Robust techniques in array signal processing

Lei, Lei

2009

Lei, L. (2009). Robust techniques in array signal processing. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/42169>

<https://doi.org/10.32657/10356/42169>

ROBUST TECHNIQUES IN ARRAY SIGNAL PROCESSING

LEI LEI

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2009

Acknowledgments

First of all, I would like to express my deepest appreciation and sincere gratitude to my supervisor, Associate Professor Henry Lew, for his guidance, invaluable advices and ideas in my study over the past four years in Nanyang Technological University. It is he who gave me strong support and sincere encouragement throughout the tough times of the research. His keen thoughts impress me so much and I think it will benefit me in my future research and work. He helps me a lot in problem solving and research orientation with his broad and profound knowledge. Within these years, I not only learned specialized knowledge from him, but also learned the spirit of exploring science. This research work will not be possible without his guidance, and encouragement. The completion of this thesis would be impossible without his close supervision and devotion to education.

I would also like to thank late Dr. Ronda Venkateswarlu, my associate supervisor, who employed me as a research student in his research team and offered me the research scholarship of Institute for Infocomm Research of Singapore. He also gave me generous help and encouragement. Unfortunately, I will have no chance

to share my research product and this thesis with him because he passed away in 2006. May his soul rest in peace!

Special thanks go to my families, who always support me and encourage me with their great patience whenever and wherever.

Contents

Table of Contents	i
Acknowledgements	ii
Abstract	xi
List of Abbreviations	xiii
List of Symbols	xv
List of Figures	xix
List of Tables	xxvi
1 Introduction	1
1.1 Motivation	1
1.2 Background	2
1.3 Objectives	7
1.4 Major Contributions of the Thesis	8
1.5 Organization of Thesis	9
2 An Overview of Array Processing	11
2.1 Introduction	11
2.2 Concepts of Array Signal Processing	12

2.3	Signals in Space	13
2.4	Beamforming System	15
2.5	Data Model of Narrowband Signals	17
2.5.1	Signal Representation	17
2.5.2	Covariance Matrix	19
2.5.3	Isotropic Noise Field	20
2.6	Array Geometry and Uniform Linear Array	22
2.7	Beamformer Classification and Robust Concepts	22
2.8	Performance Measures	24
2.8.1	Beampattern	25
2.8.2	Signal-to-Interference-plus-Noise Ratio	27
2.8.3	Array Gain	27
2.8.4	Directivity	28
3	Robust Techniques for Data-Independent Beamformers	29
3.1	Introduction	29
3.1.1	Delay-and-Sum Beamformer	31
3.1.2	Sidelobe-Control Beamformers	31
3.1.3	Supergain Endfire Beamformer	32
3.2	Performance Analysis for Perturbed Arrays	34
3.2.1	Assumptions of the Errors Model	36

3.2.2	Expected Beampattern	37
3.2.3	Directivity of Expected Beampattern	45
3.2.4	Variance of Perturbed Beampatterns	46
3.3	Performance Comparison for Different Array Syntheses	47
3.4	Robust Problems for Data-Independent Arrays	51
3.5	Robust Method I	52
3.5.1	Underlying Idea of Proposed Method	52
3.5.2	Problem Formulation	52
3.5.3	Calculation of Matrix $\mathbf{Q}$	54
3.5.4	Robust Solution for Independent Errors Case	56
3.6	Robust Method II	56
3.6.1	Underlying Idea of Proposed Method	56
3.6.2	Problem Formulation	57
3.7	Numerical Examples	58
3.7.1	Independent Errors Case	59
3.7.2	Correlated Errors Case	65
3.7.3	Design Perturbation Errors	68
3.7.4	Dolph-Chebyshev Array	71
3.8	Comparison of Two Proposed Robust Methods	75
4	Robust Techniques for Adaptive Beamformers	76

4.1	Introduction	76
4.2	Concepts of Adaptive Beamformers	78
4.2.1	Fundamentals	78
4.2.2	Capon Beamformer	79
4.3	Performance Analysis for Perturbed Capon Array	82
4.3.1	SINR of Perturbed Cases	82
4.3.2	Expected Performance of Perturbed Cases	88
4.4	Existing Robust Methods for Adaptive Beamformers	91
4.4.1	Principles of Robust Methods	91
4.4.2	Classification of Robust Capon Beamformer	93
4.5	Robust Adaptive Method 1	94
4.5.1	Phase Errors in ASV	94
4.5.2	Problem Formulation	95
4.5.3	Implementation of Robust Method	97
4.5.4	Numerical Results	100
4.5.5	Circular Array	107
4.5.6	Conclusions	111
4.6	Robust Adaptive Method 2	111
4.6.1	Robust Capon Beamformers	112
4.6.2	Overestimated Uncertainty Region in RCBs	113

4.6.3	Analysis of the Uncertainty Region	114
4.6.4	Implementation of Robust Method	118
4.6.5	Numerical Examples	121
4.6.6	Conclusion	127
5	Conclusions and Future Work	129
5.1	Conclusions	129
5.1.1	Perturbation Model	129
5.1.2	Data-independent Array	130
5.1.3	Adaptive Array	132
5.2	Recommendations	134
5.3	Future Work	135
A	Appendix of Chapter 3	138
A.1	Directivity of Standard Linear Array	138
A.2	Minimum Sensitivity Term in Independent Errors Case	139
A.3	Directivity of Expected Beampattern	140
A.4	Variance of Perturbed Beampatterns	141
A.5	Covariance Matrix of Spherically Isotropic Noise Field	143
A.6	Covariance Matrix of Cylindrically Isotropic Noise Field	144
B	Array Synthesis	145
B.1	Taylor-Kaiser Array	145

B.1.1	Kaiser Window	145
B.1.2	Taylor-Kaiser Weighting	146
B.2	Supergain Endfire Array	147
B.2.1	Visible Region	147
B.2.2	Schelkunoff's Weighting	147
B.3	Dolph-Chebyshev Array	148
B.3.1	Chebyshev Polynomials	148
B.3.2	Dolph-Chebyshev Synthesis	150
B.3.3	Synthesis Procedure	152
C	Perturbation Model for Correlated Position Errors	153
C.1	Model Assumptions	153
C.2	Generation of Correlated Sequence from IID Sequence	154
D	Appendix of Chapter 4	158
D.1	Comparison of Estimation Power	158
D.2	SINR of Perturbed Cases	159
D.3	Expected SINR of Perturbed Cases	161
D.4	Random Steering Direction Error	162
D.5	Random Position Errors	163
D.6	Derivative in Section 4.5.2	164
D.7	Distribution of ζ in Section 4.5.3	164

E Author's Publications	166
Bibliography	168

Abstract

Array signal processing, which collects and combines signals from an array, can obtain high spatial discrimination and an adaptive response that a single sensor can not achieve. Various array processing techniques were developed to enhance signal-to-noise ratio or to estimate the temporal or spatial characteristics of the observed signal in harsh environments. In practical array systems, however, some of the assumptions on the environment, sources, or array can be wrong or imprecise. Such environmental imperfections and array uncertainties will cause a mismatch between the nominal and actual data vector, which leads to a degraded performance. The impaired effects are especially fatal for adaptive beamformers.

In this thesis, we analyzed the degraded performance of the data-independent and adaptive arrays using a perturbation model. Some analytical expressions of the performance measures, e.g., the expected beam pattern and the expected SINR (signal-to-noise-plus-interference-ratio) using the proposed perturbation model are derived and discussed. This is helpful in understanding how the perturbations affect the performance of arrays qualitatively and quantitatively.

Several robust methods which combat the calibration errors in the two types of arrays are proposed in this thesis. For data-independent arrays, two methods based on the MSE (mean square error) criterion are presented. The first robust method minimizes the MSE between the perturbed and ideal responses, and it can be considered as the most insensitive array in the perturbed situation. The second robust method has a pre-defined set of quiescent weights and the robustness is obtained by minimizing the MSE between the perturbed and quiescent responses. The numerical results show that the robust II method can offer higher resolution and lower sidelobes performances when the perturbation is not very large, while the robust I method can offer a more stable performance in perturbed situations. The recommendation for the choice of the two robust methods is given according to two different environmental and perturbed situations.

Two robust methods for adaptive arrays are proposed to mitigate the degradation caused by two different types of errors. One method, which combats phase errors, can offer robustness by estimating the true phase vector of the desired signal. The proposed beamformer does not suffer from performance loss in interference rejection. Another robust method which deals with large steering errors is presented based on traditional RCBs (robust Capon beamformers). Higher SINR performance can be obtained than that of traditional RCBs since the proposed method effectively shrinks the uncertainty region by compensating the steering errors.

List of Abbreviations

Abbreviations	Full Expression
AG	array gain
ASV	array steering vector
BF	beamformer
DFT	discrete Fourier transform
DI	directivity index
DOA	direction of arrival
DOF	degree of freedom
DSB	delay-and-sum beamformer
EVD	eigenvalue decomposition
FIR	finite impulse response
FFT	fast Fourier transform
GSC	generalized sidelobe canceller
IFFT	inverse fast Fourier transform
IID	independent and identically distributed

LCMV	linearly constrained minimum variance
MSE	mean square error
PDF	probability density function
RAB1	robust adaptive beamformer 1
RAB2	robust adaptive beamformer 2
SCB	standard Capon beamformer
SINR	signal-to-interference-plus-noise ratio
SNR	signal-to-noise ratio
ULA	uniform linear array

List of Symbols

a	scalar
$\mathbf{a}$	vector
$\mathbf{A}$	matrix
$(\cdot)^*$	complex conjugate
$(\cdot)^T$	transpose of a vector or matrix
$(\cdot)^H$	complex conjugate transpose of a vector or matrix
$\approx$	approximate to
$\otimes$	Hadamard product
$\ \cdot\ $	Euclidean norm
$E\{\cdot\}$	expectation operator
$E_{error}\{\cdot\}$	expectation over perturbations
$E_{\theta,\phi}\{\cdot\}$	expectation over solid angle in beampattern
$Re\{\cdot\}$	real operator
$Im\{\cdot\}$	imaginary operator
$sinc(\cdot)$	sinc function

$trace(\cdot)$	trace operator
$det(\mathbf{A})$	determinant of a matrix
$ln(\cdot)$	natural logarithm
$\mathbf{a}_j$	data vector of the j th source signal
$\bar{\mathbf{a}}_j$	the nominal data vector of the j th source signal
$\hat{\mathbf{a}}$	the estimated ASV of the robust method
$B(\theta, \phi)$	beampattern
$B_0(\theta, \phi)$	unperturbed beampattern
$U_0(\theta, \phi)$	normalized unperturbed beampattern
$\overline{B(\theta, \phi)}$	expected beampattern over perturbations
$\overline{B_u(\theta, \phi)}$	normalized expected beampattern over perturbations
c	propagation speed of waveform
d	interspacing of the array
D	directivity
ϵ	square error
$\bar{\epsilon}$	mean square error
f	frequency of signals
$H(\theta, \phi)$	array response
$\mathbf{I}$	identity matrix
j	imaginary number = $\sqrt{-1}$
k	snapshot number

$\mathbf{k}$	wave number
$\mathbf{k}_i$	wave number of the i th source
L	correlation length
$\mathbf{n}(t)$	white noise signal
p_{out}	output power of the array
ρ	spatial correlation
$\mathbf{R}$	covariance matrix
$\hat{\mathbf{R}}$	estimated covariance matrix
$\hat{\mathbf{R}}_n$	covariance matrix of noise and interferences
θ_i	incident elevation angle of the i th source
ϕ_i	incident azimuth angle of the i th source
g_m	the gain of the m th sensor
ϕ_m	the phase of the m th sensor
$\mathbf{r}_m$	the position of the m th sensor
ϕ_m^0	original phase of the m th sensor
$\mathbf{r}_m^0$	original position of the m th sensor
Δg_m	gain error of the m th sensor
$\Delta \phi_m$	phase error of the m th sensor
$\Delta \mathbf{r}_m$	position error of the m th sensor
$\mathbf{G}$	correlation matrix of gain errors
Φ	correlation matrix of phase errors

$\mathbf{\Omega}$	correlation matrix of position errors
Ω	solid angle
α	mismatch angle between the ASV and true data vector
σ_B	standard deviation of perturbed beampatterns
σ_g	standard deviation of independent gain errors
σ_ϕ	standard deviation of independent phase errors
σ_r	standard deviation of independent position errors
$\mathbf{R}_e$	covariance matrix of array steering vector error
$\tilde{s}_i(t)$	the i th modulated source
$s_i(t)$	the i th demodulated source
t	time index
τ	time delay
$\mathbf{u}$	propagation direction unit
$\mathbf{v}(t)$	the vector of unwanted noise
$\tilde{x}_i(t)$	the signal incident at i th sensor
$x_i(t)$	the demodulated signal at i th sensor
$\mathbf{x}(t)$	array data vector (demodulated)
$y(t)$	output signal of array
δ	array steering vector error

List of Figures

2.1	Reference coordinates of an arbitrary array	14
2.2	Beamforming system	15
2.3	Narrowband beamformer with N sensors	17
2.4	Geometry of a uniform linear array	23
2.5	Measures of a beampattern	26
3.1	Beampatterns of different beamformers (delay-and-sum, Dolph-Chebyshev and Taylor-Kaiser arrays)	33
3.2	Weights of different beamformers (delay-and-sum, Dolph-Chebyshev and Taylor-Kaiser arrays)	33
3.3	Beampatterns of delay-and-sum and supergain endfire arrays	35
3.4	Individual perturbed beampatterns and ideal beampattern (delay-and-sum BF)	36

3.5	Expected beampattern vs position deviation (independent errors case)	42
3.6	Expected beampattern vs position deviation (correlated errors case $L = 5d$)	43
3.7	Expected beampattern vs position deviation (correlated errors case $L = d$)	43
3.8	Mean and mean+ σ_B beampattern ($\sigma_g = 0.1, \sigma_r = 0.2d$)	47
3.9	Mean and mean+ σ_B beampattern ($\sigma_r = 0.05d$)	48
3.10	Mean of highest sidelobe level vs standard deviation of the phase errors	50
3.11	Mean of Directivity index vs standard deviation of the phase errors	50
3.12	Beampattern comparison (standard endfire and robust II BF) . . .	59
3.13	Beampattern comparison (robust I BF in ideal and perturbed cases)	60
3.14	Array gain of the arrays in a spherically isotropic noise field (standard endfire, robust I and robust II BF)	61
3.15	Comparison of array gain in a cylindrically isotropic noise field (standard endfire, robust I and robust II BF)	62
3.16	Comparison of MSE in the independent errors case (standard endfire, robust I and robust II BF)	63

3.17 Mean highest sidelobe level in the independent errors case (standard endfire, robust I and robust II BF)	63
3.18 Non-uniform weighting function $p(\theta, \phi)$ of spatial region	64
3.19 Expected beampattern of robust II BF with uniform and non-uniform weighting functions	65
3.20 MSEs of the whole and sidelobe regions of the robust II BF with uniform and non-uniform weighting functions	66
3.21 Beampattern comparison in the correlated errors case (standard endfire and robust II BF)	67
3.22 Beampattern comparison in the correlated errors case (robust I BF in ideal and perturbed cases)	67
3.23 Comparison of MSE in the correlated errors case (standard endfire, robust I and robust II BF)	68
3.24 Array gain of the robust II array for different design perturbation errors	70
3.25 MSE of the robust II array for different design perturbation errors	70
3.26 Array gain of the robust I array for different design perturbation errors	71

3.27	MSE of the robust I array for different design perturbation errors	72
3.28	Beampattern comparison (standard Chebyshev and robust II BF) .	73
3.29	Beampattern comparison (robust I BF in ideal and perturbed cases)	74
4.1	Power estimation of the Capon and conventional beamformer	80
4.2	Individual perturbed beampatterns and ideal beampattern	82
4.3	SINR vs SNR in the ideal and perturbed cases	84
4.4	(a) Signal direction error vs mismatch angle α (b) Signal direction error vs the SINR in the ideal and perturbed cases	85
4.5	Signal direction error vs SINR (Capon and delay-and-sum beamformers)	87
4.6	(a) Expected mismatch angle α vs phase error σ_ϕ (b) Expected SINR vs phase error σ_ϕ	90
4.7	$Pr(\zeta \leq \xi')$	98
4.8	Beampattern of the beamformers (SCB, MPRBF, DLRBF and RAB1)	103
4.9	Value of $\ \phi(k+1) - \phi(k)\ $ vs iteration number	103
4.10	Histogram of the output SINR in RAB1	104

4.11	Estimated power of beamformers (SCB, MPRBF, DLRBF and RAB1)	104
4.12	Mean SINR of the beamformers vs phase deviations (SCB, MPRBF, DLRBF and RAB1)	105
4.13	Histogram of the output SINR in RAB1 (SNR = -5 dB)	105
4.14	Beampattern of the beamformers using the estimated covariance matrix (SCB, MPRBF, DLRBF and RAB1)(SNR = 10dB)	107
4.15	SINR comparison of the beamformers vs snapshots number (SCB, MPRBF, DLRBF and RAB1)	108
4.16	Mean SINR of the beamformers vs phase deviations by estimated covariance matrix (SCB, MPRBF, DLRBF and RAB1)	109
4.17	Circular Array	109
4.18	Beampattern of the beamformers using the estimated covariance matrix in a circular array (SCB, MPRBF, DLRBF and RAB1)	110
4.19	SINR comparison of the beamformers vs snapshots number in a circular array(SCB, MPRBF, DLRBF and RAB1)	110
4.20	Phase error versus sensor number	116
4.21	Radius of sphere versus errors (a) steering direction error $\Delta\theta$ (b) random phase error $\sigma_{\Delta\phi}$	117

4.22 SINR of RCBs	
(a) steering direction error $\Delta\theta$ (b) random phase error $\sigma_{\Delta\phi^\circ}$. . .	117
4.23 Qualitative comparison of the uncertainty regions	119
4.24 Estimated DOA $\hat{\theta}$ versus direction error $\Delta\theta$ in the RAB2	122
4.25 SINR comparison versus steering direction error $\Delta\theta$	
(SCB, RCB and RAB2)	122
4.26 Beampattern comparison of the SCB, RCB and RAB2	124
4.27 SINR comparison versus snapshots number	
(SCB, RCB and RAB2)	124
4.28 Estimation angle error versus input SNR	125
4.29 SINR comparison versus input SNR	
(SCB, RCB and RAB2)	125
4.30 SINR comparison versus number of sensors	
(SCB, RCB and RAB2)	126
4.31 SINR versus the size of the uncertainty region (ϵ, ϵ')	
(SCB, RCB and RAB2)	127
B.1 Endfire array zeros and visible region for $N = 6$, and $d = \lambda/4, d = \lambda/8$	148
B.2 Array weights: (a) N odd; (b) N even	149
B.3 $T_7(x)$	151

C.1	Correlated Sequence Generated by IID Sequence ($L = 5d$)	156
C.2	Positions of sensors with correlated errors of different correlation length	157

List of Tables

3.1	Main performance measures vs perturbation deviations (independent errors case)	44
3.2	Main performance measures vs correlation length (correlated errors case $\sigma_r = 0.2d$)	45
3.3	Performance measures (standard Chebyshev, robust I and robust II BF)	74

Chapter 1

Introduction

1.1 Motivation

Array signal processing, which collects and combines signals from an array (i.e., a group of sensors located at distinct spatial locations), can obtain high spatial discrimination and an adaptive response that a single sensor cannot achieve [1–4]. Various array processing techniques were developed to enhance signal-to-noise ratio (SNR) or to estimate the temporal or spatial characteristics (e.g, frequency content and location) of the observed signal in harsh environments. In recent decades, array signal processing has been widely used in RADAR, SONAR, microphone arrays, wireless communications, medical imaging and other fields [5–7].

There are two kinds of arrays, one is data-independent array, the other is adaptive arrays. For either data-independent or adaptive arrays, they are designed

based on the assumptions of perfect calibrations and ideal propagation. However, in practical applications, these assumptions may be violated by diverse errors. For example, the direction of the desired source may differ from the look direction, and the phases or gains of sensors may deviate from their nominal values. All these variations will give rise to perturbed responses and degraded performance in both kinds of arrays. Although the variations and perturbations may be different according to different applications, the framework is the same. Therefore, a detailed study of existing robust methods of array processing is needed. This thesis provides an analysis of the degraded array performance of the arrays and develops several robust algorithms to mitigate or overcome the effects of array perturbations.

The following section provides a general background of array processing and existing robust methods.

1.2 Background

Often the information we want is carried by a signal that is contaminated by noise or interferences. If the signal of interest (SOI) and noise come from different spatial regions, an array of sensors can be used as a spatial filter, which is called beamformer, to filter out the unwanted component. This filtering is done by combining the output of the array sensors with different weights such that a signal from a particular angle, or set of angles, is enhanced by a constructive combination. The unwanted signals from other angles, on the other hand, are rejected by destructive

combination. Alternatively, we may construct a model by assuming that the signal received by the array is the sum of several plane waves plus white noise. Within this model, an optimization problem is framed to find the parameter values that closely match the model. Thus, arrays can be used to estimate the characteristics of the signal source, e.g., the direction and speed.

Generally, array beamformers can be classified as either data-independent or adaptive, depending on how their weights are chosen. The weights in a data-independent beamformer are designed to present a particular response. The data-independent beamformers include the delay-and-sum approach as well as methods designed to produce a beampattern with desirable properties, e.g., a certain beamwidth or sidelobe level. The adaptive beamformers select weights based on the statistics of the array data to optimize the array response so that the output signal contains minimal contributions due to the noise and signals arriving from directions other than the desired direction.

In practical array systems, some of the assumptions on the environment, sources, or array can be wrong or imprecise. For example, the assumptions of plane wave signals and ideal propagation medium may not hold, or there may exist errors in the steering direction, phases and positions of the array sensors. Such environmental imperfections and array uncertainties will cause a mismatch between the nominal and actual data vector, which leads to a degraded performance. The impaired effects are especially fatal for adaptive beamformers, because the mismatch of the array steering vector (ASV) will make adaptive beamformers mis-

interpret the signal as interferences, and lead to "signal nulling". Whenever this happens, the performance of the adaptive beamformers may become even worse than that of a single sensor [8,9].

The data-independent and adaptive arrays have their own advantages and shortcomings. The implementation of data-independent beamformers is much simpler and without heavy computational load. However, this type of arrays cannot dynamically null out the directional interferences, and thus cannot obtain as high SINR as adaptive beamformers when there are large directional interferences. Adaptive beamformers, which give high SINR performance in error-free situations, are susceptible to the perturbations in the array parameters and the ambient environment. This will give rise to a drastically reduced array gain and an underestimation of the power of the SOI in a way that is much more serious than the data-independent array. The degraded performance is more severe especially for high SNR situations if no robust methods are implemented. Therefore, the choice of data-independent array or adaptive array is affected by the requirements of the system and the surrounding environment.

The topic of robust beamforming on data-independent array has been studied since the 1950s when Gilbert and Morgan analyzed the degraded performance for random errors present in the sensor gain, phase and position [10]. Several authors analyzed the relationship according to different types of array errors and offered corresponding robust approaches [11–14]. A closely related problem was examined by Uzsoky and Solymar [15]. Cox et al. addressed the problem of optimizing array

gain, while constraining the array's white noise gain. The algorithm gives the closed form of the robust weights by choosing the value of the assumed white noise gain to control the trade-off between directivity and robustness [16]. Unfortunately, all such robust approaches are ad hoc in nature since there is no easy and reliable way to choose an appropriate constraint value.

Due to the high sensitivity of adaptive beamformers to the array characteristics and environmental assumptions, robust techniques received much more attention after adaptive beamforming became popular in the 1970s. There are several types of robust approaches discussed in a number of papers. One common approach is to impose multiple gain constraints in different directions in the vicinity of the assumed one [8, 17, 18]. Another approach is to place derivative constraints on the steering vector [19–22]. In both of these cases, the constraints cause the beamwidth to be widened at the cost of reduced interference suppression. Another kind of robust algorithm, which is called the eigenspace-based robust method, uses the projection of the presumed steering vector onto the sample signal-plus-interference subspace instead of the presumed steering vector [23–25]. Although this robust beamformer is applicable to arbitrary steering vector mismatch, it only works well when the number of signals and interferences are exactly known and the SNR is high enough. An alternative, but popular, robust approach is diagonal loading method [26, 27], which offers robustness by adding a positive value to the diagonal terms of the covariance matrix. An explanation for the diagonal loading performance is that the added diagonal values effectively reduce the SNR of the

covariance matrix, and mitigates the sensitivity of adaptive arrays to the perturbations. Different robust methods which lead to the diagonal loading solution were proposed by several authors [28] [29]. Although the diagonal loading beamformers are applicable to arbitrary ASV mismatch, their main shortcoming is that there is no meaningful and reliable way to choose an appropriate diagonal loading factor. However, all of these robust adaptive methods are ad hoc because the robustness they offered not related to the uncertainties of the array explicitly.

In [30–33], some algorithms are proposed based on the model of the array uncertainties. The authors treated the problem of beamforming as signal estimation and found the optimal weights on maximum likelihood [30], minimum MSE (mean square error) [31, 32], or minimum tradeoff between MSE and SINR criteria [33] in the presence of array uncertainties. This type of algorithms need the preknowledge of the power of SOI. Another new class of robust algorithms, here referred to as the robust Capon beamformers (RCBs) [34–38], were developed. They assume the true data vector of the SOI is confined within an ellipsoidal uncertainty region centered at the nominal one. The true data vector is estimated by finding the maximum output power of the beamformer in the confined region. In order to avoid computational complexity, the RCBs are sometimes use a spherical uncertainty region. In this case, the RCB method produces a diagonal loading solution, and the diagonal loading factor is determined by the size of the uncertainty region which may be estimated by using the worst-case scenario of previous measurements.

1.3 Objectives

This thesis has three aims.

The first aim is to analyze how the data-independent arrays and adaptive arrays are affected by the perturbations in array characteristics. Some analytical expressions of the performance measures, e.g., the expected beampattern and the expected SINR using the proposed perturbation model are derived and discussed. Some important performance measures which cannot be expressed as explicit analytical forms are simulated and compared in detail. This is helpful in understanding how the perturbations affect the performance of arrays qualitatively and quantitatively.

Secondly, the thesis gives robust solutions to mitigate the performance degradation of data-independent arrays and adaptive arrays in perturbed situations. Two robust methods based on the MSE (mean square error) criterion are presented for data-independent arrays. Two other robust methods are proposed for adaptive arrays. One method, RAB1 (robust adaptive beamformer 1), is designed to combat phase errors. Another robust method, RAB2 (robust adaptive beamformer 2), which deals with large steering error is presented based on traditional RCB methods. Both of these two algorithms protect the adaptive arrays from "signal nulling" and provide higher SINR performance in perturbed situations.

Thirdly, a comparison of data-independent arrays and adaptive arrays according to different ambient environments and system requirements is presented.

1.4 Major Contributions of the Thesis

The main contributions of this thesis are:

1) A perturbation model which based on the position errors of the array embedded flexible fabric is proposed. From the model, a sequence of correlated errors can be obtained by an IID (independent and identically distributed) sequence.

2) Some analytical expressions of the performance measures, e.g., the expected beampattern and the variance of the beampattern using the proposed perturbation model are derived and discussed. They are helpful in understanding how the perturbations affect the performance of arrays qualitatively and quantitatively.

3) Two methods based on MSE criterion are presented for data-independent arrays. The mean square error of these arrays using the proposed perturbation model is analyzed. We give a recommendation for the choice of the two robust methods according two different environmental and perturbed situations.

4) A method which combats phase errors is proposed for an adaptive array. This method offers robustness to the perturbations that cause time-delay errors by tracking the true phase vector. Thus, the proposed beamformer does not suffer from performance loss in interference rejection.

5) Another robust method which deals with large steering errors is presented based on traditional RCBs. Higher SINR performance can be obtained than that of traditional RCBs since the proposed method effectively shrinks the uncertainty

region by compensating the steering errors.

1.5 Organization of Thesis

The remainder of the thesis is organized as follows.

Chapter 2 explains the basic principle and structure of array processing. It also introduces the terminology employed throughout the thesis.

In Chapter 3, various data-independent arrays, e.g., delay-and-sum beamformer, sidelobe-control beamformers and supergain endfire beamformer are introduced. The degraded performances of different data-independent arrays in the perturbed situations are analyzed and compared. Then, the chapter gives two robust methods based on the MSE criterion. The numerical results of the two robust methods show that both can mitigate the sensitivity to perturbations. This is especially significant for the supergain beamformer. In the last part, the chapter compares the two robust methods, and discuss which method is best suited to a given set of conditions.

Chapter 4 begins with an introduction to the fundamentals of adaptive arrays. The Capon beamformer which forms the main basis of our research is explained in detail. The degraded performance of the Capon beamformer in the perturbed situations are analyzed and compared with delay-and-sum array. Several existing robust methods are listed and discussed, with emphasis on the new kind of robust Capon beamformers. The robust adaptive arrays can be classified into two

kinds: constrained robust method and responsive robust method. The advantages and disadvantages of both kinds of robust methods are explained. Then, a robust method, RAB 1, which combats phase errors is proposed. The numerical results shows that it can offer robustness without sacrificing interference rejection capability. In the last part, another robust method, RAB 2, which deals with large steering error is presented based on the traditional RCB method. It improves the robust performance of traditional RCBs by shrinking the uncertainty region.

Finally, Chapter 5 provides some concluding remarks and possible future work.

Chapter 2

An Overview of Array Processing

2.1 Introduction

This chapter introduces the terminology employed throughout the thesis and provides an overview of the array processing techniques.

The chapter is organized as follows. In Section 2.2, some basic principles and concepts of array signal processing are introduced. In Section 2.3, the environmental assumptions and some mathematical notations used throughout the whole thesis are given. In Section 2.4, the structure of a beamforming system, including narrowband and broadband array implementations, is briefly discussed. In Section 2.5, a model of narrowband signals is presented and the formulation of an isotropic noise field is given. The uniform linear array, which is frequently used in the subsequent chapters for simulations, is introduced in Section 2.6. In Section 2.7, the

classification of beamformers based on the choice of array weights is introduced. Some fundamental robust concepts are also discussed. In the last Section 2.8, several common and important measures which are used to evaluate the performance of arrays are enumerated and explained.

2.2 Concepts of Array Signal Processing

Signal processing is concerned with the information carried by the signals of interest. By using either analog or digital equipment, signal processing techniques are used to extract the information or to transform the signals in useful ways. Array processing is a specialized branch of signal processing that is concerned with signals conveyed by propagating waves.

Often the information we want is carried by a signal degraded by noise or interferences. If the SOI and interferences occupy the same temporal frequency band, temporal filtering cannot be used to separate them from each other. However, the SOI and interferences generally originate from different spatial locations. This spatial separation can be exploited to separate them from each other by a beamformer to reduce the effects of unwanted noise. On the other hand, a beamformer can be used to extract spatial characteristics, c.g., direction or range, of signals of interest.

The goals of array processing are [1]:

- to **enhance** the signal-to-noise ratio beyond that of a single sensor's output
- to **characterize** the field by determining the *number* of sources of propagating energy, the *locations* of these sources, and the *waveforms* they are emitting
- to **track** the energy sources as they move in space

With distributed sensors, array beamforming has ability to obtain higher spatial discrimination and a more adaptive response than those of the single sensor. Therefore, typical uses of beamforming techniques arise widely in RADAR, SONAR, wireless communications, microphone arrays, medical imaging and other fields.

2.3 Signals in Space

In many signal processing applications, we are concerned with waveforms that are functions of a single variable, which usually represents time. In array processing, however, propagating waves convey signals from source to the array. Thus, these signals are a function of position as well as time. Consider the case shown in Fig. 2.1, where a source signal, $\tilde{s}(t)$, propagates in the direction $\mathbf{u}$. Note that in many array applications, e.g., telecommunications and RADAR, the original signals are modulated, and the modulated signal $\tilde{s}(t)$ is given by

$$\tilde{s}(t) = s(t)e^{j2\pi ft}, \quad (2.1)$$

Figure 2.1: Reference coordinates of an arbitrary array

where $s(t)$ is the modulating envelope signal, and f is the carrier frequency.

In this thesis, we assume that the array of sensors are located in the far field of the sources. Thus, the propagating waves incident on the array can be considered as a plane wave. We also assume that the signal propagates at a constant speed in the medium, so the signal received by each sensor is a pure time-delay form of the source signal. The signal received at the n th element of the array is

$$\tilde{x}(t, \mathbf{r}_n) = \tilde{s}(t - \tau_n), \quad (2.2)$$

where the delay τ_n is given by

$$\tau_n = \frac{\mathbf{u}^T(\theta, \phi) \cdot \mathbf{r}_n}{c}. \quad (2.3)$$

Here, $\mathbf{r}_n = [x_n \ y_n \ z_n]^T$ is the position vector of the n th sensor, c denotes the propagation speed of the source signal, and $\mathbf{u}(\theta, \phi)$ is a unit vector of the propagation direction which can be expressed as

$$\mathbf{u}(\theta, \phi) = [\cos \phi \sin \theta \quad \sin \phi \sin \theta \quad \cos \theta]^T, \quad (2.4)$$

Figure 2.2: Beamforming system

where θ, ϕ are the polar and azimuth angles respectively, and superscript T denotes vector transpose.

2.4 Beamforming System

Beamforming refers to a wide variety of array processing algorithms that focus the array's signal-capturing abilities in a particular direction. A beamformer implements beamforming techniques by processing each sensor output with a linear, time-invariant filter and summing the outputs to obtain the array output. The procedure is shown in Fig. 2.2. Note that the incident signal at each channel is demodulated to baseband before beamforming. The output $y(t)$ can be written as a convolution integral,

$$y(t) = \sum_{n=1}^N \int_{-\infty}^{\infty} x_n(\tau) h_n(t - \tau) d\tau. \quad (2.5)$$

Beamforming systems can be classified either as narrowband or broadband

depending on the frequency extent of the processed signals. If the carrier frequency f is fairly large compared to the bandwidth of the modulating signal, then $s(t)$ can be consider as a narrowband signal. In this case, $s(t)$ is slowly varying relative to f , and (2.2) reduces to

$$\tilde{x}_n(t) = s(t - \tau_n)e^{j2\pi f(t - \tau_n)} \approx s(t)e^{j2\pi f(t - \tau_n)}. \quad (2.6)$$

After demodulation, the signal at the n th element of array, $x_n(t)$, is given by

$$x_n(t) = s(t)e^{-j2\pi f\tau_n}, \quad (2.7)$$

which is a phase-shifted form of the original envelope signal.

Fig. 2.3 shows the typical beamformer used for processing narrowband signals. For each channel, the impulse response $h_n(t)$ is a complex scalar, which are called array weights. The phases of the weights are chosen to compensate the propagation delays so that the beamformer can "point" to the desired directions. The amplitude of the weights are chosen to control the shape of the array response. The output signal of a narrowband array, $y(t)$, is :

$$y(t) = \sum_{n=1}^N w_n^* x_n(t), \quad (2.8)$$

where w_n denotes the weight at the n th element and superscript $*$ represents complex conjugation.

Here, we omit the introduction of broadband array, because in this thesis we focus on the beamforming techniques for narrowband array.

Figure 2.3: Narrowband beamformer with N sensors

2.5 Data Model of Narrowband Signals

In the last section, the structure of narrowband beamforming system was introduced. In this section, we will introduce the data model of the narrowband beamforming system.

2.5.1 Signal Representation

From (2.7), the vector of array data, $\mathbf{x}(t) = [x_1(t) \ x_2(t) \ \cdots \ x_N(t)]^T$, can be written as:

$$\mathbf{x}(t) = \mathbf{a}(\theta, \phi)s(t), \quad (2.9)$$

where $\mathbf{a}(\theta, \phi)$ is define as:

$$\begin{aligned} \mathbf{a}(\theta, \phi) &= [e^{-j2\pi f\tau_1} \ e^{-j2\pi f\tau_2} \ \cdots \ e^{-j2\pi f\tau_N}]^T \\ &= [e^{-j\mathbf{k}^T \mathbf{r}_1} \ e^{-j\mathbf{k}^T \mathbf{r}_2} \ \cdots \ e^{-j\mathbf{k}^T \mathbf{r}_N}]^T, \end{aligned} \quad (2.10)$$

and $\mathbf{k}$ is the wave number of the signal from (θ, ϕ) given by

$$\mathbf{k} = \frac{2\pi f}{c} \mathbf{u}(\theta, \phi). \quad (2.11)$$

The vector $\mathbf{a}(\theta, \phi)$ incorporates all of the spatial characteristics of the array. It is referred to as the array data vector. From (2.8) and (2.9), the response of an array to a directional waveform coming from (θ, ϕ) is given by

$$H(\theta, \phi) = \mathbf{w}^H \mathbf{a}(\theta, \phi) = \sum_{n=1}^N w_n^* e^{-j\mathbf{k}^T \mathbf{r}_n}, \quad (2.12)$$

where $\mathbf{w}$ is the vector of array weights, and superscript H denotes conjugate transpose.

In a practical system, the received signal is contaminated by noise and interferences. We assume there are P sources in the environment, and the SOI, $s_0(t)$, is coming from (θ_0, ϕ_0) . The other $P-1$ sources $s_1(t), \dots, s_{P-1}(t)$, which are coming from $(\theta_0, \phi_0), \dots, (\theta_{P-1}, \phi_{P-1})$ respectively, are assumed to be interferences. Thus, the array data $\mathbf{x}(t)$ is given by

$$\mathbf{x}(t) = \mathbf{a}_0 s_0(t) + \mathbf{v}(t), \quad (2.13)$$

where

$$\mathbf{a}_0 = \begin{bmatrix} e^{-j\mathbf{k}_0^T \mathbf{r}_1} & e^{-j\mathbf{k}_0^T \mathbf{r}_2} & \dots & e^{-j\mathbf{k}_0^T \mathbf{r}_N} \end{bmatrix}^T; \quad \mathbf{k}_0 = \frac{2\pi f}{c} \mathbf{u}(\theta_0, \phi_0). \quad (2.14)$$

$\mathbf{v}(t)$ is the vector of unwanted signals in the array elements, including interference and white noise $\mathbf{n}(t)$, which is given by

$$\mathbf{v}(t) = \sum_{j=1}^{p-1} \mathbf{a}_j s_j(t) + \mathbf{n}(t), \quad (2.15)$$

where $\mathbf{a}_j$ is the array data vector of the j th source:

$$\mathbf{a}_j = \begin{bmatrix} e^{-j\mathbf{k}_j^T \mathbf{r}_1} & e^{-j\mathbf{k}_j^T \mathbf{r}_2} & \dots & e^{-j\mathbf{k}_j^T \mathbf{r}_N} \end{bmatrix}^T; \quad \mathbf{k}_j = \frac{2\pi f}{c} \mathbf{u}(\theta_j, \phi_j). \quad (2.16)$$

From (2.13), the output of narrowband beamformer is represented as

$$y(t) = \mathbf{w}^H \mathbf{x}(t) = \mathbf{w}^H \mathbf{a}_0 s_0(t) + \mathbf{w}^H \mathbf{v}(t). \quad (2.17)$$

2.5.2 Covariance Matrix

We assume all the sources and noise are stationary stochastic processes. The mean output power from the array system, p_{out} , is given by

$$p_{out} = E \{ |y(t)|^2 \} = \mathbf{w}^H \mathbf{R} \mathbf{w}, \quad (2.18)$$

where $E \{ \cdot \}$ denotes the statistical expectation, and $\mathbf{R}$ is the $N \times N$ dimensional array covariance matrix defined by

$$\mathbf{R} = E \{ \mathbf{x}(t) \mathbf{x}^H(t) \}. \quad (2.19)$$

We assume all the impinging sources are statistically uncorrelated so that

$$\begin{aligned} \mathbf{R} &= E \{ s_0(t) s_0^*(t) \} \mathbf{a}_0 \mathbf{a}_0^H + \sum_{j=1}^{p-1} E \{ s_j(t) s_j^*(t) \} \mathbf{a}_j \mathbf{a}_j^H + E \{ \mathbf{n}(t) \mathbf{n}^H(t) \} \\ &= \sigma_0^2 \mathbf{a}_0 \mathbf{a}_0^H + \mathbf{R}_{n+i}, \end{aligned} \quad (2.20)$$

where $\mathbf{R}_{n+i}$ is defined as the noise and interference covariance matrix given by

$$\mathbf{R}_{n+i} = \sum_{j=1}^{p-1} \sigma_j^2 \mathbf{a}_j \mathbf{a}_j^H + \sigma_n^2 \mathbf{I}, \quad (2.21)$$

where σ_j^2 is the signal power of the j th source, σ_n^2 is the power of white noise and $\mathbf{I}$ is identity matrix. The covariance matrix $\mathbf{R}$ is a second-order statistical property

of the demodulated signals. In practical applications, the matrix $\mathbf{R}$ is unavailable and is estimated by the array snapshots. The estimated covariance matrix, $\hat{\mathbf{R}}$, which is calculated by K snapshots, is given by

$$\hat{\mathbf{R}} = \frac{1}{K} \sum_{k=1}^K \mathbf{x}_k \mathbf{x}_k^H, \quad (2.22)$$

where $\mathbf{x}_k$ is a column vector consisting of the k th sampled data (demodulated signal $x(t)$) of all N sensors. When K approaches infinity, the estimated covariance matrix $\hat{\mathbf{R}}$ asymptotically approaches the true covariance matrix $\mathbf{R}$.

2.5.3 Isotropic Noise Field

For many array processing problems, we can reasonably assume that the observed noise field consists of many random waves propagating in all possible directions with equal probability: an isotropic noise field. Two typical isotropic noise fields, spherically and cylindrically isotropic noise fields, are often used in reverberant sound environments [39–41]. In this section, the formulation of the covariance matrix of isotropic noise field is given. The results will be used in the latter part of the thesis.

If a noise source, $x(t)$, impinges the array, the (l, m) element of the spatial covariance matrix is given by

$$r_{l,m}(\Omega_1) = E \{ \mathbf{x}_l(t) \mathbf{x}_m^*(t) \} = p(\Omega_1) e^{-j\mathbf{k}_1^T(\mathbf{r}_l - \mathbf{r}_m)}, \quad (2.23)$$

where Ω_1 represents the solid angle of the source, $\mathbf{k}_1$ is the wave number of the source, and $p(\Omega_1)$ is the power of the source. In an isotropic field, all the incident

waveforms are assumed spatially independent, and they offer same amount of contribution to the array element. Therefore, the spatial correlation of an isotropic field is the average of Eq. (2.23) over all directions:

$$r_{l,m} = p \cdot \int_{\Omega} e^{-j\mathbf{k}^T(\mathbf{r}_l - \mathbf{r}_m)} d\Omega. \quad (2.24)$$

We assume that the total power of the noise field is σ^2 , thus we have $\int_{\Omega} p d\Omega = \sigma^2$.

For example, for a spherically isotropic noise field,

$$p \cdot \int_0^{2\pi} d\phi \int_0^{\pi} d\theta \sin \theta = \sigma^2, \quad (2.25)$$

so that $p = \frac{\sigma^2}{4\pi}$. Substituting the value of p into (2.24), the (l, m) element of the covariance matrix for a spherically isotropic noise field can be calculated:

$$\mathbf{R}_{n(l,m)} = \frac{\sigma^2}{4\pi} \int_0^{2\pi} d\phi \int_0^{\pi} d\theta \sin \theta e^{-j\mathbf{k}^T(\mathbf{r}_l - \mathbf{r}_m)}. \quad (2.26)$$

For an array lying in the same plane as the wave vector directions of the cylindrically isotropic noise field, the total noise power is

$$p \cdot \int_0^{2\pi} d\phi = \sigma^2, \quad (2.27)$$

so that $p = \frac{\sigma^2}{2\pi}$. Therefore, the (l, m) element of $\mathbf{R}_n$ in a cylindrically isotropic noise is given by

$$\mathbf{R}_{n(l,m)} = \frac{\sigma^2}{2\pi} \int_0^{2\pi} e^{-j\mathbf{k}^T(\mathbf{r}_l - \mathbf{r}_m)} d\phi. \quad (2.28)$$

The covariance matrices of an ULA in spherically and cylindrically isotropic noise fields are given in A.5 and A.6 respectively.

2.6 Array Geometry and Uniform Linear Array

In array applications, a variety of array configurations are used according to different practical requirements. Generally, the array configurations can be linear, planar or volumetric. Among various array configurations in use, the uniform linear array (ULA) is a quite common and simple one. A typical uniform linear array whose elements are placed on the y axis with a uniform spacing d is shown in Fig. 2.4. For a linear array, it can resolve only one angular component, which leads to a cone of uncertainty and left/right ambiguities. Thus, it is convenient to set $\phi = \frac{\pi}{2}$ in Eq. (2.4), which becomes

$$\mathbf{u}(\theta) = [0 \quad \sin \theta \quad \cos \theta]^T. \quad (2.29)$$

The data vector in the ULA, $\mathbf{a}$, is a function of θ given by

$$\mathbf{a}(\theta) = \left[1 \cdots e^{-j \frac{2\pi f(m-1)d \sin \theta}{c}} \cdots e^{-j \frac{2\pi f(N-1)d \sin \theta}{c}} \right]. \quad (2.30)$$

In many references and this thesis, a uniform linear array with d equal to half wavelength λ is referred to as the standard linear array.

2.7 Beamformer Classification and Robust Concepts

Beamformers can be classified as either data-independent or statistically optimum, depending on how their weights are chosen. The weights in a data-independent

Figure 2.4: Geometry of a uniform linear array

beamformer are designed to present a particular response that is independent of the array data. The data-independent beamformers include the delay-and-sum approach as well as methods based on various weight vectors for sidelobe control. The statistically optimum beamformers select weights based on the statistics of the array data to optimize the array response so that the output signal contains minimal contributions due to the noise and signals arriving from directions other than the desired direction. Certain adaptive algorithms (e.g. steepest descent method) are designed so the beamformer response converges to a statistically optimum solution.

For either the data-independent or statistically optimum array, we generally assume that a propagating signal arrives at the array as an undistorted plane wave, whose data vector is correctly described by Eq. (2.10). This assumption is often used as a constraint to maintain a distortionless output of the SOI.

However, in actual implementation, this ideal assumption may be violated.

The direction of the desired source may differ from the look direction, and the phases or positions of the array sensors may also be perturbed from assumed values. Thus, the true data vector ($\mathbf{a}$) will deviate from the nominal one ($\bar{\mathbf{a}}$) in (2.10).

The mismatch between the nominal and actual data vectors leads to a degraded performance of the beamformers. The impaired effects are especially fatal for adaptive beamformers, although they can offer much higher resolution and interference nulling ability than the data-independent arrays in an ideal environment. This is because the mismatch between the nominal and true data vectors will make adaptive beamformers misinterpret the SOI as interferences, leading to ‘signal nulling’. Whenever this happens, the performance of the adaptive beamformers may become even worse than that of a single sensor [8,42]. Therefore, it is very important to analyze how the imperfections and errors affect the performance of the beamformers and propose robust methods to mitigate the influence of the non-ideal environment.

2.8 Performance Measures

In order to evaluate the performance of an array, various performance measures are used to investigate and compare. Each of these measures attempts to quantify an important aspect of either the response of an array to the signal environment or the sensitivity to an array design. Here, we define four important and commonly

used ones:

- Beampattern
- Signal-plus-Interference-and-Noise Ratio
- Array Gain
- Directivity

2.8.1 Beampattern

The beampattern $B(\theta, \phi)$ is defined as the squared magnitude of array response $H(\theta, \phi)$, i.e.,

$$B(\theta, \phi) = |H(\theta, \phi)|^2 = \mathbf{w}^H \mathbf{a}(\theta, \phi) \mathbf{a}^H(\theta, \phi) \mathbf{w}. \quad (2.31)$$

The beampattern of a delay-and-sum nine-sensor standard linear array is shown in Fig. 2.5. Two parameters of the beampattern, the 3dB beamwidth and side-lobe level, that characterize the beampattern are shown. In many applications, a beampattern with a narrow beamwidth and a low sidelobe level is desired.

- 3 dB beamwidth :

3 dB beamwidth, is a measure of the width of the mainlobe. It is defined to be the width between two points where $B(\theta, \phi)$ is a half of the maximum, and it determines how well two signals can be resolved spatially.

- Sidelobe level:

Sidelobe level is referred either as the highest or the average sidelobe level

Figure 2.5: Measures of a beampattern

depending on the application. It measures the capability of an array to reject unwanted noise and signals in directions away from the mainlobe.

Whenever an array suffers random perturbations, the beampattern will also present a random shape. Thus, a useful statistical measure is the expected value of the beampattern, which is given by

$$E_{\text{error}} \{B(\theta, \phi)\} = E_{\text{error}} \{|H(\theta, \phi)|^2\} = \mathbf{w}^H E_{\text{error}} \{\mathbf{a}(\theta, \phi) \mathbf{a}^H(\theta, \phi)\} \mathbf{w}, \quad (2.32)$$

where $E_{\text{error}} \{\cdot\}$ represents the expectation over the perturbations in the array.

2.8.2 Signal-to-Interference-plus-Noise Ratio

Signal-to-Interference-plus-Noise ratio, is always referred to output SINR. It is defined as signal power to noise-plus-interferences power in the output of the array, which is given by

$$SINR_{out} = \frac{P_{s(out)}}{P_{n(out)}} = \frac{\sigma_0^2 |\mathbf{w}^H \mathbf{a}_0|^2}{\mathbf{w}^H E \{ \mathbf{v}(t) \mathbf{v}^H(t) \} \mathbf{w}} = \frac{\sigma_0^2 |\mathbf{w}^H \mathbf{a}_0|^2}{\mathbf{w}^H \mathbf{R}_n \mathbf{w}}. \quad (2.33)$$

2.8.3 Array Gain

Array gain (AG) is defined as the ratio of the output SINR of the array to the input SINR, i.e.,

$$AG = \frac{SINR_{out}}{SINR_{in}}. \quad (2.34)$$

It measures the improvement of SINR by using the array. Here, the input SINR, is the ratio of signal and noise-plus-interferences power in one sensor:

$$SINR_{in} = \frac{E \{ s_0(t) s_0^*(t) \}}{E \{ \mathbf{v}_m(t) \mathbf{v}_m^*(t) \}} = \frac{\sigma_0^2}{\sum_{i=1}^{P-1} \sigma_i^2 + \sigma_n^2}. \quad (2.35)$$

Furthermore, Signal-to-Noise ratio (SNR), is defined as the ratio of signal power to noise power in one sensor:

$$SNR = \frac{E \{ s_0(t) s_0^*(t) \}}{E \{ \mathbf{n}_m(t) \mathbf{n}_m^*(t) \}} = \frac{\sigma_0^2}{\sigma_n^2}. \quad (2.36)$$

It is a common and important measure of the environment.

2.8.4 Directivity

The directivity is defined as the maximum of beampattern (θ_0, ϕ_0 in the Eq. (2.37)) divided by the average beampattern (averaged over the sphere), which is given by

$$D = \frac{B(\theta_0, \phi_0)}{\frac{1}{4\pi} \int_0^\pi d\theta \int_0^{2\pi} d\phi \sin \theta B(\theta, \phi)}. \quad (2.37)$$

Directivity is often used as the main measure of performance of a data-independent array since it can be calculated without knowledge of the array data.

Chapter 3

Robust Techniques for Data-Independent Beamformers

3.1 Introduction

Data-independent arrays are designed to produce a specific array response, or to achieve a beampattern with desirable properties, e.g., a certain beamwidth or sidelobe level. For an array that has a fixed shape, this is accomplished by changing the shading coefficients of array weights. Typically an uniform linear array linearly combines the spatially sampled time series from each sensor to obtain a scalar output time series in the same manner that an finite impulse response (FIR) filter linearly combines temporally sampled data. Therefore, a number of methods and techniques from FIR filtering can be used directly when we design the weights of

the uniform linear arrays [43, 44].

For many applications, the objective of data-independent arrays is to obtain a low sidelobe level with a given beamwidth. Sidelobe suppression is critical for a data-independent array to detect weak signals in the presence of strong interference and noise. A number of sidelobe control arrays are available. Some well-known examples include the Dolph-Chebyshev and Taylor-Kaiser arrays. In addition to sidelobe-control beamformers, the supergain array is another example of a data-independent array. It offers extremely high array gain and is often used for beamforming applications where the size of the array is limited. For other types of beampatterns, a number of array synthesis methods have been developed in the past 60 years. These can be found in the antenna literature [2, 45–49].

In practice it is often found that the actual performance of the beamformer is not as good as that predicted by design. A possible reason for the worse than ideal performance is the mismatch between the actual array parameters and the nominal ones. In this chapter we will analyze and simulate the effect of the random errors on the performance of data-independent arrays and give two robust methods based on the mean square error criterion which mitigate the degradation in performance.

This chapter is organized as follows. Section 3.1 discusses the data-independent arrays: delay-and-sum beamformer, sidelobe-control beamformers and supergain endfire beamformer. In Section 3.2, the performance of data-independent arrays in perturbed situations is analyzed. In Section 3.3, the performance of different array syntheses are compared in different perturbation situations. In Section 3.4, some

existing robust methods are listed and discussed. In Section 3.5 and Section 3.6, two robust methods are proposed. In Section 3.7, several numerical results which show the comparison of the non-robust and two robust arrays are illustrated and discussed. Finally, in Section 3.8, we give a suggestion on how to choose a robust method based on different system requirements and environmental situations.

3.1.1 Delay-and-Sum Beamformer

The delay-and-sum beamformer, also called the conventional beamformer, is the oldest and simplest array which is still widely used in many fields today. The delay-and-sum beamformer has weights that are uniform in magnitude for all sensors. For example, the weight vector of a linear conventional beamformer steered to θ_0 is given by

$$\mathbf{w} = [1 \quad e^{-jkd \sin \theta_0} \quad \dots \quad e^{-j(N-1)kd \sin \theta_0}]^T. \quad (3.1)$$

The corresponding beam pattern is

$$B(\theta, f) = \frac{\sin^2 [N\pi \frac{d}{c} f(\sin \theta - \sin \theta_0)]}{\sin^2 [\pi \frac{d}{c} f(\sin \theta - \sin \theta_0)]}. \quad (3.2)$$

3.1.2 Sidelobe-Control Beamformers

Among many synthesis techniques, there is a major class of pattern synthesis that achieves a high mainlobe-to-sidelobe ratio. A number of synthesis techniques have been developed for this particular problem. Typical examples are the Dolph-Chebyshev and Taylor-Kaiser arrays [50, 51]. Chebyshev synthesis [50] yields an

array with the smallest possible beamwidth for a given sidelobe level, making it a very useful and practical method for reduced sidelobe design. Taylor-Kaiser synthesis differs from Chebyshev beamformer in that the sidelobe levels decrease away from broadside, whereas Chebyshev patterns have uniform sidelobes. The design principle of the weights of Taylor-Kaiser and Dolph-Chebyshev array are introduced in Appendix B.

The comparison of the beampatterns of delay-and-sum, Dolph-Chebyshev and Taylor-Kaiser beamformer is presented in Fig. 3.1. Here, we used a nine-element standard linear array, and the highest sidelobe level is set to be -30 dB for the Dolph-Chebyshev and Taylor-Kaiser arrays. It is noted that the conventional beamformer has the narrowest beamwidth but highest sidelobe level. This is because there is a trade-off between the width of the mainbeam and the sidelobe level. The weight in each sensor is compared in Fig. 3.2. It shows that suppressed sidelobes in a ULA are obtained by setting larger weights amplitude near the center of the array.

3.1.3 Supergain Endfire Beamformer

The supergain endfire array, also called the super-directional array, can offer a significant increase in the array gain over that of other array synthesis methods. This type of array is steered to endfire, with adjacent elements separated by less than one-half wavelength. Two different design methodologies are applied to yield this super directivity performance. Schelkunoff designed the supergain array by

Figure 3.1: Beampatterns of different beamformers
(delay-and-sum, Dolph-Chebyshev and Taylor-Kaiser arrays)

Figure 3.2: Weights of different beamformers
(delay-and-sum, Dolph-Chebyshev and Taylor-Kaiser arrays)

placing all the zeros of beampattern in the visible region, and the resulting weights are known as the Jacobi polynomials (refer to Appendix B) [52, 53]. The second method is based on explicitly optimizing the directivity of an array. The two methods yield nearly identical results, and the resulting pattern has much narrower beamwidth than any other array synthesis method. It has been proved that, for N very closely spaced elements, the array gain of the supergain array approaches N^2 in a spherically isotropic noise field, and $2N - 1$ in a cylindrically isotropic noise field [16, 54]. This high array gain is much larger than the maximum array gain that a conventional beamformer can achieve. A conventional standard linear array has a directivity of N . The supergain endfire array was used in seismic and sonar applications in the early years and from the 1990's it has been widely adopted in microphone applications such as hearing-aids [55, 56]. However, the theoretical supergain is difficult to obtain in practice since the supergain array is sensitive to the small random errors that occur in real-world applications. The beampattern of a uniform linear endfire array ($N = 9$, $d = \frac{1}{5}\lambda$) is shown in Fig. 3.3.

3.2 Performance Analysis for Perturbed Arrays

Whenever the elements of an array suffer random perturbations in gain, phase and position, the true data vector of the signal, $\mathbf{a}(\theta, \phi)$, can be expressed as [12] [2]

$$\mathbf{a} = \begin{bmatrix} g_1 e^{j\phi_1} e^{-j\mathbf{k}^T \mathbf{r}_1} & g_2 e^{j\phi_2} e^{-j\mathbf{k}^T \mathbf{r}_2} & \dots & g_N e^{j\phi_N} e^{-j\mathbf{k}^T \mathbf{r}_N} \end{bmatrix}^T, \quad (3.3)$$

Figure 3.3: Beam patterns of delay-and-sum and supergain endfire arrays

which deviates from the nominal one, $\bar{\mathbf{a}}$, in ideal case

$$\bar{\mathbf{a}} = \begin{bmatrix} e^{-j\mathbf{k}^T \mathbf{r}_1^0} & e^{-j\mathbf{k}^T \mathbf{r}_2^0} & \dots & e^{-j\mathbf{k}^T \mathbf{r}_N^0} \end{bmatrix}^T. \quad (3.4)$$

In (3.3), g_m , ϕ_m , $\mathbf{r}_m$ are the gain, phase, and position of the m th array sensor respectively. We define

$$g_m = 1 + \Delta g_m; \quad \phi_m = \Delta \phi_m; \quad \mathbf{r}_m = \mathbf{r}_m^0 + \Delta \mathbf{r}_m, \quad (3.5)$$

where $\Delta \mathbf{r}_m$, Δg_m , $\Delta \phi_m$ represent the position, gain and phase errors of the m th sensor respectively and $\mathbf{r}_m^0$ represents its nominal position.

From (2.31), since the perturbations are random, the corresponding array response is unpredictable. Fig. 3.4 shows beam patterns of three independent Monte Carlo trials of a nine-element delay-and-sum standard array. In this simulation, we

Figure 3.4: Individual perturbed beampatterns and ideal beampattern
(delay-and-sum BF)

assume the array experiences Gaussian position errors with a standard deviation $\sigma_r = 0.2d$. From the figure, we find that the perturbed beampatterns always have increased sidelobes, and the maximum of the beampattern may no longer be at the steering direction (0°). Next, we will analyze the mean performance of the array that is affected by the perturbations by assuming that some statistical information of the perturbations is available.

3.2.1 Assumptions of the Errors Model

In order to proceed further and give some explicit results, we use the following statistical model of the perturbation errors. First, the variations in gain, phase

and position are independent of each other, but $\Delta g_m, \Delta g_n$ are correlated, and similarly for $\Delta \phi_m, \Delta \phi_n$ and $\Delta \mathbf{r}_m, \Delta \mathbf{r}_n$. In addition, the gain errors $\Delta \mathbf{g}$ are zero-mean jointly Gaussian variables, with correlation matrix

$$E_{error} \{ \Delta \mathbf{g} \Delta \mathbf{g}^T \} = \mathbf{G}. \quad (3.6)$$

Thus,

$$E_{error} \{ (1 + \Delta g_m)(1 + \Delta g_n) \} = 1 + \mathbf{G}_{mn}. \quad (3.7)$$

The phase errors $\Delta \phi$ are zero-mean jointly Gaussian variables, with correlation matrix

$$E_{error} \{ \Delta \phi \Delta \phi^T \} = \Phi. \quad (3.8)$$

For simplicity, the position errors along the x-axis, y-axis and z-axis are assumed to be Gaussian and each direction is independent of the other. However, $\Delta \mathbf{x}_m, \Delta \mathbf{x}_n$ are correlated, and similarly for $\Delta \mathbf{y}_m, \Delta \mathbf{y}_n$ and $\Delta \mathbf{z}_m, \Delta \mathbf{z}_n$. In addition, the position errors along all three axes are identically distributed so that

$$E_{error} \{ \Delta \mathbf{x} \Delta \mathbf{x}^T \} = E_{error} \{ \Delta \mathbf{y} \Delta \mathbf{y}^T \} = E_{error} \{ \Delta \mathbf{z} \Delta \mathbf{z}^T \} = \Omega. \quad (3.9)$$

3.2.2 Expected Beampattern

Expected Beampattern for Correlated Errors

The expected beampattern of the perturbed array is given by

$$\overline{B(\theta, \phi)} = E_{error} \{ B(\theta, \phi) \} = \mathbf{w}^H E_{error} \{ \mathbf{a}(\theta, \phi) \mathbf{a}^H(\theta, \phi) \} \mathbf{w}. \quad (3.10)$$

Here, we assume the weight vector is given by

$$\mathbf{w} = \begin{bmatrix} u_1 e^{-j\mathbf{k}_0^T \mathbf{p}_1} & u_2 e^{-j\mathbf{k}_0^T \mathbf{p}_2} & \dots & u_N e^{-j\mathbf{k}_0^T \mathbf{p}_N} \end{bmatrix}^T, \quad (3.11)$$

where u_m is the amplitude of the weight of the m th element, $\mathbf{k}_0$ is the assumed wavenumber of the look direction (θ_0, ϕ_0) and $\mathbf{p}_m$ is the assumed position of the m th element. Define matrix $\mathbf{P}$ as

$$\mathbf{P} = E_{error} \{ \mathbf{a}(\theta, \phi) \mathbf{a}^H(\theta, \phi) \}. \quad (3.12)$$

The (m, n) element of the matrix $\mathbf{P}$ is give by

$$\begin{aligned} \mathbf{P}_{m,n} &= E_{error} \{ \mathbf{a}_m \mathbf{a}_n^* \} = E_{error} \left\{ g_m g_n e^{j(\phi_m - \phi_n)} e^{-j\mathbf{k}^T(\mathbf{r}_m - \mathbf{r}_n)} \right\} \\ &= e^{-j\mathbf{k}^T(\mathbf{r}_m^0 - \mathbf{r}_n^0)} E_{error} \left\{ (1 + \Delta g_m)(1 + \Delta g_n) e^{j(\Delta \phi_m - \Delta \phi_n)} e^{-j\mathbf{k}^T(\Delta \mathbf{r}_m - \Delta \mathbf{r}_n)} \right\} \\ &= \bar{\mathbf{a}}_m \bar{\mathbf{a}}_n^* E_{error} \left\{ (1 + \Delta g_m)(1 + \Delta g_n) e^{j(\Delta \phi_m - \Delta \phi_n)} e^{-j\mathbf{k}^T(\Delta \mathbf{r}_m - \Delta \mathbf{r}_n)} \right\}, \end{aligned} \quad (3.13)$$

where $\bar{\mathbf{a}}_m$ is defined in Eq. (3.4). It is reasonable to assume that the perturbations in gain and phase are independent of the perturbations in position, so the perturbation terms can be separated into two terms, A_{mn} and B_{mn} :

$$\mathbf{P}_{m,n} = \bar{\mathbf{a}}_m \bar{\mathbf{a}}_n^* \underbrace{E_{error} \{ (1 + \Delta g_m)(1 + \Delta g_n) \}}_{A_{mn}} \underbrace{E_{error} \{ e^{j(\Delta \phi_m - \Delta \phi_n)} e^{-j\mathbf{k}^T(\Delta \mathbf{r}_m - \Delta \mathbf{r}_n)} \}}_{B_{mn}} \quad (3.14)$$

By using the result

$$E \left\{ e^{-j\mathbf{t}^T \boldsymbol{\xi}} \right\} = e^{-\frac{1}{2} \mathbf{t}^T \boldsymbol{\Lambda} \mathbf{t}}, \quad (3.15)$$

where $\boldsymbol{\xi}$ is zero-mean Gaussian distributed with covariance matrix $\boldsymbol{\Lambda}$, the expectations over the phase and position errors can be written as

$$E_{error} \left\{ e^{j(\Delta \phi_m - \Delta \phi_n)} \right\} = e^{-\frac{1}{2}(\Phi_{mm} - 2\Phi_{mn} + \Phi_{nn})}, \quad (3.16)$$

and

$$E_{error} \left\{ e^{-j\mathbf{k}^T(\Delta\mathbf{r}_m - \Delta\mathbf{r}_n)} \right\} = e^{-\frac{k^2}{2}(\Omega_{mm} - 2\Omega_{mn} + \Omega_{nn})}. \quad (3.17)$$

The (m, n) element of matrix $\mathbf{P}$ from (3.13)

$$\begin{aligned} \mathbf{P}_{m,n} &= \bar{\mathbf{a}}_m \bar{\mathbf{a}}_n^* (1 + \mathbf{G}_{m,n}) e^{-\frac{1}{2}(\Phi_{mm} - 2\Phi_{mn} + \Phi_{nn})} e^{-\frac{k^2}{2}(\Omega_{mm} - 2\Omega_{mn} + \Omega_{nn})} \\ &= \bar{\mathbf{a}}_m \bar{\mathbf{a}}_n^* + \mathbf{X}_{mn}, \end{aligned} \quad (3.18)$$

where

$$\mathbf{X}_{mn} = \bar{\mathbf{a}}_m \bar{\mathbf{a}}_n^* \left[(1 + \mathbf{G}_{m,n}) e^{-\frac{1}{2}(\Phi_{mm} - 2\Phi_{mn} + \Phi_{nn})} e^{-\frac{k^2}{2}(\Omega_{mm} - 2\Omega_{mn} + \Omega_{nn})} - 1 \right]. \quad (3.19)$$

From (3.10), the expected beampattern of correlated errors case is

$$\overline{B(\theta, \phi)} = \mathbf{w}^H \bar{\mathbf{a}} \bar{\mathbf{a}}^H \mathbf{w} + \mathbf{w}^H \mathbf{X} \mathbf{w} = B_0(\theta, \phi) + \mathbf{w}^H \mathbf{X} \mathbf{w}, \quad (3.20)$$

where $B_0(\theta, \phi)$ is the unperturbed beampattern.

Expected Beampattern for Independent Errors Case

It is interesting to study the performance if the gain, phase and position errors are uncorrelated, i.e.,

$$\mathbf{G} = \sigma_g^2 \mathbf{I}, \quad \Phi = \sigma_\phi^2 \mathbf{I}, \quad \Omega = \sigma_r^2 \mathbf{I}, \quad (3.21)$$

where σ_g^2 , σ_ϕ^2 and σ_r^2 are the variances of the gain, phase and position errors respectively. We call this perturbation situation the independent errors case, which is a special case of the proposed errors model.

For the independent errors case, A_{mn} and B_{mn} in (3.14), can be calculated as:

$$E_{error}\{A_{mn}\} = \begin{cases} 1 + \sigma_g^2 & m = n, \\ e^{-\sigma_\phi^2} & m \neq n. \end{cases} \quad (3.22)$$

$$E_{error}\{B_{mn}\} = \begin{cases} 1 & m = n, \\ e^{-k^2\sigma_r^2} & m \neq n. \end{cases} \quad (3.23)$$

Then, Eq.(3.14) becomes :

$$\mathbf{P}_{m,n} = \begin{cases} 1 + \sigma_g^2 & m = n, \\ \bar{\mathbf{a}}_m \bar{\mathbf{a}}_n^* e^{-(\sigma_\phi^2 + k^2\sigma_r^2)} & m \neq n. \end{cases} \quad (3.24)$$

Therefore, the expected beampattern of independent errors case is given by

$$\overline{B(\theta, \phi)} = \underbrace{e^{-(\sigma_\phi^2 + k^2\sigma_r^2)} B_0(\theta, \phi)}_{\text{part 1}} + \underbrace{(1 - e^{-(\sigma_\phi^2 + k^2\sigma_r^2)} + \sigma_g^2) \|\mathbf{w}\|^2}_{\text{part 2}}. \quad (3.25)$$

Analysis and Simulations

For the independent errors case, from (3.25), we find that the expected beampattern is composed of two parts. Part 1 is the nominal beampattern with uniform attenuation caused by phase and position errors. This uniform scaling does not change the shape of the pattern, and it is very weak requiring a standard deviation of phase error $\sigma_\phi = 48^\circ$ to produce a reduction of 3dB. Part 2 is the additional noise level that raises the beampattern uniformly. Since the response of mainlobe region is much larger than that of the sidelobe region, this term will greatly raise

the level of the sidelobe region. It is found that doubling the standard deviation of the phase error, σ_ϕ , increases the sidelobe level by about 6dB.

For further analysis, the expected beampattern in (3.25) is normalized by dividing $e^{-(\sigma_\phi^2 + k^2 \sigma_r^2)} B_0(\theta_0, \phi_0)$. The normalized expected beampattern, $\overline{B_u(\theta, \phi)}$, is given by

$$\overline{B_u(\theta, \phi)} = U_0(\theta, \phi) + \left(\frac{1 - e^{-(\sigma_\phi^2 + k^2 \sigma_r^2)} + \sigma_g^2}{e^{-(\sigma_\phi^2 + k^2 \sigma_r^2)}} \right) \frac{\mathbf{w}^H \mathbf{w}}{B_0(\theta_0, \phi_0)} \quad (3.26)$$

where $U_0(\theta, \phi) = \frac{B_0(\theta, \phi)}{B_0(\theta_0, \phi_0)}$ is the normalized ideal beampattern. From (3.26), it is the term $\frac{\mathbf{w}^H \mathbf{w}}{B_0(\theta_0, \phi_0)}$ that determines the increased noise level. In the same perturbed scenario, the mean pattern of the perturbed array has a high sidelobe level if this term is large.

Next, we show the perturbed performance of a delay-and-sum array through simulations. In the simulations, a nine-element standard linear array with random sensor position perturbations is considered. Fig. 3.5 shows the expected beampatterns with respect to the standard deviation of the position errors. The expected beampattern are obtained by averaging 200 Monte Carlo trials.

The analytical expected beampattern from (3.25) with $\sigma_r = 0.4d$ is plotted as a dashed line, which is very close to the simulated one (solid one). From the figure, we can easily find that the sidelobe level is almost uniform throughout the whole spatial region when the perturbation is large. The main performance measures of the average beampatterns versus different perturbation deviations are shown in Table 3.1. As expected, larger perturbations result in worse performance, i.e., the

Figure 3.5: Expected beampattern vs position deviation
(independent errors case)

beampattern has a wider beamwidth, lower directivity, and higher sidelobe level.

For the correlated errors case, from (3.20), the expected beampattern is also a combination of the error-free beampattern and a perturbation term. However, in contrast to the independent errors case, the increased level is not uniform throughout the spatial region because the matrix $\mathbf{X}$ in (3.19) depends on the signal direction. In order to apply correlated errors in simulations, a correlated position error model is proposed (refer to Appendix C). Fig. 3.6 shows the expected beampatterns with respect to the standard deviation of the position errors in the correlated errors case. In this simulation, the perturbation deviation (σ_r) is the same as the independent errors case, except that the position errors are correlated with correlation length $L = 5d$.

Figure 3.6: Expected beampattern vs position deviation
(correlated errors case $L = 5d$)

Figure 3.7: Expected beampattern vs position deviation
(correlated errors case $L = d$)

Table 3.1: Main performance measures vs perturbation deviations

(independent errors case)

perturbation deviation (σ_r)	3dB beamwidth (deg.)	directivity index (dB)	sidelobe level (dB)
0	11.3	9.57	-12.93
0.2d	11.8	8.16	-9.84
0.3d	12.7	6.12	-7.42
0.4d	14.8	4.53	-4.75
0.5d	almost isotropic	1.97	-2.47

The expected beampattern of the correlated errors case not only depends on the variance of the perturbations but the correlation length as well. Fig. 3.7 shows the expected beampatterns respect to the standard deviation of the position errors with correlation length $L = d$. Comparing Fig. 3.6 and Fig. 3.7, it is found that the errors which are strongly correlated over distance produce a broadening of the mainbeam but maintain relatively low sidelobes. When the correlation length is reduced to zero, the errors become uncorrelated, and the sidelobes increases to a uniform level. The main performance measures of the average beampatterns versus different correlation length are shown in Table 3.2. It is found that, when the array has perturbation errors with a larger correlation length, it has a slightly wider beamwidth, but with lower sidelobe level and higher directivity.

Table 3.2: Main performance measures vs correlation length

(correlated errors case $\sigma_r = 0.2d$)

correlation length (L)	3dB beamwidth (deg.)	directivity index (dB)	sidelobe level (dB)
0	11.3	8.16	-9.84
2d	11.4	8.5	-10.6
5d	11.5	9.06	-12.3

3.2.3 Directivity of Expected Beampattern

In this section, we analyze the directivity of the expected beampattern in the perturbed situations. For a standard linear array in the ideal situation, the directivity D_0 is given by (refer to Appendix A.1):

$$D_0 = \frac{B_0(\theta_0)}{\mathbf{w}^H \mathbf{w}}, \quad (3.27)$$

which is exactly the reciprocal of the sensitivity term of the expected beampattern in (3.26). In the Appendix A.2, we prove that the uniformly weighted array has the minimum value in terms of the sensitivity. Thus, for a linear standard array, uniform weighting is the best choice in the independent errors situation since it has the highest directivity ($D_0 = N$) and lowest sensitivity at the same time. The directivity of the expected beampattern $\bar{D}$ is given by (refer to Appendix A.3):

$$\bar{D} \approx \frac{e^{-(\sigma_g^2 + k^2 \sigma_r^2)}}{1 + \sigma_g^2} D_0 \quad (3.28)$$

From the above equation, we can estimate the decrease in the directivity of a standard array with independent perturbation errors. For example, a nine element standard array will decrease by 0.03 dB in directivity if the array suffers from a Gaussian random phase error of 5°.

3.2.4 Variance of Perturbed Beampatterns

In practice, the average beampattern may never be realized. The variance of the perturbed beampattern, which measures the spread of the distribution of the random beampatterns is a more important measure of the perturbed performance of the array. The variance (σ_B^2) of the random patterns is given by

$$\sigma_B^2 = E_{error} \{B^2\} - E_{error}^2 \{B\}. \quad (3.29)$$

Using the same assumptions as the previous section, the statistical expression for the variance of the perturbed beampattern is :

$$\sigma_B^2 = \sum_{m=1}^N \sum_{n=1}^N \sum_{i=1}^N \sum_{h=1}^N p_{m,n,i,h}^0 (A_{m,n,i,h} B_{m,n,i,h} C_{m,n,i,h} - D_{m,n,i,h}), \quad (3.30)$$

where $p_{m,n,i,h}^0$, $A_{m,n,i,h}$, $B_{m,n,i,h}$, $C_{m,n,i,h}$ and $D_{m,n,i,h}$ are defined in Appendix A.4.

Although the above equation is too complicated to get analytical insights, it can be evaluated numerically and can provide an useful measure of the degraded performance of the perturbed arrays. In figure 3.8, the mean (solid line) and mean+ σ_B beampatterns (dash line) are obtained using 200 Monte Carlo trials. Here, we assume the gain error ($\sigma_g = 0.1$) and the position error ($\sigma_r = 0.2d$) occur simultaneously. From the figure, a prediction can be made that most of individual

Figure 3.8: Mean and mean+ σ_B beampattern ($\sigma_g = 0.1, \sigma_r = 0.2d$)

beampatterns fall in the area between the mean and mean+ σ_B curves. In Fig. 3.9, smaller perturbation errors ($\sigma_r = 0.05d$) are applied. The corresponding mean and mean+ σ_B beampatterns are presented. It is found that the variance of the beampattern is much smaller, which implies that most perturbed beampatterns fall within a much smaller region between the curves mean and mean+ σ_B .

3.3 Performance Comparison for Different Array Syntheses

From (3.20), we know the sensitivity of the array is determined by array tapering, thus different array syntheses have different degraded performances in the same

Figure 3.9: Mean and mean+ σ_B beampattern ($\sigma_r = 0.05d$)

perturbed scenario. In the previous sections, some performance measures, e.g., expected beampattern and variance of perturbed beampattern, were presented. Comparison of these performance measures for different array syntheses can be achieved through their analytical expressions. However, for some of the measures which are also important but cannot be expressed as explicit analytical forms, numerical simulation provides a way to get some insights.

In the following simulations, delay-and-sum, Dolph-Chebyshev, Taylor-Kaiser and endfire supergain array are compared. For the first three arrays, a nine-sensor standard linear array steered to broadside is considered, and the maximum sidelobe level for Dolph-Chebyshev and Taylor-Kaiser arrays is designed to be -30 dB. For the endfire array, a nine-sensor uniform linear array with inter-sensor

spacing $d = 0.2\lambda$ is used. The error-free beampattern and weights of these arrays are shown in Figs. 3.1, 3.2 and 3.3. The performance measures for comparing the arrays are the mean directivity and the mean highest sidelobe level. These measures are different from the parameters of the expected beampattern (shown in Table 3.1) in the previous section because they are obtained by averaging the values over a number of trials.

Fig. 3.10 shows the mean value of the highest sidelobe level versus different perturbation deviations of the phase error. The mean of the highest sidelobe level of all the four arrays are getting larger when the perturbation deviation gets larger. Note that the mean highest sidelobe level of the endfire array becomes 0 dB when the phase error $\sigma_\phi > 1^\circ$. Fig. 3.11 shows the mean directivity index versus the different perturbation deviations. It is obvious that the mean directivity index of endfire array decreases much more rapidly than the other three arrays, although it has the highest directivity in the ideal case.

In a summary, of all the four arrays, the delay-and-sum beamformer has the most stable performance. Although the Dolph-Chebyshev and Taylor-Kaiser array are more sensitive to the perturbations than the conventional array, the degraded performance is not significant. The supergain endfire array, however, is seriously impaired by the perturbations. Even with relatively small perturbations, the endfire array behaves like an isotropic sensor.

Figure 3.10: Mean of highest sidelobe level vs standard deviation of the phase errors

Figure 3.11: Mean of Directivity index vs standard deviation of the phase errors

3.4 Robust Problems for Data-Independent Arrays

In Section 3.2 and 3.3, we analyzed the degraded performance of data-independent arrays. It is obvious that the supergain endfire array is extremely sensitive to small random errors or perturbations, although it is known to offer a significant increase in the array gain over that of other array synthesis methods [52]. Hence, designing a method which provides robustness against random errors is a crucial task for using a data-independent array, especially an endfire supergain array.

Since the 1950's, several people have sought to improve the robustness of array signal processing. Gilbert and Morgan [10] proposed a sensitivity function which measures the susceptibility of the beampattern to random errors in the phases and positions of the sensors. They gave a robust solution by setting a constraint on this sensitivity parameter. Several authors also analyzed the relationship according to different types of array errors. These errors include the errors due to element failure [11, 12], the errors in the amplitudes and the phases of the weights of beamformers [11–13] and the errors in the steering vectors [14, 15, 45]. A closely related problem was examined by Uzsoky and Solymar [15]. Cox et al. addressed the problem of optimizing array gain, while constraining the array's white noise gain. The algorithm gives the closed form of the robust weights by choosing the value of the assumed white noise gain to control the trade-off between directivity and robustness [16]. Unfortunately, all such robust approaches are ad hoc in nature

since there is no easy and reliable way to choose an appropriate constraint value.

In practice, some statistical knowledge about the errors can be obtained experimentally. In the next section, we assume that statistical knowledge of the errors is available, and incorporates it into the design procedure of the robust algorithms.

3.5 Robust Method I

3.5.1 Underlying Idea of Proposed Method

The expected beampattern in the perturbed situation may be never achieved. A more important measure is given by the distribution of the perturbed beampatterns. The mean square error (MSE), which is the square error between the ideal and perturbed responses averaged over variations, can be used to measure the spread of perturbed patterns. Hence, a robust method is proposed which minimizes the MSE subject to an undistorted response in the look direction.

3.5.2 Problem Formulation

From (2.12), the array response in the ideal situation is defined as $H_0(\theta, \phi) = \mathbf{w}^H \bar{\mathbf{a}}(\theta, \phi)$, and the array response perturbed situation is $H(\theta, \phi) = \mathbf{w}^H \mathbf{a}(\theta, \phi)$.

The square error between the two responses over a spatial region is given by

$$\begin{aligned}\epsilon &= E_{\theta,\phi} \{ |H(\theta, \phi) - H_0(\theta, \phi)|^2 \} \\ &= \int_{\phi} \int_{\theta} p(\theta, \phi) | \mathbf{w}^H \mathbf{a}(\theta, \phi) - \mathbf{w}^H \bar{\mathbf{a}}(\theta, \phi) |^2 \sin \theta d\theta d\phi, \quad (3.31)\end{aligned}$$

where $p(\theta, \phi)$ is a non-negative weighting function defined over a solid angle range, i.e.,

$$\int_{\phi} \int_{\theta} p(\theta, \phi) \sin \theta d\theta d\phi = 1. \quad (3.32)$$

It gives emphasis to the response in the direction (θ, ϕ) if $p(\theta, \phi)$ is large. Here, the expectation operator $E_{\theta,\phi} \{ \cdot \}$ means expectation over the spatial region. With the knowledge of the statistical perturbation information, the mean square error averaged over the perturbations is given by

$$\begin{aligned}\bar{\epsilon} &= E_{error} \{ \epsilon \} = E_{error} \{ E_{\theta,\phi} \{ |H(\theta, \phi) - H_0(\theta, \phi)|^2 \} \} \\ &= E_{\theta,\phi} \{ E_{error} \{ |H(\theta, \phi) - H_0(\theta, \phi)|^2 \} \}. \quad (3.33)\end{aligned}$$

The MSE, $\bar{\epsilon}$, can be further derived as

$$\begin{aligned}\bar{\epsilon} &= \mathbf{w}^H E_{\theta,\phi} \{ E_{error} \{ \mathbf{a} \mathbf{a}^H \} \} \mathbf{w} - \mathbf{w}^H E_{\theta,\phi} \{ E_{error} \{ \mathbf{a} \} \bar{\mathbf{a}}^H \} \mathbf{w} \\ &\quad - \mathbf{w}^H E_{\theta,\phi} \{ \bar{\mathbf{a}} E_{error} \{ \mathbf{a}^H \} \} \mathbf{w} + \mathbf{w}^H E_{\theta,\phi} \{ \bar{\mathbf{a}} \bar{\mathbf{a}}^H \} \mathbf{w}. \quad (3.34)\end{aligned}$$

Defining $\mathbf{Q}_0, \mathbf{Q}_1, \mathbf{Q}_2$ and $\mathbf{Q}$ as:

$$\begin{aligned}\mathbf{Q}_0 &= E_{\theta,\phi} \{ \bar{\mathbf{a}} \bar{\mathbf{a}}^H \}; \quad \mathbf{Q}_1 = E_{\theta,\phi} \{ E_{error} \{ \mathbf{a} \mathbf{a}^H \} \}; \\ \mathbf{Q}_2 &= E_{\theta,\phi} \{ E_{error} \{ \mathbf{a} \} \bar{\mathbf{a}}^H \}; \\ \mathbf{Q} &= \mathbf{Q}_1 - \mathbf{Q}_2 - \mathbf{Q}_2^H + \mathbf{Q}_0. \quad (3.35)\end{aligned}$$

Thus, the mean square error $\bar{\epsilon}$ from (3.34) is given by:

$$\bar{\epsilon} = \mathbf{w}^H \mathbf{Q} \mathbf{w}. \quad (3.36)$$

The proposed robust method is obtained by minimizing the mean square error $\bar{\epsilon}$ while maintaining a distortionless response at the look direction, i.e.,

$$\begin{aligned} \min_{\mathbf{w}} \quad & \bar{\epsilon} \\ \text{subject to} \quad & \mathbf{w}^H \bar{\mathbf{a}}(\theta_0, \phi_0) = 1 \end{aligned} \quad (3.37)$$

The optimal weight vector, $\mathbf{w}_o$, can be easily obtained by the Lagrange Multiplier method [57]:

$$\mathbf{w}_o = \lambda_o \mathbf{Q}^{-1} \bar{\mathbf{a}}(\theta_0, \phi_0), \quad (3.38)$$

where λ_o is calculated by the equation

$$\lambda_o = \frac{1}{\bar{\mathbf{a}}^H(\theta_0, \phi_0) \mathbf{Q}^{-1} \bar{\mathbf{a}}(\theta_0, \phi_0)}. \quad (3.39)$$

The minimum MSE, ϵ_o , from (3.36) is

$$\epsilon_o = \frac{1}{\bar{\mathbf{a}}^H(\theta_0, \phi_0) \mathbf{Q}^{-1} \bar{\mathbf{a}}(\theta_0, \phi_0)}. \quad (3.40)$$

3.5.3 Calculation of Matrix $\mathbf{Q}$

In order to calculate (3.38), matrix $\mathbf{Q}$ has to be calculated. Next, we use the errors model introduced in the Section 3.2.1 to calculate matrix $\mathbf{Q}$. The (m, n) element of $\mathbf{Q}_1$ is given by

$$\mathbf{Q}_{1(m,n)} = E_{\theta,\phi} \left\{ \bar{\mathbf{a}}_m \bar{\mathbf{a}}_n^* E_{\text{error}} \left\{ g_m g_n^* e^{j(\Delta\phi_m - \Delta\phi_n)} e^{-j\mathbf{k}^T(\Delta\mathbf{r}_m - \Delta\mathbf{r}_n)} \right\} \right\}. \quad (3.41)$$

Since the variations in the gain, phase and position are assumed to be independent of direction, $\mathbf{Q}_{1(m,n)}$ is given by

$$\mathbf{Q}_{1(m,n)} = \underbrace{E_{error} \left\{ (1 + \Delta g_m)(1 + \Delta g_n) e^{j(\Delta \phi_m - \Delta \phi_n)} e^{-j\mathbf{k}^T(\Delta \mathbf{r}_m - \Delta \mathbf{r}_n)} \right\}}_{C_{m,n}} \mathbf{Q}_{0(m,n)}, \quad (3.42)$$

where $C_{m,n}$ is given by

$$C_{m,n} = (1 + G_{m,n}) e^{-\frac{1}{2}(\Phi_{mm} - 2\Phi_{mn} + \Phi_{nn})} e^{-\frac{k^2}{2}(\Omega_{mm} - 2\Omega_{mn} + \Omega_{nn})}. \quad (3.43)$$

The (m, n) element of $\mathbf{Q}_2$ is

$$\begin{aligned} \mathbf{Q}_{2(m,n)} &= E_{\theta, \phi} \left\{ \bar{\mathbf{a}}_m \bar{\mathbf{a}}_n^* E_{error} \left\{ (1 + \Delta g_m) e^{j\Delta \phi_m} e^{-j\mathbf{k}^T \Delta \mathbf{r}_m} \right\} \right\} \\ &= \underbrace{E_{error} \left\{ (1 + \Delta g_m) e^{j\Delta \phi_m} e^{-j\mathbf{k}^T \Delta \mathbf{r}_m} \right\}}_{D_m} \mathbf{Q}_{0(m,n)}, \end{aligned} \quad (3.44)$$

where D_m is

$$D_m = e^{-\frac{1}{2}\Phi_{mm}} e^{-\frac{k^2}{2}\Omega_{mm}}. \quad (3.45)$$

From (3.42) and (3.44), the (m, n) element of matrix $\mathbf{Q}$ is given by

$$\begin{aligned} \mathbf{Q}_{(m,n)} &= (1 + G_{m,n}) e^{-\frac{\Phi_{mm} - 2\Phi_{mn} + \Phi_{nn} + k^2(\Omega_{mm} - 2\Omega_{mn} + \Omega_{nn})}{2}} \mathbf{Q}_{0(m,n)} \\ &\quad + (1 - e^{-\frac{\Phi_{mm} + k^2\Omega_{mm}}{2}} - e^{-\frac{\Phi_{nn} + k^2\Omega_{nn}}{2}}) \mathbf{Q}_{0(m,n)} \end{aligned} \quad (3.46)$$

If the $p(\theta, \phi)$ is uniform throughout the spatial region, the (m, n) element of $\mathbf{Q}_0$ is:

$$\mathbf{Q}_{0(m,n)} = \frac{1}{4\pi} \int_0^{2\pi} d\phi \int_0^\pi d\theta \sin\theta e^{-j\mathbf{k}^T(\mathbf{r}_m - \mathbf{r}_n)}. \quad (3.47)$$

For a uniform linear array, $\mathbf{Q}_{0(m,n)}$ is:

$$\mathbf{Q}_{0(m,n)} = \frac{1}{2} \int_0^\pi d\theta \sin\theta e^{-j\frac{2\pi(m-n)d\cos\theta}{\lambda}} = \text{sinc}\left(\frac{2\pi(m-n)d}{\lambda}\right). \quad (3.48)$$

The $\text{sinc}()$ function is defined in Appendix A.1.

3.5.4 Robust Solution for Independent Errors Case

It is interesting to consider the robust solution in the independent errors case as described in Section 3.2.2. From (3.46), the matrix $\mathbf{Q}$ is given by

$$\mathbf{Q}_{m,n} = \begin{cases} \mathbf{Q}_{0(m,n)} \cdot \alpha & m = n, \\ \mathbf{Q}_{0(m,n)} \cdot \beta & m \neq n. \end{cases} \quad (3.49)$$

where $\alpha = (1 + \sigma_g^2) - 2e^{-\frac{\sigma_\phi^2 + k^2 \sigma_r^2}{2}} + 1$ and $\beta = e^{-(\sigma_\phi^2 + k^2 \sigma_r^2)} - 2e^{-\frac{\sigma_\phi^2 + k^2 \sigma_r^2}{2}} + 1$. The matrix $\mathbf{Q}$ can be written as

$$\mathbf{Q} = \beta \mathbf{Q}_0 - \beta \mathbf{I} + \alpha \mathbf{I} = \beta \left(\mathbf{Q}_0 + \frac{\alpha - \beta}{\beta} \mathbf{I} \right), \quad (3.50)$$

Therefore, the solution is like diagonal loading, except the value of the diagonal loading factor can be calculated given the statistics of the array perturbations, i.e., σ_g , σ_ϕ and σ_r . For a uniform linear array with half-wavelength interspacing, the matrix $\mathbf{Q}_0$ is the identity matrix. Thus, the robust solution from (3.38) is exactly same as the conventional beamformer weights.

3.6 Robust Method II

3.6.1 Underlying Idea of Proposed Method

In the last section, we proposed the form of the beamformer which has the least MSE between the ideal and perturbed beampatterns. Actually, it is the most insensitive beamformer based on the MSE criterion. However, in some cases, the array

synthesis is predetermined in order to achieve particular parameter requirements, e.g., a Taylor array is chosen for lower sidelobes or a supergain array is selected for high array gain. For these cases, a robust approach is presented based on the idea of minimizing the MSE between the actual and quiescent responses. The difference between robust method I and robust method II is that robust method II is assumed to have a fixed response for a given ideal situation.

3.6.2 Problem Formulation

We define the quiescent response of the array as

$$H_q(\theta, \phi) = \mathbf{w}_q^H \bar{\mathbf{a}}(\theta, \phi), \quad (3.51)$$

where $\mathbf{w}_q$ are the predetermined quiescent weights which control the shape of the beampattern. The MSE, $\bar{\epsilon}'$, between the quiescent response and the actual response over a spatial region is given by

$$\begin{aligned} \bar{\epsilon}' &= E_{\theta, \phi} \{ E_{error} \{ |H(\theta, \phi) - H_q(\theta, \phi)|^2 \} \} \\ &= E_{\theta, \phi} \left\{ E_{error} \left\{ |\mathbf{w}^H \mathbf{a}(\theta, \phi) - \mathbf{w}_q^H \bar{\mathbf{a}}(\theta, \phi)|^2 \right\} \right\}. \end{aligned} \quad (3.52)$$

And $\bar{\epsilon}'$ can be further derived as:

$$\begin{aligned} \bar{\epsilon}' &= \mathbf{w}^H E_{\theta, \phi} \{ E_{error} \{ \mathbf{a} \mathbf{a}^H \} \} \mathbf{w} - \mathbf{w}^H E_{\theta, \phi} \{ E_{error} \{ \mathbf{a} \} \bar{\mathbf{a}}^H \} \mathbf{w}_q \\ &\quad - \mathbf{w}_q^H E_{\theta, \phi} \{ \bar{\mathbf{a}} E_{error} \{ \mathbf{a}^H \} \} \mathbf{w} + \mathbf{w}_q^H E_{\theta, \phi} \{ \bar{\mathbf{a}} \bar{\mathbf{a}}^H \} \mathbf{w}_q \\ &= \mathbf{w}^H \mathbf{Q}_1 \mathbf{w} - \mathbf{w}^H \mathbf{Q}_2 \mathbf{w}_q - \mathbf{w}_q^H \mathbf{Q}_2^H \mathbf{w} + \mathbf{w}_q^H \mathbf{Q}_0 \mathbf{w}_q. \end{aligned} \quad (3.53)$$

Our proposed method is to minimize the mean square error, $\bar{\epsilon}'$, and maintain a distortionless response in the desired direction (θ_0, ϕ_0) :

$$\begin{aligned} \min_{\mathbf{w}} \quad & \bar{\epsilon}' \\ \text{subject to} \quad & \mathbf{w}^H \bar{\mathbf{a}}(\theta_0, \phi_0) = 1 \end{aligned} \quad (3.54)$$

The optimal weight vector, $\mathbf{w}_o$, can be easily obtained by the Lagrange Multiplier method:

$$\mathbf{w}_o = \mathbf{Q}_1^{-1}(\mathbf{Q}_2 \mathbf{w}_q + \lambda_o \bar{\mathbf{a}}(\theta_0, \phi_0)), \quad (3.55)$$

where λ_o is calculated by the equation

$$\lambda_o = \frac{1 - \mathbf{w}_q^H \mathbf{Q}_2^H \mathbf{Q}_1^{-1} \bar{\mathbf{a}}(\theta_0, \phi_0)}{\bar{\mathbf{a}}^H(\theta_0, \phi_0) \mathbf{Q}_1^{-1} \bar{\mathbf{a}}(\theta_0, \phi_0)}. \quad (3.56)$$

3.7 Numerical Examples

We now proceed to demonstrate the proposed array methods developed in the previous sections with a couple of examples. In the following examples, a nine-element linear array steered to endfire with a nominal inter-element spacing (d) of one-fifth a wavelength (λ) is used. The supergain endfire synthesis which provides an high array gain performance in the ideal case is used as the quiescent weights to demonstrate robust method II. In the following examples, the mean performance measures are obtained by averaging 200 independent Monte-Carlo trials.

Figure 3.12: Beampattern comparison (standard endfire and robust II BF)

3.7.1 Independent Errors Case

In the first experiment, we assume the phase errors in the sensors are independent and identically Gaussian distributed with zero mean and standard deviation $\sigma_\phi = 20^\circ$. Fig. 3.12 shows the beampattern of the endfire array in the ideal case (dash-dot line) and mean beampattern of the robust method II (dash line) and standard endfire array (non-robust beamformer) in the perturbed case (solid line). It is found that the errors in sensor phases make the mean pattern of the endfire beamformer almost uniform over the whole spatial region. This implies that the non-robust endfire array is seriously degraded and behaves like an isotropic sensor in perturbed situations. In contrast, the robust II method can obtain a sidelobe level of nearly -15 dB in the expected beampattern.

Figure 3.13: Beampattern comparison (robust I BF in ideal and perturbed cases)

Then, let us compare this with the solution of the robust I method. Fig. 3.13 shows that the beampattern of the robust method I in the ideal case (solid line) and its mean beampattern in the perturbed case (dash line). It is obvious that the robust II beamformer has lower sidelobes and a narrower beamwidth than that of the robust I beamformer in both ideal and perturbed situations. This illustrates that robust II beamformer can keep a higher directivity in both ideal and perturbed situations. However, note that the deviation between the perturbed and unperturbed mean patterns of the robust II is larger than that of the robust I. This demonstrates that the robust I method is less sensitive to the perturbations.

Since the endfire array offers high array gain in spherically isotropic and cylindrically isotropic noise field in the ideal case, we now show the array gain of the

Figure 3.14: Array gain of the arrays in a spherically isotropic noise field
(standard endfire, robust I and robust II BF)

standard endfire, robust I and robust II array in both of these two situations. The isotropic noise field was introduced in Section 2.5.3, and the covariance matrices of spherically isotropic and cylindrically isotropic noise fields are calculated in Appendices A.5 and A.6. Fig. 3.14 shows array gains of the standard endfire, robust I and robust II array in a spherically isotropic noise field when the perturbation deviation in phase (σ_ϕ) changes from 0° to 10° . Fig. 3.15 shows those in a cylindrically isotropic noise field. It is found that in both situations, the array gain of robust II is much larger than that of the standard endfire array, which illustrates that the proposed robust method II maintains high resolution performance even in the perturbation situations. Although the array gain of robust I method is lower than that of robust II, it is almost constant for different perturbation standard

Figure 3.15: Comparison of array gain in a cylindrically isotropic noise field
(standard endfire, robust I and robust II BF)

deviations, which implies that it has high stability against perturbations.

Next, we evaluate the cost function, the MSE of the standard endfire, robust I and robust II arrays in different perturbation situations. Fig. 3.16 shows average MSEs of the three versions of arrays when the perturbation deviation in phase (σ_ϕ) changes from 2° to 10° . As expected, all of the three MSEs increase when the perturbation variance gets larger. However, the robust I array has the smallest MSE, which implies that the robust I array is much less sensitive to the perturbations compared to the other types of array. Next, we compare the mean of highest sidelobe level of the three versions of the array. From Fig. 3.17, it is found that the robust II array can keep a lower sidelobe level than the non-robust one. Note that when the phase error more than 15° , the mean highest sidelobe level of the

Figure 3.16: Comparison of MSE in the independent errors case
(standard endfire, robust I and robust II BF)

Figure 3.17: Mean highest sidelobe level in the independent errors case
(standard endfire, robust I and robust II BF)

Figure 3.18: Non-uniform weighting function $p(\theta, \phi)$ of spatial region

robust II method is larger than that of the robust I method.

In the above simulations, we used an uniform weighting function, $p(\theta, \phi)$ of (3.31), for the whole spatial range. In practice, we may approximately know a certain spatial region that interferences often focus on. Thus, the performance of this region is more important than that of elsewhere. By using a non-uniform weighting function, the proposed robust method II can concentrate the robustness capability on a certain region. In this simulation, we assume the scenario and perturbation is same as the above simulations.

Fig. 3.19 shows the expected beampatterns of robust method II with uniform and non-uniform weighting function. The non-uniform weighting function we used is shown in Fig. 3.18. Here, we set the weighting function of the sidelobe region ($50^\circ - 180^\circ$), w-high, 10 times as high as that of the mainlobe region (w-low). Comparing the two patterns, it is found that the non-uniform one has a lower sidelobe level but wider mainlobe than the uniform one. This shows that the non-uniform weighting improves the performance of sidelobe region by sacrificing the

Figure 3.19: Expected beam pattern of robust II BF with
uniform and non-uniform weighting functions

performance of mainlobe region. Fig. 3.20 shows the MSE of the whole region and sidelobe region of the robust II BF with uniform and non-uniform weighting functions. The uniform weighting BF has a lower MSE for the whole region but the non-uniform weighting BF has a lower MSE for the sidelobe region.

3.7.2 Correlated Errors Case

In previous simulations, we assume the perturbations present in one sensor is independent of the others. In the following experiment, we show the performance of proposed methods in the correlated perturbed cases. The correlated position errors are obtained from the perturbation model presented in Appendix C. The

Figure 3.20: MSEs of the whole and sidelobe regions of the robust II BF with uniform and non-uniform weighting functions

correlation length L is set to be 3 times the sensor interspacing and the standard deviation of the position error is $0.2d$.

Fig. 3.21 shows the average beampattern of the standard endfire array and the robust II beamformer in the correlated position errors case. It is found that both of the two beamformers are less sensitive to the correlated errors than independent errors. As in the independent errors case, the robust method II can improve the performance of the endfire array greatly by getting a much higher array gain and lower sidelobe level in the perturbed situations. Fig. 3.22 shows the average beampattern of robust I beamformer for the ideal and correlated position errors cases. Fig. 3.23 shows the MSE of three versions of the array for the correlated position errors cases. It is found from these figures that in both the correlated and

Figure 3.21: Beam pattern comparison in the correlated errors case
(standard endfire and robust II BF)

Figure 3.22: Beam pattern comparison in the correlated errors case
(robust I BF in ideal and perturbed cases)

Figure 3.23: Comparison of MSE in the correlated errors case
(standard endfire, robust I and robust II BF)

independent perturbation cases, the robust I beamformer can offer the most stable solution, while the robust II beamformer can keep comparatively high directivity and low MSE performances.

3.7.3 Design Perturbation Errors

In previous simulations, we assumed the statistics of the perturbation errors, e.g., standard deviation σ_ϕ in the independent phase errors case are precisely known. That is, the perturbation in designing robust algorithms, which is referred to as design perturbation errors, is the same as the actual perturbation errors. However, in practical applications, the statistics of perturbations can not be obtained

precisely and it may also change with time. So in the following test case, we will investigate the performance of the robust algorithms when the design perturbation errors and actual perturbation errors differ.

In this experiment, we assume that the array suffers from independent phase errors, changing from 0° to 10° , while the design perturbation deviation is fixed as 0° , 5° and 10° , respectively. First, we show the performance of the robust II method. Fig. 3.24 shows the array gain of the robust II array in a spherically isotropic noise field. Note that the solid curve represents the performance of the non-robust endfire array. It is found that when the actual perturbation is larger than 0.5° , although the design perturbation may overestimate (design $\sigma_\phi > \text{actual } \sigma_\phi$) or underestimate (design $\sigma_\phi < \text{actual } \sigma_\phi$), robust II has a much larger array gain than the non-robust one. When the actual perturbation is larger than 8° , the robust beamformer with design perturbation $\sigma_\phi = 10^\circ$ obtains a higher array gain than that with design perturbation $\sigma_\phi = 5^\circ$. This shows that a more accurate estimation of the error can offer better performance. However, the difference between the design and actual perturbation errors does not lead to a great decrease in the array gain. Fig. 3.25 shows the MSE of robust II in the same scenario. It is also found that although the design deviation differs from actual perturbation deviation, the robust II still can offer much better performance than the non-robust one.

Fig. 3.26 and Fig. 3.27 show the array gain and MSE of robust I in the same scenario. It is found that the difference between the design and actual perturbation

Figure 3.24: Array gain of the robust II array for
different design perturbation errors

Figure 3.25: MSE of the robust II array for
different design perturbation errors

Figure 3.26: Array gain of the robust I array for
different design perturbation errors

errors hardly affects the performance of the robust I method. Also, the robust I method can offer a more stable performance than the robust II method.

3.7.4 Dolph-Chebyshev Array

In the previous examples, we used a supergain endfire array to illustrate the performance of the proposed robust arrays. In the next set of simulations, we proceed to demonstrate the proposed array methods on a Dolph-Chebyshev array. Here, a nine-element standard array steering to broadside is used. The Chebyshev synthesis whose sidelobe level is -30 dB in the ideal case, is used as the quiescent weights for the robust method II. In Fig. 3.28, the beampattern of the Chebyshev array

Figure 3.27: MSE of the robust I array for
different design perturbation errors

in the ideal case (dash-dot line), the mean beampattern of robust method II (dash line) and the standard (non-robust) Chebyshev array (solid line) in the perturbed case are shown. It is found that the mean patterns of robust II and non-robust Chebyshev arrays are very close. The sidelobe level of the mean beampattern is increased about 8 dB. Fig. 3.29 shows that the beampattern of the robust method I in the ideal case (solid line) and its mean beampattern in the perturbed case (dash line). Comparing the two figures, the mean pattern of robust I has a larger sidelobe level but a narrower beamwidth. However, the deviation between the perturbed and unperturbed mean pattern of the robust I is smaller than that of the robust II.

Table 3.3 shows a comparison of the performance measures of the standard

Figure 3.28: Beampattern comparison (standard Chebyshev and robust II BF)

Chebyshev, robust I and robust II arrays. It is found that the directivity index and MSE of the expected beampattern of the standard Chebyshev array are not significantly degraded and these two measures of robust II are only a little better. However, the mean directivity index and highest sidelobe level of individual trials have a relatively larger improvement for the robust II BF. The robust I method, although offers a higher directivity and lower MSE, has a much higher sidelobe level in its expected beampattern. The standard Chebyshev array can be used to offer low mean sidelobe level. However, when a stable and high directivity performance is wanted, robust I method is recommended.

Figure 3.29: Beam pattern comparison (robust I BF in ideal and perturbed cases)

Table 3.3: Performance measures

(standard Chebyshev, robust I and robust II BF)

Measures	standard Chebyshev	Robust I	Robust II
DI of the expected beam pattern (dB)	8.34	9.08	8.44
MSE	0.0154	0.0129	0.015
Mean DI (dB)	8.04	9.02	8.56
Mean highest sidelobe-level (dB)	-13.6	-9.90	-14.4

3.8 Comparison of Two Proposed Robust Methods

The two proposed robust methods are both based on the MSE criterion. The first robust method minimizes the MSE between the perturbed and ideal responses, and it can be considered as the most insensitive array in perturbed situations. The second robust method has a pre-defined set of quiescent weights and the robustness is obtained by minimizing the MSE between the perturbed and quiescent responses. From the numerical results, we find that the robust II method can offer higher resolution and lower sidelobes when the perturbation is not very large, while the robust I method can offer a more stable performance in perturbation situations.

When we design an array, if the stability is the most important performance measure for the array, we should consider using the robust I algorithm. However, if a particular array synthesis is required to achieve certain parameter requirements, e.g., a Taylor array is chosen for lower sidelobes or a supergain array is selected for high array gain, the robust II algorithm can be used to keep high directivity and small MSE in the perturbed case.

Chapter 4

Robust Techniques for Adaptive Beamformers

4.1 Introduction

In contrast to data-independent arrays, an adaptive array chooses its weights based on the statistics of the data received at the array. Therefore, it can dynamically null out the directional interferences in a changing signal environment. This self-adjusting capability renders adaptive arrays more flexible and gives a potentially better reception performance that would be difficult to achieve for data-independent arrays. A number of adaptive approaches were proposed based on different criteria of performance measures or different knowledge of the desired signal characteristics [1, 2, 8, 42, 58, 59].

Adaptive beamformers requires some knowledge of the desired signal characteristics, either its statistics, its direction or its data vector. If the required knowledge is inaccurate, the adaptive array will null the desired signal as if it were interference. This cancellation phenomena is especially significant when SNR is large. Several approaches have been proposed to reduce the degradation caused by uncertainties in array characteristics and environmental perturbations. In this chapter, we will analyze the degraded performance of the Capon beamformer and propose two robust methods that can avoid "signal nulling" while keeping a high SINR in perturbed situations.

The chapter begins with an introduction to the fundamentals of adaptive arrays. The Capon beamformer, which is the main subject of this chapter, is explained in detail in Section 4.2.2. In Section 4.3, the degraded performance of the Capon array in perturbed situations is analyzed. In Section 4.4, some existing robust methods are listed and compared in detail, with emphasis on the new kind of robust Capon beamformers (RCBs). The robust methods are classified into two kinds: constrained robust methods and responsive robust methods. The advantages and disadvantages of both kinds of robust methods are explained. In Section 4.5, a robust method which combats phase error is proposed. This robust method obtains the true phase vector by tracking the maximum of the output power iteratively. In the Section 4.6, another method which based on RCBs and combats large direction errors is proposed. The method efficiently shrinks the uncertainty region by finding the DOA (direction-of-arrival) of the desired signal

first, and thus renders a higher SINR performance than the RCBs in perturbed situations.

4.2 Concepts of Adaptive Beamformers

4.2.1 Fundamentals

An adaptive array chooses its weights to optimize the array response so that the output signal contains minimal contributions due to the noise and signals arriving from directions other than the desired direction. Adaptive beamformers are used to find the DOA or to estimate the signal power when there are large directional interferences in the environment.

Generally, adaptive arrays are of two kinds: nonparametric and parametric [60]. The parametric method includes MUSIC [61], maximum likelihood (ML) [62, 63] methods and so on. The nonparametric methods include Generalized Side-lobe Canceler (GSC) [58], Maximum SNR [42], reference signal [59], and linear constrained minimum variance (LCMV) beamformers [1, 2, 8], based on different criteria of performance measures or different knowledge of the signal characteristics.

In narrowband applications, a well-known example of the LCMV beamformer is the Capon beamformer [64], and in broadband applications, the well studied LCMV beamformer is the Frost beamformer [65]. In this thesis, we mainly analyze

the degraded performance of the Capon beamformer in perturbed situations.

4.2.2 Capon Beamformer

The Capon beamformer assumes the DOA of the desired signal is known. It finds the optimal weight vector by minimizing the output power of the beamformer while maintaining a distortionless response in the look direction, i.e.,

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathbf{w}^H \mathbf{R} \mathbf{w} \\ \text{subject to} \quad & \mathbf{w}^H \bar{\mathbf{a}}_0 = 1 \end{aligned} \quad (4.1)$$

In the ideal case, the nominal ASV of the desired signal, $\bar{\mathbf{a}}_0$, is same as the true ASV of the desired signal, $\mathbf{a}_0$. The optimal solution of the Capon beamformer is given by

$$\mathbf{w}_o = \frac{\mathbf{R}^{-1} \mathbf{a}_0}{\mathbf{a}_0^H \mathbf{R}^{-1} \mathbf{a}_0}. \quad (4.2)$$

Hence, the output of the Capon beamformer in the ideal case is

$$p_{out} = \mathbf{w}^H \mathbf{R} \mathbf{w} = \frac{1}{\mathbf{a}_0^H \mathbf{R}^{-1} \mathbf{a}_0}. \quad (4.3)$$

In practice, the theoretical covariance matrix, $\mathbf{R}$, is always replaced by the estimated covariance matrix, $\hat{\mathbf{R}}$, from (2.22). The solution of a Capon beamformer for a finite number of samples is given by

$$\hat{\mathbf{w}}_o = \frac{\hat{\mathbf{R}}^{-1} \mathbf{a}_0}{\mathbf{a}_0^H \hat{\mathbf{R}}^{-1} \mathbf{a}_0}. \quad (4.4)$$

It converges to the theoretical solution of (4.2) when number of data samples goes to infinity.

A Capon beamformer has superior power resolving capability compared to the conventional beamformer, provided that the array steering vector (ASV), $\bar{\mathbf{a}}_0$, matches the actual data vector of the desired signal, $\mathbf{a}_0$. Fig. 4.1 shows the output power of a Capon beamformer and a delay-and-sum beamformer in the unperturbed case. Here, a nine-element standard linear array is used. The power

Figure 4.1: Power estimation of the Capon and conventional beamformer

of the desired signal is 10 dB and its assumed DOA is 0° . A strong interference with 20 dB power is assumed to be coming from 45° . The background white noise power is 0 dB. It is found that the Capon beamformer can obtain more accurate power estimation than the delay-and-sum beamformer. This is because for a delay-

and-sum beamformer, signals other than look direction leak through the sidelobes and give rise to overestimation of the power. The Capon beamformer, however, maintains uniform response in the look direction but rejects all signals from other directions. Appendix D.1 proves that power estimation of the Capon beamformer is always lower than that of a delay-and-sum beamformer.

In practice, however, because of differences between the assumed and true DOA, or because the array calibration errors are present in the sensor gain, phase and position, the array steering vector, $\bar{\mathbf{a}}_0$, always deviates from the true data vector of the desired signal. Whenever this happens, the Capon beamformer may misinterpret the desired signal as interference and give rise to 'signal nulling'. Some authors have analyzed the effects of the Capon beamformer due to perturbations and found that its performance may even be worse than an isotropic sensor [26, 66–70].

Fig. 4.2 shows beampatterns of the Capon beamformer in three independent Monte-Carlo trials. In this simulation, we assume the scenario is the same as above except that there exist random perturbations in the sensors' phases which are the same as that of Fig. 3.4. From the figure, we observe that the perturbed beampatterns of the Capon beamformer have deep nulls both at the signal and interference directions, and the sidelobes of the beampatterns are much larger than those of the ideal beampattern. No doubt, this will lead to drastically reduced array gain and underestimate the power of the desired signal, which is much more serious than that of a data-independent array. In the next section, we will analyze

Figure 4.2: Individual perturbed beampatterns and ideal beampattern

the performance of the Capon array that are affected by the perturbations.

4.3 Performance Analysis for Perturbed Capon Array

4.3.1 SINR of Perturbed Cases

In Appendix D.2, an analytical expression of the degraded SINR in the perturbed case is derived, which is given by :

$$\widetilde{SINR} = \frac{SINR_o \cos^2 \alpha}{1 + SINR_o (SINR_o + 2) \sin^2 \alpha}. \quad (4.5)$$

It is a function of the ideal SINR ($SINR_o$) and the mismatch angle (α) which measures the mismatch between the ASV and true data vector, where α is defined as

$$\alpha = \cos^{-1} \left\{ \left[\frac{|\mathbf{a}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0|^2}{(\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0)(\bar{\mathbf{a}}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0)} \right]^{\frac{1}{2}} \right\}. \quad (4.6)$$

From Eq. (4.5), it is found that the perturbed SINR reduces to $SINR_o$ when there is no error in the ASV ($\alpha = 0$). Given a specific perturbed case (angle α is fixed), if the ideal SINR is small, i.e., $SINR_o(SINR_o + 2) \ll \frac{1}{\sin^2 \alpha}$, we have

$$\widetilde{SINR} \approx SINR_o \cos^2 \alpha, \quad (4.7)$$

while if $SINR_o(SINR_o + 2) \gg \frac{1}{\sin^2 \alpha}$, we have

$$\widetilde{SINR} \approx \frac{\cos^2 \alpha}{\sin^2 \alpha (SINR_o + 2)}. \quad (4.8)$$

This means that the perturbed SINR of the Capon beamformer increases linearly with its ideal SINR when the ideal SINR is low. However, in the case of high ideal SINR, the SINR in the perturbed situations decreases rapidly with its ideal SINR.

Next, we show the relationship in an intuitive way through simulations. In the first simulation, we assume the scenario is same as that of Fig. 4.1 except that the desired signal is coming from 2° , not the assumed angle 0° . The SNR changes from 0 dB to 20 dB.

Fig. 4.3 shows the relationship between input SNR and the SINR in the ideal and perturbed cases. We observe that the higher input SNR, the higher SINR can be obtained in the ideal case, but the lower SINR is obtained in the perturbed case

Figure 4.3: SINR vs SNR in the ideal and perturbed cases

when the input SNR is larger than about -4 dB. In other words, the degraded performance is very severe when the input SNR is large.

In the second simulation, the input SNR is fixed as 10 dB, and the signal direction error (mismatch angle between the look direction and the true DOA) changes from 0° to 5° . Fig. 4.4 shows that larger direction error gives rise to large mismatch angle α of the ASV (figure a) and lower SINR in the perturbed case (figure b). Comparing the two perturbed SINR in the delay-and-sum and the Capon beamformers, we observe that both beamformers have a poorer performance when the direction error is getting large, but the delay-and-sum beamformer is less sensitive to the perturbations. When the direction error is larger than 0.3 degree, the performance of the Capon beamformer is worse than the delay-and-

Figure 4.4: (a) Signal direction error vs mismatch angle α

(b) Signal direction error vs the SINR in the ideal and perturbed cases

sum beamformer.

Next, we investigate the imperfect array problem in a specific case - no directional interference in the environment. This assumption is made by the fact that even if array imperfection occurs, the Capon beamformer can still null out interference signals (shown as Fig. 4.2). In this case, the mismatch angle of the ASV α no longer depends on the noise covariance matrix, which is usually unknown in practical situations.

When there is no directional interference, the SINR of the Capon beamformer in ideal case is:

$$SINR_o = \frac{N\sigma_0^2}{\sigma_n^2} = N \cdot SNR. \quad (4.9)$$

We define $\rho = \mathbf{a}_0^H \bar{\mathbf{a}}_0$ as the spatial correlation between the ASV and true data vector, and assume $\mathbf{a}_0^H \mathbf{a}_0 = \bar{\mathbf{a}}_0^H \bar{\mathbf{a}}_0 = N$. Thus, the mismatch angle α is given by

$$\cos^2 \alpha = \frac{|\rho|^2}{N^2}. \quad (4.10)$$

Substituting (4.10) into (4.5), the perturbed SINR of the Capon beamformer is derived as a function of SNR and spatial correlation ρ :

$$\widetilde{SINR} = \frac{SNR \frac{|\rho|^2}{N}}{(NSNR)^2(1 - \frac{|\rho|^2}{N^2}) + 2NSNR(1 - \frac{|\rho|^2}{N^2}) + 1}. \quad (4.11)$$

The perturbed SINR of a delay-and-sum beamformer in this case is given by

$$\widetilde{SINR}_{DSB} = \frac{\sigma_0^2 |\mathbf{w}^H \mathbf{a}_0|^2}{\sigma_n^2 \mathbf{w}^H \mathbf{w}} = \frac{|\rho|^2}{N} SNR. \quad (4.12)$$

It is found that the perturbed SINRs of both beamformers reduce to $N \cdot SNR$ when there is no error in the ASV ($|\rho|^2 = N^2$). Given a specific perturbed case (ρ is fixed), if SNR is small, i.e., $(NSNR)^2(1 - \frac{|\rho|^2}{N^2}) + 2NSNR(1 - \frac{|\rho|^2}{N^2}) \ll 1$, the perturbed SINR of the Capon beamformer is give by

$$\widetilde{SINR} \approx \frac{|\rho|^2}{N} SNR, \quad (4.13)$$

which implies the output SINR is proportional to input SNR and behaves like the delay-and-sum beamformer. While if SNR is large, i.e., $(NSNR)^2(1 - \frac{|\rho|^2}{N^2}) + 2NSNR(1 - \frac{|\rho|^2}{N^2}) \gg 1$, the perturbed SINR of the Capon beamformer is given by

$$\widetilde{SINR} = \frac{\frac{|\rho|^2}{N}}{N^2 SNR(1 - \frac{|\rho|^2}{N^2}) + 2N(1 - \frac{|\rho|^2}{N^2})}, \quad (4.14)$$

which means that the SINR of the Capon beamformer in the perturbed case decreases with the SNR if the SNR is high.

Figure 4.5: Signal direction error vs SINR

(Capon and delay-and-sum beamformers)

When there exist perturbations in array characteristics, the spatial correlation between the ASV and actual data vector can be calculated. For steering direction error, ρ can be obtained from (3.2)

$$|\rho(\theta_0, \theta_1)|^2 = |\mathbf{a}(\theta_0)^H \mathbf{a}(\theta_1)|^2 = \frac{\sin^2 [N \pi \frac{d}{c} f(\sin \theta_1 - \sin \theta_0)]}{\sin^2 [\pi \frac{d}{c} f(\sin \theta_1 - \sin \theta_0)]}, \quad (4.15)$$

where θ_0 is look direction and θ_1 is actual signal direction. For example, if we assume signal direction is 0 degree, while the desired signal comes from 2 degree, the spatial correlation becomes $|\rho(\theta_0, \theta_1)|^2 = 74.7$ ($\cos^2 \alpha = 0.92$). Substituting (4.15) into (4.11), an estimation of the SINR in the Capon beamformer is obtained. Fig. 4.5 shows signal direction error versus SINR in the Capon and delay-and-sum beamformers. It is found that the Capon beamformer is much more sensitive to the angular errors than the delay-and-sum beamformer especially for the large SNR

case.

4.3.2 Expected Performance of Perturbed Cases

In the previous section, the SINR of the Capon beamformer with a fixed error was analyzed. However, since the perturbations are random, the analysis of the expected performance is important. In this section, the expected performance of the Capon beamformer is deduced by assuming some statistical knowledge of the perturbations is known.

To simplify the calculation, we assume the ASV and the true data vector are related by

$$\bar{\mathbf{a}}_0 = \mathbf{a}_0 + \boldsymbol{\delta}, \quad (4.16)$$

where $\boldsymbol{\delta}$ is the ASV error and has the following statistical properties

$$E_{error}\{\boldsymbol{\delta}\} = \mathbf{0}; \quad E_{error}\{\boldsymbol{\delta}\boldsymbol{\delta}^H\} = \mathbf{R}_e. \quad (4.17)$$

It is also closely related to the perturbation model we proposed in Section 3.2.1.

From the proposed perturbation model, the m th element of the ASV and actual data vector is

$$\bar{\mathbf{a}}_{0,m} = e^{j\phi_m^0}, \quad \mathbf{a}_{0,m} = (1 + \Delta g_m)e^{j(\phi_m^0 + \Delta\phi_m)}, \quad (4.18)$$

where ϕ_m^0 represents the original phase in the sensor, which is given by

$$\phi_m^0 = \mathbf{k}_0^T \mathbf{r}_m^0. \quad (4.19)$$

The ASV error in the m th element, δ_m , can be calculated:

$$\delta_m = \bar{\mathbf{a}}_{0,m} - \mathbf{a}_{0,m} = e^{j\phi_m^0} - (1 + \Delta g_m) e^{j(\phi_m^0 + \Delta\phi_m)}, \quad (4.20)$$

If the gain and phase error is small, the above equation can be approximated by a first-order Taylor series

$$\delta_m \approx e^{j\phi_m^0} - \left[e^{j\phi_m^0} (1 + j\Delta\phi_m) - \Delta g_m e^{j\phi_m^0} (1 + j\Delta\phi_m) \right] \approx -e^{j\phi_m^0} (j\Delta\phi_m + \Delta g_m). \quad (4.21)$$

By the assumption that $E_{error} \{\Delta \mathbf{g}\} = \mathbf{0}$ and $E_{error} \{\Delta \phi\} = \mathbf{0}$, $E_{error} \{\Delta \delta\} = \mathbf{0}$, the (m, n) element of the covariance matrix of ASV error, $\mathbf{R}_e$, is given by

$$\begin{aligned} \mathbf{R}_{e(m,n)} = E_{error} \{ \delta \delta^H \}_{m,n} &= E_{error} \left\{ e^{j\phi_m^0} e^{-j\phi_n^0} (j\Delta\phi_m + \Delta g_m) (-j\Delta\phi_n + \Delta g_n) \right\} \\ &= e^{j(\phi_m^0 - \phi_n^0)} (\Phi_{m,n} + \mathbf{G}_{m,n}). \end{aligned} \quad (4.22)$$

When the phase and gain errors are independent, i.e., $E_{error} \{\Delta\phi_m \Delta\phi_n\} = 0$; $E_{error} \{\Delta g_m \Delta g_n\} = 0$, ($m \neq n$), the covariance matrix $\mathbf{R}_e$, is given by

$$\mathbf{R}_e = (\sigma_g^2 + \sigma_\phi^2) \mathbf{I}. \quad (4.23)$$

In Appendix D.3, an analytical expression of the expected SINR in the perturbed case is derived (D.26):

$$E_{error} \{ \widetilde{SINR} \} = \frac{SINR_o^2 + \sigma_0^4 e_1}{SINR_o^2 \sigma_0^2 e_2 + SINR_o (1 - \sigma_0^4 e_1 + 2\sigma_0^2 e_2) + \sigma_0^2 e_2 - 2\sigma_0^4 e_1} \quad (4.24)$$

This expression can be used to compute the expected SINR for the Capon array by given the set of parameters: $\sigma_0^2, \mathbf{R}_e, SINR_o$.

Figure 4.6: (a) Expected mismatch angle α vs phase error σ_ϕ

(b) Expected SINR vs phase error σ_ϕ

Fig. 4.6 shows the mean mismatch angle of the ASV α (figure a) and the expected SINR (figure b) versus phase error σ_ϕ . In this simulation, the scenario is same as what we described in Section 4.2.2, except that we assume there are independent zero mean Gaussian phase errors in the array sensors.

4.4 Existing Robust Methods for Adaptive Beamformers

4.4.1 Principles of Robust Methods

The analysis of the degraded performance of the Capon beamformer when it suffers different imperfections and perturbations has received much attention since the 1970s, and there are several types of robust approaches [68, 71–73]. One common method is to set multiple gain constraints at different angles in the vicinity of the assumed look direction [8, 17, 18]. These are known in the literature as multiple-point constraints, which can be expressed as

$$\min_{\mathbf{w}} \quad \mathbf{w}^H \mathbf{R} \mathbf{w} \quad \text{subject to} \quad \mathbf{C}^H \mathbf{w} = \mathbf{f}, \quad (4.25)$$

where $\mathbf{C}$ is a $N \times L$ matrix of array manifolds in the L constrained directions, and $\mathbf{f}$ is a $L \times 1$ vector specifying the desired response in each constrained direction. We may similarly constrain the derivative of the response of the array to be zero at the assumed look direction. This constraint can be expressed in the same form as (4.25). These are called derivative constraints [19–22]. The solution of the (4.25)

$$\mathbf{w}_o = \mathbf{R}^{-1} \mathbf{C} (\mathbf{C}^H \mathbf{R}^{-1} \mathbf{C})^{-1} \mathbf{f}. \quad (4.26)$$

With these constraints, the widened beamwidth is achieved at the cost of reduced capability in interference suppression because multiple constraints consume the degrees-of-freedom of the array processor. Therefore, it is particularly significant for an array with small number of elements. Furthermore, for this kind of robust

method, it is not clear how best to pick the additional constraints and the effect of the constraints is difficult to predict.

The eigenspace-based robust method [23–25] uses the projection of the presumed steering vector onto the sample signal-plus-interference subspace instead of the presumed steering vector. Although this robust beamformer is applicable to arbitrary steering vector mismatch, it works well only when the numbers of signals and interferences are exactly known and the signal-to-noise-ratio (SNR) is high enough.

The most popular robust approach is diagonal loading [20, 27], which offers robustness by adding a positive value to the diagonal terms of the covariance matrix. An explanation for the robustness of diagonal loading is that the added diagonal value effectively reduces the SNR, and thus reduces sensitivity of the array to the perturbations. Several robust approaches which set quadratic constraints have the same effect as diagonal loading, although they do not add diagonal terms explicitly [28, 74]. For example, Cox set a quadratic constraint to limit the norm of the weights [28]:

$$\min_{\mathbf{w}} \quad \mathbf{w}^H \mathbf{R} \mathbf{w} \quad \text{subject to} \quad \mathbf{w}^H \bar{\mathbf{a}}_0 = 1 \quad \text{and} \quad \mathbf{w}^H \mathbf{w} \leq \epsilon. \quad (4.27)$$

The optimal weights of the diagonal loading method is

$$\mathbf{w}_o = \frac{(\hat{\mathbf{R}} + \mu \mathbf{I})^{-1} \bar{\mathbf{a}}_0}{\bar{\mathbf{a}}_0^H (\hat{\mathbf{R}}^{-1} + \mu \mathbf{I})^{-1} \bar{\mathbf{a}}_0}, \quad (4.28)$$

where μ is the diagonal loading factor determined by the value of quadratic constraint. Although the diagonal loading robust methods are applicable to arbitrary

steering vector mismatches, their main shortcoming is that there is no easy and reliable way to choose an appropriate diagonal loading factor.

Recently, a new class of robust algorithms, here referred to as the robust Capon beamformers (RCBs) [34–38], has been developed. These algorithms assume the true data vector is confined within an uncertainty region which is an ellipsoid centered at the ASV. The true data vector is estimated by finding the maximum output power of the beamformer in the uncertainty region bounded by the ellipsoid. The RCBs offer robustness against ASV mismatch assuming the boundary of the ellipsoid represents the worst-case scenario.

Besides the robust approaches described above, there are some other robust methods based on different knowledge of the desired signals or based on specific implementations [30–33, 75–79] .

4.4.2 Classification of Robust Capon Beamformer

In summary, the robust Capon methods can be separated into two categories based on different underlying ideas. One is constrained robust methods and the other is responsive robust methods. Constrained robust methods reduce sensitivity by imposing additional constraints on the weight vector to avoid cancellation or heavy ASV distortion in the presence of a specific model mismatch. The efficiency of these methods is strictly tied to the accuracy of the mismatch assumptions. Constrained robust methods consume the degrees of freedom and results in widening and flat-

tening the main beam. The robust techniques introduced above which impose point, derivative or quadratic constraints are of this type [19–21, 27, 28, 74]. Constrained robust methods work well under a wide range of scenarios, but sacrifice the capability of interference suppression with respect to an optimal beamformer with an accurate ASV. Also, in the absence of array imperfections, the weight vector of the constrained robust method becomes biased due to additional constraints.

Responsive robust methods attempt to 'respond' to the current environment by learning the true data vector from the observations and then using the information as if it were known exactly. This type of robust method [23–25, 36–38, 80] estimate the data vector by signal subspace or by other estimation procedures. Good ASV estimates can be obtained under the preferred environment of high SNR and slowly fluctuating DOA. The responsive robust methods have as good performance as the unperturbed optimal one provided the estimated ASV is close enough to the true data vector. However, responsive methods can have imprecise estimates under harsh conditions which leads to poor performance.

4.5 Robust Adaptive Method 1

4.5.1 Phase Errors in ASV

In general, environment perturbations and array calibration errors always cause two kinds of ASV errors: amplitude errors and time-delay errors [12]. In nar-

rowband beamforming, time-delay corresponds to phase shift. Therefore, all the perturbations that lead to delay errors can be considered as phase errors. In this section, we proposed a robust method that combats phase errors in the Capon beamformer.

4.5.2 Problem Formulation

It is found that the output power of the Capon beamformer achieves a local maximum when the ASV coincides with that of the true response vector of the desired signal [80]. Based on this fact, we estimate the true phase vector $\boldsymbol{\phi} = [\phi_1, \dots, \phi_N]^T$ of ASV by finding the maximum of output power. It is note that when the desired signal is not the dominant signal, this output maximization method may yield a wrong solution. In order to avoid a wrong estimation, a feasible region of the phase vector has to be constrained. Based on the prior information of the probability density function (PDF) of the phase vector, we propose a new method on searching the phase vector with probability larger than a threshold. This robust method is expressed as:

$$\max_{\boldsymbol{\phi}} \quad \frac{1}{\hat{\mathbf{a}}_0^H \mathbf{R}^{-1} \hat{\mathbf{a}}_0} \quad \text{subject to} \quad p(\boldsymbol{\phi}) \geq \xi, \quad (4.29)$$

where $p(\boldsymbol{\phi})$ is the PDF of phase vector. Here, we assume that phase errors follow the errors model described in Section 3.2.1. The PDF of the phase vector $p(\boldsymbol{\phi})$ is given by

$$p(\boldsymbol{\phi}) = \frac{1}{\sqrt{(2\pi)^N \det(\boldsymbol{\Phi})}} \exp\left(-\frac{1}{2}(\boldsymbol{\phi} - \boldsymbol{\phi}_0)^T \boldsymbol{\Phi}^{-1}(\boldsymbol{\phi} - \boldsymbol{\phi}_0)\right), \quad (4.30)$$

where ϕ_0 represents the original phase vector in $\bar{\mathbf{a}}_0$, and the m th element of $\phi_{0,m}$ is given by

$$\phi_{0,m} = \mathbf{k}_0^T \mathbf{r}_m^0. \quad (4.31)$$

Here, the phase error is also composed of the error caused by steering error and position errors in the array. In this thesis, we assume the steering direction error $\Delta\theta$ is a zero mean Gaussian variable with standard deviation σ_θ . The position errors ($\Delta\mathbf{r}$) in array sensors are assumed to be jointly Gaussian variables with zero mean and correlation matrix $E\{\Delta\mathbf{r}\Delta\mathbf{r}^T\} = \mathbf{Q}$. Both of the two errors correspond to Gaussian phase errors (refer to Appendix D.4 and D.5). Since the linear combination of a set of Gaussian distributions is still a Gaussian distribution, the correlation matrix of generalized phase error Φ in (3.8) is a summation of the correlation matrices of three different types errors

$$\Phi = \Phi_\theta + \Phi_{pos} + \Phi_{phase}, \quad (4.32)$$

where Φ_θ , Φ_{pos} are the correlation matrices of the corresponding phase errors caused by steering error and position errors, respectively. Φ_{phase} is the correlation matrix by sensors random phase errors itself.

Taking natural logarithm of the constraint, Eq. (4.30) is equivalent to

$$\min_{\phi} \quad \hat{\mathbf{a}}_0^H \mathbf{R}^{-1} \hat{\mathbf{a}}_0 \quad \text{subject to} \quad (\phi - \phi_0)^T \Phi^{-1} (\phi - \phi_0) \leq \xi'. \quad (4.33)$$

This problem can be solved by using the Lagrange multiplier method, which is based on the objective function

$$J(\phi) = \hat{\mathbf{a}}_0^H \mathbf{R}^{-1} \hat{\mathbf{a}}_0 + \gamma [(\phi - \phi_0)^T \Phi^{-1} (\phi - \phi_0) - \xi']. \quad (4.34)$$

Differentiating respect to ϕ (refer to Appendix D.6)

$$\frac{\partial J(\phi)}{\partial \phi} = 2\text{Im}\{\hat{\mathbf{a}}_0^* \otimes (\mathbf{R}^{-1}\hat{\mathbf{a}}_0)\} + 2\gamma\Phi^{-1}(\phi - \phi_0), \quad (4.35)$$

where $*$ denotes complex conjugate, and the $\otimes$ represents Hadamard Product. It is found that $\text{Im}\{\hat{\mathbf{a}}_0^* \otimes (\mathbf{R}^{-1}\hat{\mathbf{a}}_0)\}$ is still a function of ϕ , and Eq. (4.35) can not be written as an explicit form of ϕ . Therefore, the closed form of the optimum phase vector can not be obtained. So we use an iterative method to implement updating the value of ϕ .

4.5.3 Implementation of Robust Method

We update the value of phase vector ϕ by the steepest descent method, i.e.,

$$\phi(k+1) = \phi(k) - \mu \frac{\partial J(\phi)}{\partial \phi} \Big|_{\phi=\phi(k)}, \quad (4.36)$$

where k denotes the k th iteration, and μ is a small, positive, step-size parameter which controls the rate of change of ϕ . In order to perform the iterative algorithm, some parameters such as log likelihood ξ' in (4.34) and γ in (4.35) has to be determined first. In the following parts, we introduce the criteria and method to choose these parameters.

choosing ξ'

The constraint $(\phi - \phi_0)^T \Phi^{-1}(\phi - \phi_0) \leq \xi'$ confines the phase vector in an ellipsoid centered at ϕ_0 . If the value of ξ' is small, the ellipsoid is small which only covers

the region that the phase errors frequently occur. As the value of ξ' increases, the confining ellipsoid expands. In Appendix D.7, it is derived that $\zeta = (\phi - \phi_0)^T \Phi^{-1}(\phi - \phi_0)$ follows a chi-square distribution. Its PDF is given by:

$$p(\zeta) = \frac{1}{2^{\frac{N}{2}} \Gamma(\frac{N}{2})} \zeta^{\frac{N}{2}-1} e^{-\frac{\zeta}{2}} \quad \zeta \geq 0. \quad (4.37)$$

From the PDF, the probability that the phase vector ϕ lies within the region $(\phi - \phi_0)^T \Phi^{-1}(\phi - \phi_0) \leq \xi'$ can be calculated as

$$\begin{aligned} Pr(\zeta \leq \xi') &= \int_0^{\xi'} p(\zeta) d\zeta = \int_0^{\xi'} \frac{1}{2^{\frac{N}{2}} \Gamma(\frac{N}{2})} \zeta^{\frac{N}{2}-1} e^{-\frac{\zeta}{2}} d\zeta \\ &= \int_0^{\frac{\xi'}{2}} \frac{1}{\Gamma(\frac{N}{2})} t^{\frac{N}{2}-1} e^{-t} dt. \end{aligned} \quad (4.38)$$

Fig. 4.7 shows the probability, $Pr(\zeta \leq \xi')$, when the dimension of the phase

Figure 4.7: $Pr(\zeta \leq \xi')$

vector is nine. From the figure, if the probability of the phase vector error is to be no more than 10%, then ξ' should be set less than 4.2.

Updating γ

When the phase vector is updating, the coefficient γ has to be changed to control the relative importance of the output power and the penalty function $(\phi(k) - \phi_0)^T \Phi^{-1}(\phi(k) - \phi_0)$.

For example, at the k th iteration, if $(\phi(k) - \phi_0)^T \Phi^{-1}(\phi(k) - \phi_0) \leq \xi'$, then $\gamma(k) = 0$, else, $\gamma(k) \neq 0$. Some researchers proposed different ways to update this penalty coefficient [81–83]. Here, we use a variable amount of $\gamma(k)$ like the method proposed by Kooij [81]. $\gamma(k+1)$ is updated by the following equation

$$\gamma(k+1) = (1 - \eta)\gamma(k) + \Delta\gamma [1 + \text{sign}((\phi - \phi_0)^T \Phi^{-1}(\phi - \phi_0) - \xi')], \quad (4.39)$$

where sign function is given by

$$\text{sign}(x) = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases}. \quad (4.40)$$

The parameter $\Delta\gamma$ and η are real positive constants with values smaller than or equal to unity. The variable $\gamma(k)$ increases when the value of $(\phi - \phi_0)^T \Phi^{-1}(\phi - \phi_0)$ is larger than ξ' , and decreases when the value is smaller than ξ' . Here, $\Delta\gamma$ controls the increment size and $(1 - \eta)$ is the relaxation parameter.

End Condition

To terminate the iterative procedure, an end condition has to be set. We choose the parameter ϵ , such that if $\|\phi(k+1) - \phi(k)\| \leq \epsilon$, then the iterative updating

procedure stops.

Iterative Updating Steps

In summary, the iterative updating procedure of the proposed method is as follows:

- **Step 1** Set parameters $\mu, \Delta\gamma, \epsilon, \eta, \xi'$;
- **Step 2** Initialize $\gamma(1) = 0$ and $\phi(1) = \phi_0$ by (4.31) ;
- **Step 3** Calculate $\frac{\partial J(\phi)}{\partial \phi}|_{\phi=\phi(k)}$ by (4.35);
- **Step 4** Update phase vector by $\phi(k+1) = \phi(k) - \mu \frac{\partial J(\phi)}{\partial \phi}|_{\phi=\phi(k)}$;
- **Step 5** Update $\gamma(k+1)$ by (4.39);
- **Step 6** Check if $\|\phi(k+1) - \phi(k)\| \leq \epsilon$

No, repeat Step 3-6 .

Yes, go Step 7.

- **Step 7** Update the ASV vector $\hat{\mathbf{a}}_0$ with the the latest phase vector($\phi(k+1)$)
by

$$\hat{\mathbf{a}}_0 = \exp(j\phi(k+1)). \quad (4.41)$$

4.5.4 Numerical Results

In this section, some numerical experiments are carried out to illustrate the effectiveness of the proposed robust method. In the first example, a nine-element stan-

standard linear array is used. The power of the desired signal is 10 dB and its assumed DOA is 0° . A strong interference with 20 dB power is assumed to be coming from 34° , and the background white noise power is assumed to be 0 dB. For the purpose of performance comparison, the standard Capon beamformer (SCB), multi-point robust beamformer (MPRBF) and diagonal loading robust beamformer (DLRBF) are also included in the simulations. The proposed robust method is referred to as the robust adaptive beamformer 1 (RAB1).

In all the examples, we assume phase errors are independent and identically zero mean Gaussian variables with standard deviation $\sigma_\phi = 10^\circ$. There also exists steering direction error $\Delta\theta$ which follows zero mean Gaussian distribution with standard deviation $\sigma_\theta = 3^\circ$. In numerical examples, we averaged performance measures (e.g., mean SINR) are obtained by averaging 200 Monte-Carlo trials. In each trial, the phase error is assumed not changed throughout the whole adaptive duration.

Using the theoretical covariance matrix

First, we assume the theoretical covariance matrix ($\mathbf{R}$) is available. The weights of the MPRBF is calculated by (4.26), where the constraint matrix $\mathbf{C}$ consists of array steering vector in DOA range -3° to 3° with 1° sampling. For DLRBF, we set the loading factor as N times of noise power, which is 9 in the first example. The parameters used in our proposed method are $\mu = 0.08$, $\Delta\gamma = 0.02$, $\eta = 0.1$, $\epsilon = 10^{-3}$. The value of ξ' is 12 which confines about 80% of the phase errors to be in

the ellipsoid region. In Fig. 4.8, the beampatterns of the SCB, MPRBF, DLRBF and RAB1 of one trial are shown. It is found that although the SCB still can null out the interference, it cancels the target signal as well. The RAB1 is superior than other two robust methods (MPRBF and DLRBF), because it not only has a distortionless response in the target direction and a deep null in the interference direction, it has much lower sidelobes as well. Fig. 4.9 shows the update of the value of stop condition $\|\phi(k+1) - \phi(k)\|$ of the RAB1. It meets the stop criterion after 47 iterations in this trial. In Fig. 4.10, the histogram of the output SINRs of the robust beamformer in 200 Monte-Carlo trials is given. It is observed that most of the output SINRs (about 80%) lie in a region [18 - 20] dB, which is corresponding to the probability constraint we set. The mean iteration number of the 200 Monte-Carlo trials is about 46.

Fig. 4.11 shows the estimated power of the four methods. It shows that the proposed method gives estimates with higher accuracy than the other three method do. Fig. 4.12 shows the mean SINR of three beamformers versus phase deviation ranging from 0° to 20° in 200 Monte-Carlo trials. It is note that in this simulation no steering direction error is assumed. The mean SINR of the SCB decreases rapidly when the phase deviation gets larger, while the RAB1 can keep high output SINR in all perturbation situations.

Next, we investigate the performance of the proposed algorithm in low SNR case. Fig. 4.13 shows the histogram of the output SINRs in 200 Monte-Carlo trials when SNR is -5 dB. It can be seen that there are about 80% SINRs of proposed

Figure 4.8: Beampattern of the beamformers (SCB, MPRBF, DLRBF and RAB1)

Figure 4.9: Value of $\|\phi(k+1) - \phi(k)\|$ vs iteration number

Figure 4.10: Histogram of the output SINR in RAB1

Figure 4.11: Estimated power of beamformers (SCB, MPRBF, DLRBF and RAB1)

Figure 4.12: Mean SINR of the beamformers vs phase deviations
(SCB, MPRBF, DLRBF and RAB1)

Figure 4.13: Histogram of the output SINR in RAB1 (SNR = -5 dB)

beamformer larger than 4 dB. In this case, the mean output SINR of the SCB is -1.4 dB, which reveals that the improvement of performance is not as great as high SNR case. This is understandable, because in low SNR situation, the signal nulling performance is not significant. Even the ideal Capon beamformer (the ASV is accurate) offers an output SINR as 4.5 dB, which illustrates the effectiveness of our proposed algorithm.

Using estimated covariance matrix

Here, we assume the theoretical covariance matrix is unavailable. The covariance matrix ($\hat{\mathbf{R}}$) is obtained by averaging the snapshots of the received data of the array. Here, the covariance matrix is calculated by 100 snapshots. Other environment parameters are the same as that of Fig. 4.8.

In Fig. 4.14, we compare beampatterns of the SCB, MPRBF, DLRBF and RAB1 in one trial. It is observed that the SCB has a serious signal cancellation effect, while the RAB1 can keep an uniform response around the signal direction and has a deeper null in the interference direction than DLRBF and MPRBF. In Fig. 4.15, SINRs of the four beamformers versus the number of snapshots are plotted. We note that the RAB1 has lower SINR output than DLRBF when the number of snapshots is small, but RAB1 converges to a much higher SINR as the number of snapshots gets larger.

Fig. 4.16 shows the mean SINR of the three beamformers versus phase deviation ranging from 0° to 20° in 200 Monte-Carlo trials. Here, no steering direction

Figure 4.14: Beam pattern of the beamformers using the estimated covariance matrix (SCB, MPRBF, DLRBF and RAB1) (SNR = 10dB)

error is assumed. It is found that the DLRBF has higher SINR output than RAB1 when the phase error is small, and the mean SINR of the RAB1 is almost constant in all perturbation situations.

In Fig. 4.15, SINR of the three beamformers versus the number of snapshots is plotted. The RAB1 has better performance when the number of snapshots gets larger.

4.5.5 Circular Array

In order to testify the proposed method in the array of other configurations, we study the performance of a circular array. The array geometry used in the example is

Figure 4.15: SINR comparison of the beamformers vs snapshots number
(SCB, MPRBF, DLRBF and RAB1)

shown in the Fig. 4.17. Ten sensors are distributed evenly at the circle whose radius is the wavelength of the interested signal. One desired signal of 10 dB comes from 0° , and an interference of 20dB comes form 40° . The background white noise power is 0 dB. Fig. 4.18 shows the beampatterns of SCB, RAB1, MPRBF and DLRBF. In the simultions, the estimated covariance matrix being used is averaged by 200 snapshots. Fig. 4.19 shows SINRs of the four beamformers versus the number of snapshots. We note that the RAB1 has lower SINR output than DLRBF when the number of snapshots is small, but RAB1 converges to a much higher SINR as the number of snapshots gets larger.

Figure 4.16: Mean SINR of the beamformers vs phase deviations by estimated covariance matrix (SCB, MPRBF, DLRBF and RAB1)

Figure 4.17: Circular Array

Figure 4.18: Beampattern of the beamformers using the estimated covariance matrix in a circular array (SCB, MPRBF, DLRBF and RAB1)

Figure 4.19: SINR comparison of the beamformers vs snapshots number in a circular array (SCB, MPRBF, DLRBF and RAB1)

4.5.6 Conclusions

A robust method which mitigates phase errors is proposed. The robustness is achieved by estimating the true phase vector of the desired signal. Numerical results demonstrate that the proposed beamformer can obtain a much higher SINR than the standard Capon beamformer even when the number of snapshots is small and the SNR is low. Moreover, the proposed beamformer does not suffer from performance loss in interference rejection.

4.6 Robust Adaptive Method 2

In Section 4.4, we introduced a family of robust Capon beamformers (RCBs) [34–38]. This kind of beamformer mitigates arbitrary environment perturbations and array imperfections by constraining the steering vector to be within an uncertainty region. However, the choice of the uncertainty region is crucial to the performance of RCBs. Whenever the uncertainty region is overestimated, it will give rise to a loss in SINR due to the impaired capability of the interference suppression [84,85].

In this section, we propose a robust method which can effectively reduce the uncertainty region by estimating the direction of the desired signal before imposing diagonal loading. In contrast to the proposed robust method 1, this method is based on the traditional RCBs and it can handle amplitude errors as well. The section begins with a review of existing RCBs.

4.6.1 Robust Capon Beamformers

Compared with the standard Capon beamformer, robust Capon beamformers (RCBs) offer robustness by confining the estimated ASV, $\hat{\mathbf{a}}_0$, to an uncertainty region. This can be formulated as:

$$\min_{\mathbf{w}, \hat{\mathbf{a}}_0} \quad \mathbf{w}^H \mathbf{R} \mathbf{w} \quad \text{subject to} \quad \mathbf{w}^H \hat{\mathbf{a}}_0 = 1 \quad \forall \hat{\mathbf{a}}_0 \in \eta, \quad (4.42)$$

where η is the set that contains the true data vector, $\mathbf{a}_0$. To avoid computation complexity [34,35,37,38], the uncertainty region is assumed to be a sphere centered at the nominal ASV $\bar{\mathbf{a}}_0$ with radius $\sqrt{\epsilon}$, that is:

$$\|\hat{\mathbf{a}}_0 - \bar{\mathbf{a}}_0\|^2 \leq \epsilon. \quad (4.43)$$

To solve the optimization problem in (4.42), initially we assume $\hat{\mathbf{a}}_0$ is known. Thus the output power is given by:

$$\hat{p}_{out} = \frac{1}{\hat{\mathbf{a}}_0^H \mathbf{R}^{-1} \hat{\mathbf{a}}_0}. \quad (4.44)$$

The output power of the Capon beamformer achieves a maximum when the estimated steering vector, $\hat{\mathbf{a}}_0$, coincides with that of the true ASV, $\mathbf{a}_0$. Thus, the above optimization problem can be reduced to estimating the true ASV by finding the steering vector that corresponds to the maximum output power, which is equivalent to

$$\min_{\hat{\mathbf{a}}_0} \quad \hat{\mathbf{a}}_0^H \mathbf{R}^{-1} \hat{\mathbf{a}}_0 \quad \text{subject to} \quad \|\hat{\mathbf{a}}_0 - \bar{\mathbf{a}}\|^2 \leq \epsilon. \quad (4.45)$$

By using the Lagrange multiplier method, the optimal solution, $\hat{\mathbf{a}}_0$, can be calcu-

lated:

$$\hat{\mathbf{a}}_0 = \left(\frac{1}{\lambda}\mathbf{R}^{-1} + \mathbf{I}\right)^{-1}\bar{\mathbf{a}}_0 \quad (4.46)$$

$$= \bar{\mathbf{a}}_0 - (\mathbf{I} + \lambda\mathbf{R})^{-1}\bar{\mathbf{a}}_0, \quad (4.47)$$

where (4.47) was obtained from (4.46) by the matrix inverse lemma. The Lagrange multiplier λ can be evaluated by putting (4.47) into the constraint equation:

$$\|(\mathbf{I} + \lambda\mathbf{R})^{-1}\bar{\mathbf{a}}_0\|^2 = \epsilon. \quad (4.48)$$

The weight vector, $\mathbf{w}_{RCB}$, of the RCB method is then obtained by putting the estimated ASV $\hat{\mathbf{a}}_0$ of (4.46) into (4.2):

$$\mathbf{w}_{RCB} = \frac{\mathbf{R}^{-1}\hat{\mathbf{a}}_0}{\hat{\mathbf{a}}_0^H \mathbf{R}^{-1} \hat{\mathbf{a}}_0} = \frac{(\mathbf{R} + \frac{\mathbf{I}}{\lambda})^{-1}\bar{\mathbf{a}}_0}{\bar{\mathbf{a}}_0^H (\mathbf{R} + \frac{\mathbf{I}}{\lambda})^{-1} \mathbf{R} (\mathbf{R} + \frac{\mathbf{I}}{\lambda})^{-1} \bar{\mathbf{a}}_0} \quad (4.49)$$

The optimal weight vector can be further represented as

$$\mathbf{w}_o = \beta(\mathbf{R} + \kappa\mathbf{I})^{-1}\bar{\mathbf{a}}_0, \quad (4.50)$$

where β is a normalization constant such that $\mathbf{w}^H \hat{\mathbf{a}}_0 = 1$. Observe that the optimal weights of the RCB has a diagonal loading type solution, where the diagonal loading coefficient κ is determined by the uncertainty region beforehand.

4.6.2 Overestimated Uncertainty Region in RCBs

In the RCBs, choosing the uncertainty region is a difficult problem. In order to avoid computational complexity, the RCBs are sometimes implemented using a spherical uncertainty region. In this case, the RCB method is equivalent to

the diagonal loading approach, except the diagonal loading factor is determined by the size of the uncertainty region. Therefore, the choice of the uncertainty region is crucial to the performance of RCBs. Whenever the uncertainty region is overestimated, it impairs the capability of interference rejection and gives rise to a loss of SINR. This is because the estimated ASV of RCBs is assumed to be on the boundary of the uncertainty region.

4.6.3 Analysis of the Uncertainty Region

Comparison of the direction errors and random phase errors

In most situations, the steering direction error and random calibration errors (gain, phase, and position errors) are always present in the array simultaneously. In the narrowband case, the steering direction error which gives rise to time delay errors can be regarded as phase errors. In this section, a comparison is made between the steering direction error and phase errors that occur in each sensor independently and randomly.

For a ULA, if the look direction is θ_0 , and the steering direction error is $\Delta\theta$, then the true data vector, $\mathbf{a}_0$, is given by

$$\mathbf{a}_0 = [1 \quad e^{-jkd\sin(\theta_0+\Delta\theta)} \quad \dots \quad e^{-j(N-1)kd\sin(\theta_0+\Delta\theta)}]^T. \quad (4.51)$$

The corresponding phase error in the n th element ($\Delta\phi_n$) is :

$$\begin{aligned}\Delta\phi_n &= (n-1)kd \sin(\theta_0 + \Delta\theta) - (n-1)kd \sin(\theta_0) \\ &= \frac{2\pi(n-1)d}{\lambda} [\sin(\theta_0 + \Delta\theta) - \sin(\theta_0)].\end{aligned}\quad (4.52)$$

Writing the phase error ($\Delta\phi^\circ$) in degrees:

$$\Delta\phi_n^\circ = \frac{180^\circ \Delta\phi_n}{\pi} = \frac{360^\circ(n-1)d}{\lambda} [\sin(\theta_0 + \Delta\theta) - \sin(\theta_0)] \quad (4.53)$$

This shows that the corresponding phase error caused by a steering direction error increases linearly with respect to the index of the sensor. Therefore, a steering direction error corresponds to large phase error in the sensor at the end of the array. In Fig. 4.20, we compare the steering direction and random phase errors by the simulation of a nine-element standard array. The steering direction error $\Delta\theta = 2^\circ$ which gives rise to the corresponding phase error increase of $\Delta\phi^\circ = 6.3^\circ$ is shown in solid. The independent random phase error, which is Gaussian distributed with $\sigma_{\Delta\phi^\circ} = 6.3^\circ$, is shown as a dash line for comparison. It is found that the phase errors caused by the steering error is much higher than the random phase errors, especially in sensor at the end of the array. It can be imagined that a large uncertainty region is obtained when the direction error is large or the number of array sensors is large.

Uncertainty region caused by steering and random phase errors

Next, we compare the uncertainty regions caused by steering direction error and random phase errors respectively. Eq.(4.43) is used to evaluate the radius of the

Figure 4.20: Phase error versus sensor number

uncertainty region caused by directional mismatch and random phase errors. In Fig. 4.21, the steering direction error $\Delta\theta$ changes from 0° to 6° , while the standard deviation of phase errors $\sigma_{\Delta\phi^\circ}$ changes from 0° to 6° . From the figure, it is found that the radius of the sphere caused by directional mismatch is much larger than that of random phase errors. For a direction error of $\Delta\theta = 6^\circ$, the volume of the sphere is almost 64 times of the one for the random phase error of $\sigma_{\Delta\phi^\circ} = 6^\circ$. Therefore, although the steering direction error and random calibration errors contribute to the uncertainty region, the steering direction error always dominates the uncertainty region. Fig. 4.22 shows the SINR of the RCBs versus two types of errors. We can observe that the SINR of the RCBs with direction error is much lower than that with random phase errors.

Figure 4.21: Radius of sphere versus errors

(a) steering direction error $\Delta\theta$ (b) random phase error $\sigma_{\Delta\phi}$

Figure 4.22: SINR of RCBs

(a) steering direction error $\Delta\theta$ (b) random phase error $\sigma_{\Delta\phi}$

Illustration of the uncertainty regions

In general, the true data vector is a random quantity whose fluctuations are a result of a steering direction mismatch and random calibration errors. The uncertainty region of the true data vector can be represented geometrically as a sphere centered at the nominal ASV $\bar{\mathbf{a}}(\theta_0)$. A qualitative comparison of the uncertainty region of the RCB with steering direction mismatch (large sphere in figure) and without (small sphere in figure) is shown in Fig. 4.23. For visualization, the ASV is shown in a three-dimensional space. From the numerical results of the previous section, it is clear that the uncertainty region due to random phase errors is much smaller than the case with steering direction mismatch. In addition, the difference between the two uncertainty regions is more significant when the direction error is large. Therefore, if the steering mismatch can be eliminated or reduced, a smaller uncertainty region will result in a less excessive diagonal loading for the RCB.

4.6.4 Implementation of Robust Method

The previous section showed that the steering direction error dominates the uncertainty region and leads to a significant loss of SINR especially when the direction error is large. The method proposed in this thesis seeks to reduce the uncertainty region by estimating the true DOA. The RCB method can then be applied to the reduced uncertainty region, which results in a much lower diagonal loading level. Consequently, the proposed method can result in a higher SINR than obtainable

Figure 4.23: Qualitative comparison of the uncertainty regions

from RCBs, especially in the large direction error or large array cases.

Direction estimation

Based on the assumption that the true data vector corresponds to the maximum output power, an estimate of the DOA is obtained by estimating the minimum value of the objective function in Eq. (4.45). In contrast to the RAB1 in Section 4.5, in this case we only need to estimate one parameter. Thus, a simple and intuitive way is by grid search. Firstly, a tolerance range of the DOA is assumed, e.g., from -5° to 5° (nominal direction is 0°). This range is then subdivided into equally spaced intervals. The interval spacing is determined by the desired precision of the estimated direction error. For example, if the acceptable estimated

direction error is 0.5° , a grid of 20 is set. Next, the objective function is evaluated and compared at each grid point, and the one with the global minimum is chosen to be the estimated direction. After obtaining the estimated DOA, $\hat{\theta}$, the corresponding ASV, $\hat{\mathbf{a}}(\hat{\theta})$, can be calculated.

Re-estimation of the uncertainty region

After estimating the direction of the desired signal, the corresponding ASV, $\hat{\mathbf{a}}(\hat{\theta})$, is assumed to be the center of the confined sphere. The uncertainty region (ϵ in Fig. 4.23) is largely reduced to only cover any residual steering direction estimation error and random calibration errors. Note that the uncertainty region (ϵ') depends on a number of factors such as the sensor characteristics and the operating environment of the array. However, it should be comparably smaller than the one without DOA estimation. Therefore, the performance of the robust processor will be less sensitive to the choice of the reduced uncertainty region.

From (4.46), (4.48) and (4.49), the weights of proposed robust adaptive beamformer (RAB2) are given by:

$$\mathbf{w}_o = \frac{(\mathbf{R} + \frac{\mathbf{I}}{\kappa'})^{-1} \hat{\mathbf{a}}}{\hat{\mathbf{a}}^H (\mathbf{R} + \frac{\mathbf{I}}{\kappa'})^{-1} \mathbf{R} (\mathbf{R} + \frac{\mathbf{I}}{\kappa'})^{-1} \hat{\mathbf{a}}}, \quad (4.54)$$

where the diagonal loading coefficient κ' is calculated by solving

$$\|(\mathbf{I} + \kappa' \mathbf{R})^{-1} \hat{\mathbf{a}}\|^2 = \epsilon'. \quad (4.55)$$

The proposed method is summarized as follows:

- **Step 1** Set the search grid interval according to the direction error tolerance;
- **Step 2** Estimate the true DOA, $\hat{\theta}$, using grid search;
- **Step 3** Calculate the estimated ASV $\hat{\mathbf{a}}(\hat{\theta})$;
- **Step 4** Set the spherical uncertainty region ϵ' ;
- **Step 5** Calculate the optimum weights of RAB2 by (4.54) and (4.55).

4.6.5 Numerical Examples

In this section, some numerical simulations are carried out to compare the performance of the RAB2 with the SCB and the traditional RCBs. A nine-element standard linear array is used in the simulations. The power of the desired signal is 10 dB and its assumed DOA is 0° . A strong interference with 20 dB power is assumed to be coming from 45° , and the background white noise power is assumed to be 0 dB.

In the first simulation, the direction error varies from -5° to 5° and the random phase and gain errors follow a zero mean Gaussian distribution with standard deviations $\sigma_{\Delta\phi^\circ} = 5^\circ$ and $\sigma_g = 0.05$, respectively. Here, the grid spacing is set to be 0.5° , and the estimated covariance matrix $\hat{\mathbf{R}}$ is evaluated by averaging 100 snapshots.

Figure 4.24 shows the estimated DOA $\hat{\theta}$ versus true direction error θ . It is found the the largest estimation error of DOA is 0.5° . Fig. 4.25 shows the

Figure 4.24: Estimated DOA $\hat{\theta}$ versus direction error $\Delta\theta$ in the RAB2

Figure 4.25: SINR comparison versus steering direction error $\Delta\theta$

(SCB, RCB and RAB2)

mean SINR (averaged over 200 Monte-Carlo trials) of different beamformers are compared as a function of steering direction error. The SINR of the RCB falls off with increasing direction error while the RAB2 remains fairly constant. This is because the RAB2 correctly estimates the direction of desired signal so that the initial steering direction error has negligible affect on its beamforming performance.

In the second simulation, the steering direction error is taken to be $\Delta\theta = 3^\circ$ and the random phase and gain errors are the same as the previous simulation. The beampatterns of the three beamformers are shown in Fig. 4.26. The proposed method RAB2 not only can keep a distortionless response in the target direction, but it also has a deeper null in the interference direction and lower sidelobes than those of RCB. Figure 4.27 shows the mean SINR (averaged over 200 Monte-Carlo trials) as a function of the snapshot number for the different beamformers. Note that the RAB2 has a higher SINR than either the RCB or SCB given a sufficiently large number of snapshots. In this simulation, the uncertainty region for the RCB corresponds to $\epsilon = 5.5$, while for the RAB2, $\epsilon' = 2$.

The mean estimation angle error (averaged over 200 Monte-Carlo trials) of the RAB2 when the input SNR changes from -10 dB to 20 dB is shown in Fig. 4.28 and its corresponding mean SINR is shown in Fig. 4.29. We observe that the estimation angle error is getting smaller when the SNR is getting larger, and the proposed method always has a higher SINR than traditional RCBs.

Figure 4.30 shows the mean SINR of the RCB and RAB2 as a function of the number of sensors. It is found the SINR of RCB decreases as the number of

Figure 4.26: Beampattern comparison of the SCB, RCB and RAB2

Figure 4.27: SINR comparison versus snapshots number

(SCB, RCB and RAB2)

Figure 4.28: Estimation angle error versus input SNR

Figure 4.29: SINR comparison versus input SNR

(SCB, RCB and RAB2)

Figure 4.30: SINR comparison versus number of sensors(SCB, RCB and RAB2)

sensors increase, which is the result of an increase in the uncertainty region. The SINR of RAB2, however, increases linearly with respect to the number of sensors.

In order to investigate how the choice of the uncertainty region affects the performance of the two types of robust methods, the mean SINR as a function of the size of the uncertainty regions of RCB (ϵ) and RAB2 (ϵ') is presented in figure 4.31. It is found that not only the maximum SINR obtained by the RCB is smaller than that of the RAB2, but also the output SINR is more sensitive to the size of the uncertainty region for the RCB case. In other words, the relative insensitivity of the RAB2 to the size of the uncertainty region implies a certain robustness against SINR loss.

Figure 4.31: SINR versus the size of the uncertainty region (ϵ , ϵ')
(SCB, RCB and RAB2)

4.6.6 Conclusion

Steering direction error and other calibration errors, such as gain and position errors, invariably contribute to less than ideal array processing performance in real-world applications. When a spherical uncertainty region is used to characterize the steering direction error, the RCB method results in an excessive amount of diagonal loading leading to a relatively large loss in the SINR. The proposed RAB2 method circumvents this problem by first compensating for the direction error, which effectively reduces the size of the uncertainty region, before applying diagonal loading to mitigate the effects of dynamic random errors. This leads to a better performance than would be obtained by using the RCB method alone. The

improvement is especially significant when the steering direction error is large.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, we analyzed the degraded performance of the data-independent and adaptive arrays in the proposed errors model. Several methods are proposed to make the two types of arrays robust in perturbed situations. The investigation is carried out in parallel for the data-independent array and the adaptive array.

5.1.1 Perturbation Model

In the statistical model of the perturbation errors proposed in this thesis, we assumed that the variations in gain, phase and position of each sensor are independent of each other, but the gain errors of different sensors are correlated (similarly for phase and position errors). In addition, the gain, phase and position errors are

assumed to be zero-mean Gaussian variables that are characterized by different correlation matrices.

A reasonable covariance matrix of the position errors is obtained by assuming the array is embedded in a random surface. By using the transfer function of a random surface model, a sequence of correlated position errors can be obtained from an input IID sequence. The correlation between position errors is controlled by the correlation length of the surface.

5.1.2 Data-independent Array

For a data-independent array, since its beam pattern is designed to be independent of the array data, we analyze its expected beam pattern in perturbed situations. It is found that the expected beam pattern is a combination of the error-free beam pattern and a perturbation term. Specifically, for the independent errors case, the perturbation term is uniform throughout the whole spatial region, and it is determined by the perturbation variance and the norm of the beamformer weights. When the data-independent array suffers from correlated errors, the expected beam pattern has a broader mainbeam but relatively lower sidelobes compared to the independent errors case. An expression for the variance of the perturbed beam pattern is also deduced. From that, the distribution of the perturbed beam patterns can be estimated.

By comparing four arrays: the delay-and-sum, Dolph-Chebyshev, Taylor-

Kaiser and supergain endfire arrays, we found that the delay-and-sum beamformer has the stablest performance. Although the Dolph-Chebyshev and Taylor-Kaiser arrays are more sensitive to the perturbations than the delay-and-sum array, the degraded performance is not significant. The supergain endfire array, however, is seriously impaired by the perturbations. With small perturbations, the endfire array behaves like an isotropic sensor.

In order to offer robustness to the data-independent array, two robust methods are proposed to combat arbitrary errors in the array characteristics. Assuming that perturbations follow the errors model we proposed, we incorporate this knowledge into the design of the robust procedure and obtain the optimum robust beamformers based on the mean-square-error criterion. Robust I method which minimizes the MSE between the ideal and perturbed responses, can be considered as the most insensitive array based on MSE criterion. Robust II method, which has a quiescent response specified beforehand, minimizes the MSE between the quiescent and perturbed responses. The numerical results illustrate that both robust algorithms can greatly reduce the MSE and keep a higher directivity compared with the non-robust beamformer. In addition, the robust II method can offer higher resolution and lower sidelobes performances when the perturbation is not very large, while the robust I method can offer the lowest MSE performance in the perturbed situations. Although the two proposed robust methods are both based on the MSE criterion, the choice of robust method depends on the system requirements and perturbed situations. For example, if the stability is the most

important performance measure, the robust I algorithm should be used. If a particular array synthesis is required to achieve certain parameter requirements, e.g., a Taylor-Kaiser array is chosen for lower sidelobes, or a supergain array is selected for high array gain, then robust II can be used to keep high directivity and small MSE in the perturbed case.

5.1.3 Adaptive Array

Adaptive beamformers which provide high SINR performance are susceptible to the perturbation of array characteristics and the ambient environment. Whenever this happens, adaptive beamformers may misinterpret the desired signal as interference and this leads to "signal nulling". This will give rise to a drastically reduced array gain and an underestimation of the power of the desired signal in a way which is much more serious than the data-independent array. The perturbed SINR is affected by the SINR of the ideal case and the mismatch angle between the ASV and true data vector. The degraded performance is more severe when the input SNR is large.

Due to the high sensitivity of adaptive beamformers to the array characteristics and environmental conditions, robust techniques have received much more attention for adaptive beamforming. Various methods were developed to render the adaptive array robust. These robust methods can be separated into two categories, based on different underlying ideas. One is constrained robust methods and the other is responsive robust methods. Constrained robust methods reduce sensi-

tivity by imposing additional constraints around the nominal ASV which consumes the degrees of freedom and results in widening and flattening the main beam. Constrained robust method works well under a wide range of scenarios, but sacrifices the capability of interference suppression relative to the optimal beamformer. The responsive robust methods attempt to "respond" to the current environment by learning the true data vector from the observations and then using the information as if it were known exactly. Techniques of this type estimate the data vector by signal subspace or by some other estimation procedure. Good data vector estimates can be obtained under environments with high SNR and a slowly fluctuating DOA. Under these conditions, the responsive robust methods have nearly as good performance as the optimal one in the ideal case. However, responsive methods can have imprecise estimates under harsh conditions which leads to poor performance.

A robust method (RAB1), which is one of the responsive methods, is proposed in this thesis. Robustness to phase errors is achieved by estimating the true phase vector of the desired signal. The proposed beamformer has higher robustness to array imperfections that cause time-delay error. Moreover, it does not suffer from performance loss in interference rejection, thus it can offer high performance comparable to that found in the ideal situation.

A new class of robust Capon beamformers (RCBs) has been developed recently. These algorithms assume the true data vector is confined within an uncertainty region and is estimated by finding the one that corresponds to the maximum output power of the beamformer. However, if the uncertainty region is overesti-

mated, it impairs the capability of interference rejection, and thus gives rise to a loss of SINR. We find that when the steering direction error and random calibration errors occur simultaneously, the steering direction error always dominates the uncertainty region. The loss of SINR is significant when the direction error and/or the array is large. Therefore, a robust method (RAB2) has been developed to reduce the uncertainty region by estimating the true DOA of the desired signal. Consequently, the proposed method can result in a much higher SINR than obtainable from RCBs, especially in the large direction error or large array cases.

The two robust methods, RAB1 and RAB2, aim to handle different types of errors. RAB1 can be chosen when the array suffers from delay errors. RAB2, which is based on RCBs, can be used to combat arbitrary errors. However, it has better performance when the direction error or the array is large.

5.2 Recommendations

The choice of the type of array depends on a number of factors including ambient environment, system requirements and prior information of the signals and so on. For adaptive arrays, not only the theoretical performance limits but also the finite-sample response of the beamformer has to be considered. In general, if the ambient environment has no strong directional interferences, a data-independent array is preferred. This is because the data-independent array is simple has a low computational burden and it is not so sensitive to the perturbations in the

environment or array characteristics. If there are strong directional interferences, an adaptive array can be used to suppress the interferences dynamically and offer a higher resolution performance. The use of an adaptive beamformer should be accompanied by certain robust algorithms in order to avoid "signal nulling", especially in the case of high SNR. The choice of the robust algorithms is determined by the surrounding environment and the prior knowledge of the errors. In environments with high SNR and slowly fluctuating DOA, the "responsive" robust methods can be used to obtain nearly as good performance as the optimal one in the ideal case. On the other hand, in a low SNR or less stable environment, the constrained robust methods are more suitable.

5.3 Future Work

Based on the work shown in this thesis, some parts are promising and can be further extended.

The basic idea of the RCB methods and proposed robust adaptive methods is to estimate the data vector in a constrained region. The choice of this constrained region is crucial. If the region is underestimated, it still leads to "signal nulling", and behaves like a non-robust beamformer. If the region is overestimated, it impairs the capability of interference suppression and gives rise to a loss of SINR. However, choosing an appropriate uncertainty region is a difficult problem. Generally, repeated trials are implemented to measure the data vector beforehand. Then

a minimum-volume ellipsoid which covers all the trial set is chosen as the uncertainty region. However, doing repeated trials in advance is not available for some applications. Furthermore, to construct a minimum-volume ellipsoid from a known trial set is not an easy task. Therefore, the spherical uncertainty region is always assumed and calculated by using the largest ASV mismatch. In practice, there are multiple array imperfections. Some types of imperfection may be known a priori based on previous experience or the information offered by array manufacturers. Therefore, modeling an uncertainty region accurately using the prior information on array imperfections will be important.

In this thesis, the proposed robust adaptive methods assumed that the errors that affect the array performance can be considered as an array steering vector error. However, in real-world applications, some perturbations and errors in the environment and array can not be represented as an ASV error. For some cases, e.g., wavefront distortions and local scattering, the desired signal can not be considered as a point source. Therefore, the desired signal can not be represented a data vector, but a signal subspace with a dimension larger than one. Furthermore, because of the finite size of the sample matrix, there exists a difference between the estimated and theoretical covariance matrices. This will also cause "signal nulling" even if the ASV is accurate. Moreover, multipaths or moving interferences are also factors that can degrad the performance of an adaptive array. Although many people have proposed robust methods to combat different errors in adaptive arrays, they dealt with one or another case separately. When different types of errors are

present simultaneously as they are in the real-world applications, the robust solutions is not just a simple combination of the different robust methods mitigating each type of error. It is worth studying and developing robust methods for this more complicated but general case.

Appendix A

Appendix of Chapter 3

A.1 Directivity of Standard Linear Array

From (2.37), the directivity of unperturbed beampattern, D_0 , for a linear array is given by

$$D_0 = \frac{B_0(\theta_0)}{\frac{1}{2} \int_0^\pi d\theta \sin \theta B_0(\theta)} = \frac{\mathbf{w}^H \bar{\mathbf{a}}(\theta_0) \bar{\mathbf{a}}^H(\theta_0) \mathbf{w}}{\frac{1}{2} \mathbf{w}^H \underbrace{\left(\int_0^\pi \bar{\mathbf{a}}(\theta) \bar{\mathbf{a}}^H(\theta) \sin \theta d\theta \right) \mathbf{w}}_{\mathbf{Q}_0}}. \quad (\text{A.1})$$

The (m, n) element of matrix $\mathbf{Q}_0$ equals

$$\begin{aligned} \mathbf{Q}_{0(m,n)} &= \int_0^\pi e^{-j \frac{2\pi m d \cos \theta}{\lambda}} e^{j \frac{2\pi n d \cos \theta}{\lambda}} \sin \theta d\theta \\ &= 2 \text{sinc} \left(\frac{2\pi(m-n)d}{\lambda} \right), \end{aligned} \quad (\text{A.2})$$

where $\text{sinc}(x) \equiv \frac{\sin(x)}{x}$. For a standard linear array, whose interspacing $d = \frac{\lambda}{2}$

$$\mathbf{Q}_{0(m,n)} = \begin{cases} 2 & m = n, \\ 0 & m \neq n. \end{cases} \quad (\text{A.3})$$

Then, the directivity of a standard linear array is given by

$$D_0 = \frac{\mathbf{w}^H \bar{\mathbf{a}}(\theta_0) \bar{\mathbf{a}}^H(\theta_0) \mathbf{w}}{\mathbf{w}^H \mathbf{w}} = \frac{B_0(\theta_0)}{\mathbf{w}^H \mathbf{w}}, \quad (\text{A.4})$$

A.2 Minimum Sensitivity Term in Independent Errors Case

From Eq. (3.26), the sensitivity term of the expected beampattern is $\frac{\mathbf{w}^H \mathbf{w}}{\mathbf{w}^H \bar{\mathbf{a}}(\theta_0) \bar{\mathbf{a}}^H(\theta_0) \mathbf{w}}$.

The value of the sensitivity term does not change when $\mathbf{w}$ multiplied by a scalar, thus the minimum value can be obtained by solving the optimization problem shown below

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathbf{w}^H \mathbf{w} \\ \text{subject to} \quad & \mathbf{w}^H \bar{\mathbf{a}}(\theta_0, \phi_0) = 1 \end{aligned} \quad (\text{A.5})$$

By using Lagrange multiplier method, the objective function is:

$$L(\lambda) = \mathbf{w}^H \mathbf{w} + \lambda(\mathbf{w}^H \bar{\mathbf{a}}(\theta_0, \phi_0) - 1) + \lambda^*(\bar{\mathbf{a}}(\theta_0, \phi_0)^H \mathbf{w} - 1). \quad (\text{A.6})$$

Differentiating respect to $\mathbf{w}^H$ and setting the result to zero :

$$\frac{\partial L(\lambda)}{\partial \mathbf{w}^H} = \mathbf{w} + \lambda \bar{\mathbf{a}}(\theta_0, \phi_0) = 0 \quad (\text{A.7})$$

The optimal weights, $\mathbf{w}_o$

$$\mathbf{w}_o = -\lambda_o \bar{\mathbf{a}}(\theta_0, \phi_0), \quad (\text{A.8})$$

substituting (A.8) into (A.5),

$$\lambda_o = -\frac{1}{\|\bar{\mathbf{a}}(\theta_0, \phi_0)\|^2}, \quad (\text{A.9})$$

so,

$$\mathbf{w}_o = \frac{\bar{\mathbf{a}}(\theta_0, \phi_0)}{\|\bar{\mathbf{a}}(\theta_0, \phi_0)\|^2} = \frac{1}{N} \bar{\mathbf{a}}(\theta_0, \phi_0), \quad (\text{A.10})$$

which is the same as the weights of a delay-and-sum array.

A.3 Directivity of Expected Beampattern

In a linear array, the directivity of the expected beampattern, $\bar{D}$, is given by

$$\bar{D} = \frac{\overline{B_u(\theta_0)}}{\frac{1}{2} \int_0^\pi d\theta \sin \theta \overline{B_u(\theta)}} \quad (\text{A.11})$$

Define $k_1 \equiv e^{-(\sigma_\phi^2 + k^2 \sigma_r^2)}$ and $k_2 \equiv 1 - e^{-(\sigma_\phi^2 + k^2 \sigma_r^2)} + \sigma_g^2$, Equation (3.26) becomes

$$\overline{B_u(\theta)} = k_1 U_0(\theta) + \frac{k_2}{D_0}. \quad (\text{A.12})$$

Substituting (A.12) into (A.11)

$$\bar{D} = \frac{k_1 U_0(\theta_0) + \frac{k_2}{D_0}}{\frac{1}{2} \int_0^\pi k_1 U_0(\theta) d\theta \sin \theta + \frac{k_2}{D_0}} \quad (\text{A.13})$$

Since $U_0(\theta) = \frac{B_0(\theta)}{B_0(\theta_0)}$ is the normalized beampattern, the term

$$\frac{1}{2} \int_0^\pi k_1 U_0(\theta) d\theta \sin \theta = \frac{k_1}{2} \int_0^\pi \frac{B_0(\theta)}{B_0(\theta_0)} d\theta \sin \theta = \frac{k_1}{D_0} \quad (\text{A.14})$$

(refer to Appendix A.1). Thus,

$$\overline{D} = \frac{k_1 + \frac{k_2}{D_0}}{\frac{k_1}{D_0} + \frac{k_2}{D_0}} = \frac{D_0 k_1 + k_2}{k_1 + k_2}. \quad (\text{A.15})$$

If the perturbations are small ($D_0 k_1 \gg k_2$), $\overline{D}$, is given by

$$\overline{D} \approx \frac{D_0 k_1}{k_1 + k_2}. \quad (\text{A.16})$$

A.4 Variance of Perturbed Beampatterns

Whenever the elements of an array suffer random perturbations in gain, phase and position, the true data vector of the signal, $\mathbf{a}(\theta, \phi)$, is given by

$$\mathbf{a} = \begin{bmatrix} g_1 e^{j\Delta\phi_1} e^{-j\mathbf{k}^T \mathbf{r}_1} & g_2 e^{j\Delta\phi_2} e^{-j\mathbf{k}^T \mathbf{r}_2} & \dots & g_N e^{j\Delta\phi_N} e^{-j\mathbf{k}^T \mathbf{r}_N} \end{bmatrix}^T. \quad (\text{A.17})$$

The weight vector of the array is given by

$$\mathbf{w} = \begin{bmatrix} u_1 e^{-j\mathbf{k}_0^T \mathbf{p}_1} & u_2 e^{-j\mathbf{k}_0^T \mathbf{p}_2} & \dots & u_N e^{-j\mathbf{k}_0^T \mathbf{p}_N} \end{bmatrix}^T. \quad (\text{A.18})$$

The beampattern of the array is given by

$$B(\theta, \phi) = |\mathbf{w}^H \mathbf{a}(\theta, \phi)|^2 = \sum_{m=1}^N \sum_{n=1}^N g_m g_n u_m u_n \cdot e^{j(\Delta\phi_m - \Delta\phi_n)} e^{j\mathbf{k}_0^T (\mathbf{p}_m - \mathbf{p}_n)} e^{-j\mathbf{k}^T (\mathbf{r}_m - \mathbf{r}_n)}. \quad (\text{A.19})$$

The square of the beampattern is given by

$$B^2(\theta, \phi) = \sum_{m=1}^N \sum_{n=1}^N \sum_{i=1}^N \sum_{h=1}^N g_m g_n g_i g_h u_m u_n u_i u_h e^{j\mathbf{k}_0^T (\mathbf{p}_m - \mathbf{p}_n + \mathbf{p}_i - \mathbf{p}_h)} \cdot e^{j(\Delta\phi_m - \Delta\phi_n + \Delta\phi_i - \Delta\phi_h)} e^{-j\mathbf{k}^T (\mathbf{r}_m - \mathbf{r}_n + \mathbf{r}_i - \mathbf{r}_h)}. \quad (\text{A.20})$$

It can be written as

$$B^2 = \sum_{m=1}^N \sum_{n=1}^N \sum_{i=1}^N \sum_{h=1}^N \underbrace{u_m u_n u_i u_h e^{j\mathbf{k}_0^T(\mathbf{p}_m - \mathbf{p}_n + \mathbf{p}_i - \mathbf{p}_h)} e^{-j\mathbf{k}^T(\mathbf{r}_m^0 - \mathbf{r}_n^0 + \mathbf{r}_i^0 - \mathbf{r}_h^0)}}_{\text{unperturbed term}} (1 + \Delta g_i)(1 + \Delta g_h) \cdot (1 + \Delta g_m)(1 + \Delta g_n) e^{j(\Delta\phi_m - \Delta\phi_n + \Delta\phi_i - \Delta\phi_h)} e^{-j\mathbf{k}^T(\Delta\mathbf{r}_m - \Delta\mathbf{r}_n + \Delta\mathbf{r}_i - \Delta\mathbf{r}_h)}. \quad (\text{A.21})$$

Define the unperturbed term as $p_{m,n,i,h}^0$, thus

$$\begin{aligned} \overline{B^2} = & \sum_{m=1}^N \sum_{n=1}^N \sum_{i=1}^N \sum_{h=1}^N p_{m,n,i,h}^0 \underbrace{E_{\text{error}} \{ (1 + \Delta g_m)(1 + \Delta g_n)(1 + \Delta g_i)(1 + \Delta g_h) \}}_{A_{mnih}} \\ & \cdot \underbrace{E_{\text{error}} \{ e^{j(\Delta\phi_m - \Delta\phi_n + \Delta\phi_i - \Delta\phi_h)} \}}_{B_{mnih}} \underbrace{E_{\text{error}} \{ e^{-j\mathbf{k}^T(\Delta\mathbf{r}_m - \Delta\mathbf{r}_n + \Delta\mathbf{r}_i - \Delta\mathbf{r}_h)} \}}_{C_{mnih}} \end{aligned} \quad (\text{A.22})$$

The expected term of gain error in the (m, n, i, h) th element is:

$$A_{mnih} = 1 + G_{mn} + G_{mi} + G_{mh} + G_{ni} + G_{nh} + G_{hi} + N_g, \quad (\text{A.23})$$

where $N_g = E_{\text{error}} \{ g_m g_n g_i g_h \}$, is the fourth-order moment of the gain error. For independent gain errors case, $N_g = 3\sigma_g^4$. The expected term of phase error in the m, n, i, h th element is:

$$B_{mnih} = e^{-\frac{1}{2}(\Phi_{mm} + \Phi_{nn} + \Phi_{ii} + \Phi_{hh} - 2\Phi_{mn} + 2\Phi_{mi} - 2\Phi_{mh} - 2\Phi_{ni} + 2\Phi_{nh} - 2\Phi_{ih})}. \quad (\text{A.24})$$

The expected term of position error in the m, n, i, h th element is:

$$C_{mnih} = e^{-\frac{k^2}{2}(\Omega_{mm} + \Omega_{nn} + \Omega_{ii} + \Omega_{hh} - 2\Omega_{mn} + 2\Omega_{mi} - 2\Omega_{mh} - 2\Omega_{ni} + 2\Omega_{nh} - 2\Omega_{ih})}. \quad (\text{A.25})$$

Thus the

$$\overline{B^2} = \sum_{m=1}^N \sum_{n=1}^N \sum_{i=1}^N \sum_{h=1}^N p_{m,n,i,h}^0 A_{m,n,i,h} B_{m,n,i,h} C_{m,n,i,h}, \quad (\text{A.26})$$

where $A_{m,n,i,h}$, $B_{m,n,i,h}$, $C_{m,n,i,h}$ are the values from (A.23), (A.24) and (A.25) respectively.

In section 3.2.2, the expected beampattern is:

$$\begin{aligned} \overline{B} = \sum_{m=1}^N \sum_{n=1}^N w_m^* \mathbf{P}_{m,n} w_n &= \sum_{m=1}^N \sum_{n=1}^N u_m u_n (1 + G_{m,n}) e^{j\mathbf{k}_0^T (\mathbf{p}_m - \mathbf{p}_n)} e^{-j\mathbf{k}^T (\mathbf{r}_m^0 - \mathbf{r}_n^0)} \\ &\cdot e^{-\frac{1}{2}(\Phi_{mm} - 2\Phi_{mn} + \Phi_{nn})} e^{-\frac{k^2}{2}(\Omega_{mm} - 2\Omega_{mn} + \Omega_{nn})}. \end{aligned} \quad (\text{A.27})$$

Thus

$$\begin{aligned} \overline{B}^2 &= \sum_{m=1}^N \sum_{n=1}^N \sum_{i=1}^N \sum_{h=1}^N p_{m,n,i,h}^0 (1 + G_{m,n})(1 + G_{i,h}) e^{-\frac{1}{2}(\Phi_{mm} - 2\Phi_{mn} + \Phi_{nn})} \\ &\cdot e^{-\frac{1}{2}(\Phi_{ii} - 2\Phi_{ih} + \Phi_{hh})} e^{-\frac{k^2}{2}(\Omega_{mm} - 2\Omega_{mn} + \Omega_{nn})} e^{-\frac{k^2}{2}(\Omega_{ii} - 2\Omega_{ih} + \Omega_{hh})}. \end{aligned} \quad (\text{A.28})$$

Substituting (A.28) and (A.26) into $\sigma_B^2 = \overline{B^2} - \overline{B}^2$, the variance of the perturbed patterns is given by:

$$\sigma_B^2 = \sum_{m=1}^N \sum_{n=1}^N \sum_{i=1}^N \sum_{h=1}^N p_{m,n,i,h}^0 (A_{m,n,i,h} B_{m,n,i,h} C_{m,n,i,h} - D_{m,n,i,h}) \quad (\text{A.29})$$

where

$$\begin{aligned} D_{m,n,i,h} &= (1 + G_{m,n})(1 + G_{i,h}) e^{-\frac{1}{2}(\Phi_{mm} - 2\Phi_{mn} + \Phi_{nn})} e^{-\frac{1}{2}(\Phi_{ii} - 2\Phi_{ih} + \Phi_{hh})} \\ &\cdot e^{-\frac{k^2}{2}(\Omega_{mm} - 2\Omega_{mn} + \Omega_{nn})} e^{-\frac{k^2}{2}(\Omega_{ii} - 2\Omega_{ih} + \Omega_{hh})}. \end{aligned} \quad (\text{A.30})$$

A.5 Covariance Matrix of Spherically Isotropic Noise Field

In Section 2.5.3, the (l, m) element of covariance matrix, $\mathbf{R}_n$, of a spherically isotropic noise is given by

$$\mathbf{R}_{n(l,m)} = \frac{\sigma^2}{4\pi} \int_0^{2\pi} d\phi \int_0^\pi d\theta \sin \theta e^{-j\mathbf{k}^T (\mathbf{r}_l - \mathbf{r}_m)}. \quad (\text{A.31})$$

Thus, for a uniform linear array, $\mathbf{R}_{n(l,m)}$ becomes

$$\mathbf{R}_{n(l,m)} = \frac{\sigma^2}{2} \int_0^\pi e^{-jk(l-m)d \cos \theta} \sin \theta d\theta, \quad (\text{A.32})$$

where d is the spacing between adjacent sensors. Evaluating the integral gives the following result for the (l, m) element of $\mathbf{R}_n$:

$$\mathbf{R}_{n(l,m)} = \sigma^2 \text{sinc}(kd(l-m)). \quad (\text{A.33})$$

A.6 Covariance Matrix of Cylindrically Isotropic Noise Field

If the uniform linear array is lying in the same plane as the wave vector directions of the cylindrically isotropic noise field, the (l, m) element of $\mathbf{R}_n$ from (2.28) is given by

$$\mathbf{R}_{n(l,m)} = \frac{\sigma^2}{2\pi} \int_0^{2\pi} e^{-jk(l-m)d \cos \phi} d\phi \quad (\text{A.34})$$

Evaluating the integral gives

$$\mathbf{R}_{n(l,m)} = \sigma^2 J_0(kd(l-m)), \quad (\text{A.35})$$

where $J_0(\cdot)$ is a Bessel function of the first kind of order zero [86].

Appendix B

Array Synthesis

B.1 Taylor-Kaiser Array

B.1.1 Kaiser Window

Kaiser proposed a weighting function which has found widespread usage in spectral analysis, FIR filter design, and other fields. The Kaiser weights are:

$$w_m = I_0 \left(\beta \sqrt{1 - \left(\frac{m}{M} \right)^2} \right), \quad (\text{B.1})$$

where $I_0(x)$ is the modified Bessel function of zero-order, and $m = \pm 1, \pm 2, \dots, \pm M$, or $m = 0, \pm 1, \pm 2, \dots, \pm M$, for even or odd number of array element, $N = 2M$ or $N = 2M + 1$. The parameter β specifies a beampattern trade-off between the peak height of the sidelobes and the beamwidth of the main lobe.

B.1.2 Taylor-Kaiser Weighting

Taylor was the first to use this window function in array problems and propose the relationship of β to the sidelobe level R_a (in absolute units), which is given by:

$$R_a = 4.60333 \frac{\sinh \beta}{\beta}. \quad (\text{B.2})$$

To avoid having to solve this equation for β for a given R_a , Kaiser and Schafer [87] have developed an empirical formula for β in terms of the sidelobe level in dB, $R = 20 \log_{10} R_a$, which is valid across the range $13 < R < 120$ dB:

$$\beta = \begin{cases} 0, & R < 13.26 \\ 0.76609(R-13.26)0.4 + 0.09834(R-13.26), & 13.26 < R < 60 \\ 0.12438(R+6.3) & 60 < R < 120 \end{cases} \quad (\text{B.3})$$

For example, a 9-element Taylor-Kaiser array with $R=30$ dB, β is calculated from (B.3) which equals to 4.0109. Substituting into (B.1), the designed weights steered to broadside are:

$$w_n = [1.0000, 3.7028, 7.2162, 10.2237, 11.4093, 10.2237, 7.2162, 3.7028, 1.0000]$$

B.2 Supergain Endfire Array

B.2.1 Visible Region

For uniform linear arrays, the array response is given by:

$$\begin{aligned} H(\theta) &= \sum_{n=1}^N w_n^* e^{-j \frac{2\pi(n-1)d \cos \theta}{\lambda}} \\ &= \sum_{n=1}^N u_n e^{-j \frac{2\pi(n-1)d(\cos \theta - \cos \theta_0)}{\lambda}}. \end{aligned} \quad (\text{B.4})$$

Define $\varphi = \frac{2\pi d(\cos \theta - \cos \theta_0)}{\lambda}$. The array response in φ -space is given by:

$$H(\varphi) = \sum_{n=1}^N u_n e^{-j(n-1)\varphi} = \sum_{n=1}^N u_n z^{n-1} = (z - z_1) \cdot (z - z_2) \cdots (z - z_{N-1}), \quad (\text{B.5})$$

where $z = e^{-j\varphi}$. Because the propagating direction of signals is in a region $0^\circ < \theta < 180^\circ$, φ is limited to the region $-\frac{2\pi d}{\lambda}(1 + \cos \theta_0) < \varphi < \frac{2\pi d}{\lambda}(1 - \cos \theta_0)$. This region is referred as the visible region. For example, a ULA steered to endfire with interspacing $d = \lambda/4$ has the visible region $-\pi < \varphi < 0$.

B.2.2 Schelkunoff's Weighting

Eq. (B.5) is a polynomial of degree $N - 1$ and therefore it has $N - 1$ zeros $(z_1, \dots, z_{N-1})$ which are equally spaced around the unit circle, i.e., $\varphi_i = \frac{2\pi i}{N}$, $i = 1, 2, \dots, N - 1$. Fig. B.1 A) shows the array zeros for a six-element ULA.

Schelkunoff's design idea was to place all $N - 1$ zeros of the array in the visible region by equally spacing them within it. Fig. B.1 B), C) shows endfire array zeros

Figure B.1: Endfire array zeros and visible region for $N = 6$, and $d = \lambda/4, d = \lambda/8$

and the visible region for when $d = \lambda/4$ and $d = \lambda/8$. The presence of more zeros in the visible region results in a narrower mainlobe and smaller sidelobes.

After deciding the placement of all zeros, the shading sequence u_n can be computed from (B.5).

B.3 Dolph-Chebyshev Array

B.3.1 Chebyshev Polynomials

Consider the case of an ULA steered to broadside. If the desired array response is a real symmetric function, the weights are real and symmetric (shown in Fig.B.2).

The array response is

$$H(\varphi) = a_0 + 2 \sum_{m=1}^{\frac{N-1}{2}} a_m \cos(m\varphi), \quad N \text{ is odd} \quad (\text{B.6})$$

Figure B.2: Array weights: (a) N odd; (b) N even

and

$$H(\varphi) = 2 \sum_{m=1}^{\frac{N}{2}} a_m \cos\left(\left(m - \frac{1}{2}\right)\varphi\right), \quad N \text{ is even} \quad (\text{B.7})$$

where $\varphi = \frac{2\pi d(\cos \theta)}{\lambda}$. Since

$$\cos\left(m\frac{\varphi}{2}\right) = \frac{e^{jm\varphi/2} + e^{-jm\varphi/2}}{2} = \frac{(\cos\frac{\varphi}{2} + j\sin\frac{\varphi}{2})^m + (\cos\frac{\varphi}{2} - j\sin\frac{\varphi}{2})^m}{2}, \quad (\text{B.8})$$

expanding (B.8) in a binomial series and taking the real parts gives

$$\begin{aligned} \cos\left(m\frac{\varphi}{2}\right) &= \cos^m\frac{\varphi}{2} - \frac{m(m-1)}{2!} \cos^{m-2}\frac{\varphi}{2} \sin^2\frac{\varphi}{2} \\ &+ \frac{m(m-1)(m-2)(m-3)}{4!} \cos^{m-4}\frac{\varphi}{2} \sin^4\frac{\varphi}{2} - \dots \end{aligned} \quad (\text{B.9})$$

Define $x = \cos(\frac{\varphi}{2})$, then (B.9) becomes

$$T_m(x) = \cos\left(m\frac{\varphi}{2}\right)\big|_{\cos(\frac{\varphi}{2})=x} = \begin{cases} 1 & m = 0 \\ x & m = 1 \\ 2x^2 - 1 & m = 2 \\ 4x^3 - 3x & m = 3 \end{cases}. \quad (\text{B.10})$$

The polynomials of (B.10) are Chebyshev polynomials. The m th-degree Chebyshev polynomial is defined as:

$$T_m(x) = \begin{cases} \cos(m \cos^{-1} x) & |x| \leq 1 \\ \cosh(m \cosh^{-1} x) & x > 1 \\ (-1)^m \cosh(m \cosh^{-1}(-x)) & x < -1 \end{cases} \quad (\text{B.11})$$

The Chebyshev polynomials $T_m(x)$ has alternating maximum and minima in the interval $-1 < x < 1$ that occur at $x_k = \cos(\frac{k\pi}{m})$, $k = 1, 2, \dots, m-1$. The magnitude of each maximum and minimum is unity, i.e., $T_m(x_k) = \pm 1$. $T_m(x)$ has m real roots in the interval $-1 < x < 1$ which occur at $m\frac{\varphi}{2} = (2p-1)\frac{\pi}{2}$, $p = 1, \dots, m$. Thus, the zeros are evenly spaced in φ space.

B.3.2 Dolph-Chebyshev Synthesis

Dolph [50] proposed a procedure which utilizes the above properties of the Chebyshev polynomials to develop an optimum pattern. In the Dolph-Chebyshev method, the mainlobe corresponds to the value of $T_m(x_0)$ where $x_0 > 1$ and the magnitude of sidelobes is unity (see Fig. B.3). The ratio of the mainlobe to sidelobe level R_a is:

$$T_m(x_0) = \cosh(m \cosh^{-1} x_0) = R_a. \quad (\text{B.12})$$

Thus, x_0 is given by

$$x_0 = \cosh(m \cosh^{-1} R_a). \quad (\text{B.13})$$

Figure B.3: $T_7(x)$

Change the scale of abscissa ($x' = x/x_0$) and let $x' = \cos \frac{\varphi}{2}$, then $x = x_0 \cos \frac{\varphi}{2}$.

Thus, the array response is

$$H(\varphi) = \frac{1}{R_a} T_{N-1}(x_0 \cos \frac{\varphi}{2}). \quad (\text{B.14})$$

where $\frac{1}{R_a}$ factor normalizes the response so that maximum of mainlobe is unity.

From (B.14), the $N - 1$ zeros of the response in φ -space are given by

$$\varphi_p = 2 \cos^{-1} \left(\frac{1}{x_0} \cos \left(\frac{(2p-1)\pi}{2(N-1)} \right) \right), \quad p = 1, \dots, N-1. \quad (\text{B.15})$$

After deciding the placement of all zeros, the shading sequence a_m can be computed

from (B.6) or (B.7).

B.3.3 Synthesis Procedure

The Dolph-Chebyshev synthesis procedure consists of five steps:

- **Step 1** For a N -sensor array, select $m = N - 1$ degree Chebyshev polynomial;
- **Step 2** Calculate x_0 for the specified R_a by (B.13);
- **Step 3** The desired array response is obtained by (B.14);
- **Step 4** Calculate the $N - 1$ zeros by (B.15);
- **Step 5** Calculate the array weights a_m by (B.6) or (B.7).

Appendix C

Perturbation Model for Correlated Position Errors

C.1 Model Assumptions

From observations, the position errors in each sensor of the array are not expected to be completely independent of one other. Here, we proposed a correlated position model with predefined correlation. At first, we assume the position errors in x,y and z axes are independent of each other, but the position errors of two sensors along the same axis (e.g. x_m and x_n) are correlated with respect to the distance between them. This correlated relationship is represented by the correlation length L .

C.2 Generation of Correlated Sequence from IID Sequence

Consider a linear system with an impulse response, $h(t)$. Its frequency domain transfer function is given by $H(f) = \int_{-\infty}^{\infty} h(t)e^{-j2\pi ft} dt$. For an input wave form $x(t)$, the corresponding output to the linear system is $y(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau$. If $x(t)$ is a stationary random process, then the following relations hold:

$$R_{yy}(\tau) = \int \int h(\zeta)h(\eta)R_{xx}(\tau + \zeta + \eta)d\zeta d\eta \quad (C.1)$$

$$S_{yy}(f) = |H(f)|^2 S_{xx}(f) \quad (C.2)$$

where $R_{xx}(\tau) = E \{x(t)x(t + \tau)\}$ and $S_{xx}(f)$ are the autocorrelation function and power spectral density of $x(t)$ respectively, and similarly, $R_{yy}(\tau)$ and $S_{yy}(f)$ are the autocorrelation function and power spectral density of $y(t)$ [88].

If $x(t)$ is a white random process, then $S_{xx}(f) = 1$ and $R_{xx}(\tau) = \delta(\tau)$, so that $S_{yy}(f) = |H(f)|^2$. The filter response in the frequency domain (modulo the phase) can be estimated as $H(f) = \sqrt{(S_{yy}(f))}$ and its corresponding impulse response is:

$$h(t) = \int_{-\infty}^{\infty} \sqrt{S_{yy}(f)} e^{j2\pi ft} df. \quad (C.3)$$

The above ideas can be applied to the case of one-dimensional random rough surfaces. Let the height of the random surface, $z(x)$, be generated by filtering a white process, i.e.,

$$z(x) = \int_{-\infty}^{\infty} w(\eta)v(x - \eta)d\eta, \quad (C.4)$$

where $w(x)$ is a linear spatial filter and $v(x)$ is a white random process, which is a function of position along the x direction. By using the argument given above, the filter is given by

$$w(x) = \int_{-\infty}^{\infty} \sqrt{S_{zz}(\varsigma)} e^{j2\pi\varsigma x} d\varsigma . \quad (\text{C.5})$$

where $S_{zz}(\varsigma)$ is the power spectral density of the height function and is related to the spatial correlation function by

$$R_{zz}(x) = \int_{-\infty}^{\infty} S_{zz}(\varsigma) e^{j2\pi\varsigma x} d\varsigma . \quad (\text{C.6})$$

The spatial correlation function or its power spectral density of the surface height can be measured from experiment. A useful empirical form of the correlation function is given by

$$R_{zz}(x) = \exp \left[-\left(\frac{x}{L} \right)^2 \right] . \quad (\text{C.7})$$

where L is the correlation length, which controls the rate of change of surface height with distance along the surface. By taking the Fourier transform of the correlation function (Weiner-Khinchin theorem), the power spectral density is given by

$$S_{zz}(f) = \sqrt{\pi} L \exp(-\pi^2 L^2 f^2) . \quad (\text{C.8})$$

This then results in the following form of the filter response:

$$w(x) = \left(\frac{2}{\sqrt{\pi} L} \right)^{\frac{1}{2}} \exp \left[-2 \left(\frac{x}{L} \right)^2 \right] . \quad (\text{C.9})$$

For a numerical implementation, the discrete version of the height function is

$$z_i = \sum_k w_k v_{i-k} . \quad (\text{C.10})$$

Figure C.1: Correlated Sequence Generated by IID Sequence ($L = 5d$)

One can show that the variances of z_i and v_i are related as follows: $\sigma_z^2 = E\{z_i^2\} = \sigma_v^2 \sum_k w_k^2$. It is computationally convenient to choose $\sigma_z^2 = \sigma_v^2$. This then requires $\sum_k w_k^2 = 1$. For the Gaussian correlation above, the discrete form is

$$w_i = w_0 \exp \left[-2 \left(\frac{i \Delta x}{L} \right)^2 \right], \quad (\text{C.11})$$

where w_0 is a normalization factor required to make $\sum_k w_k^2 = 1$.

So once the correlation function has been specified, the random surface heights can be generated using a realization of v_i , which can be obtained from a random number generator pertaining to a specific distribution, e.g., the Gaussian distribution. An example of a realization of a random surface in one dimension is shown in fig. C.1.

We implement three independent IID sequences into the above filters to generate correlated positions of x, y and z respectively. The positions of the sensors with correlated errors in three dimensions is shown in C.2.

Figure C.2: Positions of sensors with correlated errors of different correlation length

Appendix D

Appendix of Chapter 4

D.1 Comparison of Estimation Power

In the ideal case, the weight vector of a conventional beamformer which has uniform gain in the look direction is given by

$$\mathbf{w}_c = \frac{1}{N} \mathbf{a}_0. \quad (\text{D.1})$$

The output power of the conventional beamformer, P_c , is given by

$$P_c = \mathbf{w}_c^H \mathbf{R} \mathbf{w}_c = \frac{\mathbf{a}_0^H \mathbf{R} \mathbf{a}_0}{N^2}. \quad (\text{D.2})$$

From Eq. (4.3), the output power of the Capon beamformer is $P_{\text{capon}} = \frac{1}{\mathbf{a}_0^H \mathbf{R}^{-1} \mathbf{a}_0}$. If we define $\mathbf{R}^{\frac{1}{2}}$ as the square root of $\mathbf{R}$, and it satisfies $\mathbf{R}^{\frac{1}{2}} \mathbf{R}^{\frac{1}{2}} = \mathbf{R}$ and $\mathbf{R}^{\frac{1}{2}} \mathbf{R}^{-\frac{1}{2}} = \mathbf{I}$, then

$$\mathbf{a}_0^H \mathbf{a}_0 = (\mathbf{a}_0^H \mathbf{R}^{\frac{1}{2}})(\mathbf{R}^{-\frac{1}{2}} \mathbf{a}_0) = N. \quad (\text{D.3})$$

According to Cauchy-Schwarz' inequality theorem,

$$\left\| \mathbf{a}_0^H \mathbf{R}^{\frac{1}{2}} \right\| \left\| \mathbf{R}^{-\frac{1}{2}} \mathbf{a}_0 \right\| \geq \mathbf{a}_0^H \mathbf{R}^{\frac{1}{2}} \mathbf{R}^{\frac{1}{2}} \mathbf{a}_0 = N. \quad (\text{D.4})$$

The above equation can be further derived as

$$\begin{aligned} (\mathbf{a}_0^H \mathbf{R} \mathbf{a}_0)(\mathbf{a}_0^H \mathbf{R}^{-1} \mathbf{a}_0) &\geq N^2, \\ \frac{\mathbf{a}_0^H \mathbf{R} \mathbf{a}_0}{N^2} &\geq \frac{1}{\mathbf{a}_0^H \mathbf{R}^{-1} \mathbf{a}_0}, \end{aligned} \quad (\text{D.5})$$

which proves that $P_c \geq P_{\text{capon}}$.

D.2 SINR of Perturbed Cases

From Eq. (2.20), by using the matrix inverse lemma

$$\mathbf{R}^{-1} = (\sigma_0^2 \mathbf{a}_0 \mathbf{a}_0^H + \mathbf{R}_n)^{-1} = \mathbf{R}_n^{-1} - \frac{\sigma_0^2 \mathbf{R}_n^{-1} \mathbf{a}_0 \mathbf{a}_0^H \mathbf{R}_n^{-1}}{1 + \sigma_0^2 \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0}. \quad (\text{D.6})$$

From above equation, the denominator of the weight vector of the Capon beamformer from (4.2) is given by

$$\mathbf{a}_0^H \mathbf{R}^{-1} \mathbf{a}_0 = \frac{\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0}{1 + \sigma_0^2 \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0}, \quad (\text{D.7})$$

and the numerator of the weights is given by

$$\mathbf{R}^{-1} \mathbf{a}_0 = \frac{\mathbf{R}_n^{-1} \mathbf{a}_0}{1 + \sigma_0^2 \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0}. \quad (\text{D.8})$$

Thus, the optimal weights of Capon beamformer is given by

$$\mathbf{w}_o = \frac{\mathbf{R}^{-1} \mathbf{a}_0}{\mathbf{a}_0^H \mathbf{R}^{-1} \mathbf{a}_0} = \frac{\mathbf{R}_n^{-1} \mathbf{a}_0}{\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0}. \quad (\text{D.9})$$

From Eq. (D.9), the optimal SINR in the ideal case, $SINR_o$, can be calculated as

$$SINR_o = \frac{\sigma_0^2 |\mathbf{w}_o^H \mathbf{a}_0|^2}{\mathbf{w}_o^H \mathbf{R}_n \mathbf{w}_o} = \sigma_0^2 \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0. \quad (\text{D.10})$$

When there exist perturbations in the array, the data vector deviates from the nominal ASV, and the weight vector of the Capon beamformer is given by

$$\mathbf{w} = \frac{\mathbf{R}^{-1} \bar{\mathbf{a}}_0}{\bar{\mathbf{a}}_0^H \mathbf{R}^{-1} \bar{\mathbf{a}}_0}. \quad (\text{D.11})$$

Thus, the output SINR in the perturbed case is given by

$$\widetilde{SINR} = \frac{\sigma_0^2 |\mathbf{w}^H \mathbf{a}_0|^2}{\mathbf{w}^H \mathbf{R}_n \mathbf{w}} = \frac{\sigma_0^2 |\bar{\mathbf{a}}_0^H \mathbf{R}^{-1} \mathbf{a}_0|^2}{\bar{\mathbf{a}}_0^H \mathbf{R}^{-1} \mathbf{R}_n \mathbf{R}^{-1} \bar{\mathbf{a}}_0}. \quad (\text{D.12})$$

Substituting (D.6) into (D.12), and the numerator is calculated

$$\sigma_0^2 |\bar{\mathbf{a}}_0^H \mathbf{R}^{-1} \mathbf{a}_0|^2 = \frac{\sigma_0^2 |\bar{\mathbf{a}}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0|^2}{(1 + \sigma_0^2 \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0)^2}, \quad (\text{D.13})$$

and the denominator is calculated

$$\begin{aligned} \bar{\mathbf{a}}_0^H \mathbf{R}^{-1} \mathbf{R}_n \mathbf{R}^{-1} \bar{\mathbf{a}}_0 &= \bar{\mathbf{a}}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0 - \frac{2\sigma_0^2 |\bar{\mathbf{a}}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0|^2}{(1 + \sigma_0^2 \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0)} \\ &\quad + \frac{\sigma_0^4 (\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0) |\bar{\mathbf{a}}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0|^2}{(1 + \sigma_0^2 \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0)^2}. \end{aligned} \quad (\text{D.14})$$

In order to simplify the analytical expression, we define angle α as

$$\cos^2(\alpha) = \frac{|\mathbf{a}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0|^2}{(\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0)(\bar{\mathbf{a}}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0)}, \quad (\text{D.15})$$

which measures the mismatch between ASV and true data vector. Using (D.10)

and (D.15), we can get

$$\bar{\mathbf{a}}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0 = \frac{\sigma_0^2 |\mathbf{a}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0|^2}{SINR_o \cdot \cos^2 \alpha}. \quad (\text{D.16})$$

Substituting (D.10) into (D.13),

$$\sigma_0^2 |\bar{\mathbf{a}}_0^H \mathbf{R}^{-1} \mathbf{a}_0|^2 = \frac{\sigma_0^2 |\mathbf{a}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0|^2}{(1 + \text{SINR}_o)^2}. \quad (\text{D.17})$$

Substituting (D.16) into (D.14),

$$\bar{\mathbf{a}}_0^H \mathbf{R}^{-1} \mathbf{R}_n \mathbf{R}^{-1} \bar{\mathbf{a}}_0 = \frac{\sigma_0^2 |\bar{\mathbf{a}}_0^H \mathbf{R}^{-1} \mathbf{a}_0|^2 (1 + 2\text{SINR}_o \sin^2 \alpha + \text{SINR}_o^2 \sin^2 \alpha)}{\text{SINR}_o (1 + \text{SINR}_o)^2 \cos^2 \alpha}. \quad (\text{D.18})$$

From (D.17) and (D.18), the perturbed output SINR can be calculated

$$\widetilde{\text{SINR}} = \frac{\text{SINR}_o \cos^2 \alpha}{1 + 2\text{SINR}_o \sin^2 \alpha + \text{SINR}_o^2 \sin^2 \alpha}. \quad (\text{D.19})$$

D.3 Expected SINR of Perturbed Cases

Assuming the ASV ($\bar{\mathbf{a}}_0$) and true data vector ($\mathbf{a}_0$) has the relationship:

$$\bar{\mathbf{a}}_0 = \mathbf{a}_0 + \boldsymbol{\delta}, \quad (\text{D.20})$$

where $\boldsymbol{\delta}$ is the random ASV mismatch which satisfies:

$$E_{\text{error}} \{\boldsymbol{\delta}\} = \mathbf{0}; \quad E_{\text{error}} \{\boldsymbol{\delta} \boldsymbol{\delta}^H\} = \mathbf{R}_e. \quad (\text{D.21})$$

From (D.15), the numerator of mismatch angle $\cos^2 \alpha$ is given by

$$\begin{aligned} E_{\text{error}} \left\{ |\mathbf{a}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0|^2 \right\} &= E_{\text{error}} \left\{ |\mathbf{a}_0^H \mathbf{R}_n^{-1} (\mathbf{a}_0 + \boldsymbol{\delta})|^2 \right\} \\ &= (\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0)^2 + \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{R}_e \mathbf{R}_n^{-1} \mathbf{a}_0. \end{aligned} \quad (\text{D.22})$$

The denominator of $\cos^2 \alpha$ from (D.15) is given by

$$\begin{aligned} E_{\text{error}} \left\{ \bar{\mathbf{a}}_0^H \mathbf{R}_n^{-1} \bar{\mathbf{a}}_0 \right\} (\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0) &= E_{\text{error}} \left\{ (\mathbf{a}_0 + \boldsymbol{\delta})^H \mathbf{R}_n^{-1} (\mathbf{a}_0 + \boldsymbol{\delta}) \right\} (\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0) \\ &= (\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0) (\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0 + \text{Tr} \{ \mathbf{R}_n^{-1} \mathbf{R}_e \}). \end{aligned} \quad (\text{D.23})$$

From (D.22) and (D.23), an analytical expression for the expected mismatch angle is given by

$$E_{error} \{ \cos^2 \alpha \} = \frac{(\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0)^2 + \mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{R}_e \mathbf{R}_n^{-1} \mathbf{a}_0}{(\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0)(\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0 + \text{Tr} \{ \mathbf{R}_n^{-1} \mathbf{R}_e \})} \quad (\text{D.24})$$

Define $\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{R}_e \mathbf{R}_n^{-1} \mathbf{a}_0 = e_1$ and $\text{Tr} \{ \mathbf{R}_n^{-1} \mathbf{R}_e \} = e_2$, and $\mathbf{a}_0^H \mathbf{R}_n^{-1} \mathbf{a}_0 = \frac{\text{SINR}_o}{\sigma_0^2}$, and the above equation can be written as

$$E_{error} \{ \cos^2 \alpha \} = \frac{\text{SINR}_o^2 + \sigma_0^4 e_1}{\text{SINR}_o^2 + \text{SINR}_o \sigma_0^2 e_2} \quad (\text{D.25})$$

Substituting (D.25) into the (D.19), the expected SINR of the Capon beamformer in perturbed case is given by:

$$E_{error} \{ \widetilde{\text{SINR}} \} = \frac{\text{SINR}_o^2 + \sigma_0^4 e_1}{\text{SINR}_o^2 \sigma_0^2 e_2 + \text{SINR}_o (1 - \sigma_0^4 e_1 + 2\sigma_0^2 e_2) + \sigma_0^2 e_2 - 2\sigma_0^4 e_1} \quad (\text{D.26})$$

When the phase and gain errors are independent, the covariance matrix $\mathbf{R}_e = (\sigma_g^2 + \sigma_\phi^2) \mathbf{I}$. The corresponding expected SINR can be calculated by substituting $e_1 = \mathbf{a}_0^H \mathbf{R}_n^{-2} \mathbf{a}_0$ and $e_2 = (\sigma_g^2 + \sigma_\phi^2) \text{Tr} \{ \mathbf{R}_n^{-1} \}$ into (D.26).

D.4 Random Steering Direction Error

When there exists steering direction error, $\Delta\theta$, the array response vector $\mathbf{a}$ is given by

$$\mathbf{a}(\theta_0) = \mathbf{a}(\theta_s + \Delta\theta) = \begin{bmatrix} e^{jkr_1^0 \sin(\theta_s + \Delta\theta)} & \dots & e^{jkr_N^0 \sin(\theta_s + \Delta\theta)} \end{bmatrix}^T, \quad (\text{D.27})$$

where θ_s is the look direction, θ_0 is the signal direction. Using a first-order Taylor-series approximation, we get

$$\sin(\theta_s + \Delta\theta) \approx \sin(\theta_s) + \cos(\theta_s) \Delta\theta. \quad (\text{D.28})$$

Thus,

$$\mathbf{a}(\theta_0) \approx \begin{bmatrix} e^{j\phi_1^0} e^{j\Delta\phi_1} & e^{j\phi_2^0} e^{j\Delta\phi_2} & \dots & e^{j\phi_N^0} e^{j\Delta\phi_N} \end{bmatrix}^T, \quad (\text{D.29})$$

where $\Delta\phi_n = kr_n^0 \Delta\theta \cos \theta_s$. We assume the steering direction errors follows a zero mean Gaussian distribution with standard deviation σ_θ , the corresponding phase errors $\Delta\phi$ are Gaussian variables with covariance matrix $\Phi_\theta = \text{diag} \{\sigma_1^2, \dots, \sigma_N^2\}$, where $\sigma_n^2 = (kr_n^0 \cos \theta_s)^2 \sigma_\theta^2$.

D.5 Random Position Errors

When there exist position errors in the array elements (assuming no steering direction error, i.e., $\theta_s = \theta_0$), the array response vector $\mathbf{a}$ is given by

$$\begin{aligned} \mathbf{a}(\theta_0) &= \begin{bmatrix} e^{jkr_1 \sin \theta_0} & \dots & e^{jkr_N \sin \theta_0} \end{bmatrix}^T \\ &= \begin{bmatrix} e^{jkr_1^0 \sin \theta_0} e^{jk\Delta r_1 \sin \theta_0} & \dots & e^{jkr_N^0 \sin \theta_0} e^{jk\Delta r_N \sin \theta_0} \end{bmatrix}^T, \end{aligned} \quad (\text{D.30})$$

where Δr_n is position error in the n th sensor. We assume the position errors $\Delta \mathbf{r} = [\Delta r_1, \dots, \Delta r_N]^T$ follows zero-mean jointly Gaussian distribution with covariance matrix $\mathbf{Q}$, i.e., $E \{\Delta \mathbf{r} \Delta \mathbf{r}^T\} = \mathbf{Q}$, the corresponding phase errors follows zero-mean jointly Gaussian distribution $\Phi_{pos} = (k \sin \theta_0)^2 \mathbf{Q}$.

D.6 Derivative in Section 4.5.2

Define $f(\phi) = \hat{\mathbf{a}}_0^H \mathbf{R}^{-1} \hat{\mathbf{a}}_0$ and the (m, n) element of $\mathbf{R}^{-1}$ as $r_{m,n}$, then

$$f(\phi) = \sum_{m=1}^N \sum_{n=1}^N e^{-j\phi_m} r_{m,n} e^{j\phi_n}. \quad (\text{D.31})$$

Differentiating $f(\phi)$ respect to ϕ_k gives:

$$\frac{\partial f(\phi)}{\partial \phi_k} = -j e^{-j\phi_k} \sum_{n=1}^N e^{-j\phi_n} r_{k,n} + j e^{-j\phi_k} \sum_{m=1}^N e^{-j\phi_m} r_{m,k}. \quad (\text{D.32})$$

The term

$$e^{-j\phi_k} \sum_{n=1}^N e^{-j\phi_n} r_{k,n} = \hat{\mathbf{a}}_0^* \otimes (\mathbf{R}^{-1} \hat{\mathbf{a}}_0), \quad (\text{D.33})$$

where $*$ denotes complex conjugate, and the $\otimes$ represents Hadamard product.

Since R is a hermitian matrix, $r_{m,n} = r_{n,m}^*$, (D.32) becomes:

$$\begin{aligned} \frac{\partial f(\phi)}{\partial \phi_k} &= -j [\hat{\mathbf{a}}_0^* \otimes (\mathbf{R}^{-1} \hat{\mathbf{a}}_0)] + j [\hat{\mathbf{a}}_0^* \otimes (\mathbf{R}^{-1} \hat{\mathbf{a}}_0)]^* \\ &= 2\text{Im} \{ \hat{\mathbf{a}}_0^* \otimes (\mathbf{R}^{-1} \hat{\mathbf{a}}_0) \}. \end{aligned} \quad (\text{D.34})$$

D.7 Distribution of ζ in Section 4.5.3

Define $\Delta\phi = \phi - \phi_0$. From (4.30), the phase error vector $\Delta\phi$ is a zero mean Gaussian vector with covariance Φ . Since Φ is a real symmetric matrix, it can be eigen-decomposed as

$$\Phi = \mathbf{P} \Lambda_\phi \mathbf{P}^{-1} \quad (\mathbf{P}^{-1} = \mathbf{P}^T), \quad (\text{D.35})$$

where $\mathbf{\Lambda}_\phi = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_N)$ is the eigenvalue matrix, and $\mathbf{P}$ is the corresponding eigenvector matrix. Thus,

$$\mathbf{\Phi}^{-1} = \mathbf{P}\mathbf{\Lambda}_\phi^{-1}\mathbf{P}^{-1} = \mathbf{P}\mathbf{D}_\phi\mathbf{D}_\phi\mathbf{P}^{-1}, \quad (\text{D.36})$$

where $\mathbf{D}_\phi$ is also a diagonal matrix and $\mathbf{D}_\phi\mathbf{D}_\phi = \mathbf{\Lambda}_\phi^{-1}$. From the above equation,

$$\zeta = \Delta\phi^T \mathbf{\Phi}^{-1} \Delta\phi = \Delta\phi^T \mathbf{P}\mathbf{D}_\phi\mathbf{D}_\phi\mathbf{P}^{-1} \Delta\phi. \quad (\text{D.37})$$

Define $\mathbf{u} = \mathbf{D}_\phi\mathbf{P}^{-1}\Delta\phi$, then

$$\zeta = \mathbf{u}^T \mathbf{u} = \sum_{i=1}^N u_i^2, \quad (\text{D.38})$$

where u_i are independent Gaussian random variables with zero mean and unit variance. Then the random variable ζ follows a chi-square distribution, given by

$$p(\zeta) = \frac{1}{2^{\frac{N}{2}} \Gamma(\frac{N}{2})} \zeta^{\frac{N}{2}-1} e^{-\frac{\zeta}{2}} \quad \zeta \geq 0 \quad (\text{D.39})$$

Appendix E

Author's Publications

1. Lei Lei, Yu Zhuliang, "Robust Capon beamformer against generalized phase errors", submitted to Signal Processing on.
2. Lei Lei, "Robust Capon beamformer against generalized phase errors", submitted to ICASSP 2009, Conference on, Taiwan Apr. 2009.
3. Lei Lei, Henry Lew, "Robust Supergain Array in Noise Fields", Information, Communications and Signal Processing, 2007 6th International Conference on, 10-13 Dec. 2007, pp. 1-4.
4. Lei Lei, Henry Lew, "Robust Adaptive Vertical Array in the Ocean", Information, Communications and Signal Processing, Oceans'06 Asia Pacific IEEE - Singapore, Conference on, 16-19 May. 2006, pp. 256-259.
5. Lei Lei, Henry Lew, Venkateswarlu Ronda, "Robust Data-independent Beamformer", Information, Communications and Signal Processing, 2005 Fifth

International Conference on 06-09 Dec. 2005 , pp. 1274-1277.

Bibliography

- [1] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing: Concepts and Techniques*. New Jersey: Prentice-Hall, 1993.
- [2] H. L. Van Trees, *Optimum Array Processing*. New York: John Wiley and Sons, 2002.
- [3] P. S. Naidu, *Sensor Array Signal Processing*. New York: CRC Press, 2001.
- [4] B. D. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Mag.*, vol. 5, no. 2, pp. 4–24, 1988.
- [5] S. S. Haykin and J. H. Justice, Eds., *Array Signal Processing*. New Jersey: Prentice-Hall, 1985.
- [6] R. O. Nielsen, *Sonar Signal Processing*. Boston: Artech House, 1991.
- [7] M. Brandstein and D. Ward, Eds., *Microphone Arrays: Signal Processing Techniques and Applications*. New York: Springer, 2001.
- [8] J. E. Hudson, *Adaptive Array Principles*. London: Peter Peregrinus Ltd., 1981.

-
- [9] C. D. Seligson, "Comments on high resolution frequency-wavenumber spectrum analysis," *Proc. IEEE*, vol. 58, no. 6, pp. 947–949, 1970.
 - [10] E. N. Gilbert and S. P. Morgan, "Optimum design of directive antenna arrays subject to random variations," *Bell Sys. Tech. J.*, vol. 34, pp. 637–663, July 1955.
 - [11] D. J. Ramsdale and R. A. Howerton, "Effect of element failure and random errors in amplitude and phase on the sidelobe level attainable with a linear array," *J. Acoust. Soc. Amer.*, vol. 68, pp. 901–907, 1980.
 - [12] A. H. Quazi, "Array beam response in the presence of amplitude and phase fluctuations," *J. Acoust. Soc. Amer.*, vol. 72, pp. 171–180, 1982.
 - [13] H. G. Berman and A. Berman, "Effect of correlated phase fluctuation on array performance," *J. Acoust. Soc. Amer.*, vol. 34, pp. 555–562, 1962.
 - [14] D. R. Farrier, "Gain of an array of sensors subjected to processor perturbations," *IEE Proc.*, vol. 130, no. 4, pp. 251–254, 1983.
 - [15] M. Uzsokly and L. Solymar, "Theory of super-directive linear arrays," *Acta Phys. Acad. Sci. Hungary*, vol. 6, no. 2, pp. 185–204, 1956.
 - [16] H. Cox and R. Zeskind, "Practical supergain," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, no. 3, pp. 393–398, 1986.

-
- [17] K. Takao, M. Fujita, and T. Nishi, "An adaptive antenna array under directional constraint," *IEEE Trans. Antennas Propagat.*, vol. 24, no. 5, pp. 662–669, 1976.
- [18] J. L. Krolik, "The performance of matched-field beamformers with Mediterranean vertical array data," *IEEE Trans. Signal Processing*, vol. 44, no. 10, pp. 2605–2611, 1996.
- [19] M. H. Er and A. Cantoni, "Derivative constraints for broad-band element space antenna array processors," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 31, no. 6, pp. 1378–1393, 1983.
- [20] K. M. Buckley and L. J. Griffiths, "An adaptive generalized sidelobe canceller with derivative constraints," *IEEE Trans. Antennas Propagat.*, vol. 34, no. 3, pp. 311–319, 1986.
- [21] S. Zhang and I. Thng, "Robust presteering derivative constraints for broad-band antenna arrays," *IEEE Trans. Signal Processing*, vol. 50, no. 1, pp. 1–10, 2002.
- [22] A. K. Steele, "Comparison of directional and derivative constraints for beamformers subject to multiple linear constraints," *IEE Proc.*, vol. 130, no. 1, pp. 41–45, 1983.
- [23] D. D. Feldman and L. J. Griffiths, "A projection approach for robust adaptive beamforming," *IEEE Trans. Signal Processing*, vol. 42, no. 4, pp. 867–876, 1994.

-
- [24] J. F. Bull, M. A. Arnao, and L. R. Burgess, "Hypersensitivity effects in adaptive antenna arrays," in *Antennas and Propagation Society International Symposium*, vol. 1, May 1990, pp. 396–399.
- [25] W. S. Youn and C. K. Un, "Robust adaptive beamforming based on the eigenstructure method," *IEEE Trans. Signal Processing*, vol. 42, no. 6, pp. 1543–1547, 1994.
- [26] N. K. Jablon, "Adaptive beamforming with the generalized sidelobe canceller in the presence of array imperfections," *IEEE Trans. Antennas Propagat.*, vol. 34, no. 8, pp. 996–1012, 1986.
- [27] B. D. Carlson, "Covariance matrix estimation errors and diagonal loading in adaptive arrays," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 24, no. 4, pp. 397–401, 1988.
- [28] H. Cox, R. M. Zeskind, and M. M. Owen, "Robust adaptive beamforming," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 35, no. 10, pp. 1365–1376, 1987.
- [29] M. H. Er and A. Cantoni, "A unified approach to the design of robust narrow-band antenna array processor," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 1, pp. 17–23, 1990.
- [30] O. Besson, A. A. Monakov, and C. Chalus, "Signal waveform estimation in the presence of uncertainties about the steering vector," *IEEE Trans. Signal Processing*, vol. 52, no. 9, pp. 2432–2440, 2004.

- [31] Y. C. Eldar, A. Nehorai, and P. S. L. Rosa, "An expected least-squares beamforming approach to signal estimation with steering vector uncertainties," *IEEE Signal Processing Lett.*, vol. 13, no. 5, pp. 288–291, 2006.
- [32] —, "An competitive mean-squared error approach to beamforming," *IEEE Trans. Signal Processing*, vol. 55, no. 11, pp. 5143–5154, 2007.
- [33] Y. Rong, Y. C. Eldar, and A. B. Gershman, "An competitive mean-squared error approach to beamforming," *IEEE Trans. Signal Processing*, vol. 1, no. 4, pp. 651–659, 2007.
- [34] S. A. Vorobyov, A. B. Gershman, and Z. Q. Luo, "Robust adaptive beamforming using worst-case performance optimization: a solution to the signal mismatch problem," *IEEE Trans. Signal Processing*, vol. 51, no. 2, pp. 313–324, 2003.
- [35] S. Shahbazpanahi, A. B. Gershman, Z. Q. Luo, and K. M. Wong, "Robust adaptive beamforming for general-rank signal models," *IEEE Trans. Signal Processing*, vol. 51, no. 9, pp. 2257–2269, 2003.
- [36] R. G. Lorenz and S. P. Boyd, "Robust minimum variance beamforming," *IEEE Trans. Signal Processing*, vol. 53, no. 5, pp. 1684–1696, 2005.
- [37] J. Li, P. Stoica, and Z. Wang, "On robust Capon beamforming and diagonal loading," *IEEE Trans. Signal Processing*, vol. 51, no. 7, pp. 1702–1715, 2003.
- [38] —, "Doubly constrained robust Capon beamforming," *IEEE Trans. Signal Processing*, vol. 52, no. 9, pp. 2407–2423, 2004.

-
- [39] R. Cook, R. Waterhouse, R. Berendt, S. Edelman, and M. Thompson, "Measurement of correlation coefficients in reverberant sound fields," *J. Acoust. Soc. Am.*, vol. 27, no. 6, pp. 1072–1077, 1955.
- [40] S. L. Gay and J. Benesty, Eds., *Acoustic Signal Processing for Telecommunication*. Kluwer Academic Publishers, 2000, ch. 10, pp. 181–237.
- [41] H. Cox, "Spatial correlation in arbitrary noise field with application to ambient sea noise," *J. Acoust. Soc. Am.*, vol. 54, no. 5, pp. 1289–1301, 1973.
- [42] R. A. Monzingo and T. W. Miller, *Introduction to Adaptive Arrays*. New York: John Wiley and Sons, 1980.
- [43] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*. New Jersey: Prentice-Hall, 1989.
- [44] J. G. Proakis, C. M. Rader, F. Ling, and C. L. Nikias, *Advanced Digital Signal Processing*. New York: Macmillan Publishing Company, 1992.
- [45] B. D. Steinberg, *Principles of Aperture and Array System Design*. New York: John Wiley and Sons, 1976.
- [46] R. S. Elliott, *Antenna Theory and Design*. New Jersey: Prentice-Hall, 1981.
- [47] J. D. Kraus, *Antennas*. New York: McGraw-Hill, 1988.
- [48] C. A. Balanis, *Antenna Theory Analysis and Design*. New York: John Wiley and Sons, 2005.

-
- [49] J. L. Volakis, *Antenna Engineering Handbook*. New York: McGraw-Hill, 2007.
- [50] C. L. Dolph, "A current distribution for broadside arrays which optimizes the relationship between beamwidth and sidelobe level," *Proc. IRE*, vol. 34, pp. 335–348, 1946.
- [51] T. T. Taylor, "Design of line source antennas for narrow beamwidth and low sidelobe," *IRE Trans. Antennas Propagat.*, vol. 3, pp. 16–28, 1955.
- [52] S. A. Schelkunoff, "A mathematical theory of linear arrays," *Bell Sys. Tech. J.*, vol. 22, no. 1, pp. 80–107, 1943.
- [53] D. E. Weston, "Jacobi sensor arrangement for maximum array directivity," *J. Acoust. Soc. Am.*, vol. 80, no. 4, pp. 1170–1181, 1986.
- [54] R. W. Taylor and G. E. Dailey, "The super-directional acoustic sensor," in *OCEANS '92. 'Mastering the Oceans Through Technology'. Proceedings.*, 1992, pp. 386–391.
- [55] J. M. Kates and M. R. Weiss, "A comparison of hearing-aid array-processing techniques," *J. Acoust. Soc. Amer.*, vol. 99, no. 5, pp. 3138–3148, 1996.
- [56] W. Soede, A. J. Berkhout, and F. A. Bilsen, "Development of a directional hearing instrument based on array technology," *J. Acoust. Soc. Amer.*, vol. 94, no. 2, pp. 785–798, 1993.

-
- [57] M. S. Bazarraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*. New York: John Wiley and Sons, 2006.
- [58] L. J. Griffiths and C. W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas Propagat.*, vol. 30, no. 1, pp. 27–34, 1982.
- [59] B. Widrow, P. E. Mantey, L. J. Griffiths, and B. B. Goode, "Adaptive antenna systems," *Proc. IEEE*, vol. 55, no. 12, pp. 2143–2159, 1967.
- [60] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Processing Mag.*, vol. 13, no. 4, pp. 67–94, 1996.
- [61] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propagat.*, vol. 34, no. 3, pp. 276–280, 1986.
- [62] B. Ottersten and T. Kailath, "Maximum likelihood localization of multiple sources by alternating projection," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, no. 10, pp. 1553–1560, 1988.
- [63] A. L. Swindlehurst and P. Stoica, "Maximum likelihood methods in radar array signal processing," *Proc. IEEE*, vol. 86, no. 2, pp. 421–441, 1998.
- [64] J. Capon, "High-resolution frequency wavenumber spectrum analysis," *Proc. IEEE*, vol. 57, no. 8, pp. 1408 – 1418, 1969.

-
- [65] O. L. F. III, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, no. 8, pp. 926–935, 1972.
- [66] H. Cox, "Resolving power and sensitivity to mismatch of optimum array processors," *J. Acoust. Soc. Amer.*, vol. 54, pp. 771–785, 1973.
- [67] L. C. Godara, "Error analysis of the optimal antenna array processors," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 22, no. 4, pp. 395–409, 1986.
- [68] M. Wax and Y. Anu, "Performance analysis of the minimum variance beamformer in the presence of steering vector errors," *IEEE Trans. Signal Processing*, vol. 44, no. 4, pp. 928–937, 1996.
- [69] J. K. Yang and A. L. Swindlehurst, "The effects of array calibration errors on DF-based signal copy performance," *IEEE Trans. Signal Processing*, vol. 43, no. 11, pp. 2724–2732, 1995.
- [70] R. T. Compton, "The effect of random steering vector errors in the Applebaum adaptive array," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 18, no. 5, pp. 392–400, 1982.
- [71] H. Cox, "Resolving power and sensitivity to mismatch of optimum array processors," *J. Acoust. Soc. Amer.*, vol. 54, no. 3, pp. 771–785, 1973.
- [72] C. L. Zahm, "Effects of errors in the direction of incidence on the performance of an adaptive array," *Proc. IEEE*, vol. 60, no. 8, pp. 1008–1009, 1972.

-
- [73] A. M. Vural, "Effects of perturbations on the performance of optimum adaptive arrays," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 15, no. 1, pp. 76–87, 1979.
- [74] B. D. Van Veen, "Minimum variance beamforming with soft response constraints," *IEEE Trans. Signal Processing*, vol. 39, no. 9, pp. 1964–1972, 1991.
- [75] W. S. Youn and C. K. Un, "A linearly constrained beamforming robust to array imperfections," *IEEE Trans. Signal Processing*, vol. 41, no. 3, pp. 1425–1428, 1993.
- [76] J. W. Kim and C. K. Un, "An adaptive array robust to beam pointing error," *IEEE Trans. Signal Processing*, vol. 40, no. 6, pp. 1582–1584, 1992.
- [77] K. L. Bell, Y. Ephraim, and H. L. Van Trees, "A bayesian approach to robust adaptive beamforming," *IEEE Trans. Signal Processing*, vol. 48, no. 2, pp. 386–398, 2000.
- [78] N. Kikuma and K. Takao, "Broadband and robust adaptive antenna under correlation constraint," *Microwaves, Antennas and Propagation, IEE Proc.*, vol. 136, no. 2, pp. 85–89, 1989.
- [79] O. Hoshuyama, A. Sugiyama, and A. Hirano, "A robust adaptive beamformer for microphone arrays with a blocking matrix using constrained adaptive filters," *IEEE Trans. Signal Processing*, vol. 47, no. 10, pp. 2677–2684, 1999.

-
- [80] M. H. Er and B. C. Ng, "A new approach to robust beamforming in the presence of steering vector errors," *IEEE Trans. Signal Processing*, vol. 42, no. 7, pp. 1826–1829, 1994.
 - [81] T. Kooij, "Adaptive array processors for sensitivity constrained optimization," Ph.D. dissertation, Catholic Univ. America, Washington, DC, 1977.
 - [82] L. P. Winkler and M. Schwartz, "Adaptive nonlinear optimization of the signal-to-noise ratio of an array subject to a constraint," *J. Acoust. Soc. Am.*, vol. 52, no. 1, pp. 39–51, 1972.
 - [83] —, "Constrained array optimization by penalty function techniques," *J. Acoust. Soc. Am.*, vol. 55, no. 5, pp. 1042–1048, 1974.
 - [84] Z. L. Yu and M. H. Er, "A robust capon beamformer against uncertainty of nominal steering vector," *EURASIP Journal on Applied Signal Processing*, vol. 1, pp. 29–37, 2006.
 - [85] S. E. Nai, W. Ser, Z. L. Yu, and S. Rahardja, "Iterative robust capon beamformer," in *SSP '07, IEEE/SP 14th Workshop on*, 2007, pp. 542–545.
 - [86] M. Abramowitz and A. Stegun, *Handbook of Mathematical Functions: with formulas, graphs, and mathematical tables*. New York: Dover Publications, 1965.
 - [87] J. F. Kaiser and R. W. Schafer, "On the use of the I_0 -sinh window for spectrum analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, no. 1, pp. 105–107, 1980.

-
- [88] Y. Viniotis, *Probability and Random Processes for Electrical Engineers*. New York: McGraw-Hill, 1998.