convergence compared to

conventional LMS beamformer. Computational complexity analysis of R-1D-MRT LMS beamformer is discussed in the next section.

#### 4.1.2 Computational Complexity of R-1D-MRT LMS Beamformer

Computational operations are performed in the R-1D-MRT transformation, weight adaptation, and weight multiplication and summing sections of the R-1D-MRT LMS beamformer. The weight adaptation section of the LMS algorithm with  $N_s$  antennas/sensors requires  $5N_s$  real multiplications and  $6N_s$  real additions for a single iteration. The total number of computations for weight adaptation is calculated as the product of computational complexity of a single iteration and the minimum number of iterations for convergence of LMS BF ( $I_{\min(\text{LMS})}$ ) and R-1D-MRT LMS BF ( $I_{\min(\text{R-1D-MRT LMS})}$ ) specified in Table 4.1. For the R-1D-MRT LMS BF, the conversion to R-1D-MRT requires  $2N_s(N-1)$  real additions. For multiplication of the signals and weights, the LMS BF requires  $N_s(4N-4)$  real multiplications and  $N_s(2N-2)$  real additions while the R-1D-MRT LMS BF requires only  $N_s(2N-4)$  real multiplications and  $N_s(N-2)$  real additions. Summation requires  $N-2$  real additions for the R-1D-MRT LMS BF and  $2N-2$  real additions for LMS BF.

The total number of computations required for LMS beamformer is  $N_s(4N-4+5I_{\min(\text{LMS})})$  real multiplications and  $N_s(4N-4+6I_{\min(\text{LMS})})$  real additions while the R-1D-MRT LMS beamformer requires  $N_s(2N-4+5I_{\min(\text{R-1D-MRT-LMS})})$  real multiplications and  $N_s(4N-4+6I_{\min(\text{R-1D-MRT-LMS})})$  real additions. The computational complexity of LMS and R-1D-MRT LMS beamformer for 4 antennas/sensors ( $N_s$ )

and 8 antennas/sensors ( $N_s$ ) with minimum number of iterations required for convergence is presented in Table 4.2.

Table 4.2 Comparison of computational complexity of LMS BF and R-1D-MRT LMS BF

N (Data size)	$N_s = 4$				$N_s = 16$			
	Number of real multiplications		Number of real additions		Number of real multiplications		Number of real additions	
	LMS BF	R- 1D- MRT LMS BF	LMS BF	R-1D- MRT LMS BF	LMS BF	R-1D- MRT LMS BF	LMS BF	R-1D- MRT LMS BF
8192	201056	85520	215056	155056	700224	310080	735424	581824
4096	93520	40352	99120	74640	294080	140608	300480	273600
2048	40752	18368	42352	35152	143008	67872	145408	133888
1024	22768	10176	24048	18768	72432	34544	73824	67680
512	9336	4560	9568	8752	35904	17760	36544	34432
256	5120	2432	5328	4560	19360	9408	19968	17856
128	3032	1408	3232	2512	10528	5232	11008	9568
64	1568	756	1680	1320	6112	2864	6528	5088

Table 4.2 validates the computational efficiency of the R-1D-MRT LMS beamformer. The reduction in the number of multiplications and additions in the R-1D-MRT NLMS BF should in turn reduce its computation time. In the next section the computation time of the LMS and R-1D-MRT LMS beamformers are compared.

### 4.1.3 Computation Time of R-1D-MRT LMS Beamformer

To further validate the efficiency of the R-1D-MRT LMS beamformer, the computation time of the R-1D-MRT LMS beamformer and the LMS beamformer for varying number of antennas/sensors ( $N_s$ ) and data sizes (N) are compared. Table 4.3

and Table 4.4 display a comparison of the computation time of the R-1D-MRT LMS beamformer for varying number of antennas/sensors ( $N_s$ ) and data sizes ( $N$ ).

Table 4.3 Comparison of computation time of LMS BF and R-1D-MRT LMS BF (seconds) [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $\mu = 0.005$ ]

N (Data size)	$N_s = 4$			$N_s = 16$		
	LMS BF	R-1D-MRT LMS BF	% Reduction	LMS BF	R-1D-MRT LMS BF	% Reduction
8192	0.5270	0.1594	69	2.3698	0.4734	80
4096	0.1460	0.0476	67	0.4564	0.1371	70
2048	0.0480	0.0133	72	0.1359	0.0426	68
1024	0.0126	0.0048	62	0.0430	0.0160	63
512	0.0045	0.0024	46	0.0152	0.0049	68
256	0.0021	0.0014	33	0.0049	0.0025	48
128	0.0013	0.0011	15	0.0023	0.0016	30
64	0.0012	0.0007	42	0.0015	0.0013	13
32	0.0008	0.0006	25	0.0012	0.0012	0
16	0.0006	0.0005	16	0.0011	0.0011	0

Table 4.4 Comparison of computation time of LMS BF and R-1D-MRT LMS BF (seconds) [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $\mu = 0.005$ ]

N (Data size)	$N_s = 50$			$N_s = 100$		
	LMS BF	R-1D-MRT LMS BF	% Reduction	LMS BF	R-1D-MRT LMS BF	% Reduction
8192	15.5609	3.1118	80	32.5548	7.7344	76
4096	3.0514	0.4214	86	7.6684	1.6774	78
2048	0.3377	0.1034	69	1.5880	0.2497	84
1024	0.1117	0.0355	68	0.2299	0.0668	71
512	0.0348	0.0145	58	0.0614	0.0244	60
256	0.0136	0.0061	55	0.0228	0.0111	51
128	0.0059	0.0039	34	0.0101	0.0069	32
64	0.0035	0.0035	0	0.0060	0.0057	5
32	0.0032	0.0032	0	0.0047	0.0046	2
16	0.0031	0.0030	0	0.0045	0.0042	6

The computation time is calculated for simulation with source at 30 degrees and 2 interferences at 0 and -50 degrees and SINR of 30 dB. The simulations were performed with a computer using Intel Core i5 CPU working at 2.20 GHz and RAM of 4GB. Tables 4.3 and 4.4 indicate that the R-1D-MRT LMS beamformer offers more than 50 percent reduction in computation time for sampling sizes of 256 and above. The computational efficiency of the R-1D-MRT LMS beamformer is thus validated.

The next section focuses on the performance of the R-1D-MRT LMS beamformer in various communication channels.

#### 4.1.4 Performance of R-1D-MRT LMS Beamformer in Various Communication Channels

The performance of the R-1D-MRT LMS beamformer in AWGN, Rayleigh and Rician channels is analyzed in this section. One method of analysis is based on Euclidean distance measurement and the other is a visual one centered on scatterplots.

The Euclidean distance between the desired signal and the output of LMS and R-1D-MRT LMS beamformer is presented in Table 4.5.

Table 4.5 Comparison of Euclidian distance parameters of LMS BF and R-1D-MRT LMS BF for various communication channels for varying SINR [ $N_s = 6$ , Source angle  $0^\circ$ , Interference angles  $-20^\circ$  and  $40^\circ$ ,  $N = 512$ ,  $\mu = 0.01$ ]

SINR	AWGN		Rayleigh		Rician	
	LMS BF	R-1D-MRT LMS BF	LMS BF	R-1D-MRT LMS BF	LMS BF	R-1D-MRT LMS BF
10 dB	1.1908	1.1829	0.2611	0.2292	0.1526	0.1316
20 dB	1.6942	1.5306	1.4701	1.4540	0.6229	0.6195
30 dB	1.9363	1.8443	0.5959	0.5820	0.4613	0.4409

The Euclidean distance analysis of R-1D-MRT LMS beamformer in the communication channels indicate that the R-1D-MRT LMS beamformer has more proximity to the desired signal compared to LMS beamformer. Visual confirmation is obtained by comparing the scatterplots of the R-1D-MRT LMS beamformer and LMS beamformer for various communication channels in varying levels of noise plus interference. The scatterplots are represented in Fig 4.4 - 4.6.

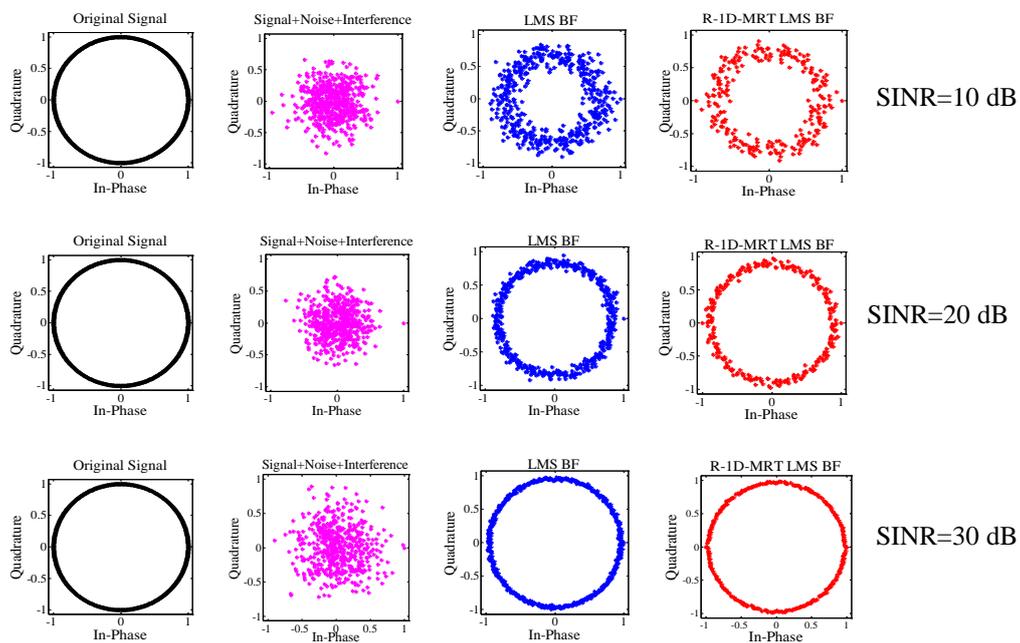


Fig 4.4 Scatterplots of R-1D-MRT LMS BF for AWGN channel for varying SINR [ $N_s = 6$ , Source angle =  $0^\circ$ , Interferences at  $-20^\circ$  and  $40^\circ$ ,  $N = 512$ ,  $\mu = 0.01$ ]

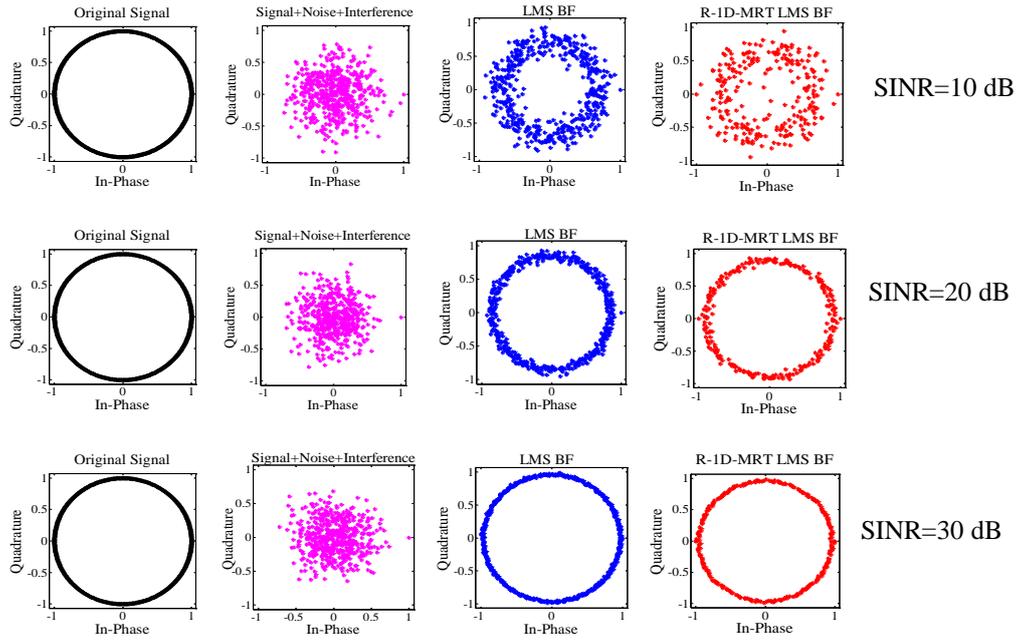


Fig 4.5 Scatterplots of R-1D-MRT LMS BF for Rayleigh channel for varying SINR [ $N_s = 6$ , Source angle =  $0^\circ$ , Interferences at  $-20^\circ$  and  $40^\circ$ ,  $N = 512$ ,  $\mu = 0.01$ ]

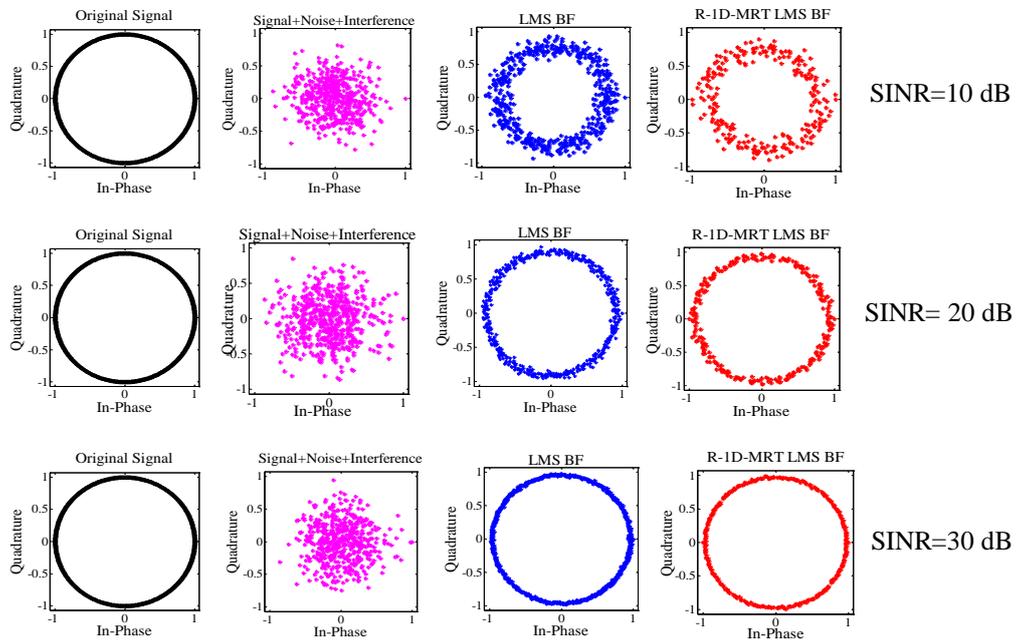


Fig 4.6 Scatterplots of R-1D-MRT LMS BF for Rician channel for varying SINR [ $N_s = 6$ , Source angle =  $0^\circ$ , Interferences at  $-20^\circ$  and  $40^\circ$ ,  $N = 512$ ,  $\mu = 0.01$ ]

The scatterplots of Fig 4.4 - 4.6 indicate that the R-1D-MRT LMS beamformer output is closer to the desired signal (original signal) than the LMS beamformer output. The dissimilarity between the beamformer outputs becomes less pronounced as SINR value increases.

The reduction in convergence rate, computational complexity, simulation time and better performance in communication channels for the R-1D-MRT LMS beamformer observed in the preceding sections validates the efficiency of the R-1D-MRT algorithm for use with LMS beamformers. The next section discusses the validation of the R-1D-MRT LMS beamformer using FPGA implementation.

#### **4.1.5 FPGA Implementation of R-1D-MRT LMS Beamformer**

For the FPGA implementation of R-1D-MRT LMS beamformer, XSG models of the training phase and beamforming phase is created. The optimum weights are computed in the training phase based on a training or pilot sequence. Once the optimum weights are computed, the received signal is multiplied with these weights in the beamforming phase. The XSG models designed for FPGA implementation of the R-1D-MRT LMS beamformer are presented in the subsequent sections.

##### **4.1.5.1 Xilinx System Generator model of R-1D-MRT LMS beamformer**

The Xilinx System Generator model for generating the source and interfering signals for the R-1D-MRT LMS beamformer is represented in Fig 4.7.

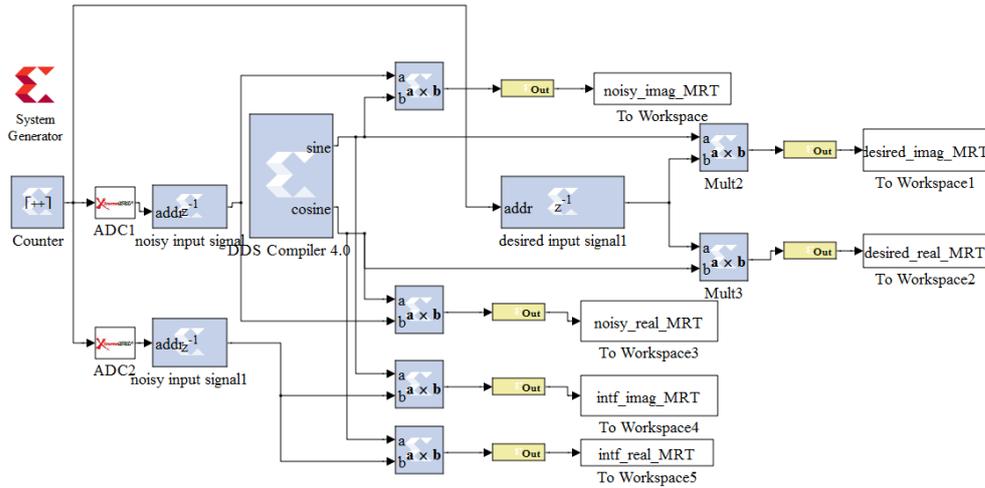


Fig 4.7 XSG model for generating desired and interfering signals of R-1D-MRT LMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu = 0.05$ ]

The Xilinx System Generator model of the training phase for estimating the optimum weights of the R-1D-MRT LMS beamformer is shown in Fig 4.8.

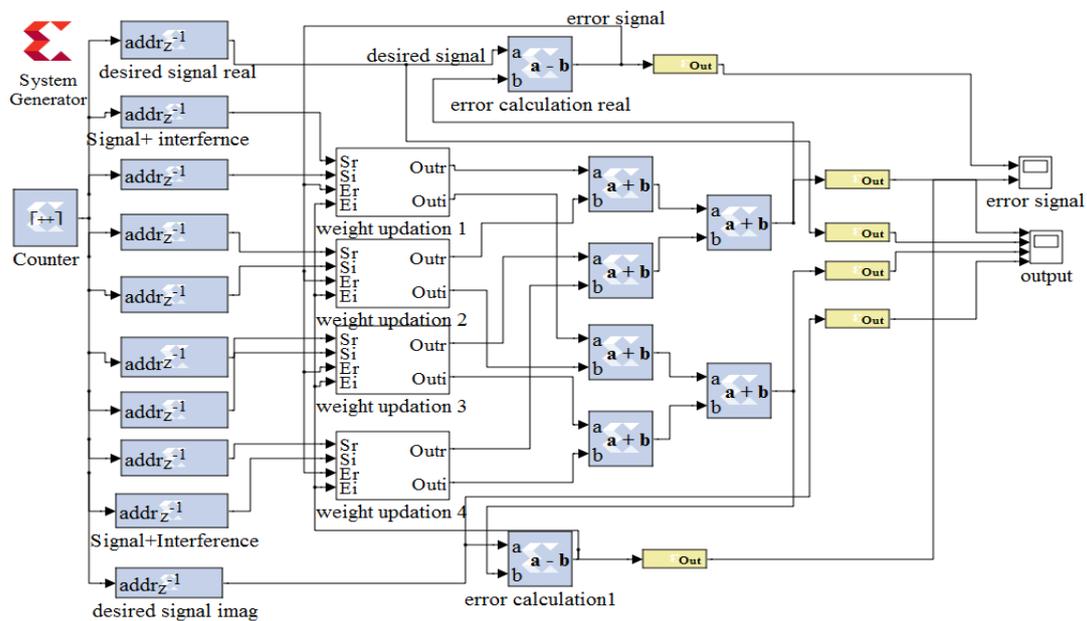


Fig 4.8 Training phase XSG model of R-1D-MRT LMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu = 0.05$ ]

The weight adaptation subsystem of the training model is illustrated in Fig 4.9.

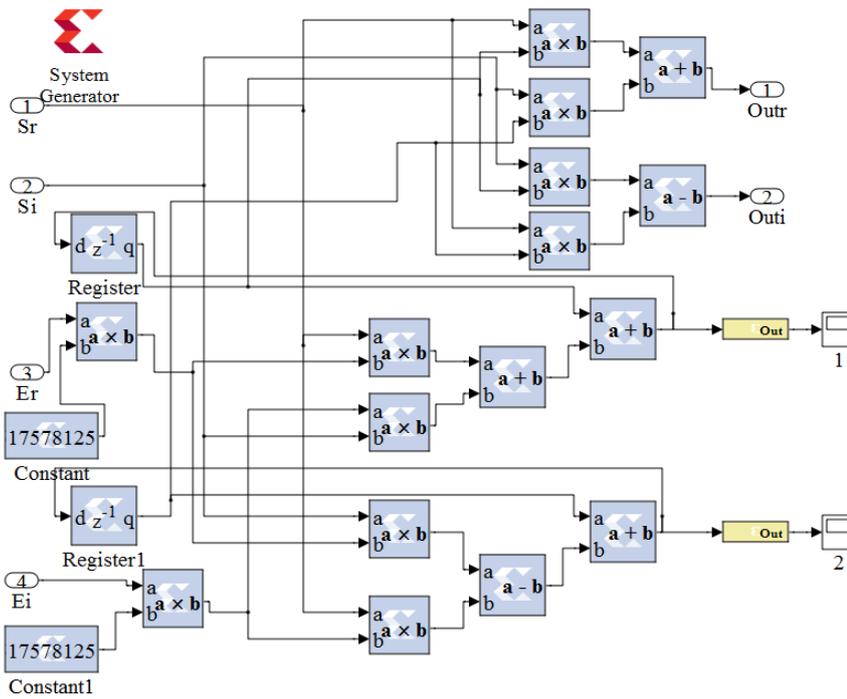


Fig 4.9 Weight adaptation subsystem of R-1D-MRT LMS BF training phase  
 $[N_s = 4, \text{Source angle} = 30^\circ, \text{Interference angle} = -50^\circ, N = 256, \text{SINR} = 30 \text{ dB}, \mu = 0.05]$

The model for obtaining the beamformer output of R-1D-MRT LMS beamformer is shown in Fig 4.10.

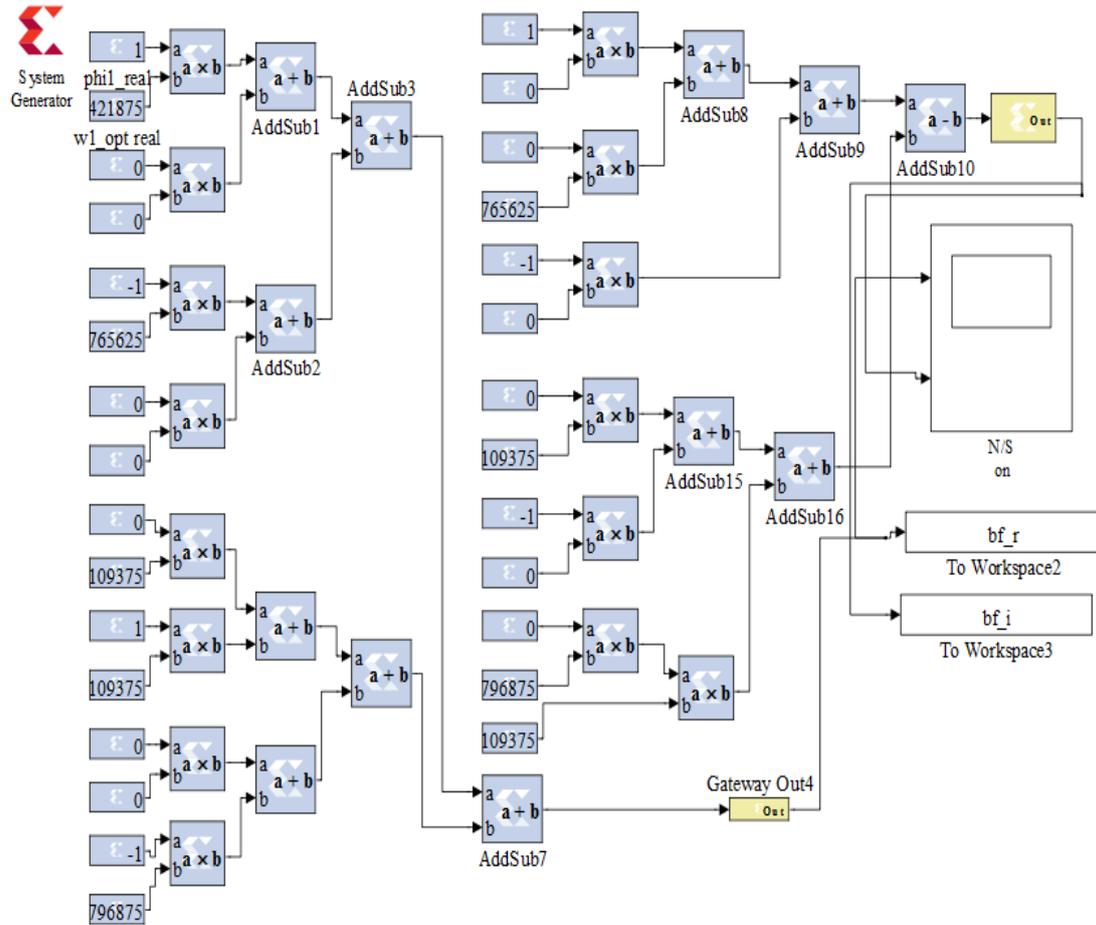


Fig 4.10 XSG model for obtaining beamformer output of R-1D-MRT LMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu = 0.05$ ]

#### 4.1.5.2 Beam pattern and beamformer output of XSG model R-1D-MRT LMS beamformer

The results of the simulation of XSG models of Fig 4.7- Fig 4.10 is presented in Fig 4.11 and 4.12.

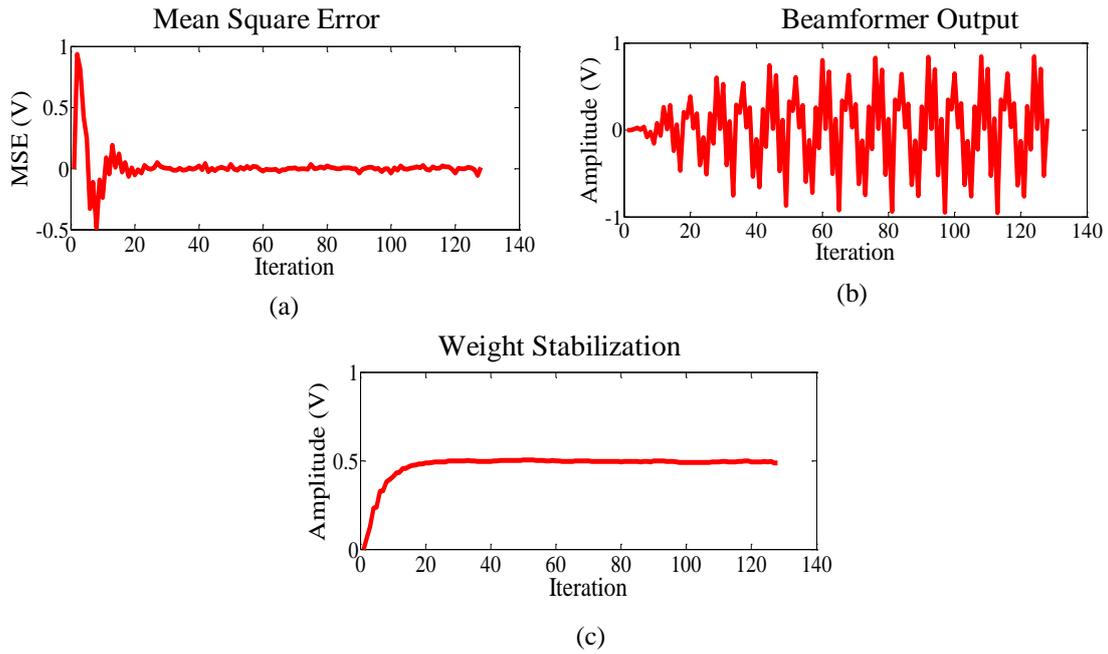


Fig 4.11 R-1D-MRT LMS beamformer output with XSG modelling [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu = 0.05$ ]

The mean square error signal is plotted in Fig 4.11(a). The mean square error value increases in the initial phase and then stabilizes after 30 iterations. The beamformer output illustrated in Fig 4.11(b) also stabilizes after 50 iterations. The convergence of the weight of the XSG model of R-1D-MRT LMS beamformer to a steady value in 30 iterations is shown in Fig 4.11(c).

The normalized beam pattern of the XSG simulation of R-1D-MRT LMS beamformer is shown in Fig 4.12(a). It exhibits a peak at the desired angle of  $30^\circ$  and a null at the interfering direction of  $-50^\circ$ . The XSG models created are then synthesized, built and implemented on FPGA. The beam pattern obtained for the FPGA implementation of R-1D-MRT LMS beamformer is shown in Fig 4.12(b).

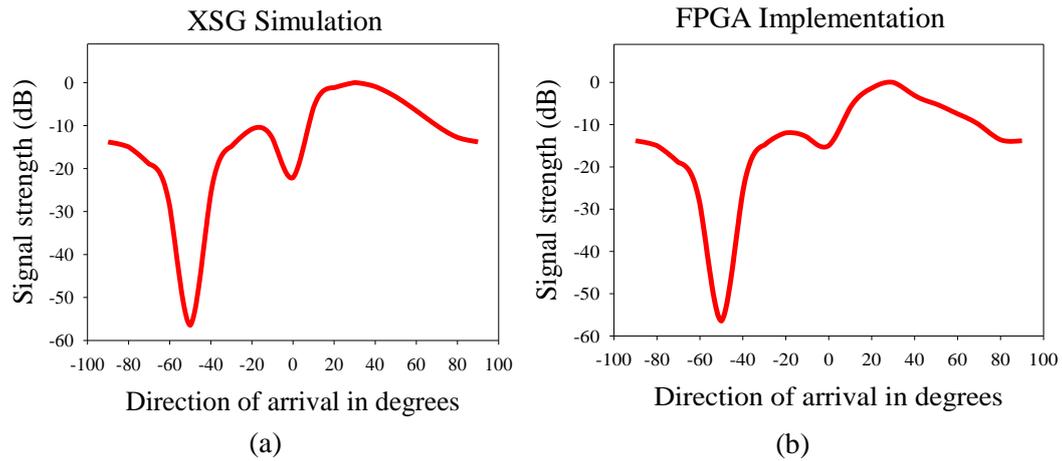


Fig 4.12 Normalized beam pattern of XSG simulation and FPGA implementation of R-1D-MRT LMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu = 0.05$ ]

The normalized beam pattern obtained in the FPGA implementation of R-1D-MRT LMS BF also displays the peak of the radiation pattern at the desired angle of  $30^\circ$  while simultaneously nulling the interference coming from  $-50^\circ$ . This demonstrates the competence of the R-1D-MRT LMS BF for beamforming. In the next section, the resource utilization of R-1D-MRT LMS BF and LMS BF are compared.

#### 4.1.5.3 FPGA resource utilization of R-1D-MRT LMS beamformer.

A comparison of the FPGA resource utilization for the R-1D-MRT LMS BF and LMS BF is presented in Table 4.6.

Table 4.6 Comparison of FPGA resources of LMS BF and R-1D-MRT LMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu = 0.05$ ]

Resources	N = 256		N = 1024	
	LMS BF	R-1D-MRT LMS BF	LMS BF	R-1D-MRT LMS BF
Slices	3444	3442	3446	3446
Flip Flops	4938	4926	4932	4930
Block RAM	22	12	64	32
Look Up Tables	5766	5762	5770	5768
Input Output blocks	400	400	400	400
Multipliers	164	164	160	160
Tristate Buffers	0	0	0	0

The FPGA resources required by the R-1D-MRT LMS beamformer are significantly lower than that utilized by the LMS beamformer.

Thus the efficiency of R-1D-MRT LMS beamformer is also validated for FPGA implementation. The next beamformer to be analysed is the Normalized Least Mean Square beamformer.

## 4.2 R-1D-MRT NLMS BEAMFORMER

The Least Mean Square (LMS) beamformer is popular for adaptive beamforming because of its simplicity and robustness but its convergence rate is slow (Widrow and Streams, 1985). The Normalized Least Mean Square (NLMS) beamformer overcomes the drawbacks of LMS beamformer and exhibits faster convergence with the same structure as LMS beamformer. The faster convergence is normally achieved by using a variable step size parameter, which is obtained by dividing a fixed step size parameter by the input vector power for all iterations (Jimaa et al., 2009). The overall

performance of NLMS beamformer is governed by the step size parameter. Faster convergence with large mean square error results when large step size is chosen, while slow convergence with small mean square error is obtained with small step size (Homana et al., 2011; Li and Tian, 2009; Lin et al., 2010; Huang and Lee, 2012; Shengkui et al., 2006; Shin et al., 2004).

In the training stage, the weights are calculated using the NLMS algorithm on the training data and once the optimum weights are arrived at, beamforming is performed using the optimum weights. In all iterations of the NLMS algorithm the weight vector  $\mathbf{W}$  is updated by using a gradient descent method which reduces the squared estimate error (Dai and Wang, 2010).

$$\mathbf{Y}(n) = \mathbf{W}^H \mathbf{X}(n) \quad (4.5)$$

Here  $\mathbf{Y}$  is the beamformer output,  $\mathbf{W}$  is the weight vector and  $\mathbf{X}$  is the input signal.

The weights are updated based on the relationship specified in equation 4.6.

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \mu \mathbf{X}(n) \mathbf{e}^*(n) \quad (4.6)$$

$\mu$  is a positive constant which functions as step size parameter in the NLMS algorithm.

$$\mu = \mu_1 / (\varepsilon + \|\mathbf{X}(n)\|^2) \quad (4.7)$$

where  $\varepsilon$  is a very small number added to avoid instability and  $\mu_1$  is a constant ( $0 < \mu_1 < 2$ ). The step size  $\mu$  varies adaptively by following the changes in the input signal level. This prevents the update weights from diverging and makes the algorithm more stable and faster converging than when a fixed step size is used. The schematic diagram of a conventional NLMS beamformer is shown in Fig 4.13(a).

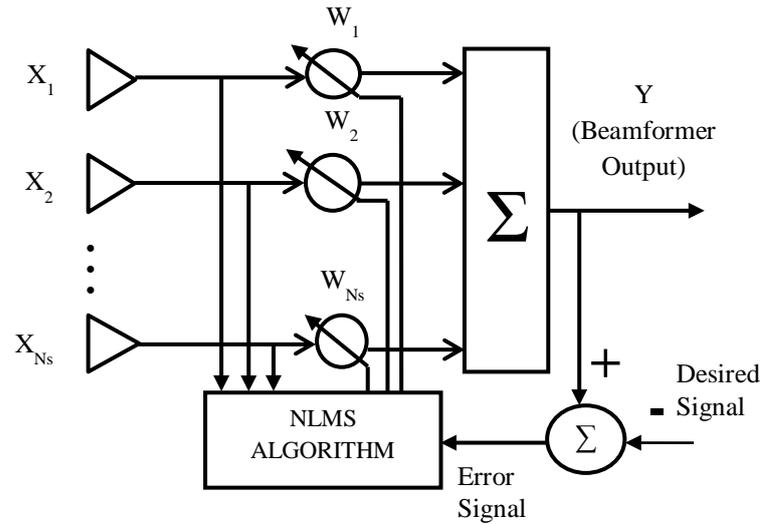


Fig 4.13(a) Schematic diagram of conventional NLMS beamformer

In the R-1D-MRT NLMS beamformer, the signal received at the array is transformed using the R-1D-MRT algorithm and the NLMS beamformer performs beamforming with this transformed data. The beamformer output is compared with a desired response and an estimation error is generated. The weights are automatically adjusted using the NLMS algorithm to minimize this estimation error and the adaptation process continues till the estimation error becomes zero.

The schematic diagram of the R-1D-MRT NLMS beamformer is illustrated in Fig 4.13(b).

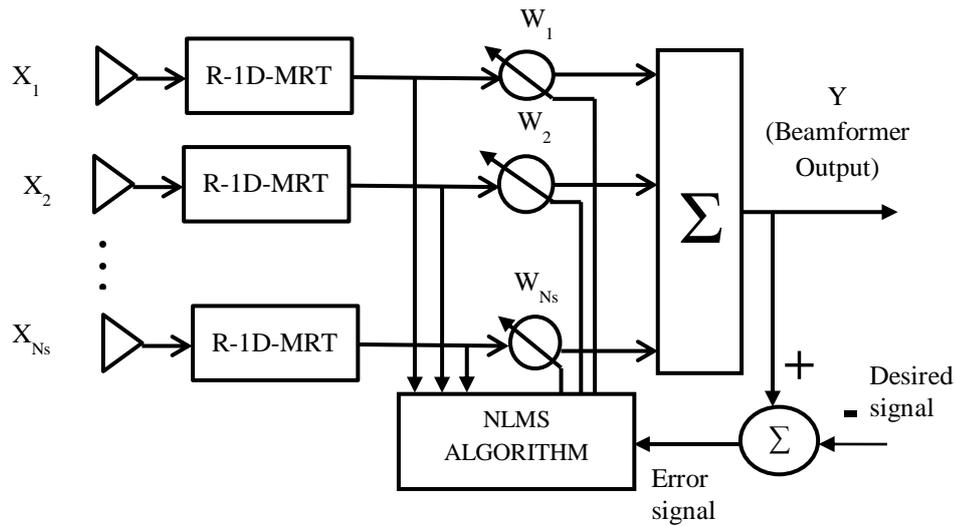


Fig 4.13(b) Schematic diagram of R-1D-MRT NLMS beamformer

The validation of the R-1D-MRT NLMS beamformer considers the MATLAB<sup>®</sup> simulations of beam pattern, beamformer output, convergence behaviour, computational complexity and performance in various communication channels. The next section examines the beam pattern and beamformer output of the R-1D-MRT NLMS beamformer.

#### 4.2.1 Beam Pattern and Beamformer Output of R-1D-MRT NLMS Beamformer

This section presents the normalized beam pattern and beamformer output of the R-1D-MRT NLMS beamformer for various locations of source and interferences and varying number of array elements. Fig 4.14 shows the simulation figures for the R-1D-MRT NLMS BF with 8 array elements, a source at  $30^\circ$ , interference angles at  $0^\circ$  and  $-50^\circ$  and SINR of 30 dB.

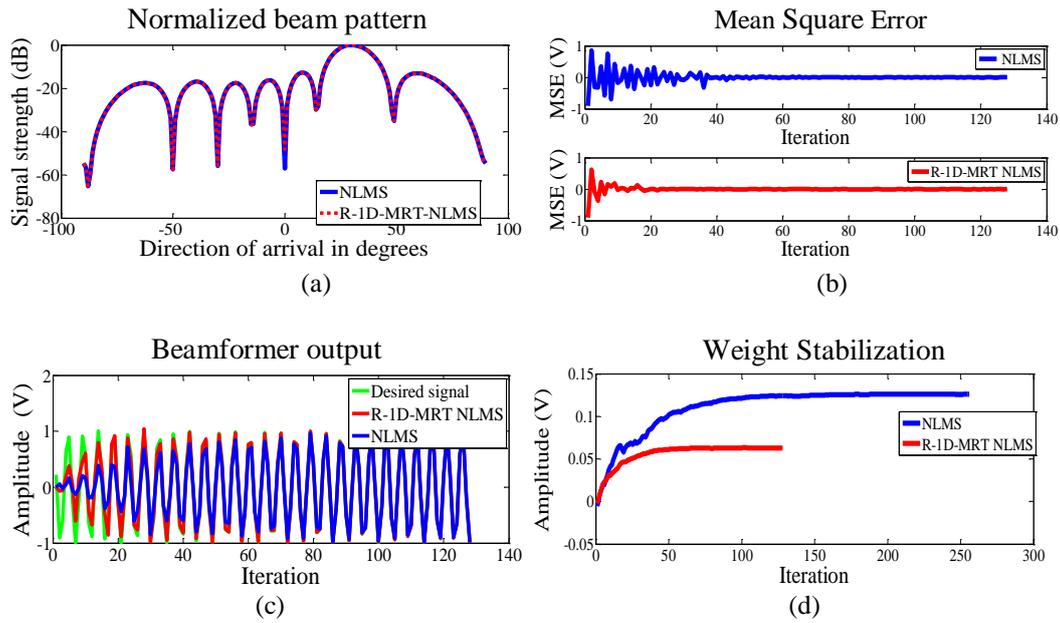


Fig 4.14 R-1D-MRT NLMS beamformer output [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angle  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30dB,  $\mu_1 = 0.008$ ]

The simulations are repeated for 12 antennas/sensors ( $N_s$ ) and different directions of source and interferences as depicted in Fig 4.15.

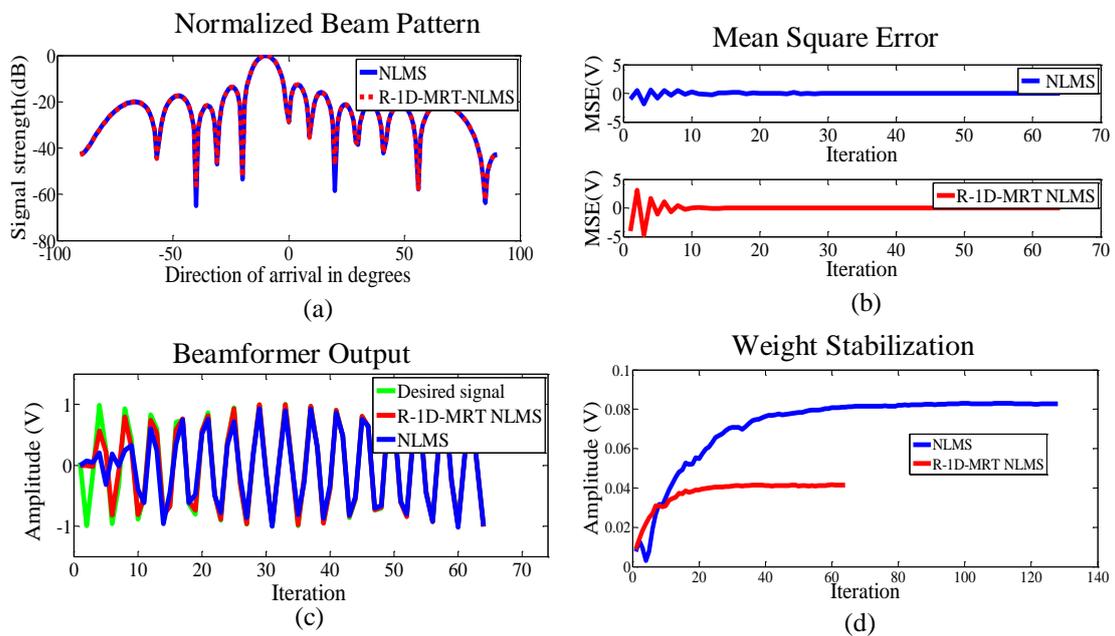


Fig 4.15 R-1D-MRT NLMS beamformer output [ $N_s = 12$ , Source angle  $-10^\circ$ , Interference angle  $20^\circ$  and  $-20^\circ$ ,  $N = 128$ , SINR = 30 dB,  $\mu_1 = 0.009$ ]

The normalized beam patterns exhibit a peak at the desired direction and nulls in the interfering directions as shown in Fig 4.14(a) and 4.15(a). The error signal of the R-1D-MRT NLMS beamformer also converges to zero quicker than NLMS beamformer as depicted in Fig 4.14(b) and 4.15(b). The output signal of the R-1D-MRT NLMS beamformer approaches the desired signal value much ahead of the R-1D-MRT NLMS beamformer as seen in Fig 4.14(c) and Fig 4.15(c). The convergence of the weights is more rapid for the R-1D-MRT NLMS beamformer compared to the NLMS beamformer as represented in Fig 4.14(d) and 4.15(d).

A comparison of the minimum number of iterations required by the R-1D-MRT NLMS beamformer and NLMS beamformer for convergence is presented in Table 4.7.

Table 4.7 Comparison of minimum number of iterations ( $I_{\min}$ ) for NLMS BF and R-1D-MRT NLMS BF for convergence [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 40 dB]

N (Data size)	$N_s = 4$			$N_s = 16$		
	$\mu_1$	NLMS BF ( $I_{\min}$ - NLMS)	R-1D-MRT NLMS BF ( $I_{\min}$ -R-1D-MRT- NLMS)	$\mu_1$	NLMS BF ( $I_{\min}$ -NLMS)	R-1D-MRT NLMS BF ( $I_{\min}$ -R-1D-MRT- NLMS)
8192	0.0005	2400	1000	0.0001	1320	630
4096	0.0010	1100	450	0.0003	740	380
2048	0.004	400	160	0.0007	400	150
1024	0.005	200	75	0.004	250	80
512	0.008	100	35	0.005	50	20
256	0.02	30	16	0.009	60	30
128	0.03	37	16	0.009	30	15

The R-1D-MRT NLMS beamformer achieves convergence with lesser numbers of iterations compared to NLMS beamformer. The next section discusses the computational complexity of the R-1D-MRT NLMS beamformer.

#### 4.2.2 Computational Complexity of R-1D-MRT NLMS Beamformer

The difference between the weight adaptation process of the NLMS and LMS beamformers is in the computation of the variable step size. For LMS beamformer the step size is fixed, while for NLMS beamformer the variable step size is obtained by normalizing the fixed step size with the power of the input signal. The variable step size calculation requires one complex multiplication and one division for a single iteration.

The weight adaptation stage of the NLMS algorithm with  $N_s$  antennas/sensors requires  $9N_s$  real multiplications and  $8N_s$  real additions for a single iteration along with a real division operation. The minimum number of iterations required for the NLMS beamformer is  $I_{\min\text{-NLMS}}$  while for the R-1D-MRT NLMS beamformer, the minimum number of iterations is  $I_{\min\text{-R-1D-MRT NLMS}}$  [Table 4.7]. Thus the required number of computations for NLMS beamformer in the weight adaptation stage is  $9N_s I_{\min\text{-NLMS}}$  real multiplications and  $8N_s I_{\min\text{-NLMS}}$  real additions while the R-1D-MRT NLMS beamformer requires  $9N_s I_{\min\text{-R-1D-MRT NLMS}}$  real multiplications and  $8N_s I_{\min\text{-R-1D-MRT NLMS}}$  real additions. For the R-1D-MRT NLMS beamformer, the conversion to R-1D-MRT requires  $2(N-1)$  real additions. For multiplication of the signals and weights, the NLMS beamformer requires  $N_s(4N-4)$  real multiplications and  $N_s(2N-2)$  real additions while the R-1D-MRT NLMS beamformer requires only  $N_s(2N-4)$  real multiplications and  $N_s(N-2)$  real additions. Totally, the NLMS

beamformer requires  $N_s(4N-4+9 I_{\min\text{-NLMS}})$  real multiplications and  $N_s(4N-4+8 I_{\min\text{-NLMS}})$  real additions and  $I_{\min\text{-NLMS}}$  real divisions. The R-1D-MRT NLMS beamformer requires  $N_s(2N-4+9 I_{\min\text{-R-1D-MRT NLMS}})$  real multiplications and  $N_s(4N-4+8 I_{\min\text{-R-1D-MRT NLMS}})$  real additions and  $I_{\min\text{-R-1D-MRT NLMS}}$  real divisions.

The computational complexity of NLMS and R-1D-MRT NLMS beamformer for 4 antennas/sensors ( $N_s$ ) and 16 antennas/sensors ( $N_s$ ) for minimum number of iterations is represented in Tables 4.8 and 4.9 respectively.

Table 4.8 Comparison of computational complexity of NLMS BF and R-1D-MRT NLMS BF ( $N_s = 4$ )

N (Data size)	Number of real multiplications		Number of real additions		No of real divisions	
	NLMS BF	R-1D-MRT NLMS BF	NLMS BF	R-1D-MRT NLMS BF	NLMS BF	R-1D-MRT NLMS BF
8192	217456	101520	207856	163056	2400	1000
4096	105120	48952	100720	79920	1100	450
2048	47152	22128	45552	37872	400	160
1024	23568	10876	22768	18768	200	75
512	11776	5340	11376	9296	100	35
256	5160	2608	5040	4592	30	16
128	3364	1584	3216	2544	37	16

Table 4.9 Comparison of computational complexity of NLMS BF and R-1D-MRT NLMS BF ( $N_s = 16$ )

N (Data size)	Number of real multiplications		Number of real additions		No of real divisions	
	NLMS BF	R-1D-MRT NLMS BF	NLMS BF	R-1D-MRT NLMS BF	NLMS BF	R-1D-MRT NLMS BF
8192	714304	352800	693184	606108	1320	630
4096	368640	185728	356800	311464	740	380
2048	188608	87072	182208	150492	400	150
1024	101472	44224	97472	75856	250	80
512	39904	19200	39104	35288	50	20
256	24960	12448	24000	20204	60	30
128	12448	6192	11968	10062	30	15

The computational efficiency of the R-1D-MRT NLMS beamformer is thus validated. The computation time of the R-1D-MRT NLMS beamformer should also be considerably lower as the number of real multiplications and real additions is reduced in the R-1D-MRT NLMS beamformer. The effect of the reduction in computational complexity on the computation time of R-1D-MRT NLMS beamformer is analyzed in the next section.

### 4.2.3 Computation Time of R-1D-MRT NLMS Beamformer

To further validate the efficiency of the R-1D-MRT NLMS beamformer, the computation time of the R-1D-MRT NLMS beamformer and the NLMS beamformer for varying number of antennas/sensors ( $N_s$ ) and data sizes ( $N$ ) are compared in Tables 4.10 and 4.11.

Table 4.10 Comparison of computation time of NLMS BF and R-1D-MRT NLMS BF (seconds) [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $\mu_1 = 0.03$ ,  $\varepsilon = 0.05$ ]

N (Data size)	$N_s = 4$			$N_s = 16$		
	NLMS BF	R-1D-MRT NLMS BF	% Reduction	NLMS BF	R-1D-MRT NLMS BF	% Reduction
8192	0.4746	0.1493	68	3.0018	0.4412	85
4096	0.1474	0.0486	67	0.4449	0.1335	70
2048	0.0471	0.0138	70	0.1317	0.0406	69
1024	0.0139	0.0055	60	0.0417	0.0143	65
512	0.0053	0.0026	51	0.0146	0.0052	64
256	0.0027	0.0015	44	0.0050	0.0026	48
128	0.0016	0.0010	38	0.0024	0.0016	33
64	0.0011	0.0008	27	0.0018	0.0013	27
32	0.0009	0.0007	22	0.0014	0.0011	21
16	0.0008	0.0006	25	0.0011	0.0011	0

Table 4.11 Comparison of computation time of NLMS and R-1D-MRT NLMS BF (seconds) [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $\mu_1 = 0.03$ ,  $\varepsilon = 0.05$ ]

N (Data size)	$N_s = 50$			$N_s = 100$		
	NLMS BF	R-1D-MRT NLMS BF	% Reduction	NLMS BF	R-1D-MRT NLMS BF	% Reduction
8192	15.6481	3.3926	78	32.1719	7.9414	75
4096	3.3301	0.3817	88	7.8994	1.7826	77
2048	0.4012	0.1035	74	1.6869	0.2335	86
1024	0.1032	0.0334	67	0.2359	0.0635	73
512	0.0357	0.0139	61	0.0602	0.0219	63
256	0.0124	0.0065	47	0.0202	0.0109	46
128	0.0064	0.0047	27	0.0101	0.0076	25
64	0.0034	0.0031	9	0.0069	0.0047	32
32	0.0031	0.0030	0.03	0.0051	0.0046	10
16	0.0025	0.0023	0.08	0.0040	0.0039	0.02

The computation time is calculated for a simulation with source at 30 degrees and 2 interferences at 0 and -50 degrees and SINR of 30 dB. The simulations were performed with a computer using Intel Core i5 CPU working at 2.20 GHz and RAM of 4GB. The R-1D-MRT NLMS beamformer offers a considerable reduction in computation time over that of NLMS beamformer when the data sizes (N) are 256 and above.

The next section discusses the performance of the R-1D-MRT NLMS beamformer in various communication channels.

#### 4.2.4 Performance of R-1D-MRT NLMS Beamformer in Various Communication Channels

An analysis of the performance of the R-1D-MRT NLMS beamformer in AWGN, Rayleigh and Rician channels, based on Euclidean distance measurement and scatterplots is presented in this section.

Table 4.12 displays the Euclidean distance between the original signal and the output of NLMS and R-1D-MRT NLMS beamformers.

Table 4.12 Comparison of Euclidian distance parameters of NLMS BF and R-1D-MRT NLMS BF for various communication channels for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.03$ ,  $\varepsilon = 0.005$ ]

SINR	AWGN		Rayleigh		Rician	
	NLMS BF	R-1D-MRT NLMS BF	NLMS BF	R-1D-MRT NLMS BF	NLMS BF	R-1D-MRT NLMS BF
10 dB	0.9974	0.9166	2.6695	2.1407	2.0394	1.9054
20 dB	0.2886	0.2087	1.3205	1.0082	1.8601	1.8140
30 dB	0.0595	0.0549	0.8360	0.6355	1.1216	1.0397

The R-1D-MRT NLMS beamformer output is more similar than the NLMS beamformer output to the desired signal as indicated by the Euclidean distance analysis in various communication channels. A visual evaluation is performed by comparing the scatterplots of the R-1D-MRT NLMS beamformer and NLMS beamformer for various communication channels for varying SINR. The scatterplots are represented in Fig 4.16 - 4.18.

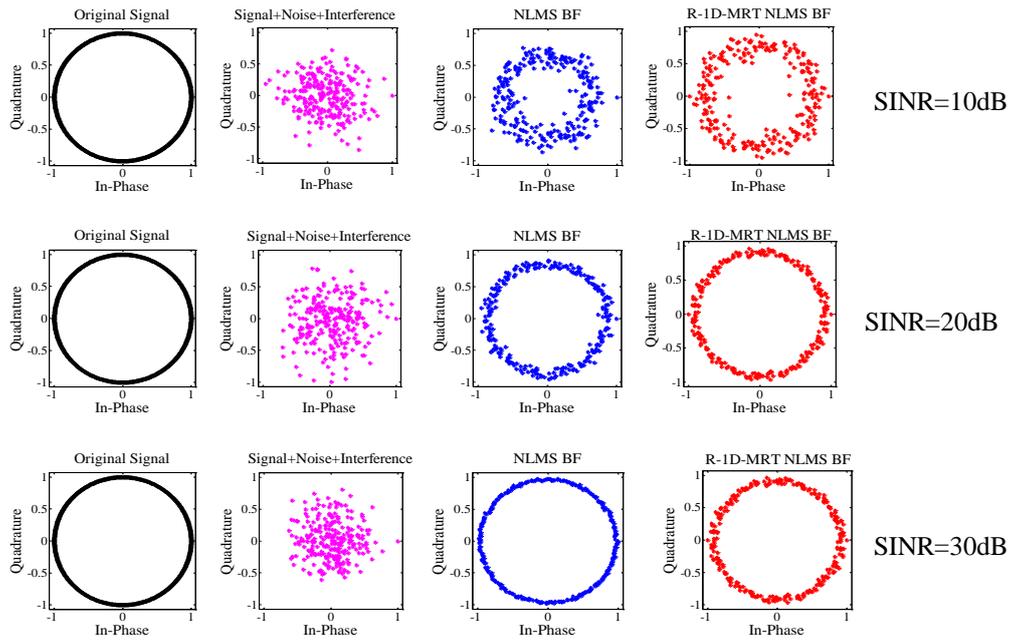


Fig 4.16 Scatterplots of R-1D-MRT NLMS BF for AWGN channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.03$ ,  $\varepsilon = 0.005$ ]

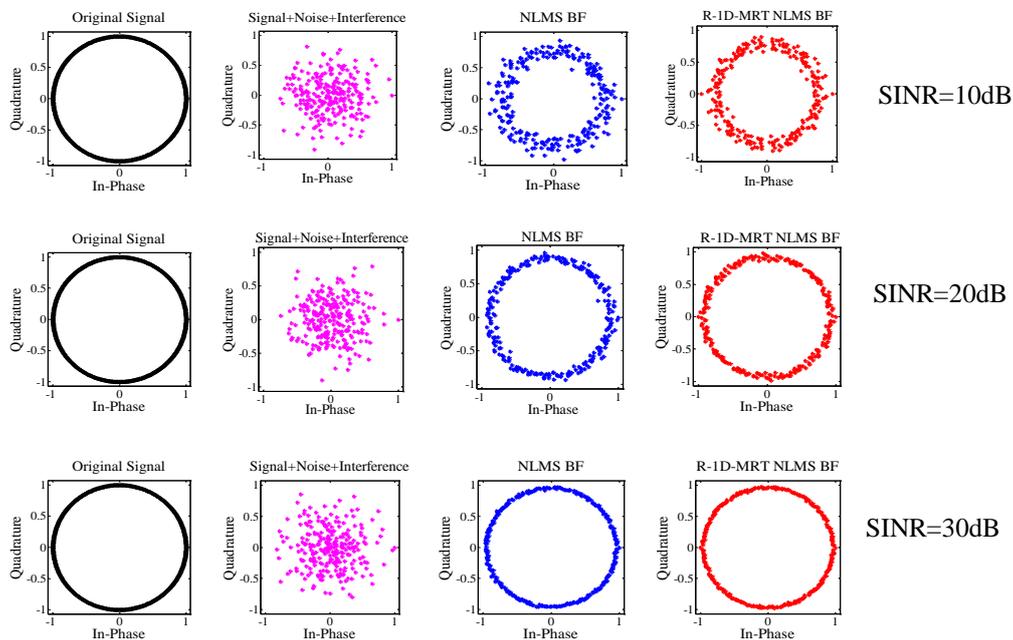


Fig 4.17 Scatterplots of R-1D-MRT NLMS BF for Rayleigh channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.03$ ,  $\varepsilon = 0.005$ ]

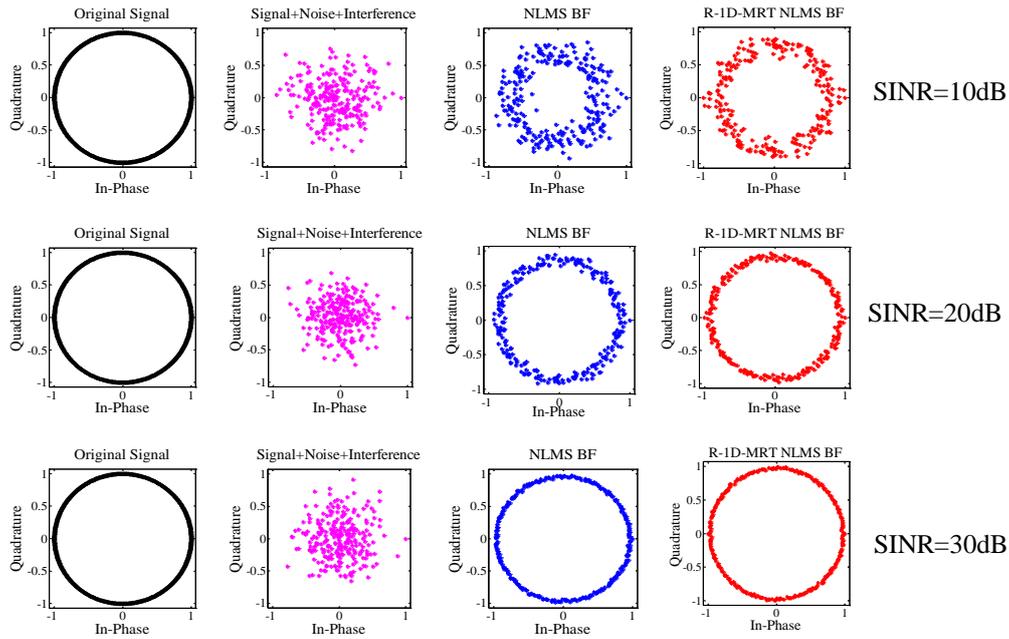


Fig 4.18 Scatterplots of R-1D-MRT NLMS BF for Rician channel for varying SINR [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angles =  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.03$ ,  $\varepsilon = 0.005$ ]

The scatterplots of Fig 4.16-4.18 indicate that the R-1D-MRT NLMS beamformer output has more proximity than the NLMS beamformer output to the desired (original) signal.

The parameters like convergence rate, computational complexity, computation time and performance in communication channels evaluated for the beamformer in the preceding sections indicate that the R-1D-MRT NLMS beamformer exhibits better performance compared to NLMS beamformer. This validates the efficiency of the R-1D-MRT algorithm for use with NLMS beamformers. The next section discusses the FPGA implementation of the R-1D-MRT NLMS beamformer.

## 4.2.5 FPGA Implementation of R-1D-MRT NLMS Beamformer

The optimum weights of the R-1D-MRT NLMS beamformer is computed in the training phase using the training or pilot sequence. The received signal is multiplied with the optimum weights for beamforming. The Xilinx System Generator models designed for FPGA implementation of the R-1D-MRT NLMS beamformer is illustrated in the following sections.

### 4.2.5.1 Xilinx System Generator model of R-1D-MRT NLMS beamformer

The Xilinx System Generator model of the R-1D-MRT NLMS beamformer is similar to the model of R-1D-MRT LMS beamformer except for the weight adaptation block. The Xilinx System Generator model for generating the source and interfering signals for the R-1D-MRT NLMS beamformer is shown in Fig 4.19.

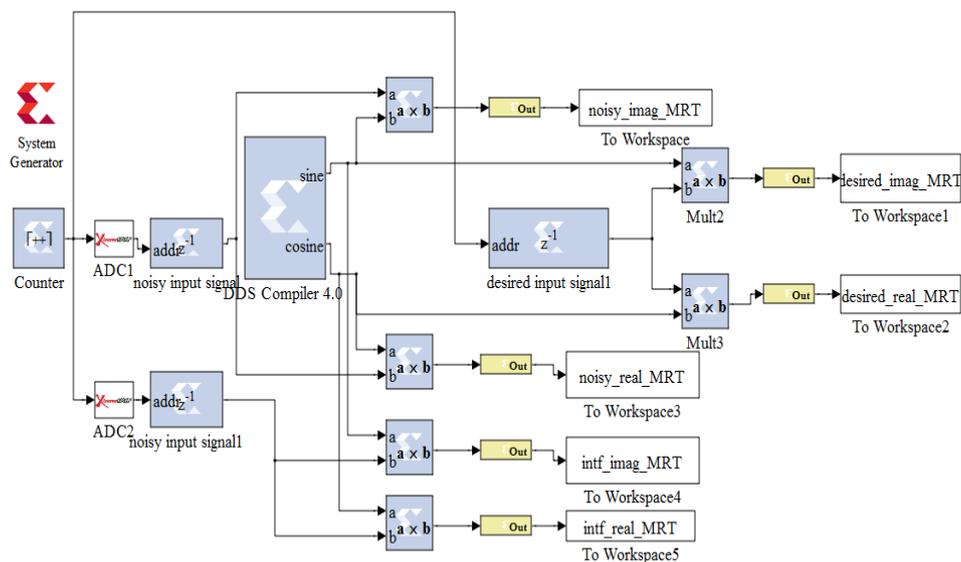


Fig 4.19 XSG Model of source and interfering signal generation of R-1D-MRT NLMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.05$ ,  $\varepsilon = 0.005$ ]

The training model for estimating the optimum weights of the R-1D-MRT NLMS beamformer is shown in Fig 4.20.

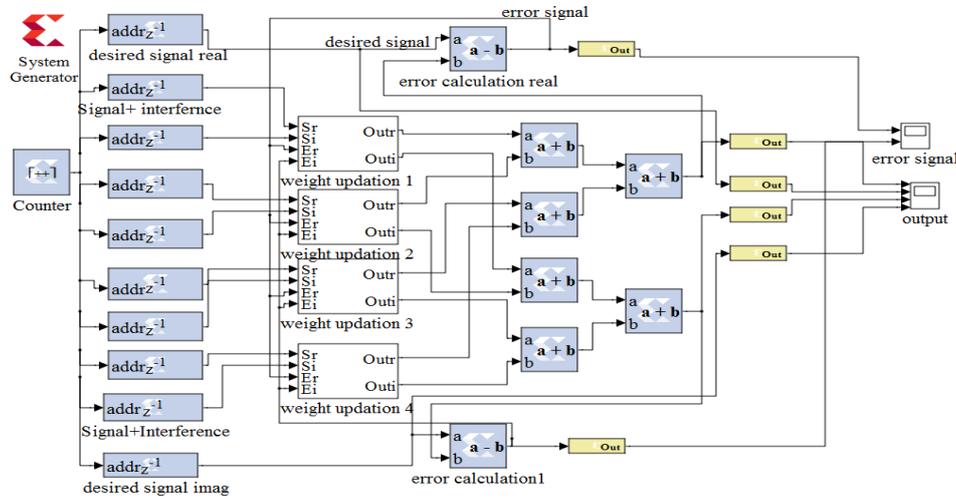


Fig 4.20 Training model for R-1D-MRT NLMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.05$ ,  $\varepsilon = 0.005$ ]

Fig 4.21 presents the weight adaptation subsystem of the training model of R-1D-MRT NLMS beamformer.

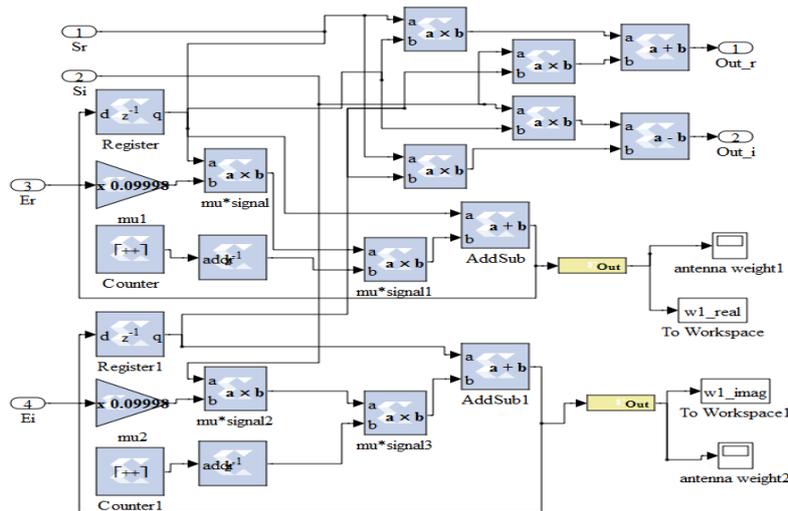


Fig 4.21 Weight adaptation subsystem for training model of R-1D-MRT NLMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.05$ ,  $\varepsilon = 0.005$ ]

The output of R-1D-MRT NLMS beamformer is obtained using the model represented in Fig 4.22.

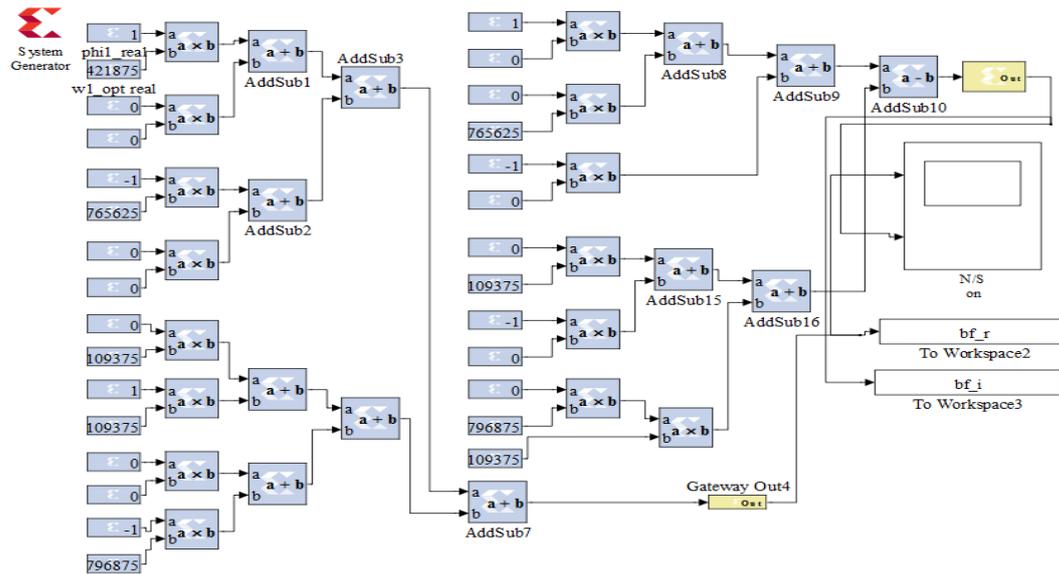


Fig 4.22 XSG model for obtaining output of R-1D-MRT NLMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.05$ ,  $\varepsilon = 0.005$ ]

The XSG models shown in Fig 4.19 - 4.22 are simulated using Altium Designer<sup>®</sup> and the results obtained are presented in the next section.

#### 4.2.5.2 Beam pattern and beamformer output of XSG model of R-1D-MRT NLMS beamformer

The normalized beam pattern, beamformer output, error signal and weights of the R-1D-MRT NLMS beamformer obtained after simulation of its XSG models are presented in Fig 4.23 and Fig 4.24.

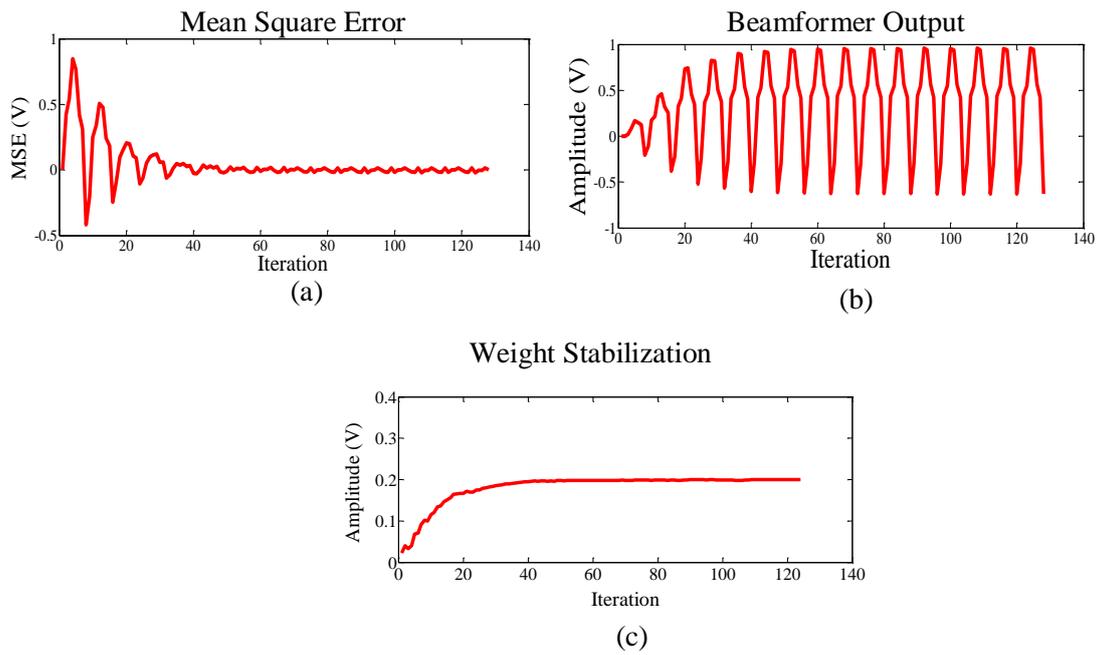


Fig 4.23 R-1D-MRT NLMS BF output with XSG modelling [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.05$ ,  $\varepsilon = 0.005$ ]

In 40 iterations, the mean square error converges to zero as shown in Fig 4.23(a). The beamformer output of Fig 4.23(b) illustrates that the output signal approaches the maximum value in 60 iterations. The weights also stabilize at a steady value in 40 iterations as depicted in Fig 4.23(c). The normalized beam pattern of the XSG simulation of R-1D-MRT NLMS beamformer is represented in Fig 4.24(a). After synthesizing the XSG models of R-1D-MRT NLMS beamformer using Xilinx ISE<sup>®</sup> and Altium Designer<sup>®</sup>, it is built and implemented on the Altium NB3000 Nanoboard. The normalized beam pattern obtained for the FPGA implementation is shown in Fig 4.24(b). Both the beam patterns exhibit a peak at the desired angle of  $30^\circ$  and null at interference angle of  $-50^\circ$ .

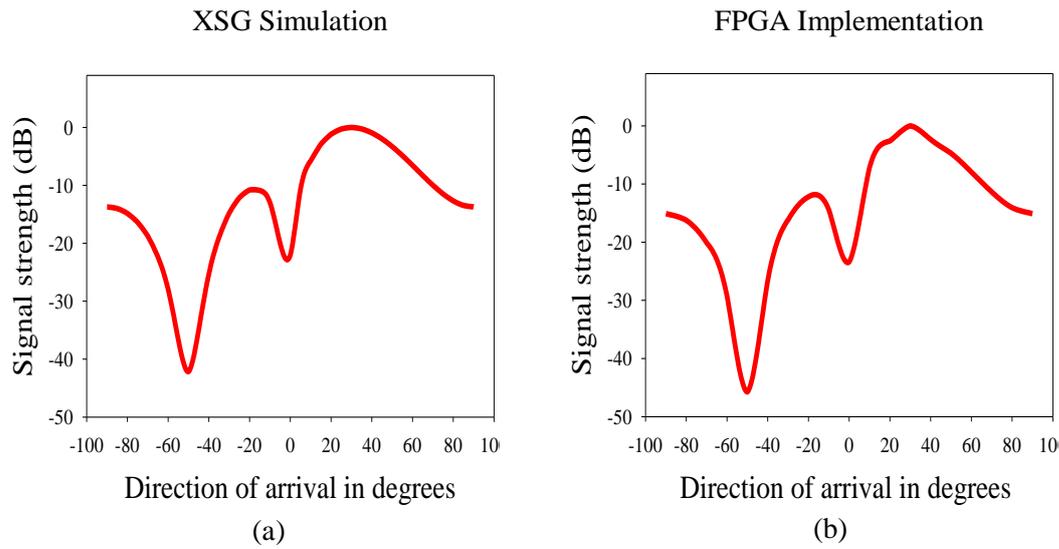


Fig 4.24 Normalized beam pattern of XSG simulation and FPGA implementation of R-1D-MRT NLMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.05$ ,  $\varepsilon = 0.005$ ]

The normalized beam pattern of the FPGA implementation also exhibits a peak in the direction of the desired signal of  $30^\circ$  and reduced output at interference angle of  $-50^\circ$ , thus proving the competence of the R-1D-MRT NLMS beamformer for adaptive beamforming. The next section discusses the FPGA resource utilization of the R-1D-MRT NLMS beamformer.

#### 4.2.5.3 FPGA resource utilization of R-1D-MRT NLMS beamformer

The FPGA resource utilization of the R-1D-MRT NLMS beamformer is presented in Table 4.13.

Table 4.13 Comparison of FPGA resources of NLMS BF and R-1D-MRT NLMS BF [ $N_s = 4$ , Source angle =  $30^\circ$ , Interference angle =  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\mu_1 = 0.05$ ,  $\varepsilon = 0.005$ ]

Resources	N = 256		N = 1024	
	NLMS BF	R-1D-MRT NLMS BF	NLMS BF	R-1D-MRT NLMS BF
Slices	1510	1508	1512	1502
Flip Flops	1769	1767	1773	1771
Block RAM	26	13	78	39
Look Up Tables	2675	2671	2705	2677
Input Output blocks	392	392	392	392
Multipliers	84	84	84	84
Tristate Buffers	0	0	0	0

### 4.3 CHAPTER SUMMARY

The computational efficiency of the R-1D-MRT LMS and R-1D-MRT NLMS beamformers has been validated by comparing the computational complexity and computation time of these beamformers with conventional LMS and NLMS beamformers. Faster convergence exhibited by the R-1D-MRT LMS and R-1D-MRT NLMS beamformers confirms the advantage offered by the R-1D-MRT algorithm for beamforming. The performance of the R-1D-MRT LMS and R-1D-MRT NLMS beamformers is enhanced in various communication channels which are also validated by measurements of the Euclidean distance parameters. Simulations of the XSG models and the FPGA implementation further endorse the suitability of the R-1D-MRT LMS and R-1D-MRT NLMS beamformers.

The efficiency of the R-1D-MRT algorithm for beamforming is validated on 3 more beamformers, namely Recursive Least Square, Sample Matrix Inversion and Generalized Sidelobe Canceler in the next chapter.

## CHAPTER 5

### R-1D-MRT RLS, SMI AND GSC BEAMFORMERS

CONTENTS	☆ R-1D-MRT RLS beamformer	☆ R-1D-MRT SMI beamformer	☆ R-1D-MRT GSC beamformer
	<ul style="list-style-type: none"><li>◆ Beam pattern and beamformer output</li><li>◆ Computational complexity</li><li>◆ Computation time</li><li>◆ Performance in various communication channels</li></ul>	<ul style="list-style-type: none"><li>◆ Beam pattern and beam output</li><li>◆ Computational complexity</li><li>◆ Computation time</li><li>◆ Performance in various communication channels</li></ul>	<ul style="list-style-type: none"><li>◆ Beam pattern and beam output</li><li>◆ Computational complexity</li><li>◆ Computation time</li><li>◆ Performance in various communication channels</li></ul>

The previous chapter had demonstrated the efficiency of the R-1D-MRT algorithm for beamforming with LMS and NLMS beamformers. This chapter validates its efficiency on Recursive Least Square (RLS), Sample Matrix Inversion (SMI) and Generalized Sidelobe Canceller (GSC) beamformers. The use of recursive equations in RLS beamformer ensures faster convergence but at the cost of higher computational complexity. The Sample Matrix Inversion beamformer achieves fast convergence by applying a block adaptive method. The Generalized Sidelobe Canceller beamformer is a multiple constrained version of the LCMV beamformer. In the following section, the validation of R-1D-MRT RLS beamformer is presented.

#### 5.1 R-1D-MRT RLS BEAMFORMER

The weights of the RLS beamformer are adapted using the RLS algorithm. This algorithm approximates the Weiner solution by directly using the method of least squares to adjust the weight vector. Compared to Least Mean Square Algorithm, the RLS algorithm exhibits faster convergence but has higher computational complexity.

The weights are updated (Simon Haykin, 2012; Lei Wang, 2009) based on the equations 5.1-5.5.

The initial conditions are set as  $\mathbf{W}(0) = \mathbf{0}$ ,  $\mathbf{P}(0) = \delta^{-1}\mathbf{I}$ , where  $\mathbf{W}$  is the weight vector,  $\mathbf{P}$  is a  $N_s \times N_s$  inverse correlation matrix for an array of  $N_s$  elements,  $\mathbf{I}$  is a  $N_s \times N_s$  identity matrix and  $\delta$  is a positive real number called regularization parameter. For each sample  $n = 1, 2, \dots, N$ , the following equations are computed

$$\boldsymbol{\pi}(n) = \mathbf{P}(n-1)\mathbf{X}(n) \quad (5.1)$$

where  $\mathbf{X}(n)$  is the input signal.

$$\mathbf{k}(n) = \boldsymbol{\pi}(n)/(\zeta + \mathbf{X}^H(n)\boldsymbol{\pi}(n)) \quad (5.2)$$

where  $\zeta$  is the forgetting factor which is a positive constant close to but less than unity.

$$\boldsymbol{\xi}(n) = \mathbf{d}(n) - \mathbf{W}^H(n-1)\mathbf{X}(n) \quad (5.3)$$

where  $\mathbf{d}(n)$  is the desired signal and  $\mathbf{W}(n)$  is the weight vector.

$$\mathbf{W}(n) = \mathbf{W}(n-1) + \mathbf{k}(n)\boldsymbol{\xi}^*(n) \quad (5.4)$$

$$\mathbf{P}(n) = \zeta^{-1}\mathbf{P}(n-1) - \zeta^{-1}\mathbf{k}(n)\mathbf{X}^H(n)\mathbf{P}(n-1) \quad (5.5)$$

The schematic diagram of a conventional RLS beamformer is shown in Fig 5.1(a).

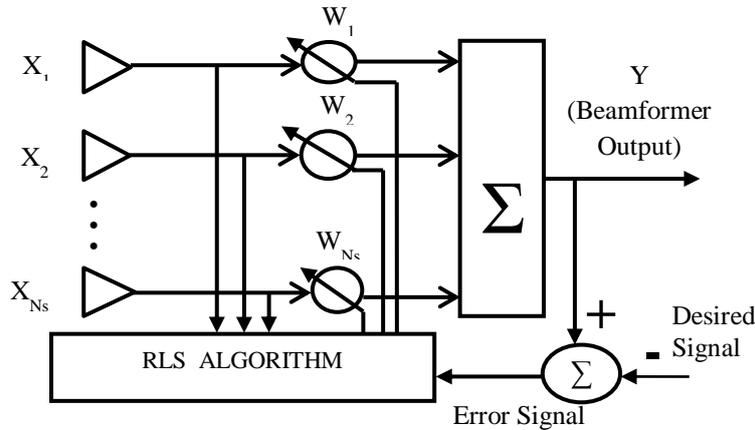


Fig 5.1(a) Schematic diagram of conventional RLS beamformer

In R-1D-MRT RLS beamformer, the R-1D-MRT algorithm is applied to the signals arriving at the array inputs. Beamforming is then performed on the transformed data.

Fig 5.1(b) represents the schematic diagram of R-1D-MRT RLS beamformer.

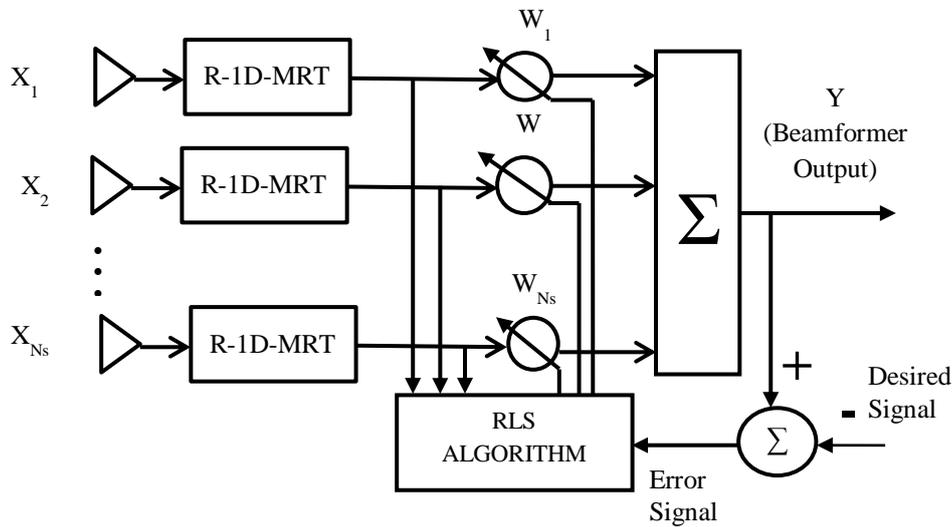


Fig 5.1(b) Schematic diagram of R-1D-MRT RLS beamformer

The analysis of the beam pattern and beamformer output of the R-1D-MRT RLS beamformer is presented in the following section.

### 5.1.1 Beam Pattern and Beamformer Output of R-1D-MRT RLS Beamformer

The desired beam pattern should exhibit nulls in the direction of interfering signals and a peak in the direction of the desired signal. The normalized beam pattern, MSE, beamformer output and weight stabilization of the R-1D-MRT RLS beamformer obtained with MATLAB<sup>®</sup> simulations for varying parameters are illustrated in Fig 5.2 and Fig 5.3.

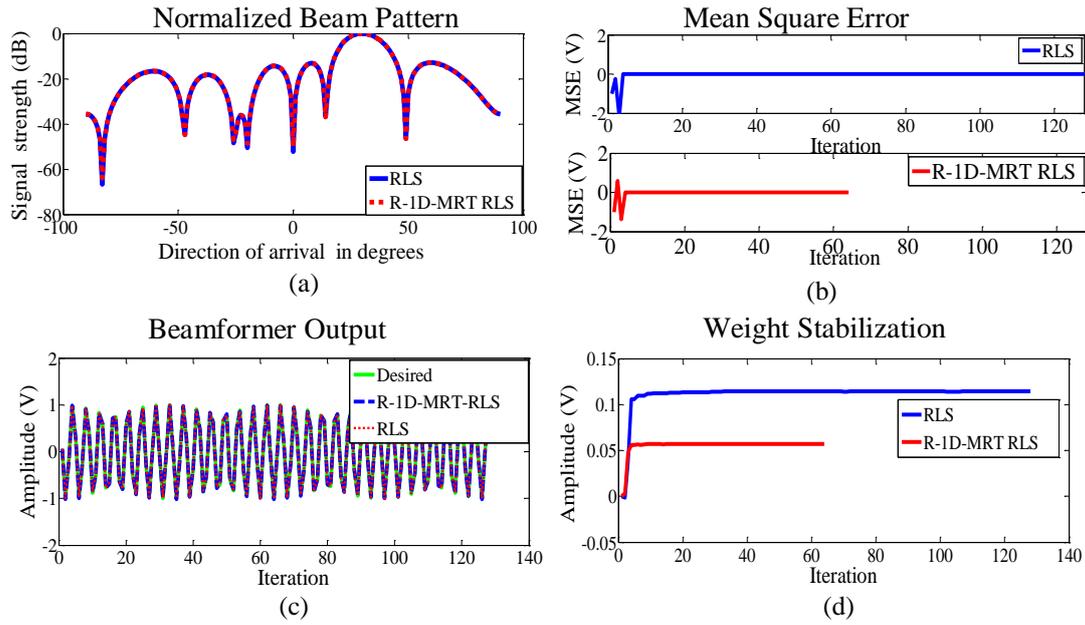


Fig 5.2 R-1D-MRT RLS beamformer output [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 128$ , SINR = 30 dB,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

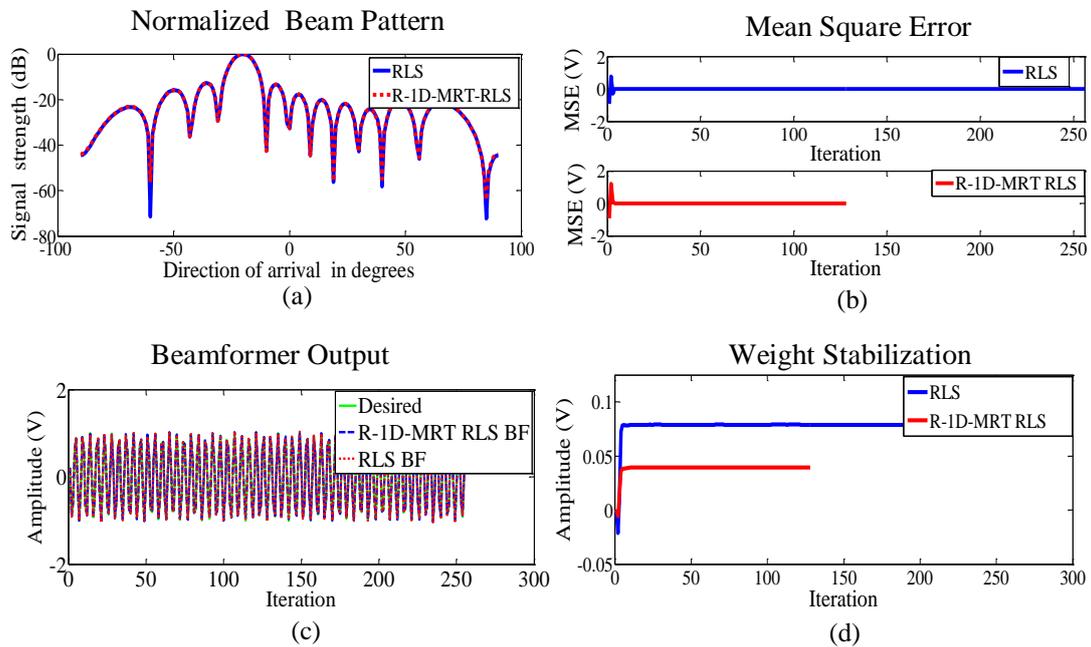


Fig 5.3 R-1D-MRT RLS beamformer output [ $N_s = 12$ , Source angle  $-20^\circ$ , Interference angles  $40^\circ$  and  $-60^\circ$ ,  $N = 256$ , SINR = 30 dB,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

The normalized beam patterns of Fig 5.2(a) and 5.3(a) exhibit the desired pattern with main lobes in the direction of the desired signal and nulls in the direction of the interfering signals. The rapid convergence characteristic of the RLS algorithm is evident in Fig 5.2(b,d) and Fig 5.3(b,d) which show that within a few iterations, the error signal converges to zero and the weights stabilize to a steady value for both the R-1D-MRT RLS beamformer and RLS beamformer. Fig 5.2(c) and Fig 5.3(c) show that the beamformer output of both RLS beamformer and R-1D-MRT RLS beamformer, track the desired signal almost instantaneously.

The minimum number of iterations required for convergence by the RLS beamformer and R-1D-MRT RLS beamformer for varying data sizes ( $N$ ) and varying number of antennas/sensors ( $N_s$ ) is presented in Table 5.1.

Table 5.1 Comparison of minimum number of iterations ( $I_{\min}$ ) for RLS BF and R-1D-MRT RLS BF for convergence [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

N (Data size)	$N_s = 4$		$N_s = 16$	
	RLS BF	R-1D-MRT RLS BF	RLS BF	R-1D-MRT RLS BF
8192	4	4	4	4
4096	4	4	4	4
2048	4	4	4	4
1024	4	4	4	4
512	4	4	4	4
256	4	4	4	4
128	4	4	4	4
64	4	4	4	4

The RLS and R-1D-MRT RLS beamformers exhibit rapid convergence irrespective of the data size ( $N$ ).

This section shows that the R-1D-MRT RLS beamformer exhibits a performance comparable to a conventional RLS beamformer. The discussion proceeds to evaluate the effect of the R-1D-MRT algorithm on the computational complexity of RLS beamformer in the next section.

### 5.1.2 Computational Complexity of R-1D-MRT RLS Beamformer

The conversion of the input signal to R-1D-MRT form requires  $2(N-1)$  real additions. For a single iteration of the RLS beamformer, equation 5.1 requires  $N_s^2$  complex multiplications and  $N_s(N_s-1)$  complex additions. Equation 5.2 requires 1 complex division and  $N_s$  complex multiplications and  $N_s$  complex additions. Equations 5.3 and 5.4 require  $N_s$  complex multiplications and  $N_s$  complex additions each. Equation 5.5 requires  $2N_s$  complex multiplications,  $N_s$  complex additions and  $2N_s$  real multiplications. A single iteration thus requires  $(4(N_s^2-1)+19N_s)$  complex multiplications and  $(3N_s(N_s-1)+15N_s)$  complex additions. The iterations have to be repeated  $N$  times for a RLS beamformer while the R-1D-MRT RLS beamformer requires it to be repeated only  $N/2$  times. Hence optimum weight calculation of RLS beamformer requires  $N \times (2N_s^2+5N_s)$  real multiplications and  $N \times (3N_s^2+2N_s-1)$  real additions while R-1D-MRT RLS beamformer requires only  $N_s(2N-4)+(N/2) \times (4(N_s^2-1)+19N_s)$  real multiplications and  $N_s(3N-5)+N/2(3N_s(N_s-1)+15N_s)$  real additions.

Multiplication with the weights require  $N \times N_s$  complex multiplications for RLS beamformer and  $(N/2) \times N_s$  complex multiplications for R-1D-MRT RLS beamformer. The summing stage requires  $N \times N_s$  complex additions for RLS beamformer and  $N/2 \times N_s$  complex additions for R-1D-MRT RLS beamformer. Each complex multiplication requires 4 real multiplications and 3 real additions while each

complex addition requires 2 real additions. Hence the total computational load on the RLS beamformer is  $(8N_s^2N+24N_sN)$  real multiplications and  $(6N_s^2N+20N_sN)$  real additions while the R-1D-MRT RLS beamformer requires only  $(4N_s^2N+12N_sN)$  real multiplications and  $(6N_s^2N+18N_sN-2N_s)$  real additions. A comparison of the computational complexity of RLS beamformer and R-1D-MRT RLS beamformer is for varying number of antennas/sensors ( $N_s$ ) and varying data sizes ( $N$ ) is shown in Table 5.2.

Table 5.2 Comparison of computational complexity of RLS BF and R-1D-MRT RLS BF

N (Data size)	$N_s = 4$				$N_s = 8$			
	No of real multiplications		No of real additions		No of real multiplications		No of real additions	
	RLS BF	R-1D- MRT RLS BF	RLS BF	R-1D- MRT RLS BF	RLS BF	R-1D- MRT RLS BF	RLS BF	R-1D- MRT RLS BF
8192	1835008	917504	655456	589912	5767168	2883584	1311104	1180016
4096	917504	458752	327776	295000	2883584	1441792	655744	590192
2048	458752	229376	163936	147544	1441792	720896	328064	295280
1024	229376	114688	82016	73816	720896	360448	164224	147824
512	114688	57344	41056	36952	360448	180224	82304	74096
256	57344	28672	20576	18520	180224	90112	41344	37232
128	28672	14336	10336	9304	90112	45056	20864	18800
64	14336	7168	5216	4696	45056	22528	10624	9584
32	7168	3584	2656	2392	22528	11264	5504	4976
16	3584	1792	1376	1240	11264	5632	2944	2672

A significant reduction in the number of multiplications and additions are observed for the R-1D-MRT RLS beamformer, thus validating its computational efficiency. The next section examines the computation time of the R-1D-MRT RLS beamformer.

### 5.1.3 Computation Time of R-1D-MRT RLS Beamformer

The computation time of the RLS beamformer and R-1D-MRT RLS beamformers for beamforming with 4, 16, 50 and 100 antennas/sensors ( $N_s$ ) is considered in this section. The data size is varied from 16 to 8192 in each case and the results are presented in Table 5.3 and Table 5.4.

Table 5.3 Comparison of computation time of RLS BF and R-1D-MRT RLS BF (seconds) [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

N (Data size)	$N_s = 4$			$N_s = 16$		
	RLS BF	R-1D-MRT RLS BF	% Reduction	RLS BF	R-1D- MRT RLS BF	% Reduction
8192	0.3123	0.2552	18	0.3605	0.1768	51
4096	0.1447	0.0978	32	0.1652	0.0860	48
2048	0.0654	0.0404	38	0.0822	0.0419	49
1024	0.0304	0.0211	31	0.0416	0.0259	37
512	0.0170	0.0123	28	0.0258	0.0169	34
256	0.0117	0.0100	14	0.0122	0.0089	27
128	0.0111	0.0077	30	0.0087	0.0072	17
64	0.0066	0.0056	16	0.0070	0.0062	11
32	0.0059	0.0056	5	0.0062	0.0056	9
16	0.0054	0.0054	0	0.0056	0.0055	2

Table 5.4 Comparison of computation time of RLS BF and R-1D-MRT RLS BF (seconds) [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

N (Data size)	$N_s = 50$			$N_s = 100$		
	RLS BF	R-1D-MRT RLS BF	% Reduction	RLS BF	R-1D- MRT RLS BF	% Reduction
8192	0.7224	0.3838	46	1.9352	1.0208	47
4096	0.3334	0.1981	40	0.9451	0.4913	48
2048	0.1687	0.0936	54	0.4580	0.2519	45
1024	0.0897	0.0505	43	0.2424	0.1252	48
512	0.0453	0.0276	39	0.1177	0.0661	43
256	0.0276	0.0168	39	0.0644	0.0390	39
128	0.0191	0.0113	40	0.0356	0.0206	42
64	0.0122	0.0078	36	0.0197	0.0141	28
32	0.0076	0.0067	11	0.0157	0.0112	28
16	0.0066	0.0063	4	0.0108	0.0094	12

The simulations were performed with a computer using Intel Core i5 CPU working at 2.20 GHz and RAM of 4GB. The computation time is reduced with the R-1D-MRT RLS beamformer by virtue of the reduced computational complexity. This further validates the computational efficiency of the R-1D-MRT RLS beamformer.

The next section reflects on the performance of R-1D-MRT RLS beamformer in various communication channels.

#### 5.1.4 Performance of R-1D-MRT RLS Beamformer in Various Communication Channels

The validation of the R-1D-MRT RLS beamformer in this section evaluates its performance in AWGN, Rayleigh and Rician channels. The Euclidian distance is

compared for conventional RLS and R-1D-MRT RLS beamformers under different noisy scenarios. The results are presented in Table 5.5.

Table 5.5 Comparison of Euclidian distance parameters of RLS BF and R-1D-MRT RLS BF for various communication channels for varying SINR [ $N_s = 8$ , Source angle =  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

SINR	AWGN		Rayleigh		Rician	
	RLS BF	R-1D-MRT RLS BF	RLS BF	R-1D-MRT RLS BF	RLS BF	R-1D-MRT RLS BF
10 dB	0.5599	0.2947	1.4471	1.2859	0.6906	0.4316
20 dB	0.1659	0.0642	0.1837	0.1310	0.2469	0.1831
30 dB	0.0741	0.0395	0.0756	0.0672	0.1369	0.1133

The above observations show that the Euclidian distance is lesser for the R-1D-MRT RLS beamformer for all the channels compared, indicating a higher proximity of the R-1D-MRT RLS beamformer output to the original signal. To further confirm the finding, a visual comparison is presented using scatterplots. The scatterplots of the original signal, noisy signal with interference, RLS beamformer output and R-1D-MRT RLS beamformer output for various communication channels are illustrated in Fig 5.4 - Fig 5.6.

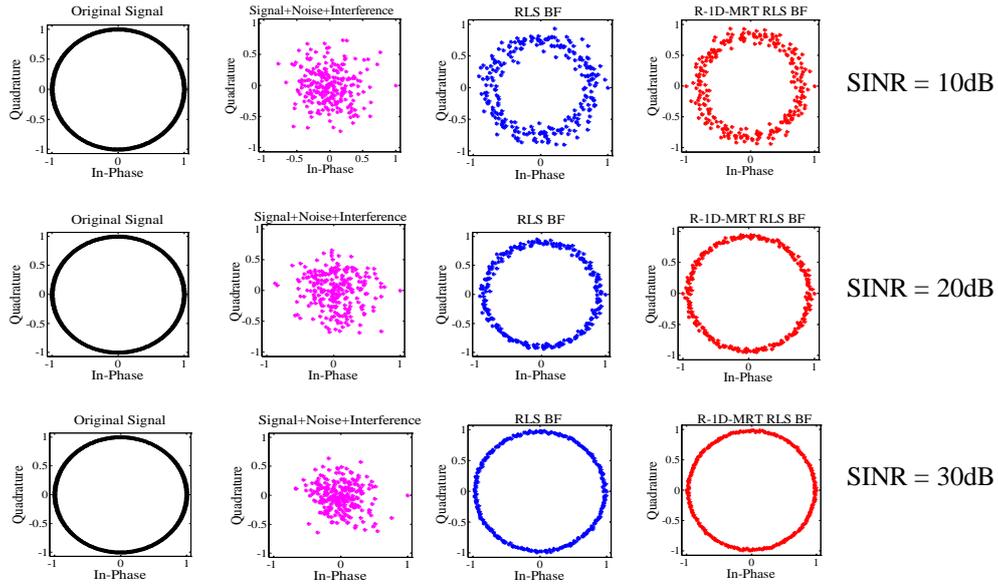


Fig 5.4 Scatterplots of R-1D-MRT RLS BF for AWGN channel for varying SINR [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

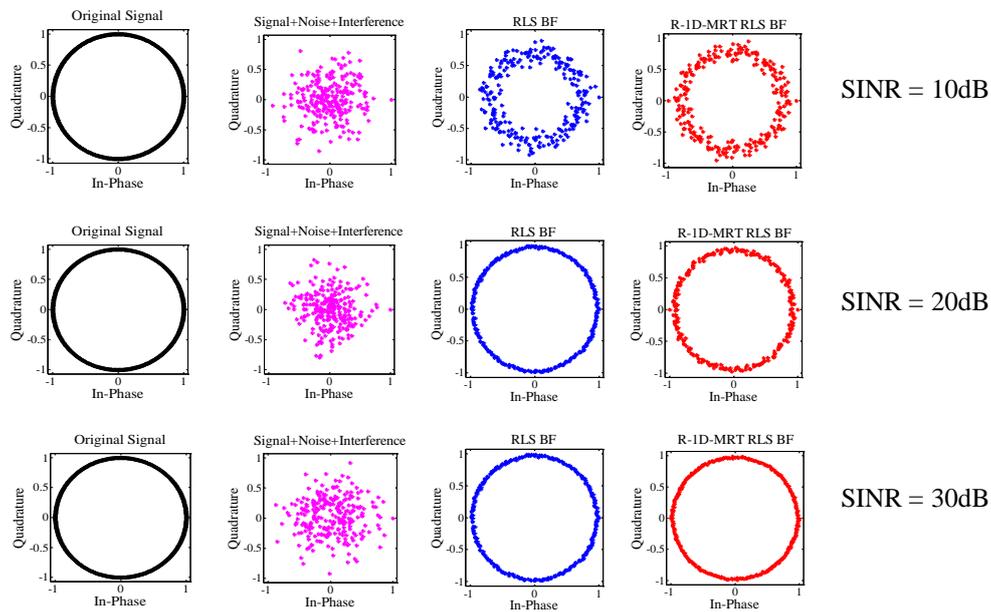


Fig 5.5 Scatterplots of R-1D-MRT RLS BF for Rayleigh channel for varying SINR [ $N_s = 8$ , Source angle =  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

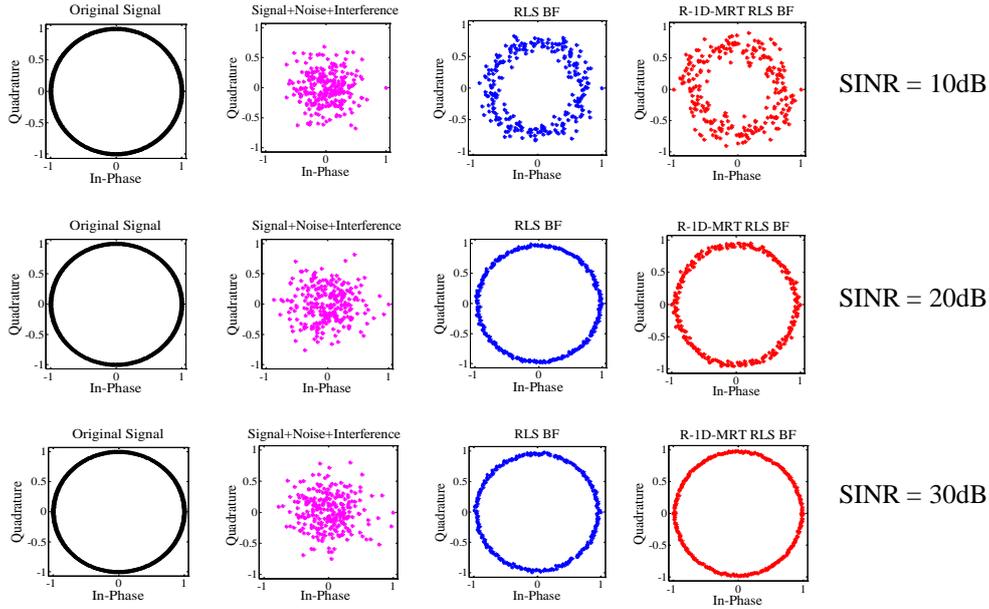


Fig 5.6 Scatterplots of R-1D-MRT RLS BF for Rician channel for varying SINR [ $N_s = 8$ , Source angle =  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ ,  $\zeta = 0.99$ ,  $\delta = 0.89$ ]

The scatterplots of the R-1D-MRT RLS beamformer for the various communication channels illustrated above suggest that in noisy environments, the R-1D-MRT RLS beamformer output exhibits higher fidelity compared to the conventional RLS beamformer output.

Section 5.1.1 - 5.1.5 has discussed the validation of the R-1D-MRT RLS beamformer. The computational efficiency and performance improvement in various communication channels of the R-1D-MRT RLS beamformer is also proved. The next section describes the R-1D-MRT SMI beamformer and its validation.

## 5.2 R-1D-MRT SMI BEAMFORMER

The R-1D-MRT RLS beamformer considered in the previous section had fast convergence but it involved recursive computations. In this section, the

R-1D-MRT algorithm is validated on SMI beamformers, which employ SMI algorithm for weight adaptation. The SMI beamformer achieves fast convergence, using sample matrices instead of the recursive computations used by the RLS beamformer. The sample matrix is a time average estimate of the array correlation matrix using  $K$ -time samples. The faster convergence rate is achieved as the algorithm employs direct inversion of the correlation matrix  $C$ , where

$$C = E[\mathbf{X}\mathbf{X}^H] \quad (5.6)$$

Here  $\mathbf{X}$  represents array signal vector of size  $1 \times N_s$ , where  $N_s$  represents number of antennas/sensors in the array.  $\mathbf{X}^H$  represents Hermitian transpose of  $\mathbf{X}$ . SMI is also known as Direct Matrix Inversion (DMI). The algorithm divides the input samples into “ $k$ ” number of blocks, each of length  $K$ . The optimal beamforming weights of the SMI beamformer are computed based on the estimates of the covariance matrix  $C$  and the cross correlation vector  $r$  for an input signal block of size  $K$ .

The correlation matrix  $C$  is given by

$$\hat{C}(k) = \frac{1}{K} \sum_{k=0}^{K-1} \mathbf{X}(k)\mathbf{X}^H(k) \quad (5.7)$$

The cross correlation vector  $r$  is given by

$$\hat{r}(k) = \frac{1}{K} \sum_{k=0}^{K-1} \mathbf{d}^*(k)\mathbf{X}(k) \quad (5.8)$$

where  $d(k)$  represents the desired signal.

The optimum weight vector is given by

$$\hat{\mathbf{w}}(k) = C^{-1}(k)r(k) \quad (5.9)$$

The signal received at the array is weighted with the optimum weights obtained using the SMI algorithm and summed to give the beamformer output (K.R. Zalawadia et al., 2011; Xiaojun Mao et al., 2015; Michael L. Picciolo, 2003; Xin Song et al., 2009).

The schematic diagram of the conventional SMI beamformer is shown in Fig 5.7(a).

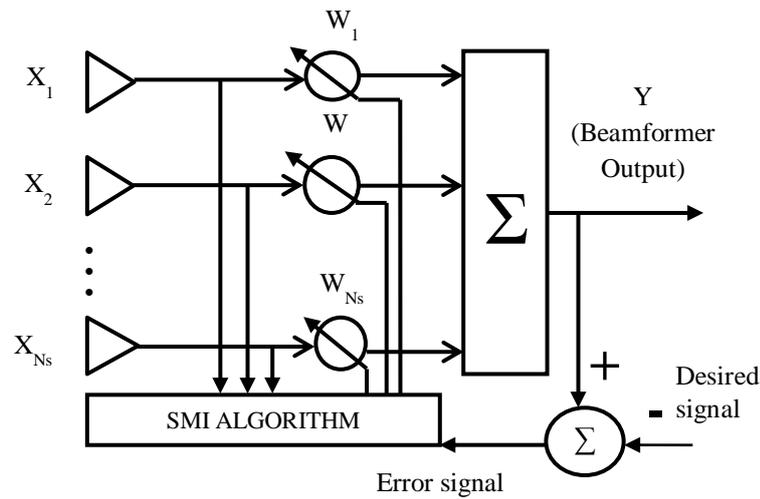


Fig 5.7(a) Schematic diagram of conventional SMI beamformer

In the R-1D-MRT SMI beamformer, the data received at the array elements is transformed using the R-1D-MRT algorithm and then multiplied with the weights obtained using the SMI algorithm and finally summed to give the R-1D-MRT SMI beamformer output. The schematic diagram of the R-1D-MRT SMI beamformer is shown in Fig 5.7(b).

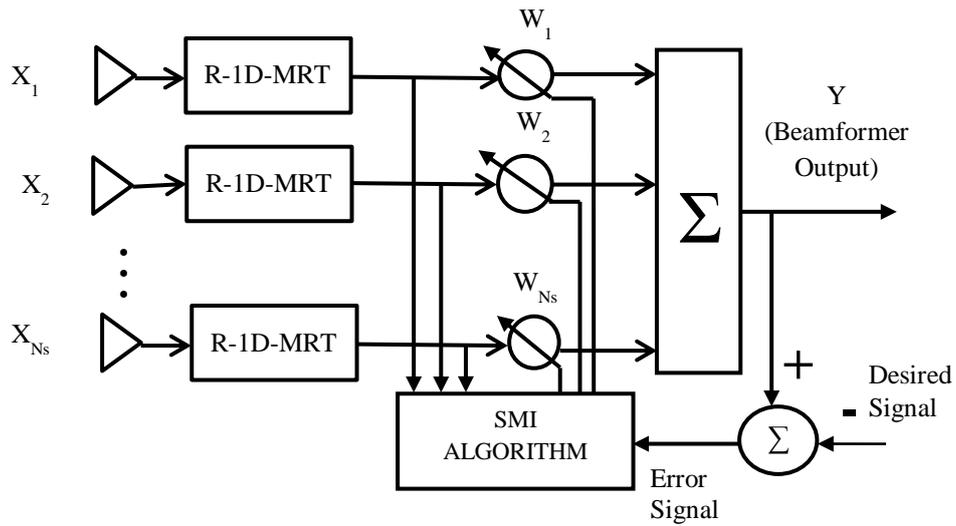


Fig 5.7(b) Schematic diagram of R-1D-MRT SMI beamformer

The validation of the R-1D-MRT SMI beamformer considering its beam pattern, computational complexity, computation time etc. is presented in the following sections.

### 5.2.1 Beam Pattern and Beamformer Output of R-1D-MRT SMI Beamformer

The normalized beam pattern, mean square error, beamformer output and weight stabilization of the R-1D-MRT SMI beamformer obtained with different simulations is presented in Fig 5.8 and Fig 5.9.

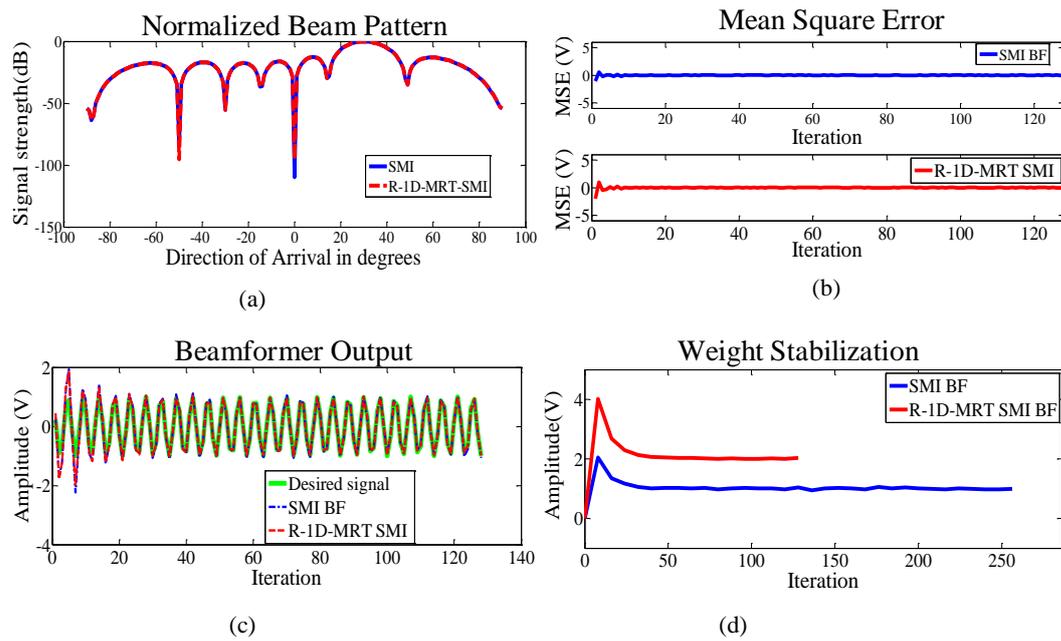


Fig 5.8 R-1D-MRT SMI BF output [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB, Block length = 8]

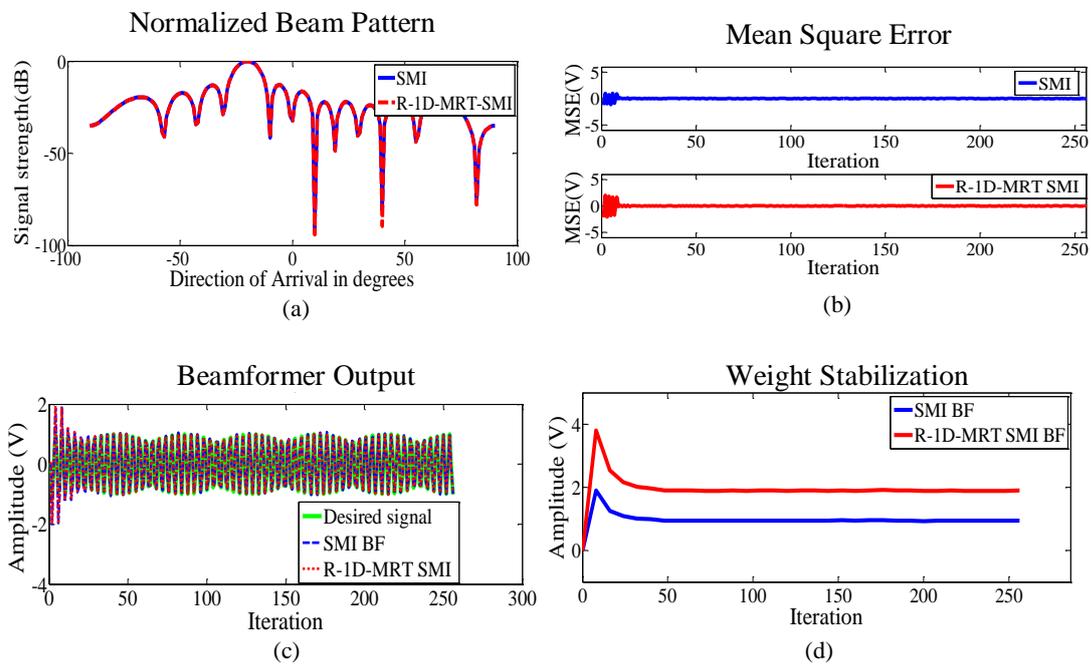


Fig 5.9 R-1D-MRT SMI BF output [ $N_s = 12$ , Source angle  $-20^\circ$ , Interference angles  $50^\circ$  and  $-40^\circ$ ,  $N = 512$ , SINR = 30 dB, Block length = 8]

The normalized beam patterns of Fig 5.8(a) and 5.9(a) are of the desired form showing main lobe in the desired direction of  $30^\circ$  and  $-20^\circ$  respectively and nulls in the interference directions of  $(0^\circ, -50^\circ)$  and  $(50^\circ, -40^\circ)$  respectively. The mean square error (MSE) of the SMI beamformer and the R-1D-MRT SMI beamformer converges to zero rapidly as can be seen in Fig 5.8(b) and 5.9(b). The initial value of mean square error is higher for R-1D-MRT SMI beamformer. The outputs of the SMI beamformer and the R-1D-MRT SMI beamformer track the input signal quickly as can be seen in Fig 5.8(c) and 5.9(c). The weights of the SMI beamformer and R-1D-MRT SMI beamformer stabilize to a steady value in 50 iterations as observed in Fig 5.8(d) and Fig 5.9(d) respectively. The convergence of SMI and R-1D-MRT SMI beamformer becomes faster as the block length reduces. The MSE for varying block lengths is shown in Fig 5.10.

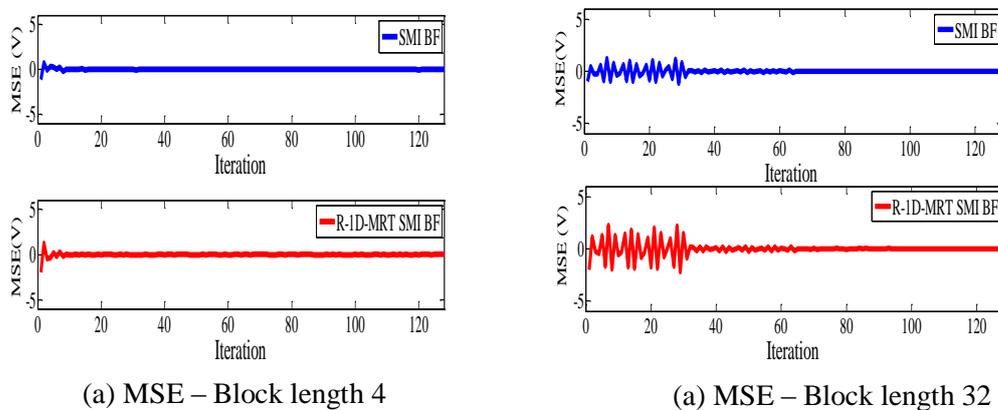


Fig 5.10 MSE of SMI BF and R-1D-MRT SMI BF for varying block lengths [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB]

The MSE convergences faster for a block length of 4 than for a block length of 32 for both SMI and R-1D-MRT SMI beamformers as indicated by Fig 5.10(a) and 5.10(b)

respectively. The minimum number of iterations required for convergence for various block lengths is presented in Table 5.6.

Table 5.6 Comparison of minimum number of iterations ( $I_{\min}$ ) of SMI BF and R-1D-MRT SMI BF for convergence for varying block lengths [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , SINR = 30 dB]

Block length	SMI BF	R-1D-MRT SMI BF
64	66	66
32	65	65
16	43	44
8	17	24
4	8	9

The convergence of MSE is delayed as the block length increases for both SMI BF and R-1D-MRT SMI BF. The number of iterations required for convergence for varying data sizes ( $N$ ) and varying number of antennas/sensors ( $N_s$ ) with fixed block length is presented in Table 5.7.

Table 5.7 Comparison of minimum number of iterations ( $I_{\min}$ ) of SMI BF and R-1D-MRT SMI BF for convergence [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB, Block length = 8]

N (Data size)	$N_s = 8$		$N_s = 12$	
	SMI BF	R-1D-MRT SMI BF	SMI BF	R-1D-MRT SMI BF
8192	34	34	30	30
4096	30	30	25	25
2048	26	26	23	23
1024	24	24	20	20
512	18	18	18	18
256	18	18	18	18
128	18	18	17	17
64	18	18	15	15

The minimum number of iterations for convergence is the same for both SMI BF and R-1D-MRT SMI BF for varying data sizes ( $N$ ) for a block length of 8 with 8 antennas/sensors ( $N_s$ ) in the array. Similar result is observed for an array of 12 antennas/sensors ( $N_s$ ).

In the next section, the computational complexity of the R-1D-MRT SMI beamformer is discussed.

### 5.2.2 Computational Complexity of R-1D-MRT SMI Beamformer

SMI beamformers use blocks of data for beamforming. For a comparative analysis, the same block length is considered for both SMI and R-1D-MRT SMI beamformers. Since the input data size is reduced by half for R-1D-MRT SMI beamformer, the number of blocks will be half that of the SMI beamformer. For calculating Eqn 5.7,  $N_s^2 BL$  complex multiplications and  $N_s^2 (BL-1)$  complex additions are required by both the beamformers, where  $BL$  is the block length. For Eqn 5.8,  $N_s \times BL$  complex multiplications and  $N_s \times (BL - 1)$  complex additions are required. For computing Equation 5.9,  $N_s^3$  complex multiplications and  $N_s^2 - 1$  complex additions are required. This has to be repeated  $N/BL$  times for SMI beamformer and  $M/BL$  times for R-1D-MRT SMI beamformer, where  $M = N/2$ . Multiplying the data with the weights requires  $4N_s \times (N-1)$  real multiplications and  $2N_s \times (N-2)$  real additions for SMI beamformer and  $2N_s \times (N-2)$  real multiplications and  $N_s \times (N-2)$  real additions for R-1D-MRT SMI beamformer. The summation stage requires  $2N_s \times (N-2)$  real additions for SMI beamformer and  $N_s \times (N-2)$  real additions for R-1D-MRT SMI beamformer. The transformation to R-1D-MRT requires  $N_s \times (2N-1)$  real additions for the R-1D-MRT SMI beamformer. The computational complexity of the SMI beamformer

and R-1D-MRT SMI beamformer for 4 and 8 antennas/sensors ( $N_s$ ) is compared in Table 5.8.

Table 5.8 Comparison of computational complexity of SMI BF and R-1D-MRT SMI BF [Block length = 16]

N (Data size)	$N_s = 4$				$N_s = 8$			
	No of real multiplications		No of real additions		No of real multiplications		No of real additions	
	SMI BF	R-1D- MRT SMI BF	SMI BF	R-1D- MRT SMI BF	SMI BF	R-1D- MRT SMI BF	SMI BF	R-1D- MRT SMI BF
8192	950256	475120	797688	454372	3014624	1507296	2677744	1469928
4096	472150	237552	398840	231280	1507296	753632	1338864	734952
2048	237552	118768	199416	116084	753632	376800	669424	367464
1024	118768	59376	99704	58036	376800	188384	334704	183720
512	59376	29680	49848	29012	188384	94176	167344	91848
256	29680	14832	24920	14500	94176	47072	83644	45912
128	14832	7408	12456	7244	47072	23520	41824	22944
64	7408	3996	6224	3616	23520	11744	20904	11460

The computational complexity of R-1D-MRT SMI beamformer is lower than that of SMI beamformer and the resultant reduction in computation time is analyzed in the next section.

### 5.2.3 Computation Time of R-1D-MRT SMI Beamformer

This section discusses the validation of R-1D-MRT SMI beamformer on the basis of computation time. Arrays of 4, 16, 50 and 100 antennas/sensors ( $N_s$ ) with varying input data sizes are considered. The computation time is presented in Tables 5.9 and 5.10.

Table 5.9 Comparison of computation time of SMI BF and R-1D-MRT SMI BF (seconds) [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB, Block length = 16]

N (Data size)	$N_s = 4$			$N_s = 16$		
	SMI BF	R-1D-MRT SMI BF	% Reduction	SMI BF	R-1D-MRT SMI BF	% Reduction
8192	0.0651	0.0290	55	0.1582	0.0734	54
4096	0.0249	0.0141	43	0.0604	0.0373	38
2048	0.0098	0.0060	39	0.0301	0.0155	49
1024	0.0047	0.0030	36	0.0158	0.0081	49
512	0.0025	0.0018	28	0.0071	0.0054	24
256	0.0017	0.0012	29	0.0033	0.0029	9
128	0.0010	0.0008	20	0.0020	0.0020	0
64	0.0007	0.0008	-14	0.0013	0.0012	8
32	0.0006	0.0007	-17	0.0013	0.0011	16
16	0.0006	0.0006	0	0.0011	0.0008	27

Table 5.10 Comparison of computation time of SMI BF and R-1D-MRT SMI BF (seconds) [Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ , SINR = 30 dB, Block length = 16]

N (Data size)	$N_s = 50$			$N_s = 100$		
	SMI BF	R-1D-MRT SMI BF	% Reduction	SMI BF	R-1D-MRT SMI BF	% Reduction
8192	0.5865	0.3219	45	3.5642	1.0598	70
4096	0.2678	0.1419	47	2.0550	0.4878	76
2048	0.1304	0.0682	48	0.9462	0.2496	74
1024	0.0638	0.0337	47	0.4679	0.1233	74
512	0.0315	0.0172	45	0.2360	0.0600	75
256	0.0137	0.0091	34	0.1185	0.0311	74
128	0.0074	0.0052	30	0.0552	0.0163	70
64	0.0040	0.0035	13	0.0274	0.0083	80
32	0.0023	0.0021	9	0.0152	0.0040	74
16	0.0016	0.0015	6	0.0039	0.0027	31

The simulations were performed with a computer using Intel Core i5 CPU working at 2.20 GHz and RAM of 4GB.

The computation time of R-1D-MRT SMI beamformer is lower than that required by the SMI beamformer. The significant reduction in the number of multiplication operations results in reduced computation complexity and computation time for the R-1D-MRT SMI beamformer as observed in Tables 5.8 - 5.10, thus validating the computational efficiency of the R-1D-MRT SMI beamformer.

The next section considers the performance of R-1D-MRT SMI beamformer in various communication channels.

#### 5.2.4 Performance of R-1D-MRT SMI Beamformer in Various Communication Channels

The performance of the R-1D-MRT SMI beamformer in AWGN, Rayleigh and Rician channels are considered in this section. The Euclidian distance of the SMI and R-1D-MRT SMI beamformers for varying SINR is presented in Table 5.11.

Table 5.11 Comparison of Euclidian distance of SMI BF and R-1D-MRT SMI BF for various communication channels for varying SINR [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 1024$ , Block Length = 8]

SNR	AWGN		Rayleigh		Rician	
	SMI BF	R-1D-MRT SMI BF	SMI BF	R-1D-MRT SMI BF	SMI BF	R-1D-MRT SMI BF
10 dB	0.0111	0.0109	0.0021	0.0012	0.0017	0.0013
20 dB	0.0045	0.0038	0.0012	0.0011	0.0013	0.0011
30 dB	0.0028	0.0023	0.0011	0.0009	0.0011	0.0005

The Euclidean distance between the R-1D-MRT SMI beamformer output and original signal is lesser compared to the Euclidean distance between the SMI beamformer output and the original signal.

Fig 5.11 – Fig 5.13 illustrate the scatterplots of the original signal, signal with noise and interference, SMI beamformer output and R-1D-MRT SMI beamformer output for various SINR for AWGN, Rayleigh and Rician channels respectively.

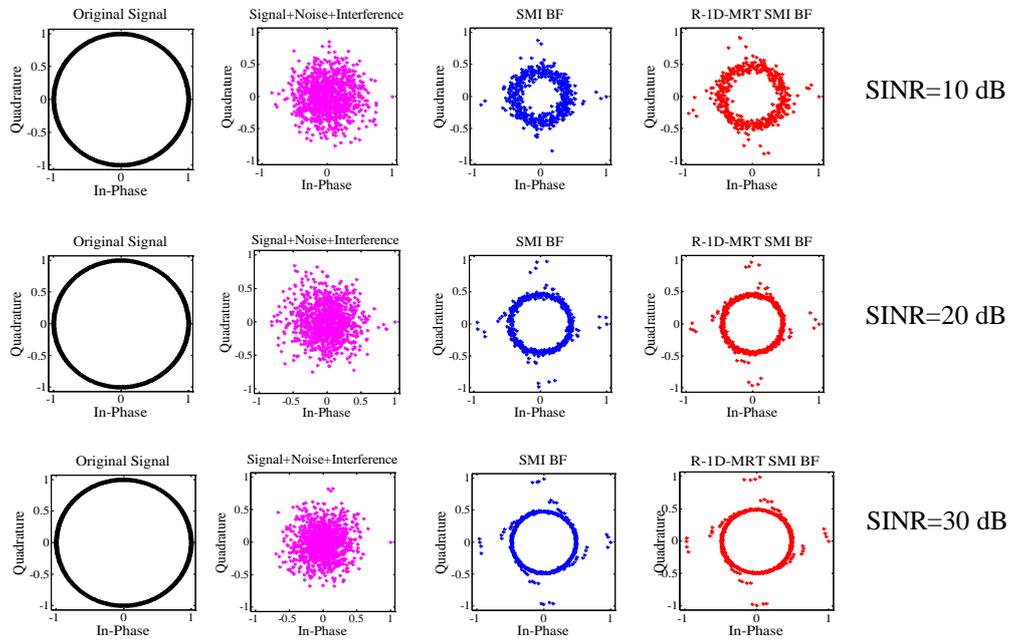


Fig 5.11 Scatterplots of R-1D-MRT SMI BF for AWGN channel for varying SINR [ $N_s = 8$ , Source angle  $30^\circ$ , Interferences angles  $0^\circ$  and  $-50^\circ$ ,  $N = 1024$ , Block Length = 8]

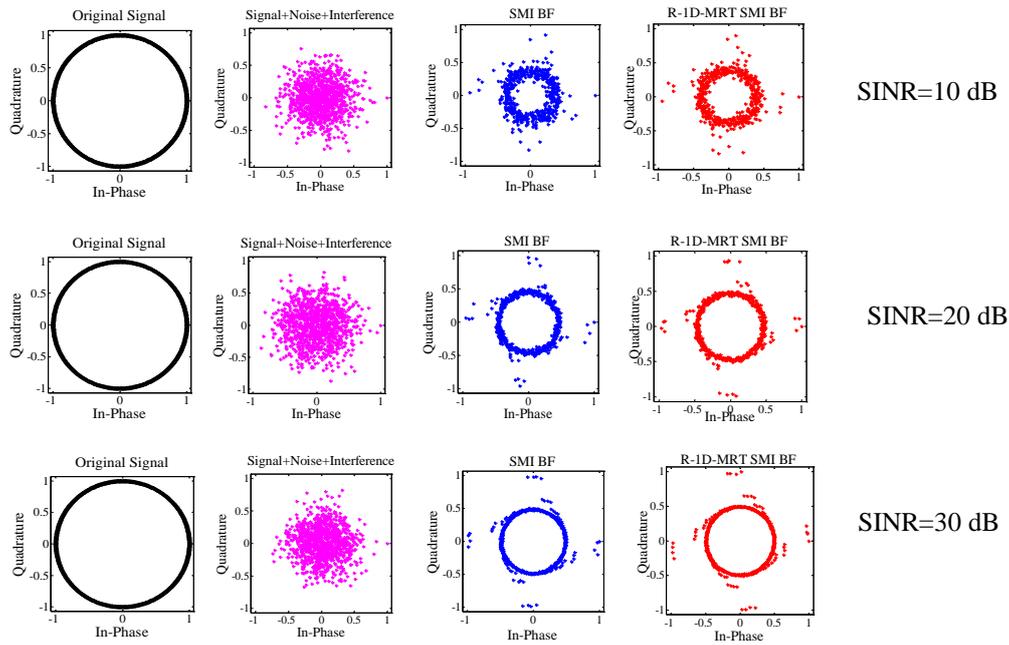


Fig 5.12 Scatterplots of R-1D-MRT SMI BF for Rayleigh channel for varying SINR [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 1024$ , Block Length = 8]

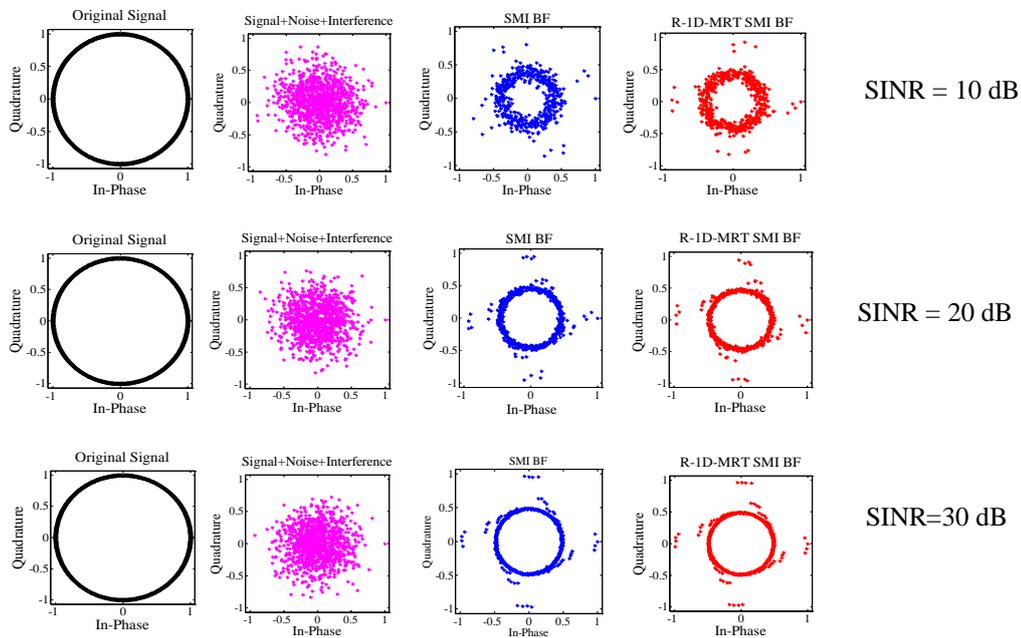


Fig 5.13 Scatterplots of R-1D-MRT SMI BF for Rician channel for varying SINR [ $N_s = 8$ , Source angle  $30^\circ$ , Interference angles  $0^\circ$  and  $-50^\circ$ ,  $N = 256$ , Block Length = 8]

The scatterplots of the R-1D-MRT SMI beamformer show more likeness to the original signal compared to that of the SMI beamformer.

This section thus confirms the validity of the R-1D-MRT SMI beamformer for beamforming. In the next section, the validation of R-1D-MRT algorithm with Generalized Sidelobe Cancellers is discussed.

### 5.3 R-1D-MRT GSC BEAMFORMER

The MVDR and LCMV beamformers discussed in chapter 3, imposed a single constraint on the beamformer. The output power of the beamformer was minimized subject to a unity gain constraint for the MVDR beamformer and gain  $g$  for the LCMV beamformer in the desired direction. The Generalized Sidelobe Canceller (GSC beamformer) imposes multiple linear constraints (Simon Haykin, 2012). The operation of the GSC depends on the combined effect of 2 actions:

- a) The imposition of linear multiple constraints so as to preserve an incident signal along a direction of interest.
- b) The adjustment of some weights, in accordance with the LMS algorithm so as to minimize the effects of interference and noise at the beamformer output.

The multiple linear constraints are described by a  $N_s$  by  $L$  matrix  $D$  (Constraint matrix), where  $L$  is the number of linear constraints and  $N_s$  is the number of antenna/sensor elements in the array (Talmon et al., 2009; Lu et al., 2012).

A signal blocking matrix  $D_a$  of size  $N_s$  by  $N_s-L$  is defined by

$$D_a^H D_a = 0 \quad (5.10)$$

The weight vector is given by

$$\mathbf{w}(n) = \mathbf{w}_q - \mathbf{D}_a \mathbf{w}_a(n) \quad (5.11)$$

$\mathbf{W}_a(n)$  is the adjustable weight vector.

$\mathbf{W}_q$  is the quiescent weight vector given by

$$\mathbf{W}_q = \mathbf{D}(\mathbf{D}^H \mathbf{D})^{-1} \mathbf{g} \quad (5.12)$$

where  $\mathbf{g}$  is the prescribed gain vector.

The beamformer output ( $Y$ ) is given by the error signal

$$\mathbf{Y}(n) = \mathbf{W}^H(n) \mathbf{X}(n) = (\mathbf{W}_q - \mathbf{D}_a^H \mathbf{W}_a(n))^H \mathbf{X}(n) = \mathbf{W}_q^H \mathbf{X}(n) - \mathbf{D}_a^H \mathbf{W}_a^H(n) \mathbf{X}(n) \quad (5.13)$$

where  $\mathbf{X}(n)$  is the input signal.

The schematic diagram of a conventional GSC beamformer is shown in Fig 5.14(a).

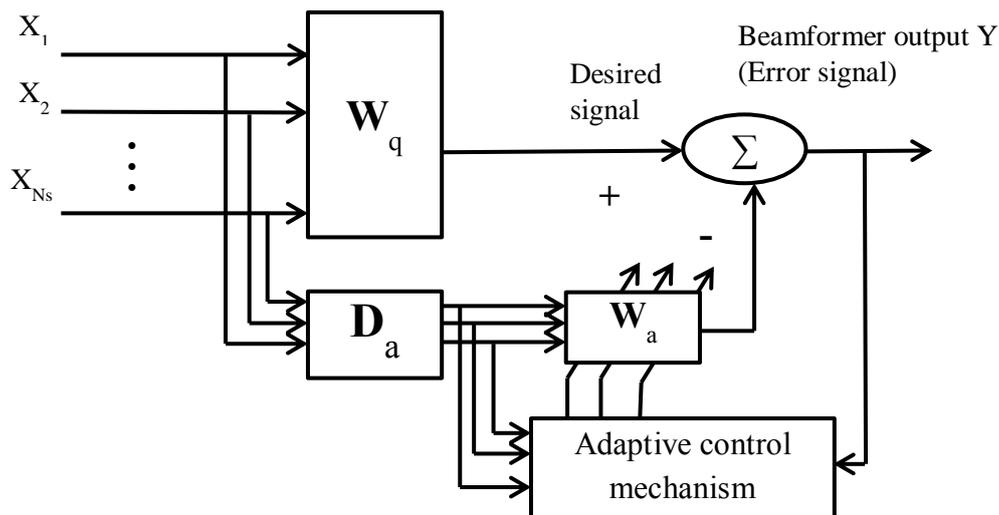


Fig 5.14(a) Schematic diagram of conventional GSC beamformer

For the R-1D-MRT GSC beamformer, the GSC beamforming is performed using R-1D-MRT transformed input signals. Fig 5.14(b) illustrates the schematic diagram of R-1D-MRT GSC beamformer.

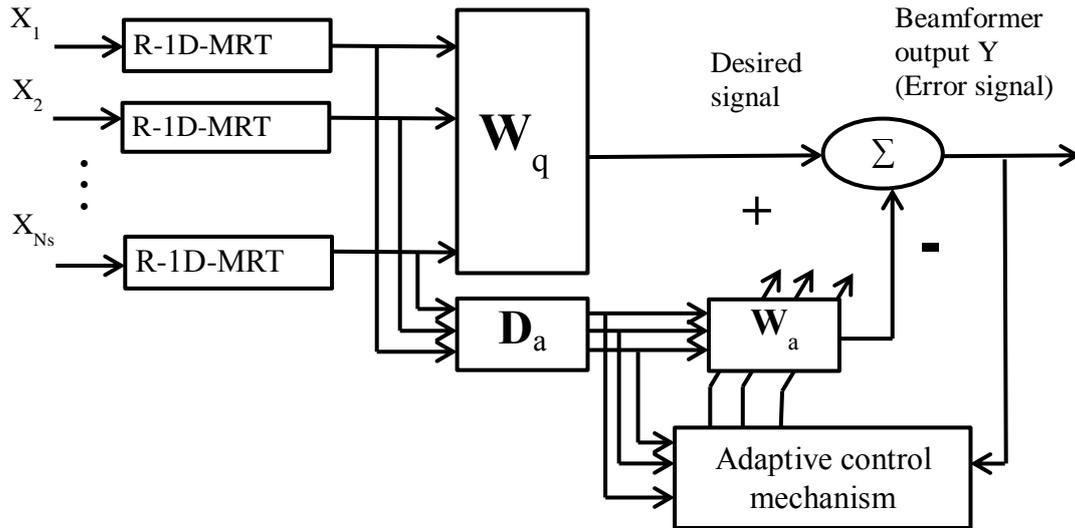


Fig 5.14(b) Schematic diagram of R-1D-MRT GSC beamformer

The beam pattern and beamformer output of R-1D-MRT GSC beamformer is analyzed in the next section.

### 5.3.1 Beam Pattern and Beamformer Output of R-1D-MRT GSC Beamformer

The normalized beam pattern and beamformer output of the R-1D-MRT GSC beamformer is investigated for varying number of sensors and constraints and varying directions of source and interference. The results of MATLAB<sup>®</sup> simulation is presented in Fig 5.15-5.17.

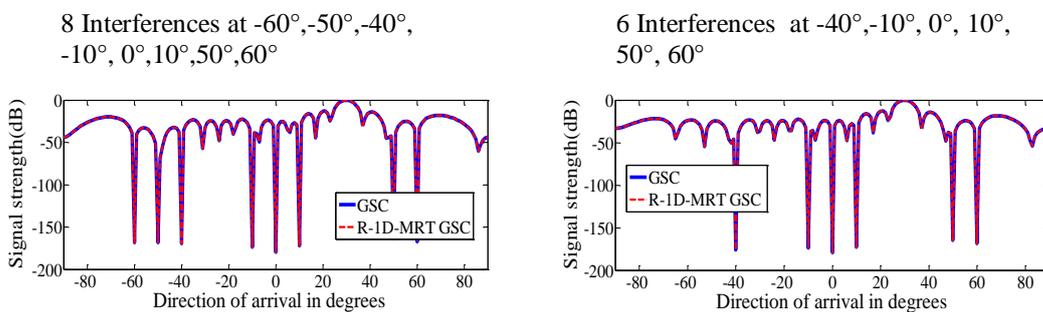


Fig 5.15 Normalized beam pattern of R-1D-MRT GSC BF for varying constraints [ $N_s = 20$ , Source angle  $30^\circ$ ,  $N = 256$ , SINR = 30 dB]

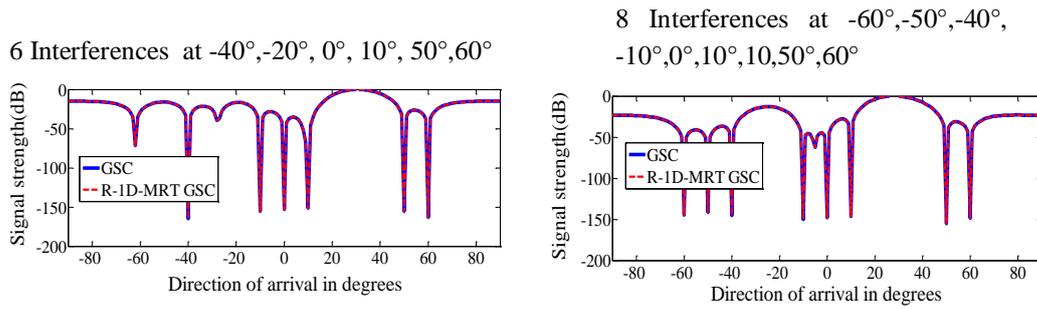


Fig 5.16 Normalized beam pattern of R-1D-MRT GSC BF for varying constraints [ $N_s = 10$ , Source angle  $30^\circ$ ,  $N = 256$ , SINR = 30 dB]

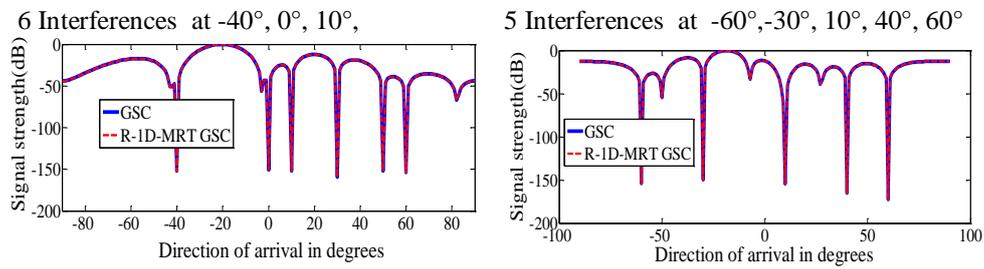


Fig 5.17 Normalized beam pattern of R-1D-MRT GSC BF for varying constraints [ $N_s = 10$ , Source angle  $-20^\circ$ ,  $N = 256$ , SINR = 30 dB]

The normalized beam patterns of the GSC beamformer and R-1D-MRT GSC beamformer exhibit a peak at direction of source and nulls in the directions of interfering signals.

The beamformer output of R-1D-MRT GSC beamformer for varying gain constraints (g) is presented in Fig 5.18.

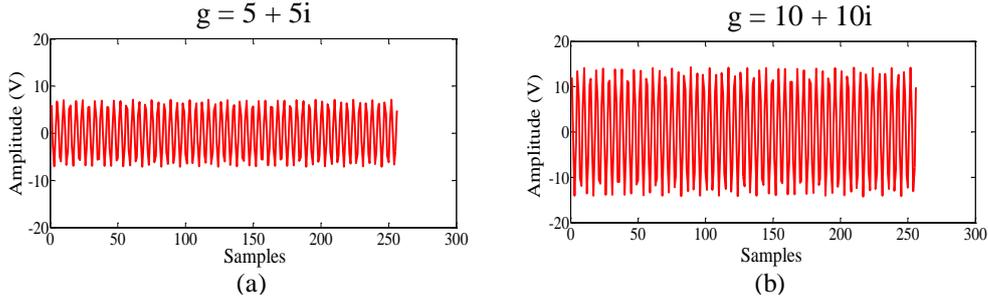


Fig 5.18 Beamformer output of R-1D-MRT GSC BF for varying gain constraints [ $N_s = 10$ , Source angle  $30^\circ$ , Interferences at  $-60^\circ$ ,  $-50^\circ$ ,  $-4^\circ$ ,  $-20^\circ$ ,  $0^\circ$ ,  $10^\circ$ ,  $50^\circ$ ,  $60^\circ$ ,  $N = 256$ , SINR = 30 dB]

The output of R-1D-MRT GSC beamformer for a gain constraint of  $5+5i$  is depicted in Fig 5.18(a) and in 5.18(b) for a gain constraint of  $10+10i$ .

The validation of R-1D-MRT GSC on the basis of computational complexity is analyzed in the following section.

### 5.3.2 Computational Complexity of R-1D-MRT GSC Beamformer

The equations for computing the weights of the GSC BF and R-1D-MRT GSC BF (Eqn 5.10-5.12) are independent of the input data. Hence the computational complexity for calculating the weights is the same for both the beamformers and therefore only the computational load for the remaining sections of the beamformers is compared. The GSC BF requires  $N_s(4N-4)$  real multiplications and  $N_s(2N-2)$  real additions for multiplication of the input signal with the weights and  $N_s(2N-2)$  real additions for summing up the weighted signals. The R-1D-MRT GSC BF requires  $N_s(2N-4)$  real multiplications and  $N_s(N-2)$  real additions for multiplication of the input signal with the weights,  $N_s(N-2)$  real additions for summing the weighted signals and  $N_s(2N-2)$  real additions for transforming the input signal to R-1D-MRT

form. Table 5.12 presents the computational load for an array of 10 and 20 antennas/sensors ( $N_s$ ) for varying data sizes ( $N$ ).

Table 5.12 Comparison of computational complexity of GSC BF and R-1D-MRT GSC BF

N (Data size)	$N_s = 10$				$N_s = 20$			
	No of real multiplications		No of real additions		No of real multiplications		No of real additions	
	GSC BF	R-1D- MRT GSC BF	GSC BF	R-1D- MRT GSC BF	GSC BF	R-1D- MRT GSC BF	GSC BF	R-1D- MRT GSC BF
8192	327640	163800	327640	327620	655280	327600	655280	655240
4096	163800	81880	163800	163780	327600	163760	327600	327560
2048	81880	40920	81880	81860	163760	81840	163760	163720
1024	40920	20440	40920	40900	81840	40880	81840	81800
512	20440	10200	20440	20420	40880	20400	40880	40840
256	10200	5080	10200	10180	20400	10160	20400	20360
128	5080	2520	5080	5060	10160	5040	10160	10120
64	2520	1240	2520	2500	5040	2480	5040	5000
32	1240	600	1240	1220	2480	1200	2480	2440
16	600	280	600	580	1200	560	1200	1160

The R-1D-MRT GSC beamformer has 40 real additions lesser than GSC beamformer while the number of real multiplications is significantly reduced. The computation time of the GSC and R-1D-MRT GSC beamformers are compared in the next section.

### 5.3.3 Computation Time of R-1D-MRT GSC Beamformer

The computation time of the GSC and R-1D-MRT GSC beamformers for an array of 10, 20, 50 and 100 antennas/sensors ( $N_s$ ) is presented in Tables 5.13 and 5.14.

Table 5.13 Comparison of computation time of GSC BF and R-1D-MRT GSC BF (seconds) [Source angle = 30°, Interference angles = -40°, -20°, 0°, 10°, -50°, 60°, SINR = 30 dB]

N (Data size)	$N_s = 10$			$N_s = 20$		
	GSC BF	R-1D-MRT GSC BF	% Reduction	GSC BF	R-1D-MRT GSC BF	% Reduction
8192	0.0128	0.0090	30	0.0208	0.0127	39
4096	0.0068	0.0051	25	0.0102	0.0071	30
2048	0.0035	0.0027	23	0.0054	0.0039	27
1024	0.0022	0.0017	23	0.0031	0.0031	0
512	0.0015	0.0015	0	0.0022	0.0020	9
256	0.0012	0.0012	0	0.0021	0.0020	9
128	0.0010	0.0008	20	0.0013	0.0012	0
64	0.0009	0.0008	11	0.0013	0.0011	8
32	0.0009	0.0008	11	0.0012	0.0011	8
16	0.0008	0.0008	0	0.0011	0.0011	0

Table 5.14 Comparison of computation time of GSC BF and R-1D-MRT GSC BF (seconds) [Source angle = 30°, Interference angles = -40°, -20°, 0°, 10°, -50°, 60°, SINR = 30 dB]

N (Data size)	$N_s = 50$			$N_s = 100$		
	GSC BF	R-1D-MRT GSC BF	% Reduction	GSC BF	R-1D-MRT GSC BF	% Reduction
8192	0.0547	0.0316	43	0.1207	0.0568	57
4096	0.0251	0.0117	53	0.0605	0.0310	48
2048	0.0106	0.0076	28	0.0293	0.0165	44
1024	0.0062	0.0045	27	0.0176	0.0114	35
512	0.0046	0.0043	6	0.0112	0.0106	5
256	0.0041	0.0037	9	0.0090	0.0084	6
128	0.0036	0.0036	0	0.0083	0.0079	5
64	0.0031	0.0026	16	0.0079	0.0075	5
32	0.0025	0.0022	12	0.0075	0.0075	0
16	0.0024	0.0022	8	0.0069	0.0069	0

The simulations were performed with a computer using Intel Core i5 CPU working at 2.20 GHz and RAM of 4GB. The computation time for R-1D-MRT GSC

beamformer is lesser than that of GSC beamformer. The validation of the computational efficiency of the R-1D-MRT GSC beamformer in terms of computational complexity and computation time is thus completed.

The next section discusses the performance of the R-1D-MRT GSC beamformer in various communication channels.

### 5.3.4 Performance of R-1D-MRT GSC Beamformer in Various Communication Channels

The performance of the R-1D-MRT GSC beamformer is analyzed in AWGN, Rayleigh and Rician communication channels. The Euclidean distance of the beamformer output with the desired signal is tabulated for the various channels and presented in Table 5.15.

Table 5.15 Comparison of Euclidian distance parameters of GSC BF and R-1D-MRT GSC BF for various communication channels for varying SINR [ $N_s = 10$ , Source angle =  $30^\circ$ , Interference angles =  $-40^\circ, -20^\circ, 0^\circ, 10^\circ, -50^\circ, 60^\circ$ ,  $N = 256$ ]

SNR	AWGN		Rayleigh		Rician	
	GSC BF	R-1D-MRT GSC BF	GSC BF	R-1D-MRT GSC BF	GSC BF	R-1D-MRT GSC BF
10 dB	0.0846	0.0802	0.0936	0.0864	0.0804	0.0773
20 dB	0.0562	0.0543	0.0735	0.0704	0.0635	0.0618
30 dB	0.0278	0.0273	0.0295	0.0286	0.0213	0.0211

The Euclidean distance between R-1D-MRT GSC beamformer output and the desired signal is lesser compared to the Euclidean distance between GSC beamformer output and the desired signal, indicating that the R-1D-MRT GSC beamformer output has more similarity to the desired signal.

The scatterplots of the GSC beamformer output and R-1D-MRT GSC beamformer output for AWGN, Rayleigh and Rician channels for varying SINR are illustrated in Fig 5.19 - Fig 5.21.

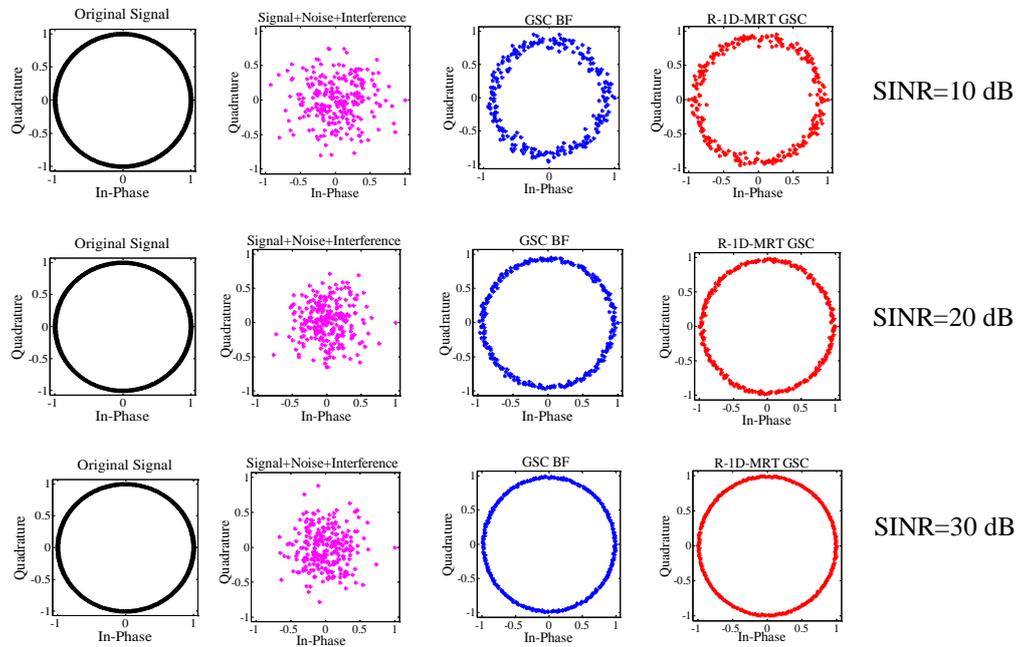


Fig 5.19 Scatterplots of R-1D-MRT GSC BF for AWGN channel for varying SINR [ $N_s = 10$ , Source angle =  $30^\circ$ , Interference angles =  $-40^\circ$ ,  $-20^\circ$ ,  $0^\circ$ ,  $10^\circ$ ,  $-50^\circ$ ,  $60^\circ$ ,  $N = 256$ ]

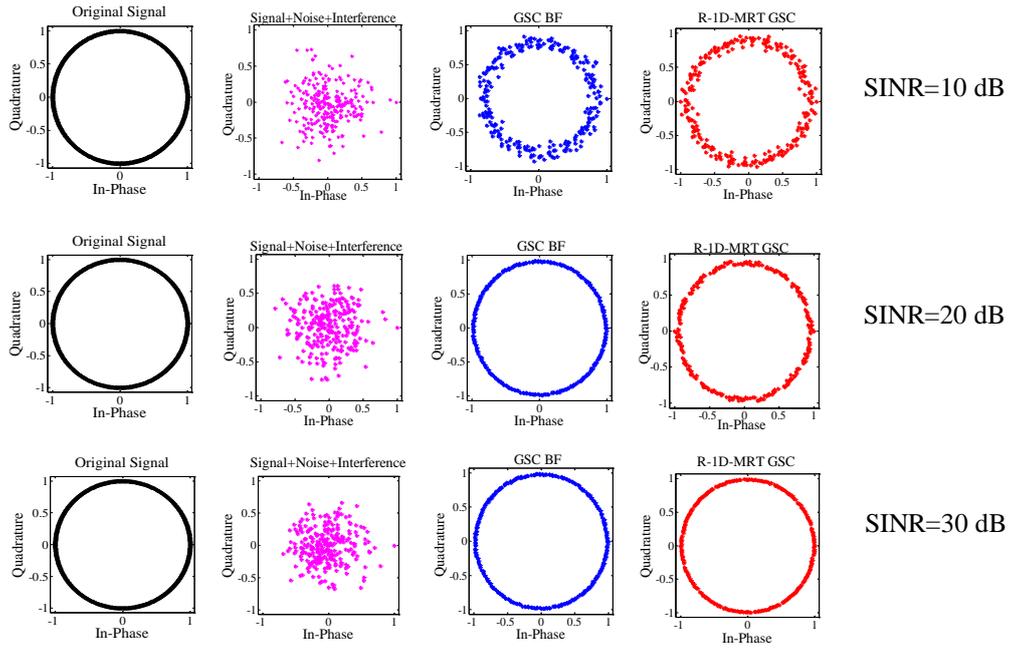


Fig 5.20 Scatterplots of R-1D-MRT GSC BF for Rayleigh channel for varying SINR [ $N_s = 10$ , Source angle =  $30^\circ$ , Interference angles =  $-40^\circ, -20^\circ, 0^\circ, 10^\circ, -50^\circ, 60^\circ$ ,  $N = 256$ ]

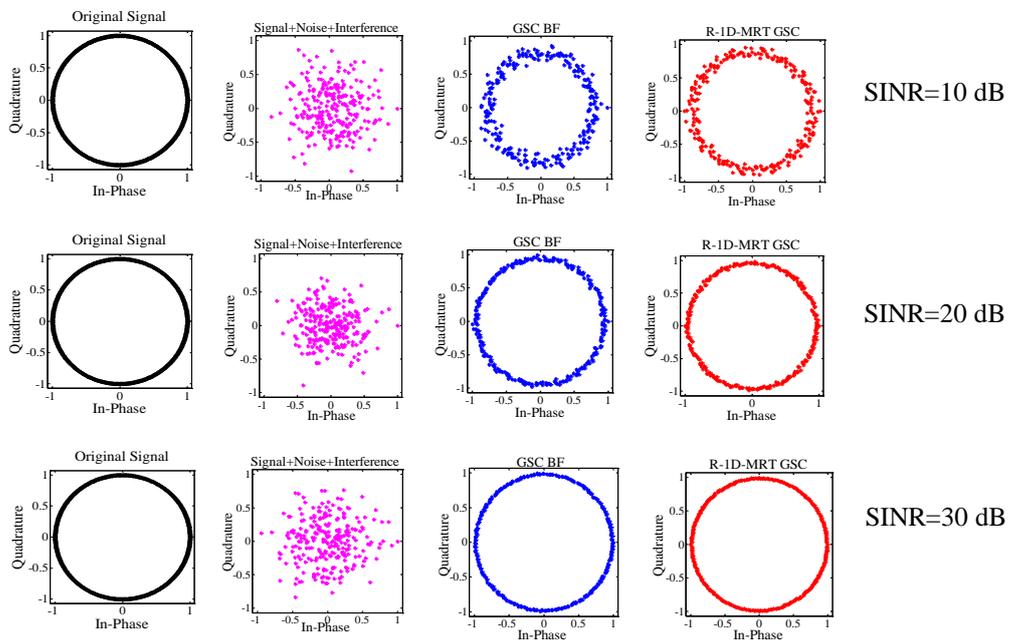


Fig 5.21 Scatterplots of R-1D-MRT GSC BF for Rician channel for varying SINR [ $N_s = 10$ , Source angle =  $30^\circ$ , Interference angles =  $-40^\circ, -20^\circ, 0^\circ, 10^\circ, -50^\circ, 60^\circ$ ,  $N = 256$ ]

The scatterplot of the R-1D-MRT GSC BF has higher similarity to the scatterplot of the original signal at low SINR values while at high SINR, scatterplots of both the beamformer outputs have high level of similarity to the scatterplot of the original signal. The lower Euclidean distance and higher similarity of the R-1D-MRT GSC BF to the original signal thus validates the efficiency of the R-1D-MRT algorithm for GSC beamforming in various communication channels.

A summary of the discussions of this chapter is presented in the following section.

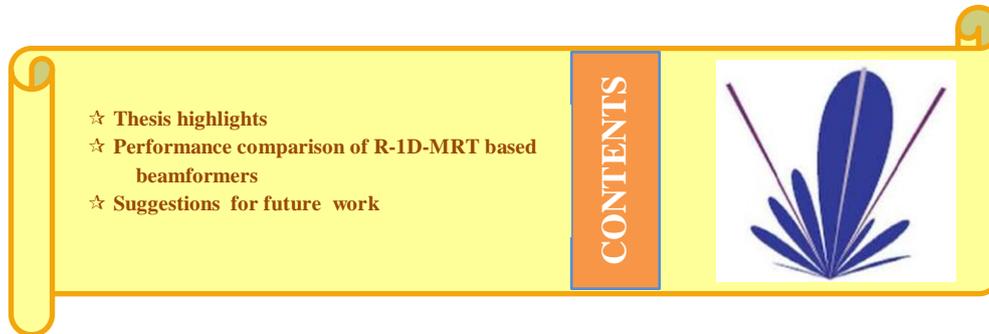
#### **5.4 CHAPTER SUMMARY**

This chapter discussed the validation of the R-1D-MRT algorithm on RLS, SMI and GSC beamformers. The use of recursive equations results in quick convergence in the R-1D-MRT RLS BF besides reduction in computational complexity and computation time compared to conventional RLS beamformer. The R-1D-MRT SMI beamformer achieves fast convergence by adopting a block adaptive method for calculation of weights. Reduced computational complexity and computation time along with better performance in various communication channels in comparison with conventional SMI beamformers are observed. The R-1D-MRT GSC beamformer specifies multiple constraints to null several interferers simultaneously. The computational efficiency of this beamformer is validated in terms of its computational complexity and computation time. The performance of R-1D-MRT GSC beamformer in AWGN, Rayleigh and Rician channels is also seen to be better than that of the conventional GSC beamformer.

The summary of the discussions pertaining to this research work is presented in the next chapter.

## CHAPTER 6

### CONCLUSIONS AND FUTURE SCOPE



The development of a computationally efficient antenna beamforming algorithm, its validation on various beamformers and its implementation using FPGA are described in this thesis. The salient features of the research work are presented in this chapter.

#### 6.1 THESIS HIGHLIGHTS

The highlights of the thesis are:

- ◆ Development of R-1D-MRT, a computationally efficient algorithm for beamforming applications.
- ◆ Validation of R-1D-MRT algorithm on conventional beamformers and its implementation using FPGA.
- ◆ Validation of R-1D-MRT algorithm on constrained beamformers - MVDR beamformer and LCMV beamformer - and their implementation using FPGA.
- ◆ Validation of R-1D-MRT algorithm on stochastic gradient beamformers - LMS beamformer and NLMS beamformer - and their FPGA implementation.
- ◆ Validation of R-1D-MRT algorithm on RLS, SMI and GSC beamformers.

Validation of these beamformers is performed on the basis of normalized beam pattern, beamformer output, computational complexity, computation time and performance in AWGN, Rayleigh and Rician communication channels. FPGA implementation of the beamformers is based on simulations and synthesis of Xilinx System Generator models created for each beamformer. FPGA resource utilization of the XSG models is also investigated.

The computation of the algorithm involves only addition operations and avoids complex multiplications. Application of the R-1D-MRT algorithm reduces the size of the data required to be processed by the beamformer to half the size of the input data. This leads to effective reduction in the computational complexity and computation time of the beamformer in comparison to conventional beamformers using raw data and is demonstrated in chapters 2 - 6 for various beamformers.

The normalized beam pattern and the beamformer output of the various beamformers using R-1D-MRT transformed data is observed to be at par with those of the respective beamformers processing raw data. The normalized beam patterns exhibit peak at the desired direction in the case of conventional beamformers as seen in chapter 2. The normalized beampatterns exhibit peak at the desired direction along with nulls in the undesired directions as demonstrated by the adaptive beamformers considered in chapters 3-5. The performance of the beamformers using R-1D-MRT transformed data is validated on AWGN, Rayleigh and Rician channels for varying SINR, using Euclidean distance measurement and scatterplots in chapters 2 – 5. In all the cases the performance of R-1D-MRT based beamformers is observed to be better than that of the beamformers using raw data, especially at low SINR values.

The validation of the R-1D-MRT Conventional Digital beamformers using FPGA is presented in Chapter 2. The simulation and synthesis of the XSG models show that the normalized beam pattern and beamformer output obtained with the FPGA implementation of R-1D-MRT CDBF is at par with that of a CDBF using raw data. The FPGA resource utilization is lower for the R-1D-MRT CDBF.

The beamformer output obtained satisfies the unity gain constraint for R-1D-MRT MVDR beamformer and gain constraint  $g$  for R-1D-MRT LCMV beamformers in both MATLAB<sup>®</sup> simulations and XSG simulation as discussed in chapter 3. The FPGA implementation is carried out using Modified Gaussian Elimination method for QR decomposition which is required for computing the inverse of the correlation matrix. The FPGA resources required by the R-1D-MRT MVDR and R-1D-MRT LCMV beamformers are seen to be lower than that required by MVDR and LCMV beamformers using raw data.

The validation of R-1D-MRT on stochastic gradient based LMS and NLMS beamformers shows faster convergence as detailed in chapter 4. Beamformer output, normalized beam pattern of XSG simulations and FPGA implementation of these beamformers also exhibit the desired characteristics.

The R-1D-MRT RLS beamformer uses recursive equations to achieve fast convergence. The R-1D-MRT SMI beamformer adapts the data block by block and also achieves fast convergence. The R-1D-MRT GSC beamformer satisfies multiple constraints and its beam pattern shows peak at the desired direction with the desired gain and nulls at the undesired directions as specified in the constraint vector. These results are presented in chapter 5.

## **6.2 PERFORMANCE COMPARISON OF R-1D-MRT BASED BEAMFORMERS**

The validations discussed in chapters 2 – 5 prove that the application of R-1D-MRT algorithm reduces the computational complexity and computation time while giving a normalized beam pattern and beamformer output comparable with beamformers using raw data. The performance in AWGN, Rayleigh and Rician channels is enhanced in R-1D-MRT beamformers. Convergence rate is improved for R-1D-MRT LMS and R-1D-MRT NLMS beamformers compared to conventional LMS and NLMS beamformers. Convergence rate of R-1D-MRT RLS and R-1D-MRT SMI beamformers match those of conventional RLS and SMI beamformers. R-1D-MRT GSC beamformer output is at par with that of conventional GSC beamformer. FPGA implementations of R-1D-MRT beamformers exhibit the required beam patterns and beamformer output with lesser requirement of FPGA resources.

The performance comparison of R-1D-MRT based beamformers considered in this thesis is illustrated in Table 6.1.

Table 6.1 Performance comparison of R-1D-MRT based beamformers and conventional beamformers

Parameter	R-1D-MRT CBF	R-1D-MRT MVDR	R-1D-MRT LCMV	R-1D-MRT LMS	R-1D-MRT NLMS	R-1D-MRT RLS	R-1D-MRT SMI	R-1D-MRT GSC
Computational Complexity ( $N_s = 4, N = 512$ )	25% reduction (Table 2.3)	50% reduction (Table 3.1)	50% reduction (Table 3.8)	30% reduction (Table 4.2)	36% reduction (Table 4.8)	39% reduction (Table 5.2)	46% reduction (Table 5.8)	25% reduction ( $N_s = 10$ ) (Table 5.12)
Computational Time ( $N_s = 4, N = 512$ )	15% reduction (Table 2.4)	37% reduction (Table 3.2)	14% reduction (Table 3.9)	46% reduction (Table 4.3)	51% reduction (Table 4.10)	28% reduction (Table 5.3)	28% reduction (Table 5.9)	11% reduction ( $N_s = 10$ ) (Table 5.13)
Euclidean distance (AWGN, $N_s = 4, N = 256$ ) (SNR/SINR = 10dB)	40% reduction (Table 2.6)	20% reduction (Table 3.4)	12% reduction (Table 3.11)	6% reduction (Table 4.5)	8% reduction (Table 4.12)	47% reduction (Table 5.5)	21% reduction (Table 5.11)	5% reduction (Table 5.15)
Convergence Rate $N_s = 4, N = 512$	Not Applicable	Not Applicable	Not Applicable	58% Faster (Table 4.1)	65% Faster (Table 4.7)	Same as conventional RLS (Table 5.1)	Same as conventional SMI (Table 5.6)	Not Applicable
Other features	Cannot reject interferences (Fig 2.9)	Passes the desired signal with unity gain (Fig 3.5)	Passes the desired signal with gain $g$ (Fig 3.28)	Converges to optimum weight (Fig 4.2)	Faster convergence to optimum weight (Fig 4.14)	Rapid convergence (Fig 5.2)	Rapid convergence (Fig 5.8)	Multiple constraints (Fig 5.15)

### 6.3 SUGGESTIONS FOR FUTURE WORK

The 1D-MRT is a versatile transform which requires only addition operations for computing its coefficients. Its resourcefulness has been proven in applications like texture descriptor analysis for images, study of chaotic behaviour etc. The R-1D-MRT algorithm presented in this thesis has shown effective reduction in computational complexity for beamforming by using only one column of the 1D-MRT transform. The possibility of beamforming using combinations of other coefficients of 1D-MRT may be explored.

As a future work this algorithm can be tested using antenna/sensor arrays for real time beamforming applications. The computational efficiency of R-1D-MRT can be validated for underwater beamforming using sonar, since sonar uses a large number of hydrophones in many applications.

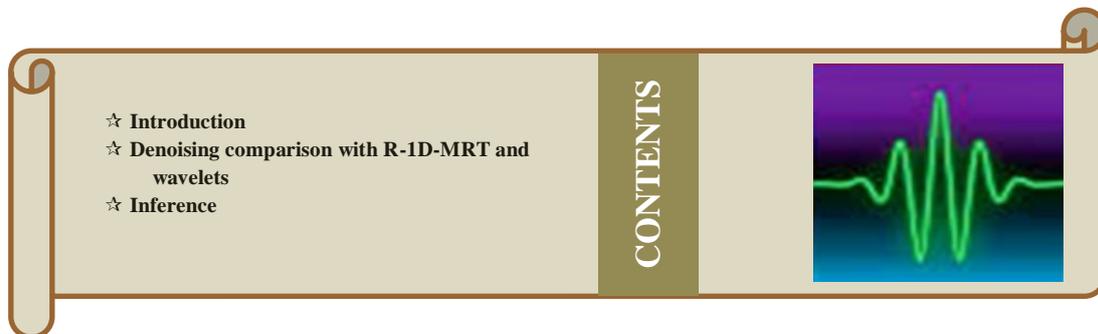
The rapid strides being made in science and technology are merging all the activities of human life into a highly interconnected smart world. As more and more smart devices become part of our daily life, digital processing requires voluminous amounts of data to be exchanged. The computational reduction offered by the R-1D-MRT algorithm may be tested for applications like data mining, big data, IOT and similar applications which process huge amounts of data. Software defined control and real time beamforming are becoming a vital part of sensor control in smart homes. Similarly massive beamforming is estimated to be an essential part of 5G radio access. Application of R-1D-MRT for these purposes may be explored as a future scope.

The R-1D-MRT RLS beamformer requires computation of recursive equations which can be evaluated using QR decomposition and implemented in FPGA using systolic arrays. This method of matrix inversion can also be used for FPGA implementation of R-1D-MRT GSC beamformers. The R-1D-MRT SMI can also be implemented in FPGA by adapting the weights block by block.

The R-1D-MRT algorithms may also be extended to other beamformers not considered in this research work.

## APPENDIX A

### R-1D-MRT BASED DENOISING IN CONVENTIONAL BEAMFORMERS



This chapter validates the use of R-1D-MRT algorithm for denoising of signals.

#### A.1 INTRODUCTION

A popular method for noise reduction is wavelet denoising which consists of 3 steps:

- 1) Application of a linear forward wavelet transform to the noisy signals
- 2) Nonlinear shrinkage denoising using thresholding
- 3) Application of a linear inverse wavelet transform

Denoising based on thresholding in the wavelet domain was proposed by Donoho and Johnstone (1994). Thresholding involves a binary decision. Two thresholding techniques are soft and hard thresholding. For each wavelet coefficient, if its amplitude is smaller than a predefined threshold, it will be set to zero, otherwise it will be kept unchanged (hard thresholding), or shrunk in the absolute value by an amount of the threshold (soft thresholding) (K.I. Ramachandran and K.P. Soman, 2010).

$$\text{Hard thresholding: } y = \begin{cases} x, & \text{if } |x| > T \\ 0, & \text{if } |x| < T \end{cases} \quad (\text{A.1})$$

$$\text{Soft thresholding: } y = \begin{cases} \text{sign}(x) (|x| - T), & \text{if } |x| > T \\ 0, & \text{if } |x| < T \end{cases} \quad (\text{A.2})$$

where,  $x$  is the input signal,  $y$  is the output signal and  $T$  is the threshold.

The four methods of selecting the threshold for wavelet denoising are Rigrsure, Heursure, Sqtwolog and Minimaxi.

Rigrsure method uses a threshold selection rule based on Stein's Unbiased Estimate of Risk. An estimate of the risk for a particular threshold value  $T$  is obtained. Minimizing the risks in  $T$  gives a selection of the threshold value. Sqtwolog is based on the universal threshold  $\sigma\sqrt{2\log n}$  where  $n$  is the size of the signal and  $\sigma$  is the rms value of noise. Heursure is a mixture of Rigrsure method and Sqtwolog method. Minimaxi method uses minmax thresholding i.e., selects the largest threshold that minimizes the maximum relative risk with respect to the optimum relative risk.

## **A.2 COMPARISON OF R-1D-MRT BASED DENOISING AND WAVELET DENOISING**

Signals denoised with various wavelets and threshold functions are compared with R-1D-MRT denoised signals for different levels of Additive white Gaussian Noise and the resultant SNR values are shown in Tables A.1 - A.8.

Table A.1 Comparison of R-1D-MRT denoising and wavelet denoising using Daubechies wavelet with Heursure thresholding in conventional beamformers [ $N_s = 8$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR Input	SNR R-1D-MRT	SNR Wavelet (Daubechies 4)	SNR R-1D-MRT	SNR Wavelet (Daubechies 8)	SNR R-1D-MRT	SNR Wavelet (Daubechies 16)
5	8.01	7.95	8.09	8.11	7.97	7.94
10	12.96	12.82	12.98	12.95	13.04	13.10
15	17.76	17.29	17.94	17.93	17.74	17.85
20	22.38	21.17	22.32	22.44	22.43	22.66
25	26.22	23.78	26.23	26.46	26.26	27.05
30	28.81	25.08	28.81	29.32	28.81	30.42

Table A.2 Comparison of R-1D-MRT denoising and wavelet denoising using Symlet wavelet with Heursure thresholding in conventional beamformers [ $N_s = 8$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR Input	SNR R-1D-MRT	SNR Wavelet (Symlet 4)	SNR R-1D-MRT	SNR Wavelet (Symlet 8)	SNR R-1D-MRT	SNR Wavelet (Symlet 16)
5	8.04	7.93	7.93	7.97	8.10	8.01
10	12.97	12.71	12.92	12.95	12.84	12.91
15	17.76	17.19	12.87	12.90	17.72	17.93
20	22.37	20.79	22.31	22.57	22.40	22.67
25	26.14	23.07	26.15	26.83	26.17	27.00
30	28.80	24.14	28.79	29.88	28.79	30.49

Table A.3 Comparison of R-1D-MRT denoising and wavelet denoising using Daubechies wavelet with Rigrsure thresholding in conventional beamformers [ $N_s = 8$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR Input	SNR R-1D-MRT	SNR Wavelet (Daubechies 4)	SNR R-1D-MRT	SNR Wavelet (Daubechies 8)	SNR R-1D-MRT	SNR Wavelet (Daubechies 16)
5	8.01	7.87	8.05	7.99	8.02	8.07
10	12.94	12.82	13.00	12.99	12.93	13.03
15	17.89	17.31	17.83	17.92	17.91	18.05
20	22.43	21.15	22.34	22.49	22.32	22.68
25	26.81	23.77	26.16	26.39	26.18	27.02
30	28.76	25.07	28.79	29.24	28.80	30.37

Table A.4 Comparison of R-1D-MRT denoising and wavelet denoising using Symlet wavelet with Rigrsure thresholding in conventional beamformers [ $N_s = 8$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR Input	SNR R-1D-MRT	SNR Wavelet (Symlet 4)	SNR R-1D-MRT	SNR Wavelet (Symlet 8)	SNR R-1D-MRT	SNR Wavelet (Symlet 16)
5	8.14	8.02	8.09	8.07	8.12	8.06
10	12.86	12.72	12.97	13.07	13.01	13.04
15	17.84	17.28	17.83	17.85	17.79	18.00
20	22.34	20.74	33.40	22.60	22.44	22.74
25	26.20	23.06	26.28	26.85	26.28	27.16
30	28.76	24.12	28.77	29.81	28.82	30.59

Table A.5 Comparison of R-1D-MRT denoising and wavelet denoising using Daubechies wavelet with Sqtwolog thresholding in conventional beamformers [ $N_s = 8$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR Input	SNR R-1D-MRT	SNR Wavelet (Daubechies 4)	SNR R-1D-MRT	SNR Wavelet (Daubechies 8)	SNR R-1D-MRT	SNR Wavelet (Daubechies 16)
5	8.07	7.96	7.90	7.88	8.17	8.21
10	12.81	12.62	13.06	13.11	13.07	13.11
15	17.74	17.18	17.74	17.96	17.80	18.02
20	22.41	21.08	22.40	22.54	22.23	22.52
25	26.21	23.78	26.25	26.56	26.20	26.99
30	28.77	25.08	26.78	29.32	28.81	30.42

Table A.6 Comparison of R-1D-MRT denoising and wavelet denoising using Symlet wavelet with Sqtwolog thresholding in conventional beamformers [ $N_s = 8$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR Input	SNR R-1D-MRT	SNR Wavelet (Symlet 4)	SNR R-1D-MRT	SNR Wavelet (Symlet 8)	SNR R-1D-MRT	SNR Wavelet (Symlet 16)
5	7.96	7.89	7.97	7.93	8.21	8.03
10	12.92	12.79	13.01	12.99	12.95	13.06
15	17.82	17.18	17.89	17.89	17.81	17.85
20	22.40	20.82	22.49	22.67	22.37	22.63
25	26.27	23.09	26.23	26.83	26.24	27.07
30	28.80	24.14	28.77	29.89	29.77	30.54

Table A.7 Comparison of R-1D-MRT denoising and wavelet denoising using Daubechies wavelet with Minimaxi thresholding in conventional beamformers [ $N_s = 8$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR Input	SNR R-1D-MRT	SNR Wavelet (Daubechies 4)	SNR R-1D-MRT	SNR Wavelet (Daubechies 8)	SNR R-1D-MRT	SNR Wavelet (Daubechies 16)
5	7.96	7.81	7.84	7.96	8.04	7.98
10	13.14	12.72	12.97	12.97	12.91	13.05
15	17.86	17.37	17.86	17.87	17.70	17.82
20	22.46	21.20	22.39	22.45	22.43	22.77
25	26.16	23.77	26.24	26.49	26.13	27.00
30	28.79	25.09	28.75	29.22	28.81	30.38

Table A.8 Comparison of R-1D-MRT denoising and wavelet denoising using Symlet wavelet with Minimaxi thresholding in conventional beamformers [ $N_s = 8$ , Source angle =  $30^\circ$ ,  $N = 256$ ]

SNR Input	SNR R-1D-MRT	SNR Wavelet (Symlet 4)	SNR R-1D-MRT	SNR Wavelet (Symlet 8)	SNR R-1D-MRT	SNR Wavelet (Symlet 16)
5	8.08	7.93	8.09	8.00	7.97	8.01
10	13.02	12.74	12.99	12.99	13.05	13.08
15	17.85	17.18	17.87	17.97	17.90	17.99
20	22.38	20.78	22.41	20.60	22.41	22.66
25	26.19	23.03	26.17	26.71	26.18	27.07
30	28.81	24.15	28.79	29.84	28.76	30.48

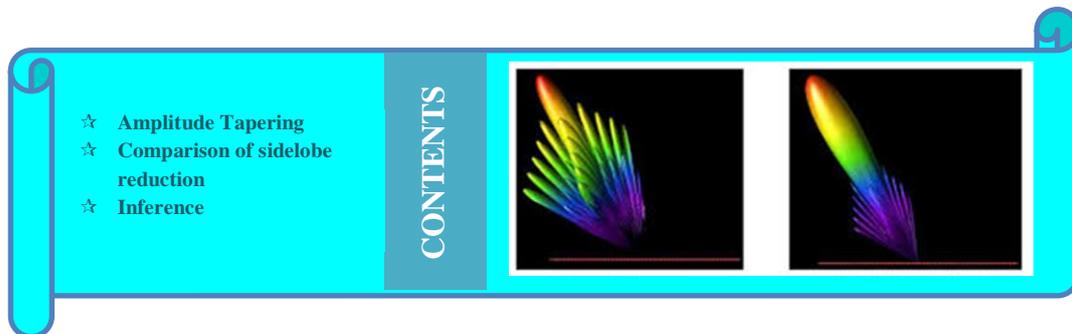
### A.3 INFERENCE

From Tables A1-A8, it can be inferred that the SNR at the output of the beamformer is higher than the SNR at the beamformer input for an input SNR of upto 25dB when wavelets are used. This indicates that effective denoising is obtained with the various wavelets and thresholding functions. Similar increase in output SNR is obtained when R-1D-MRT is applied to the noisy signal. This implies that R-1D-MRT also effectively denoises the signal. The SNR of the R-1D-MRT denoising signal has

higher output SNR than wavelet denoising with Daubechies 4 and Symlet 4 wavelets for low values of input SNR while it gives a comparable output SNR with Daubechies 8, Daubechies 16, Symlet 8 and Symlet 16 wavelets for all the four thresholding methods. The wavelet decomposition, thresholding and wavelet reconstruction requires  $N$  operations each, where  $N$  is the data size. Hence wavelet denoising requires  $3N$  operations while R-ID-MRT requires just  $N/2$  operations for obtaining a similar output. This gives a substantial improvement in computational complexity also. Thus the denoising capability of the R-1D-MRT algorithm in conventional beamformers is validated.

## APPENDIX B

# R-1D-MRT BASED SIDELOBE REDUCTION IN CONVENTIONAL BEAMFORMERS



Radiation patterns of beamformers exhibit main lobe and side lobes. The main lobe is in the direction of maximum radiation while the side lobes represent unwanted radiation in undesired directions. The presence of excessive side lobe radiation results in wastage of energy during transmission. During reception, interfering signals may be picked up by the antennas/sensors and noise level also increases when side lobe levels are high. Hence it is desirable to have low side lobe levels in beamformers.

### B.1 AMPLITUDE TAPERING

Sidelobe levels can be reduced using amplitude tapering of the elements of the array (Donoho and Johnstone, 1994; Roohollah Ghavamirad et al., 2017). Application of wavelet denoising along with amplitude tapering further enhances the level of reduction of sidelobes.

In appendix A, it was shown that the application of R-1D-MRT gives almost same denoising effect as wavelet denoising. This appendix analyzes the application of amplitude tapering along with R-1D-MRT instead of amplitude tapering with wavelet

denoising for side lobe reduction and a comparison of the reduction in side lobes obtained using both these methods is presented.

Amplitude tapering is the most commonly used method for reducing the side lobe levels of a uniform array. The excitation amplitudes of the array elements are varied using taper windows. Different taper windows are used for varying the amplitude of the array elements. The sidelobe reduction using Hamming, Hanning, Kaiser, Dolph-Chebyshev, Taylor, Blackmann, Bartlett and Gaussian windows are analyzed here.

The Hamming window is given by

$$\Delta w = 0.54 + 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \text{ for } -\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \quad \text{B.1}$$

The Hanning window is given by

$$\Delta w = 0.5 + 0.5 \cos\left(\frac{2\pi n}{N-1}\right) \text{ for } -\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \quad \text{B.2}$$

The Kaiser window is given by

$$\Delta w = \frac{I_0\left[\alpha\sqrt{1-\left(\frac{2n}{N-1}\right)^2}\right]}{I_0(\alpha)} \text{ for } |n| \leq \frac{N-1}{2} \quad \text{B.3}$$

where  $\alpha$  is the adjustable parameter and  $I_0(x)$  is the zeroth order Bessel function of the first kind of order zero.

The Dolph-Chebyshev window is given by

$$\Delta w = \cos^{-1}\left[\left(\cosh\frac{\cosh^{-1}10^{A/20}}{N-1}\right)^{-1}\right] \quad \text{B.4}$$

The Taylor window is given by

$$\Delta w = 1 - \sum_{m=1}^{n-1} F_m \cos\left(\frac{2\pi mn}{N-1}\right), \quad -\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \quad \text{B.5}$$

$F_m = F(m, \bar{n}, \eta)$  are the Taylor coefficients of the  $m^{\text{th}}$  order,  $\eta$  and  $\bar{n}$  show ratio of mainlobe over sidelobe level and number of sidelobes at equal level respectively.

The Blackman window is given by

$$\Delta w = 0.42 - 0.5 \cos\left(\frac{2\pi n}{N}\right) + 0.08 \cos\left(\frac{4\pi n}{N}\right) \quad 0 \leq n \leq N \quad \text{B.6}$$

The Bartlett window is given by

$$\Delta w = 1 - \frac{2|n|}{N-1}, \quad -\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \quad \text{B.7}$$

The Gaussian window is given by

$$\Delta w = e^{-\frac{1}{2}\left(\frac{n-(N-1)/2}{\sigma(N-1)/2}\right)^2}, \quad \sigma \leq 0.5 \quad \text{B.8}$$

## **B.2 COMPARISON OF R-1D-MRT BASED SIDELOBE REDUCTION AND WAVELET BASED SIDELOBE REDUCTION**

Wavelet denoising along with amplitude tapering is applied to a conventional beamformer. The normalized beam pattern obtained is compared with the normalized beam pattern of amplitude tapered R-1D-MRT conventional beamformer. The comparison is repeated for various taper windows. The normalized beam patterns obtained are plotted in Fig B.1.

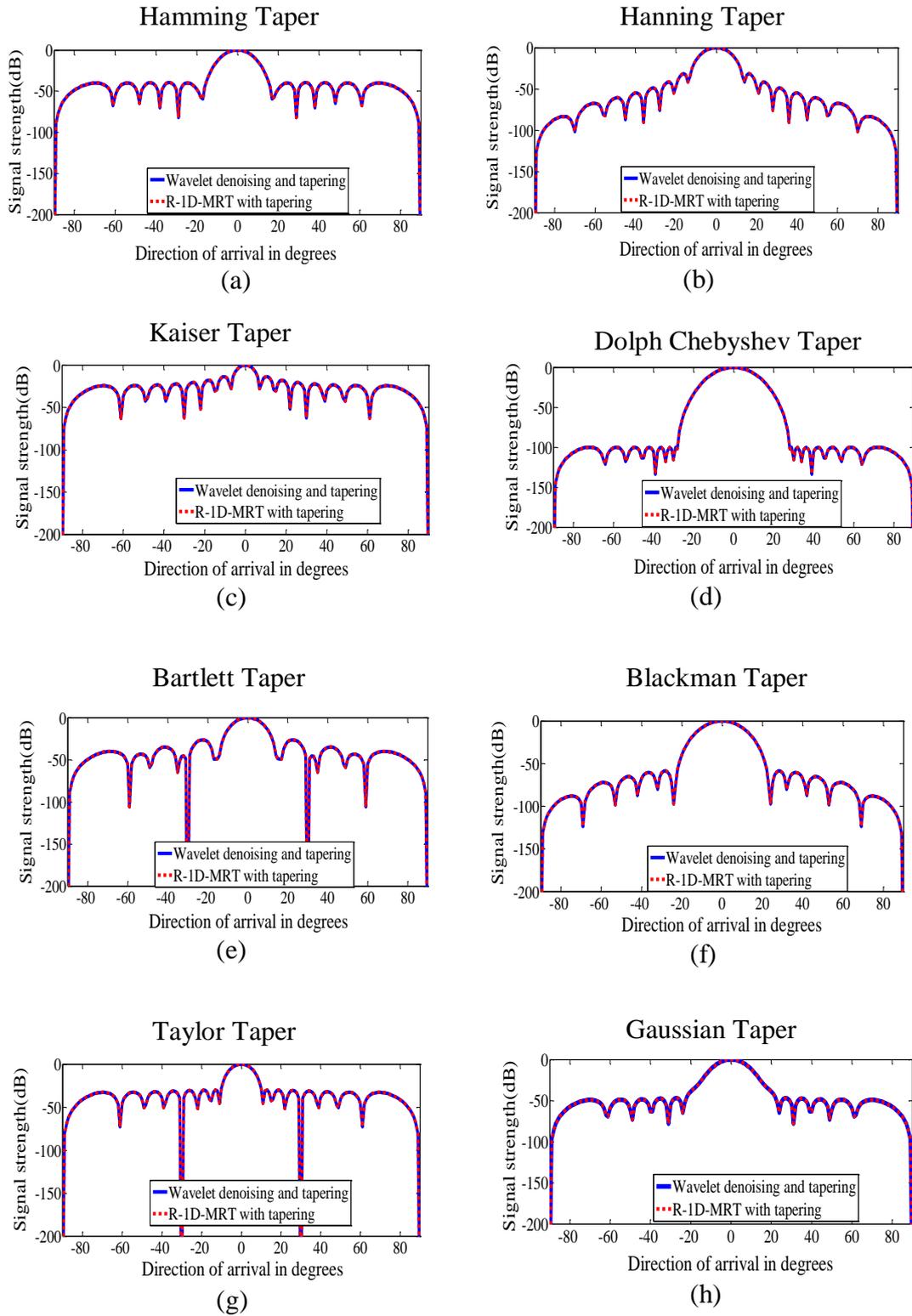


Fig B.1 Comparison of normalized beam patterns of CBF with Rigrsure wavelet denoising and various taper windows and R-1D-MRT with various taper windows [ $N_s = 16$ , Source angle =  $0^\circ$ ,  $N = 256$ , SNR = 30dB]

The comparison of the reduction in sidelobe levels of the normalized beam patterns using wavelet denoising with tapering and R-1D-MRT with tapering is presented in Table B.1.

Table B.1 Comparison of sidelobe reduction

Taper window	Reduction in sidelobe level (dB) (Conventional Beamformer with Rigrsure wavelet denoising and tapering)	Reduction in sidelobe level(dB) (R-1D-MRT based Conventional Beamformer with tapering)
Hamming	39	39
Hanning	31	31
Kaiser	13.68	13.68
Chebyshev	100.1	100.1
Taylor	13	13
Blackman	58.66	58.66
Bartlett	26.32	26.32
Gaussian	46.53	46.53

### B.3 INFERENCE

The normalized beam patterns obtained using R-1D-MRT and tapering are at par with those obtained using wavelet denoising and tapering as seen in Fig B.1.

The normalized beam patterns of wavelet denoised cum tapered signals and R-1D-MRT transformed cum tapered signals exhibit almost equivalent reduction in sidelobe levels as observed from Fig B.1 and Table B.1. This validates the fact that R-1D-MRT with tapering can be used as a viable alternative to wavelet denoising and

tapering for sidelobe reduction in conventional beamformers. Wavelet denoising of signals requires more processing as 3 processes are to be performed namely 1) application of wavelet transform, 2) denoising using thresholds and 3) application of inverse wavelet transform. These three processes must be performed before tapers are applied, each of which requires  $N$  operations, where  $N$  is the data size. In comparison lesser processing is required with the R-1D-MRT method as the R-1D-MRT transformation requires only an addition process of  $N/2$  operations and the taper can be applied immediately after this process, thus offering significant reduction in computational complexity. Thus the use of R-1D-MRT with tapering offers a less complex method of reducing sidelobes in conventional beamformers.

## REFERENCES

- [1] **Alwan, E.A., M. LaRue, W. Khalil and J.L. Volakis** (2013) Experimental validation of a coding-based digital beamformer, *IEEE Antennas and Propagation Society International Symposium (APSURSI)*, Orlando, FL, 398-399.
- [2] **Bhadran, V.** *Development and Implementation of Visual Approach and Parallel Distributed Architecture for 2-D DFT and UMRT Computation*. Ph.D dissertation, Cochin University of Science and Technology, Kochi, 2009.
- [3] **Bhadran, V., C. R. Rajesh, and R. Gopikakumari** (2008), Visual representation of 2-D DFT in terms of 2x2 data: A pattern analysis, Proceedings of *International Conference on Computing, Communication and Networking (ICCCN)*, 1.
- [4] **Bloemendal, B., J. van de Laar and P. Sommen** (2012) Beamformer design exploiting blind source extraction techniques, Proceedings of *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, 2589-2592.
- [5] **Bucris, Y., I. Cohen and M. A. Doron** (2010) Robust focusing for wideband MVDR beamforming, Proceedings of *2010 IEEE Sensor Array and Multichannel Signal Processing Workshop*, Israel, 1-4.
- [6] **Chang, D.C. and C.N. Hu** (2012) Smart Antennas for Advanced Communication Systems, *Proceedings of the IEEE*, **100**, 2233-2249.
- [7] **Chen, Y., F. Wang, J. Wan and K. Xu** (2016) Robust adaptive beamforming based on matched spectrum processing with little prior information, Proceedings of *2016 IEEE 13th International Conference on Signal Processing (ICSP)*, China, 404-408.
- [8] **Dai, J. and Y. Wang** (2010) NLMS adaptive algorithm implement based on FPGA, Proceedings of *3rd International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, Shenyang, 422-425.
- [9] **Di Martino, G., and A. Iodice** (2017), Passive beamforming with coprime arrays, in *IET Radar, Sonar & Navigation*, **11- 6**, 964-971.
- [10] **Dmochowski, J., J. Benesty and S. Affes** (2008) Linearly Constrained Minimum Variance Source Localization and Spectral Estimation, *IEEE Transactions on Audio, Speech, and Language Processing*, **16**, 1490-1502.
- [11] **Do-Hong, T. and P. Russer** (2003) A new design method for digital beamforming using spatial interpolation, *IEEE Antennas and Wireless Propagation Letters*, **2**, 177-181.
- [12] **Donadio, M.P.**, <https://pdfs.semanticscholar.org/CIC> filter introduction.

- [13] **Donoho, D.L. and I.M. Johnstone** (1994) Ideal spatial adaptation via wavelet shrinkage *Biometrika*, **81**, 425-455.
- [14] **Elamaran, V., R. Vaishnavi, A.M. Rozario, S.M. Joseph and A. Cherian** (2013) CIC for decimation and interpolation using Xilinx system generator, Proceedings of *International Conference on Communication and Signal Processing*, Melmaruvathur, 622-626.
- [15] **El-Khamy., S.E. and M. Shokry** (2013) Enhanced performance of blind and non-blind adaptive arrays using wavelet beamforming, Proceedings of *International Conference on High Performance Computing & Simulation (HPCS)*, Helsinki, 347-351.
- [16] **Eugene E.** <https://www.sigmaplus.fr>, Simulation of Communication Systems, 2014.
- [17] **Girma, S.** (2012) *Investigation of Adaptive Beamforming Algorithm for Smart Antenna System to Improve the Effect of Angle Separation*, MS Thesis, Addis Ababa University, Addis Ababa, Ethiopia.
- [18] **Hema Singh and Rakesh Mohan Jha** (2012) Trends in Adaptive Array Processing, *International Journal of Antennas and Propagation*.
- [19] **Homana, I., Topa, M., Botond, K. and Contan, C.** (2011) Behaviour of variable step size NLMS algorithms for acoustic echo cancellation, Proceedings of *10th International Symposium on Signals, Circuits and Systems (ISSCS)*, Lasi, 1-4.
- [20] **Huang, H.C., and J. Lee** (2012) A new variable step-size NLMS algorithm and its performance analysis', *IEEE Transactions on Signal Processing*, **60**, 2055-2060.
- [21] **Imtiaj, S.K., I.S. Misra, and R. Biswas** (2012) A comparative study of beamforming techniques using LMS and SMI algorithms in smart antennas, Proceedings of *International Conference on Communications, Devices and Intelligent Systems*, 246-249.
- [22] **Jakobsson, A. and S.R. Alty** (2006) On the efficient implementation and time-updating of the linearly constrained minimum variance beamformer, *14th European Signal Processing Conference*, Florence, 1-5.
- [23] **Jiang, M., W. Liu and Y. Li** (2016) Adaptive Beamforming for Vector-Sensor Arrays Based on a Reweighted Zero-Attracting Quaternion-Valued LMS Algorithm, *IEEE Transactions on Circuits and Systems II: Express Briefs*, **63**, 274-278.
- [24] **Jimaa, S.A., A. Al-Simiri, R.M. Shubair, and T. Shimamura** (2009) Convergence evaluation of variable step-size NLMS algorithm in adaptive

- channel equalization, Proceedings of *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Ajman, 145–150.
- [25] **JinYou Q., Z. JianYun and Z. Xin'an** (2007) A Widely-Linear LMS Algorithm for Adaptive Beamformer, Proceedings of *International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications*, Hangzhou, 1060-1063.
- [26] **Kevin McClaning**, Tom Vito, Radio Receiver Design, Noble Publishing Corporation, 2000.
- [27] **Khanna,R., R. Mehra and Chandni** (2017), FPGA based implementation of pulsed radar with time delay in digital beamforming using partially serial architecture, Proceedings of *3rd International Conference on Computational Intelligence & Communication Technology (CICT)*, Ghaziabad, India, 1-6.
- [28] **Khedekar,S., and M. Mukhopadhyay** (2016), Digital beamforming to reduce antenna side lobes and minimize DOA error, Proceedings of *6 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPE5)*, Paralakhemundi, Odisha, India, 1578-1583.
- [29] **Khumane, D.D., A.N. Jadhav, and S.D. Bhosali** (2011) Dynamic analysis for direction of arrival estimation and adaptive beamforming for smart antennas, *International Journal of Electronics and Communication Engineering*, **4**, 415–423.
- [30] **Kim S., and Y.E. Wang** (2009) Two-Dimensional Planar Array for Digital Beamforming and Direction-of-Arrival Estimations, *IEEE Transactions on Vehicular Technology*, **58**, 3137-3144.
- [31] **Kiong, T.S., S. Balasem Salem, J.K.S. Paw, K.P. Sankar, and Soodabeh Darzi** (2014) Minimum variance distortionless response beamformer with enhanced nulling level control via dynamic mutated artificial immune system, *The Scientific World Journal*, 1-9.
- [32] **Koh, C.L., and S. Weiss** (2005) Constant beam width generalised sidelobe canceller, Proceedings of *IEEE/SP 13th Workshop on Statistical Signal Processing*, Novosibirskpp, 283-288.
- [33] **Krim, H and M.Viberg** (1996) Two decades of array signal processing research: the parametric approach, *IEEE Signal Processing Magazine*, **13**, 67-94.
- [34] **Lei Wang**, Array Signal Processing Algorithms for Beamforming and Direction Finding, PhD Thesis, University of New York, 2009.
- [35] **Li, H. and Tian, H.** (2009) A new VSS-LMS adaptive filtering algorithm and its application in adaptive noise jamming cancellation system, Proceedings of

- IEEE Circuits and Systems International Conference on Testing and Diagnosis*, Chengdu, 1–4.
- [36] **Lin, G., Y. Li, and B. Jin** (2010) Research on the algorithms of robust adaptive beamforming, Proceedings of *International Conference on Mechatronics and Automation (ICMA)*, Xi'an, 751–755.
- [37] **Lin, G., Y. Li and B. Jin** (2010) Research on the algorithms of robust adaptive beamforming, Proceedings of *IEEE International Conference on Mechatronics and Automation*, Xi'an, 751-755.
- [38] **Liu, F.L., J.K. Wang, C.Y. Sun, and R.Y. Du** (2011) Robust MVDR beamformer for nulling level control via multi-parametric quadratic programming, *Progress In Electromagnetics Research C*, vol. 20, pp. 239-254, 2011.
- [39] **Loehning, M., T. Hentschel and G. Fettweis** (2000), Digital Down Conversion in Software Radio Terminals, Proceedings of *10th European Signal Processing Conference, EUSIPCO 2000*, 1517-1520.
- [40] **Long, M. and W. Xin-yue** (2010) Robust adaptive beamforming based on sliding-window RLS algorithm, Proceedings of *2nd International Conference on Signal Processing Systems*, Dalian, 3, 793-797.
- [41] **Lu, S., J. Sun, G. Wang and J. Tian** (2012) A Novel GSC Beamformer Using a Combination of Two Adaptive Filters for Smart Antenna Array, *IEEE Antennas and Wireless Propagation Letters*, 11,377-380.
- [42] **Lu, S., J. Sun, G. Wang and J. Tian** (2012) A Novel GSC Beamformer Using a Combination of Two Adaptive Filters for Smart Antenna Array, *IEEE Antennas and Wireless Propagation Letters*, 11, 377-380.
- [43] **Ma, C. Y., T.C. Hsung, D.P.K. Lun, K.C. Ho and H.K. Kwan**(2007) Denoising for Generalized Sidelobe Canceller, Proceedings of *IEEE International Symposium on Circuits and Systems*, New Orleans, LA, 1581-1584.
- [44] **Manolakis, D.G., V.K. Ingle, S.M. Kogon** Statistical and adaptive signal processing: spectral estimation, signal modelling, adaptive filtering, and array processing, Artech House, 2005.
- [45] **Mao, X., W. Li, Y. Li, Y.Sun, and Z. Zhai** (2015), Robust Adaptive Beamforming against Signal Steering Vector Mismatch and Jammer Motion, *International Journal of Antennas and Propagation*, 1-12.
- [46] **Markovich-Golan, S., S. Gannot and I. Cohen** (2012) A sparse blocking matrix for multiple constraints GSC beamformer, Proceedings of *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, 197-200.

- [47] **Markovich-Golan, S., S. Gannot and I. Cohen** (2012) Low-Complexity Addition or Removal of Sensors/Constraints in LCMV Beamformers, *IEEE Transactions on Signal Processing*, **60**, 1205-1214.
- [48] **Mathews C.P. and M.D. Zoltowski** (2002) Beamspace ESPRIT for multi-source arrival angle estimation employing tapered windows, Proceedings of *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, FL, USA, **3**,3009-3012.
- [49] **Mathias Weiß** (2009), Digital Antennas, <https://www.sto.nato.int/publications>.
- [50] **Meenakshy K. and R. Gopikakumari** (2009) Texture descriptors based on 1-D MRT, *International Journal of recent trends in Engineering*, **2**, 1-3.
- [51] **Michael, L.P.**, Robust Adaptive Signal Processors, PhD Thesis, Virginia Polytechnic Institute and State University, 2003.
- [52] **Mohammad, T.I. and A.A.R. Zainol** (2006) MI-NLMS adaptive beamforming algorithm for smart antenna system applications, *Journal of Zhejiang University Science*, **7**, 1709-1716.
- [53] **Pan, C., J. Chen, and J. Benesty** (2014) Performance Study of the MVDR Beamformer as a Function of the Source Incidence Angle, Proceedings of *IEEE / ACM Transactions on Audio, Speech and Language Processing*, **22**, 67-79.
- [54] **Preetha, B., V. Bhadrans, and R. Gopikakumari** (2012) A new algorithm to compute forward and inverse 2-D UMRT for N-a power of 2, Proceedings of *Second Int. Conference on Power, Signals, Control and Computation (EPSICON)*, 1.
- [55] **Qin, B., Y. Cai, B. Champagne, M. Zhao and S. Yousefi** (2013) Low-complexity variable forgetting factor constant modulus RLS-based algorithm for blind adaptive beamforming, Proceedings of *Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, 171-175.
- [56] **Raghunath, K. J., and V.U. Reddy** (1992) Finite data performance analysis of MVDR beamformer with and without spatial smoothing, *IEEE Transactions on Signal Processing*, **40**, 2726-2736.
- [57] **Rahmani, M and M.H. Bastani** (2014) Robust and rapid converging adaptive beamforming via a subspace method for the signal-plus-interference covariance matrix estimation, *IET Signal Processing*, **8**, 507-520.
- [58] **Rajesh, C.R.** *Development of a new transform MRT*, PhD Dissertation, Cochin University of Science and Technology, Kochi, 2009.
- [59] **Rajesh, C.R., M.S. AnishKumar, and R. Gopikakumari** (2007), An invertible transform for image representation and its application to image

- compression, Proceedings of *the Int. Symposium on Signal Processing and its Applications (ISSPA)*, 1.
- [60] **Ramachandran K. I. and K.P. Soman**, Insight into wavelets: From theory to practice, PHI learning 2010.
- [61] **Razia, S., T. Hossain and M.A.Matin** (2012) Performance analysis of adaptive beamforming algorithm for smart antenna system, Proceedings of *International Conference on Informatics, Electronics & Vision (ICIEV)*, Dhaka, 946-949.
- [62] **Roncero D.F.**, A study of QR decompositions and Kalman Filter implementation, M Tech Thesis, Signal Processing, School of Electrical Engineering, Kungliga Tekniska Hgskolan, Sweden, 2014.
- [63] **Roohollah, G., B. Hossein and A.S. Mohammad** (2017), Nonlinear FM Waveform Design to Reduction of sidelobe level in Autocorrelation Function, Proceedings of *25<sup>th</sup> Iranian Conference on Electrical Engineering*, Tehran.
- [64] **Sajith Mohan, C., S.B. Menon, K.E. Nissar and K.P. Arun Kumar** (2012) Adaptive beamforming with instantaneous convergence & its efficient implementation in DSP hardware, Proceedings of *2012 Annual IEEE India Conference (INDICON)*, Kochi, 700-704.
- [65] **Sathish, R. and C.V. Anand** (2004) Spatial wavelet packet denoising for improved DOA estimation, Proceedings of *14th IEEE Signal Processing Society Workshop Machine Learning for Signal Processing*, Sao Luis, 745-754.
- [66] **Serra J. and M. Nájar** (2014) Asymptotically Optimal Linear Shrinkage of Sample LMMSE and MVDR Filters, *IEEE Transactions on Signal Processing*, **62**, 3552-3564.
- [67] **Shen, F., F. Chen and J. Song** (2015) Robust Adaptive Beamforming Based on Steering Vector Estimation and Covariance Matrix Reconstruction, *IEEE Communications Letters*, **19**, 1636-1639.
- [68] **Shengkui, Z., M. Zhihong, and K. Suiyang** (2006) Modified LMS and NLMS algorithms with a new variable step size, Proceedings of *9<sup>th</sup> International Conference on Control, Automation, Robotics and Vision*, Singapore, 1-6.
- [69] **Shi, Y. M., L. Huang, C. Qian and H.C. So** (2015) Shrinkage Linear and Widely Linear Complex-Valued Least Mean Squares Algorithms for Adaptive Beamforming, *IEEE Transactions on Signal Processing*, **63**, 119-131.
- [70] **Shin, H.C., A.H. Sayed, and W.J. Song** (2004) Variable step-size NLMS and affine projection algorithms, *Signal Processing Letters*, **11**, 132 - 135.

- [71] **Shu, X. and S. Han** (2009) Improvement of DOA Estimation Using Wavelet Denoising, Proceedings of *First International Conference on Information Science and Engineering*, Nanjing, 587-590.
- [72] **Shubair, R.M. and A. Merri** (2005) A convergence study of adaptive beamforming algorithms used in smart antenna systems, Proceedings of *11th International Symposium on Antenna Technology and Applied Electromagnetics [ANTEM 2005]*, France, 1-5.
- [73] **Simon Haykin** Adaptive Filter Theory, Pearson, 4<sup>th</sup> Edition, 2012.
- [74] **Song, X., J. Wang, and Y. Han** (2009), Progress In Electromagnetics Research Symposium, Beijing, China, 405-411.
- [75] **Souden, M., J. Benesty and S. Affes** (2010) A Study of the LCMV and MVDR Noise Reduction Filters, *IEEE Transactions on Signal Processing*, **58**, 4925-4935.
- [76] **Srar, J.A., K.S. Chung and A. Mansour** (2010) Adaptive Array Beamforming Using a Combined LMS-LMS Algorithm, *IEEE Transactions on Antennas and Propagation*, **58**, 3545-3557.
- [77] **Srar, J.A., K.S. Chung, and A. Mansour** (2010) Adaptive array beamforming using a combined LMS-LMS algorithm, *IEEE Transactions on Antennas and Propagation*, **58**, 3545-3557.
- [78] **Talmon, R., I. Cohen and S. Gannot** (2009), Convolutional Transfer Function Generalized Sidelobe Canceler, *IEEE Transactions on Audio, Speech, and Language Processing*, **17**, 1420-1434.
- [79] **Torres Rossario, J.A.** *Implementation of a phased array antenna using digital beamforming*, (Master of Sciences Thesis), University of Puerto Rico, 2005.
- [80] **Torres-Rossario J. A.**, *Implementation of a phased array antenna using digital beamforming*. (Master of Sciences Thesis). University of Puerto Rico, 2005.
- [81] **Tuthill, J., Y.H. Leung and I. Thng** (1995) Adaptive RLS filters with linear and quadratic constraints, Proceedings of *International Conference on Acoustics, Speech, and Signal Processing*, Detroit, MI, **2**, 1424-1427.
- [82] **VanVeen, B.D. and K.M. Buckley** (1988) Beamforming: a versatile approach to spatial filtering, *IEEE ASSP Magazine*, **5**, 4-24.
- [83] **Waheed, O. T., A. Shabra and I. M. Elfadel** (2015) FPGA methodology for power analysis of embedded adaptive beamforming, Proceedings of *International Conference on Communications, Signal Processing and their Applications (ICCSPA-2015)* Sharjah, 1-6.

- [84] **Wang L. and R.C. de Lamare** (2007) Constrained Constant Modulus RLS-based Blind Adaptive Beamforming Algorithm for Smart Antennas, Proceedings of *4th International Symposium on Wireless Communication Systems*, Trondheim, 657-661.
- [85] **Widrow, B. and Streams, S.** (1985) *Adaptive Signal Processing*, Prentice Hall, Englewood Cliffs, NJ.
- [86] **Xin, D., L. Guisheng, L. Hongqing and T. Haihong** (2006) Robust Constrained LMS Adaptive Beamformer, Proceedings of *CIE International Conference on Radar*, Shanghai, 1-4.
- [87] **Xin, X., Y. Qinghong, S. Zhaolin, L. Qingjiang, W. Yinan and L. Wenli** (2010) The Realization of Digital Beamforming Based on FPGA and DSP, Proceedings of *International Conference on Intelligent System Design and Engineering Application*, Changsha, 713-716.
- [88] **Yang, J., H. Xi and F. Yang** (2005) RLS-based blind adaptive beamforming algorithm for antenna array in CDMA systems, Proceedings of *IEEE International Conference on Information Acquisition*, Hong Kong, 6.
- [89] **Yasin M. and P. Akhtar** (2012) Performance analysis of Bessel beamformer with LMS algorithm for smart antenna array, Proceedings of *International Conference on Open Source Systems and Technologies*, Lahore, 1-5.
- [90] **Yu, L., W. Liu and R. Langley** (2010) SINR Analysis of the Subtraction-Based SMI Beamformer, *IEEE Transactions on Signal Processing*, **58**, 5926-5932.
- [91] **Zaharov V.V. and M. Teixeira** (2008) SMI-MVDR Beamformer Implementations for Large Antenna Array and Small Sample Size, *IEEE Transactions on Circuits and Systems I: Regular Papers*, **55**, 3317-3327.
- [92] **Zaharov V.V. and M. Teixeira** (2005) Modified Implementations of RLS Beamformer, Proceedings of *IEEE Antennas and Propagation Society International Symposium*, Washington, 3A, 275-278.
- [93] **Zaharov, V.V., A. Gonzalez, J. Acosta and M. Teixeira** (2007) Implementing a Vector RLS Smart Antenna Beamformer Using Xilinx System Generator, Proceedings of *2nd International Symposium on Wireless Pervasive Computing*, San Juan, 654-657.
- [94] **Zalawadia, K.R., T.V. Doshi and U.D. Dalal** (2011) Adaptive Beam Former Design Using RLS Algorithm for Smart Antenna System, Proceedings of *International Conference on Computational Intelligence and Communication Networks*, Gwalior, 102-106.

- [95] **Zhen-Hai Xu, Ming-Zui Chen and Bin Rao** (2013) The optimal LCMV beamformer under multiple desired signals case, Proceedings of *IET International Radar Conference*, Xi'an,1-4.

## List of papers published based on this research

### International Journals

1. **Mini P. R., S. Mridula, Binu Paul, P. Mohanan** (2016), A novel approach for improving performance of LMS beamformer, *International Journal of Signal Processing Systems*, **4**, 118 - 123.
2. **Mini P. R., S. Mridula, Binu Paul, P. Mohanan** (2016), FPGA Implementation of One Dimensional Reduced Mapped Real Transform based digital beamformer, *International Journal of Electronics Letters (Taylor and Francis)*, **5**, 1 - 12.
3. **Mini P. R., S. Mridula, Binu Paul, P. Mohanan** (2016), A novel algorithm for adaptive NLMS beamformer, *International Journal of Wireless and Mobile Computing (Inderscience)*, **10**, 122 - 129.

### International Conferences

1. **Mini P. R., S. Mridula, Binu Paul, P. Mohanan**, An approach for sidelobe reduction in conventional beamformers, *INDICON 2012*, Cochin, Dec 2012, 794 - 797.
2. **Mini P. R., S. Mridula, Binu Paul, P. Mohanan**, Conventional Beamformers based on One Dimensional Reduced Mapped Real Transform, *APSYM 2014*, Dec 2014, 103 - 106.

### National Conference

1. **Mohammed Yaseen, Mini P. R., S. Mridula**, Implementation of array theory of antennas using FPGA, 7th IETE ICON RFW 2014, Bangalore.

# CURRICULUM VITAE

**MINI P. R.**

ASSISTANT PROFESSOR  
DEPT. OF ELECTRONICS AND  
COMMUNICATION ENGINEERING  
FEDERAL INSTITUTE OF SCIENCE  
AND TECHNOLOGY  
ANGAMALY - 683577, KERALA



## Education

Degree	Institution	Period	Remarks
X	Alvernia Matric H.S.S., Coimbatore	1984-1985	First Class 85.6 %
XII	Avila Convent M.H.S.S., Coimbatore	1986-1987	First Class 87.4 %
BE	Government College of Technology, Coimbatore	1987 - 1991	First Class 72.6 %
M Tech	National Institute of Technology, Calicut	2006 - 2008	First Class CGPA 7.2
PhD	Cochin University of Science and Technology	Registered On 04.08.2009	

## Teaching Experience

Institution	Designation	Period
Federal Institute of Science and Technology, Angamaly	Assistant Professor	Sep. 2002 - till date
Amrita Viswa Vidyapeedom, Coimbatore	Lecturer	Aug. 2000 - Aug. 2002 Oct. 1997 - Jan. 1998
Government College of Technology, Coimbatore	Lecturer	Jul. 1993 - Oct. 1997 Aug. 1991 - Mar. 1992

## Industrial Experience

Institution	Designation	Period
Cavalcade Softwares Pvt Ltd, Perumbavoor	Faculty	Feb. 1998 - Jul. 2000
West Coast Engineering Corporation, Coimbatore	Design Engineer	Apr. 1992 - Jun. 1993

## Fields of research interest

Adaptive Signal Processing, Digital System Design, Smart Antennas.

## Publications

International Journal : 3  
International Conference : 2  
National Conference : 1

## Awards

Won the Best Teacher Award of FISAT in 2009

## Software skill set

Languages : VHDL, MATLAB®  
Software Packages : Xilinx System Generator, Altium Designer®, HFSS

## Personal details

Age and date of birth : 47, 21.10.1969  
Gender : Female  
Nationality : Indian  
Languages known : English, Malayalam, Hindi, Tamil  
E-mail : mini@fisat.ac.in  
Mobile : 94470 49355  
Permanent Address : Apt. No. 4C, Rajmandir Apartments, Kaduval,  
Perumbavoor, Ernakulam - 683542, Kerala.