TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

DISSERTATION

A Contribution to the Design of Sparse Arrays for Data-Independent and Adaptive Broadband Beamformers

This dissertation is approved by the Department of Electrical and Computer Engineering for the award of the Doctoral Degree.

Author: Phan Le Son

Dean: Prof. Dr.-Ing. Ralph Urbansky

Reviewer: Prof. Dr.-Ing. Alexander Potchinkov
Reviewer: Prof. Dr.-Ing. Christian Langen

Date of defense: 19 November, 2021

D386

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Abstract

A Contribution to the Design of Sparse Arrays for Data-Independent and Adaptive Broadband Beamformers

by Phan Le Son

Beamforming performs spatial filtering to preserve the signal from given directions of interest while suppressing interfering signals and noise arriving from other directions. For example, a microphone array equipped with beamforming algorithm could preserve the sound coming from a target speaker and suppress sounds coming from other speakers. Beamformer has been widely used in many applications such as radar, sonar, communication, and acoustic systems. A *data-independent beamformer* is the beamformer whose coefficients are independent on sensor signals, it normally uses less computation since the coefficients are computed once. Moreover, its coefficients are derived from the well-defined statistical models, then it produces less artifacts. The major drawback of this beamforming class is its limitation to the interference suppression. On the other hand, an *adaptive beamformer* is a beamformer whose coefficients depend on or adapt to sensor signals. It is capable of suppressing the interference better than a data-independent beamforming but it suffers from either too much distortion of the signal of interest or less noise reduction when the updating rate of coefficients does not synchronize with the changing rate of the noise model. Besides, it is computationally intensive since the coefficients need to be updated frequently. In acoustic applications, the bandwidth of signals of interest extends over several octaves, but we always expect that the characteristic of the beamformer is invariant with regard to the bandwidth of interest. This can be achieved by the so-called *broadband beamforming*. Since the beam pattern of conventional beamformers depends on the frequency of the signal, it is common to use a dense and uniform array for the broadband beamforming to guarantee some essential performances together, such as frequency-independence, less sensitive to white noise, high directivity factor or high front-to-back ratio. In this dissertation, we mainly focus on the sparse array of which the aim is to use fewer sensors in the array, while simultaneously assuring several important performances of the beamformer. In the past few decades, many design methodologies for sparse arrays have been proposed and were applied in a variety of practical applications. Although good results were presented, there are still some restrictions, such as the number of sensors is large, the designed beam pattern must be fixed, the steering ability is limited and the computational complexity is high.

In this work, two novel approaches for the sparse array design taking a hypothesized uniform array as a basis are proposed, that is, one for data-independent beamformers and the another for adaptive beamformers. As an underlying component of the proposed methods, the dissertation introduces some new insights into the uniform array with broadband beamforming. In this context, a function formulating the relations between the sensor coefficients and its beam pattern over frequency is proposed. The function mainly contains the coordinate transform and inverse Fourier transform. Furthermore, from the bijection of the function and broadband beamforming perspective, we propose the lower and upper bounds for the interdistance of sensors. Within these bounds, the function is a bijective function that can be utilized to design the uniform array with broadband beamforming.

For data-independent beamforming, many studies have focused on optimization procedures to seek the sparse array deployment. This dissertation presents an alternative approach to determine the location of sensors. Starting with a weight spectrum of a virtual dense and uniform array, some techniques are used, such as analyzing a weight spectrum to determine the critical sensors, applying the clustering technique to group the sensors into different groups and selecting representative sensors for each group. After the sparse array deployment is specified, the optimization technique is applied to find the beamformer coefficients. The proposed method helps to save the computation time in the design phase and its beamformer performance outperforms other state-of-the-art methods in several aspects such as the higher white noise gain, higher directivity factor or more frequency-independence.

For adaptive beamforming, the dissertation attempts to design a versatile sparse microphone array that can be used for different beam patterns. Furthermore, we aim to reduce the number of microphones in the sparse array while ensuring that its performance can continue to compete with a highly dense and uniform array in terms of broadband beamforming. An irregular microphone array in a planar surface with the maximum number of distinct distances between the microphones is proposed. It is demonstrated that the irregular microphone array is well-suited to *sparse recovery* algorithms that are used to solve underdetermined systems with subject to sparse solutions. Here, a sparse solution is the sound source's spatial spectrum that need to be reconstructed from microphone signals. From the reconstructed sound sources, a method for array interpolation is presented to obtain an interpolated dense and uniform microphone array that performs well with broadband beamforming.

In addition, two alternative approaches for generalized sidelobe canceler (GSC) beamformer are proposed. One is the data-independent beamforming variant, the other is the adaptive beamforming variant. The GSC decomposes beamforming into two paths: The upper path is to preserve the desired signal, the lower path is to suppress the desired signal. From a beam pattern viewpoint, we propose an improvement for GSC, that is, instead of using the blocking matrix in the lower path to suppress the desired signal, we design a beamformer that contains the nulls at the look direction and at some other directions. Both approaches are simple beamforming design methods and they can be applied to either sparse array or uniform array.

Lastly, a new technique for direction-of-arrival (DOA) estimation based on the annihilating filter is also presented in this dissertation. It is based on the idea of finite rate of innovation to reconstruct the stream of Diracs, that is, identifying an annihilating filter/locator filter for a few uniform samples and the position of the Diracs are then related to the roots of the filter. Here, an *annihilating filter* is the filter that suppresses the signal, since its coefficient vector is always orthogonal to every frame of signal. In the DOA context, we regard an active source as a Dirac associated with the arrival direction, then the directions of active sources can be derived from the roots of the annihilating filter. However, the DOA obtained by this method is sensitive to noise and the number of DOAs is limited. To address these issues, the dissertation proposes a robust method to design the annihilating filter and to increase the degree-of-freedom of the measurement system (more active sources can be detected) via observing multiple data frames. Furthermore, we also analyze the performance of DOA with diffuse noise and propose an extended multiple signal classification algorithm that takes diffuse noise into account. In the simulation, it shows, that in the case of diffuse noise, only the extended multiple signal classification algorithm can estimate the DOAs properly.

TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Zusammenfassung

A Contribution to the Design of Sparse Arrays for Data-Independent and Adaptive Broadband Beamformers

von Phan Le Son

Ein Beitrag zum Design dünn besetzter Mikrofonarrays für datenunabhängige und adaptive Breitband-Beamformer Beamforming ist eine Technik, die eine räumliche Filterung durchführt, um gewünschte Signale aus bestimmten Richtungen zu erhalten und gleichzeitig Störsignale und Rauschen aus anderen Richtungen zu unterdrücken. Beamforming wird in vielen Anwendungen wie beispielsweise in Radar-, Sonar-, Kommunikations- und akustischen Systemen eingesetzt. Ein datenunabhängiger Beamformer ist ein Beamformer, dessen Koeffizienten unabhängig von den Messsignalen sind, und der normalerweise einen geringeren Rechenaufwand erfordert und weniger Betriebsartefakte erzeugt. Der Nachteil dieser Klasse von Beamformern ist die eingeschränkte Unterdrückung von Störsignalen. Im Gegensatz dazu ist ein adaptiver (datenabhängiger) Beamformer ein Beamformer, dessen Koeffizienten von den Messsignalen abhängig sind oder sich an diese anpassen. Ein solcher Beamformer kann Störsignale besser unterdrücken als der datenunabhängige Beamformer, erfordert jedoch einen hohen Rechenaufwand und kann zu Störungen des Signals führen. Bei akustischen Anwendungen erstreckt sich die Bandbreite der interessierenden Signale über mehrere Oktaven und die Eigenschaften des Beamformers sollen über den interessierenden Frequenzbereich möglichst konstant sein, was mit sogenannten Breitband-Beamformern erreicht werden kann. Da die Richtcharakteristik herkömmlicher Beamformer von der Frequenz des Signals abhängig ist, verwendet man üblicherweise ein dicht und gleichmäßig besetztes Array für das Breitband-Beamforming, um einige wesentliche Anfordeungen wie z.B. Frequenzunabhängigkeit, white noise gain, directivity factor und front-to-back ratio gerecht zu werden. In dieser Dissertation konzentrieren wir uns auf dünn besetzte Arrays, deren Ziel es ist, mit wenigen Sensoren gleichzeitig mehrere wichtige Anforderungen des Beamformers zu erfüllen. In den letzten Jahrzehnten wurden viele Entwurfsmethoden für dünn besetzte Arrays vorgeschlagen und in einer Vielzahl von praktischen Anwendungen genutzt. Obwohl gute Ergebnisse erzielt wurden, gibt es immer noch einige Einschränkungen, z.B., dass die Anzahl der Sensoren groß ist, der entworfene Beamformer statisch ist und die Komplexität der erforderlichen Berechnungen hoch ist.

In dieser Arbeit werden zwei neue Ansätze für das Design dünn besetzter Arrays vorgeschlagen, die auf dem hypothetischen gleichmäßig besetztem Array basieren, einer für datenunabhängige Beamformer und einer für adaptive Beamformer. Als zugrunde liegende Komponente der vorgeschlagenen Methoden bietet die Dissertation einige neue Einblicke in das gleichmäßig besetzte Array mit Breitband-Beamforming. In diesem Zusammenhang wird eine Funktion vorgeschlagen, die die Beziehungen zwischen den Sensorkoeffizienten und der Richtcharakterisitk formuliert. Die Funktion wird als inverse beam pattern transform bezeichnet. Sie enthält hauptsächlich die Koordinatentransformation und die inverse Fourier Transformation. Darüber hinaus legen wir aus der Perspektive der inverse beam pattern transform

und dem breitbandigem Beamforming die unteren und oberen Grenzen für den Abstand der Sensoren fest. Innerhalb dieser Grenzen ist die inverse beam pattern function die bijektive Funktion, die verwendet werden kann, um das gleichmäßig besetzte Array mit Breitband-Beamforming zu entwerfen.

Für die datenunabhängigen Beamformer haben sich viele Studien auf Optimierungsverfahren darauf konzentriert, ein dünn besetztes Array zu entwerfen. Diese Dissertation präsentiert einen alternativen Ansatz zur Bestimmung der Position der Sensoren, der ausgehend von einem Gewichtsspektrum eines virtuellen gleichmäßig besetzten Arrays aus mehreren Schritten besteht. Dazu gehören die Analyse des Gewichtsspektrums zur Bestimmung der kritischen Sensoren, die Anwendung der Clustering-Technik zur Gruppierung der Sensoren in verschiedene Gruppen und die Auswahl der repräsentativen Sensoren für jede Gruppe. Nachdem das dünn besetzte Array spezifiziert wurde, wird eine Optimierungstechnik angewendet, um die Beamformer-Koeffizienten zu ermitteln. Das vorgeschlagene Verfahren hilft, Rechenzeit in der Entwurfsphase einzusparen und die Ergebnisse übertreffen andere Verfahren nach dem Stand der Technik in mehreren Aspekten.

Hinsichtlich adaptiver Beamformer zielt die Dissertation darauf ab, ein vielseitiges dünn besetztes Mikrofonarray zu entwerfen, das für verschiedene Richtcharakteristiken verwendet werden kann. Darüber hinaus wird angestrebt, die Anzahl der Mikrofone in dem dünn besetzten Array klein zu halten und gleichzeitig sicherzustellen, dass die Güte weiterhin mit einem dicht und gleichmäßig besetzten Array zum Breitband-Beamforming konkurrieren kann. Dazu wird eine unregelmäß- ige Anordnung der Sensoren in einer Ebene mit der maximalen Anzahl unterschiedlicher Abstände zwischen den Sensoren vorgeschlagen. Es wird gezeigt, dass das unregelmäßige Mikrofonarray für die Algorithmen, die zur Rekonstruktion der Schallquellen verwendet werden, gut geeignet ist. Ausgehend von den rekonstruierten Schallquellen wird ein Interpolationsverfahren vorgestellt, um ein interpoliertes dichtes und gleichmäßig besetztes Array zu erhalten, das für das Breitband-Beamforming geeignet ist.

Darüber hinaus werden zwei alternative Ansätze für das GSC-Beamforming (Generalized Sidelobe Canceler) vorgeschlagen. Einer für das datenunabhängige Beamforming, ein anderer für das adaptive Beamforming. Der Ansatz GSC-Beamforming zerlegt das Beamforming in zwei Pfade: Der obere Pfad dient zur Erhaltung des gewünschten Signals, der untere Pfad dient zur Unterdrückung der unerwünschten Signale. Unter dem Gesichtspunkt des Beamformings, schlagen wir eine Verbesserung für GSC vor. Anstatt eine Blocking-Matrix im unteren Pfad zu verwenden, um das gewünschte Signal zu unterdrücken, entwerfen wir einen Beamformer, der die Nullen in interessierenden Richtungen und einigen anderen Richtungen enthält. Beide Ansätze sind einfache Entwurfsmethoden, die auf dünn besetzte und gleichmäßig besetzte Arrays angewendet werden können.

Schließlich wird eine neue Technik zur Schätzung der Einfallsrichtung (DOA) vorgestellt, die auf einem sogenannten Annihilating Filter basiert. Die durch dieses Verfahren ermittelte Einfallsrichtung ist jedoch empfindlich gegenüber Rauschen und die Anzahl der gleichzeitig geschätzten Einfallsrichtungen ist begrenzt. Um diese Probleme zu beheben, wird eine robuste Methode vorgeschlagen, um das Annihilating Filter zu entwerfen und den Freiheitsgrad des Messsystems (wie viele Quellen können gleichzeitig erkannt werden) durch Beobachtung mehrerer Datenfenster zu erhöhen. Darüber hinaus analysieren wir auch die Fähigkeit, die Einfallswinkel bei diffusem Rauschen zu ermitteln und schlagen einen erweiterten MUSIC-Algorithmus vor, der das diffuse Rauschen berücksichtigt. Im Experiment zeigt sich,

dass bei diffusem Rauschen nur mit dem erweiterten MUSIC-Algorithmus die Einfallswinkel korrekt ermittelt werden können.

Acknowledgements

Firstly, I acknowledge with thanks my supervisor, Prof. Alexander Potchinkov, for his invaluable advice and guidance in my research. I am very much thankful to him for spending time on discussing ideas, explaining the concepts and supporting in the coding/hardware implementations. Also, Prof. Potchinkov always respects my ideas, even sometimes they were not really good. Moreover, he offers a very flexible working environment.

Secondly, I would like to thank Dr. Stephan Herzog for his invaluable advice and his supporting in many cases, even outside of the office. I am also thankful for my colleagues who have made the Digital Signal Processing chair in TU Kaiserslautern a friendly and comfortable working environment. Especially, I would like to thank Mr. Natanael Nieland for his supporting in this writing.

Finally, I need to thank Prof. Christian Langen from Karlsruhe University of Applied Sciences for his recommendation of my PhD study, that is a wonderful journey, and his supporting during my early days in Germany.

Contents

Acknowledgements **ix**

1 Introduction to the Thesis Topic **1**
1.1 Background and Motivation . . . 1
1.2 Main Contributions . . . 3
1.3 Organization . . . 4
1.4 Published and Submitted Papers . . . 5

2 Scientific Background **7**
2.1 Fundamentals of Microphone Array . . . 7
2.1.1 Sound Propagation . . . 7
Far Field . . . 8
Near Field . . . 8
Superposition Property . . . 8
2.1.2 Microphone . . . 9
Condenser Microphone . . . 9
Dynamic Microphone . . . 9
MEMS Microphone . . . 10
2.2 Review of Beamforming Techniques . . . 11
2.2.1 Characteristics of Beamforming . . . 12
Beam Pattern . . . 12
Beam-width . . . 13
Steering . . . 13
Directivity Factor . . . 14
White Noise Gain . . . 14
Front-to-Back Ratio . . . 14
2.2.2 Data-independent Beamforming . . . 15
Delay and Sum Beamforming . . . 15
Superdirective Beamforming . . . 16
Differential Beamforming . . . 16
2.2.3 Adaptive Beamforming . . . 20
Linearly Constrained Adaptive Beamforming . . . 20
Generalized Sidelobe Canceler Beamformer . . . 23
2.3 Sparse Array and Broadband Beamforming . . . 25

3 Using Inverse Fourier Transform to Obtain the Beamformer Weights **29**
3.1 Introduction . . . 29
3.2 Continuous Sensor . . . 31
3.3 Discrete Sensor . . . 34
3.4 Imperfections of Applying Inverse Beam Pattern Transform . . . 38
3.4.1 Non-uniform Discrete Direction Angle . . . 38
3.4.2 The Number of Encoding Points at Low Frequencies . . . 39
3.5 Numerical Simulations . . . 40

3.6 Conclusions . . . 48
3.7 Appendix: The Number of Encoding Points . . . 49

4 Sparse Arrays with Frequency-Independent Beam Pattern 51
4.1 Introduction . . . 51
4.2 Signal Model . . . 54
4.3 Design Method . . . 55
4.3.1 Analysis . . . 56
4.3.2 Selecting . . . 59
4.3.3 Optimization . . . 61
4.4 Evaluation metrics . . . 62
4.5 Numerical Simulations . . . 63
4.5.1 Sparse Broad-side Linear Array . . . 63
Analysis Step . . . 63
Selecting Step . . . 64
Optimization Step . . . 65
4.5.2 Sparse End-fire Linear Array . . . 68
Analysis Step . . . 69
Selecting Step . . . 69
Optimization Step . . . 70
4.5.3 Planar Symmetric Sparse Array . . . 72
Analysis Step . . . 73
Selecting Step . . . 75
Optimization Step . . . 75
4.6 Conclusions . . . 78

5 Irregular Microphone Array Design for Broadband Beamforming 79
5.1 Introduction . . . 79
5.2 Signal Model . . . 82
5.3 Design Irregular Microphone Array . . . 83
5.3.1 Irregular Microphone Array Deployment . . . 83
5.3.2 Recovery Algorithms for Sound Source Reconstruction . . . 88
5.3.3 Microphone Interpolation . . . 94
5.3.4 Broadband Beamforming for Dense and Uniform Array . . . 94
5.4 Numerical Simulations . . . 94
5.5 Experimental Results . . . 97
5.5.1 Experiment 1 . . . 98
5.5.2 Experiment 2 . . . 101
5.6 Conclusions . . . 103

6 Discussion and Future Directions 105
6.1 Discussion . . . 105
6.2 Future Directions . . . 106

A Alternative Approaches to Generalized Sidelobe Canceler 109
A.1 Introduction . . . 110
A.2 Generalized Sidelobe Canceler . . . 112
A.3 Alternative Approaches to Generalized Sidelobe Canceler Beamformer 115
A.3.1 Fixed Beamforming with Suppressed Sidelobes . . . 116
A.3.2 Adaptive Beamforming with Suppresed Sidelobe . . . 117
A.4 Numerical Simulation . . . 119

A.4.1 Fixed Beamforming with Suppressed Sidelobe 119
A.4.2 Adaptive Beamforming with Suppressed Sidelobe 122
A.5 Conclusions 125
A.6 Appendix: Phase Response of Beamforming 125

B DOA Estimation via an Annihilating Filter **127**
B.1 Introduction 127
B.2 MUSIC with Sound Noise Model 128
B.3 DOA Estimation with Annihilating Filter 130
B.4 Numerical Simulations 133
B.4.1 Simulations in White Noise Environment 133
B.4.2 Simulations in White Noise and Diffuse Noise Environments . 133
B.5 Conclusions 135

C Matlab Code: Simulated Annealing for Sensor Array Deployment **137**

D Matlab code: Orthogonal Matching Pursuit (OMP) **141**

List of Figures

2.1 Condenser microphone. 9
2.2 Dynamic microphone. 10
2.3 MEMS microphone [Pfl17]. 10
2.4 Spatial filtering of beamforming [BJ12]: Blue is the audio signal, a dash-red box is a frame of the STFT. 11
2.5 Cardioid beam pattern. 12
2.6 Beam-width of a beam pattern. 13
2.7 Null, mainlobe, sidelobe of beam pattern. 13
2.8 Electronic steering ability of a linear array. 14
2.9 Uniform linear array configuration. 15
2.10 DSB Beam patterns of the ULA in Fig. 2.9 at different frequencies. . . . 16
2.11 SD Beam patterns of the ULA in Fig. 2.9 at different frequencies. 17
2.12 The implementation of third-order DMA in time-domain [BJ12]. 17
2.13 Beam patterns of the third-order DMA. 18
2.14 Beam patterns of the third-order DMA are designed in Fourier domain. 19
2.15 MVDR beamforming at 1.6 kHz focuses on the broadside direction and the interference presents at 45^o and 135^o. 20
2.16 MVDR beamforming with broadband focus on the broadside direction and interference present at 45^o and 135^o. 21
2.17 Beam pattern with LCMV over frequency. 22
2.18 Schematic description of the decomposition of the optimal weight vector into two orthogonal parts. 24
2.19 0.5 second waveform: Microphone (blue line) contains the interference, white noise and interested signal; the SOI (green line) contains a impulse response; GSC (red line) beams to the SOI. 25
2.20 Zoom-in the waveform: Microphone (blue line) contains the interference, white noise and interested signal; the SOI (green line) contains a impulse response; GSC (red line) beams to the SOI. 25
2.21 An example of nested array. 26
2.22 Sparse array vs uniform array. 27

3.1 The steps inside the proposed method to obtain the beamformer coefficients from an arbitrary beam pattern. 31
3.2 Example of gain functions associated to different frequencies in the spherical coordinate. 32
3.3 An example of the gain function in the Cartesian coordinate at one frequency. 33
3.4 Inside inverse beam pattern transform at a frequency. 34
3.5 Discrete sensor array in the Cartesian coordinate. 34
3.6 Sensor indices in the u-v plane of the Cartesian coordinate and the radius of the hemispheres 36

3.7 Transform from the polar coordinate to the Cartesian coordinate and the non-linear resolution of direction angles. 39
3.8 The number of encoding points is changed corresponding to the frequency. 39
3.9 (a) Desired beam pattern (in dB scale). (b) Cross-section of beam pattern. 41
3.10 (a) Beam pattern at 16 kHz, (b) Beam patterns over frequency from 200 Hz to 20 kHz, (c) WNGs over frequency ($N = 200$, $\Delta_d = 0.01$ m and $c = 343.2$ m/s). 42
3.11 Desired Beam-Pattern in dB scale (maximum dB scale is 40dB). 43
3.12 A gain function in the Cartesian coordinate at 16 kHz. 43
3.13 (a) Beam pattern at 16 kHz (in dB scale). (b) Beam patterns over frequency from 250 Hz to 16 kHz. (c) WNGs over frequency ($N = 100$, $\Delta_d = 0.015$ m and $c = 343.2$ m/s). 45
3.14 (a) Blue line is the beam pattern (BP) in Eq. (3.19), red "*" is the desired beam pattern without side-lobe. (b) Beam pattern at 16 kHz (in dB scale). (c) Beam patterns over frequency from 350 Hz to 20 kHz. (d) WNGs over frequency ($N = 101$, $\Delta_d = 0.01$ m and $c = 343.2$ m/s). . . 46
3.15 (a) Blue line is the beam pattern in Eq. (3.19) steering to $\theta = -70^o$, red "*" is the desired beam pattern without side-lobe. (b) Beam pattern at 16 kHz (in dB scale). (c) Beam patterns over frequency from 350 Hz to 20 kHz. (d) WNGs over frequency ($N = 101$, $\Delta_d = 0.01$ m and $c = 343.2$ m/s). 47
3.16 The number of integer points on the line $u = k$. 49

4.1 Flowchart for the proposed sparse array design. 53
4.2 Example of gain functions associate with different frequencies in the spherical coordinate: Concentric hemispheres with different radii. . . . 57
4.3 A gain function in the Cartesian coordinate at a frequency. 57
4.4 Sensor indices in the plan u-v of the Cartesian coordinate and the radius of the hemispheres. 58
4.5 Planar array: Analyzing the DUA. 59
4.6 Broad-side linear array: Analyzing the DUA. 64
4.7 The broad-side linear array deployments: proposed symmetric sparse array (red circle), SUA (green circle) and BUA (blue circle) and coherence sparse array design (black *). 65
4.8 Broad-side linear array: The beam patterns versus frequency. 65
4.9 Broad-side linear array: Beam patterns (dB) at 10 kHz for different broad-band linear arrays, the optimization of the BUA (c) is infeasible. 66
4.10 Side-lobe constraints are relaxed, the performance comparison of different broad-side linear array layouts (the disconnection lines mean that the optimizations are infeasible). 67
4.11 The side-lobe constraints are strictly set, the performance comparison of different Broad-side linear array layouts. 68
4.12 Broad-side linear array: beam patterns (dB) at 10 kHz for different look directions: (a) 90^o, (b) 95^o, (c) 100^o, (d) 105^o, (e) 110^o, (f) 115^o. . . 69
4.13 End-fire linear array: Analyzing the DUA with broadband beamforming. 70
4.14 The end-fire linear array deployments: New sparse array (red circle), SUA (green circle), BUA (blue circle) and incoherence sparse array design (black *). 70
4.15 End-fire linear array: Beam patterns versus frequency. 71

4.16 End-fire linear array: Beam patterns at 1 kHz (for the case the error tolerance of the BPE is strictly set). The optimization of the BUA (c) is infeasible. . . . 72
4.17 Beam pattern error-tolerances are strictly set, the performance comparison of different end-fire linear array layouts. . . . 73
4.18 The beam pattern error-tolerances are relaxed, the performance comparison of different end-fire linear array layouts. . . . 74
4.19 Symmetric sparse array (81 sensors constitute an irregular layout but having four symmetric quarters), SUV (121 sensors) and BUA (121 sensors). . . . 75
4.20 Planar array: (left column) beam patterns at 1.7 kHz and (right column) their cross-sections over frequency. . . . 76
4.21 Comparison of the performances of different planar array layouts. . . . 77
4.22 Beam patterns (top view) at 1.7 kHz for different look directions. . . . 78

5.1 The steps of the proposed method. . . . 81
5.2 Segment layout for microphones. . . . 84
5.3 The possible positions of microphones in a segment. . . . 84
5.4 The final layout of irregular microphone array. . . . 86
5.5 The histogram of distinct distances of the microphone array in Fig. 5.4. 87
5.6 The updated cost values of the simulated annealing over the iterations. 87
5.7 The spectrum of the real part of the transformation matrix at 2.532 kHz of the irregular array (25 microphones). The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$. . . . 88
5.8 The imaginary part of the transformation matrix at 2.532 kHz of the irregular array (25 microphones). The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$. . . . 88
5.9 The spectrum of the real part of the transformation matrix at 2.532 kHz of the uniform array (25 microphones). The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$. . . . 89
5.10 The spectrum of the imaginary part of the transformation matrix at 2.532 kHz of the uniform array (25 microphones). The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$. . 89
5.11 Analyzing the maximum columns' coherence of submatrices of the transformation matrix. . . . 90
5.12 Spectrum of the Gram matrix ($\mathbf{A}^H\mathbf{A}$), the axes' values from 1 to N are the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$. . . . 91
5.13 The simulated sound sources impinge on the array: The X-axis is the elevation angle with the range of $(0^o, 90^0)$, the Y-axis is the azimuth angle with the range of $(0^o, 360^o)$ and the Z-axis is the amplitude of the active sounds. . . . 91
5.14 Comparison of sparse recovery algorithms for the sound source reconstruction: X-axis is the elevation angle with the range $(0^o, 90^0)$, Y-axis is the azimuth angle with the range $(0^o, 360^o)$ and Z-axis is the amplitude of active sounds. . . . 92
5.15 Comparison of sparse recovery algorithms. . . . 93

5.16 The reconstructed incident sound by the OMP: X-axis is the elevation angle, Y-axis is the azimuth angle, Z-axis is the strength of the incident sound. 95
5.17 The reconstructed incident sounds by the l_2-norm: the X-axis is the elevation angle, the Y-axis is the azimuth angle and the Z-axis is the strength of the incident sound. 96
5.18 In the case of noiseless, using the OMP algorithm for the sound source reconstruction: Beam pattern of the irregular microphone array (left) and beam pattern of the full microphone array (right). 96
5.19 In the case of white noise, SNR = 20 dB, using the OMP algorithm for the sound source reconstruction: Beam pattern of the irregular microphone array (left) and beam pattern of the full microphone array (right). 97
5.20 In the case of noiseless, using LSE minimization for the sound source reconstruction: Beam patterns of the irregular array (left) and beam patterns of the DUA (right). 97
5.21 In the case of white noise, SNR = 20 dB, using LSE minimization for the sound source reconstruction: Beam patterns of the irregular array (left) and beam patterns of the DUA (right). 98
5.22 Experiment with the microphone array in Fig. 5.4. 99
5.23 Spectrogram and waveform of the audio signal from microphone 5. The horizontal axis is the time, the vertical axis of the top and bottom figures are the frequency and amplitude, respectively. 99
5.24 Spectra of the signals (from top to bottom of the figure): microphone 5, microphone 26, interpolated audio via the OMP, and interpolated audio via the LSE (l_2-norm). The horizontal axes are the time (s), the vertical axes are the frequency (kHz). 100
5.25 Waveform of the signals (from top to bottom of the figure): microphone 5, verification microphone, interpolated microphone via the OMP, and interpolated microphone via the LSE (l_2-norm). The horizontal axes are the time (s), the vertical axes are the amplitude. 100
5.26 Waveform of the signals (from top to bottom of the figure): microphone 5, microphone 26 (verification microphone), interpolated microphone by the OMP, and beamforming. The horizontal axes are the time (s), the vertical axes are the amplitude. 101
5.27 Waveform of the signals (from top to bottom of the figure): microphone 5, the beam signal steers to the direction of the source of interest, and the beam signal steers to the inverted direction of the source of interest. The horizontal axes are the time (s), the vertical axes are the amplitude. 102
5.28 Spectrogram of the signals (from top to bottom of the figure): microphone 5, beam signal steers to direction of source of interest, and the beam signal steers to the inverted direction of the source of interest. The horizontal axes are the time (s), the vertical axes are the frequency (kHz). 102
5.29 Spectrogram of the signals (from top to bottom of the figure): the spectrum of the beamforming steering to the speaker 1 at $(\phi, \theta) = (0^o, 0^o)$ and the spectrum of microphone 5 (there are overlapping sounds on zone 1, 2). The horizontal axes are the time (s), the vertical axes are the frequency (kHz). 103

A.1 Overview of generalize sidelobe canceler. 109
A.2 Schematic description of the decomposition of the optimal weight vector into two orthogonal parts. 112
A.3 An output of the blocking matrix. 114
A.4 Example for the DSB's beam patterns of the ULA at different frequencies. 114
A.5 Fixed beamforming with suppressed sidelobes. 116
A.6 Adaptive beamforming with suppressed sidelobes. 118
A.7 Beam patterns at some frequencies: (left) upper path beam pattern, FBSS and SDM, (right) lower path beam pattern. 120
A.8 The broadband beam pattern of the upper path. 121
A.9 The broadband beam pattern of the lower path. 121
A.10 The broadband beam pattern of the FBSS. 122
A.11 WNGs and DFs over frequency. 122
A.12 Comparison between the GSC and ABSS. 123
A.13 Comparison between the GSC and ABSS from 362.5 ms to 427.5 ms. . . 124
A.14 SEs over SIR for the ABSS, FBSS and GSC, two interferences at 90^o and 150^o (a null position). 124
A.15 SEs over SIR for the ABSS, FBSS and GSC, two interferences at 90^o and 180^o. 125

B.1 Power Spectra of different methods: MUSIC, AFS, AFM and the true DOAs. 134
B.2 Power Spectra of different methods (SNR = 20 dB, N = 10): MUSIC, AFS, AFM and the true DOAs (on-grid). 134
B.3 RMSE over SNR of MUSIC and AFM, $N = 10$, 1000 Monte Carlo trials. 135
B.4 Power Spectra of different methods (SNR = 20 dB, N = 3): MUSIC, annihilating filter for single frame, annihilating filter for multiple frames and the true DOAs (off-grid). 135
B.5 Power Spectra of different methods (SNR = 20 dB contains the white noise and diffuse noise): MUSIC, extended MUSIC for diffuse noise, and the true DOAs are presented in the top figure, the blow is for the AFM. 136

List of Abbreviations

ABSS	**A**daptive **B**eamforming with **S**uppressed **S**idelobe
AF	**A**nnihilating **F**ilter
ASP	**A**rray **S**ignal **P**rocessing
BCS	**B**ayesian **C**ompressive **S**ensing
BP	**B**eamPattern
BPE	**B**eamPattern **E**rror
BUA	**B**ig **U**niform **A**rray
CoSaMP	**C**ompressive **Sa**mpling **M**atching **P**ursuit
CS	**C**ompressed **S**ensing
DF	**D**irectivity **F**actor
DMA	**D**ifferential **M**icrophone **A**rray
DOA	**D**irection **O**f **A**rrival
DSB	**D**elay and **S**um **B**eamforming
DUA	**D**ense and **U**niform **A**rray
ESPRIT	**E**stimation of **S**ignal **P**arameters via **R**otational **I**nvariant **T**echniques
FBR	**F**ront-to-**B**ack **R**atio
FBSS	**F**ixed **B**eamforming with **S**uppressed **S**idelobe
FI	**F**requency-**I**ndepedent
FIR	**F**inite **I**mpulse **R**esponse
GSC	**G**eneralized **S**idelobe **C**anceler
IHT	**I**terative **H**ard **T**hreshold
LMS	**L**east **M**ean **S**quare
MEMS	**M**icro**E**lectrical-**M**echanical **S**ystem
MV	**M**inimum **V**ariance
MVDR	**M**inimum **V**ariance **D**istortionless **R**esponse
MUSIC	**MU**ltiple **SI**gnal **C**lassification
LCMV	**L**inearly **C**onstrained **M**inimum **V**ariance
OMP	**O**rthogonal **M**atching **P**ursuit
PDF	**P**robability **D**ensity **F**unction
PCM	**P**ulse **C**ode **M**odulation
RADAR	**RA**dio **D**etection **A**nd **R**anging
RIC	**R**estricted **I**sometry **C**onstant
RIP	**R**estricted **I**sometry **P**roperties
RMSE	**R**oot **M**ean **S**quare **E**rror
SA	**S**parse **A**rray
SD	**S**uper**D**irective beamforming
SDM	**S**uper**D**irective with **M**ultiple Constraints
SE	**S**ignal's **E**rror
SONAR	**SO**und **NA**vigation and **R**anging
SIR	**S**ignal-to-**I**nterference **R**atio
SNR	**S**ignal-to-**N**oise **R**atio
SOI	**S**ignal **O**f **I**nterest
SP	**S**ubspace **P**ursuit

SSV	**S**ound **S**ource **V**ector
STFT	**S**hort-**T**ime **F**ourier **T**ransform
SUA	**S**mall **U**niform **A**rray
WNG	**W**hite **N**oise **G**ain
ULA	**U**niform **L**inear **A**rray

List of Notations

$(.)^T$	transpose
$(.)^*$	conjugate
$(.)^H$	conjugate transpose
j	imaginary unit
$*$	convolution
$[\mathbf{A}]_{i,j}$	(i,j) entry of $\mathbf{A}$
$\mathbf{A}(i,:)$	row i^{th} of $\mathbf{A}$
$\mathbf{A}(:,i)$	column i^{th} of $\mathbf{A}$
$\text{diag}(\mathbf{A})$	a column vector formed from the main diagonal of $\mathbf{A}$
$\text{mean}(\mathbf{v})$	mean value of the elements in vector $\mathbf{v}$
$\|.\|_1$	l_1-norm
$\|.\|_2$	l_2-norm
$\|.\|$	absolute operator of number or cardinality of set
$\mathbb{C}$	set of complex numbers
$\lfloor . \rfloor$	the floor function

Chapter 1

Introduction to the Thesis Topic

1.1 Background and Motivation

A microphone is a transducer that converts sound pressure to an electrical signal. Microphones are used in many applications such as telephones, hearing aids, consumer devices, video conference rooms, radio, television broadcasting and many others. Normally, the electrical signal from a microphone includes additive noise such as electrical noise, mechanical imprecise, interference, acoustic environment which are disturbing in most applications. There is a strong motivation to improve the quality of the signal, also be flexible to select a subset of the signal such as the frequency of interest, the direction of interest. One of the most promising approaches is to use a microphone array that contains multiple microphones to measure the spatial information of sound pressure. If the microphone array, where the temporal and spatial information of the sound pressure are sampled together, equips with wise algorithms, it could perform many interesting functions which a single microphone could not, such as beamforming, sound source localization, sound source separation, sound field reconstruction, or suppressing different kind of ambient noises.

Among several classes of algorithms, beamforming is widely used in practical applications and it is also attracted a lot of interest in the literature. The topic of beamforming is not restricted to audio applications, it is also a key technology in other applications such as radar, sonar, medical machines, communication, radioastronomy, etc. For microphone arrays, conventional beamforming performs spatial filtering to preserve the desired signal while suppressing interfering signals and noise arriving from directions other than the directions of interest. For speaker arrays, the beamforming is applied to produce a sound field at certain area satisfying several characteristics such as personal sound zone, sound pressure equalization and active noise cancellation to list a few. Moreover, beamforming is indispensable for the sound source localization of moving objects, flying aircraft, high-speed trains, motor cars in motion, open rotors like helicopter and wind turbine rotors [Mic06]. However, the design and implementation of microphone/speaker arrays with beamforming algorithms is not a trivial task because a beamformer tends to have a frequency-dependent response, while many applications need to deal with broadband signals.

Back to history, one of the earliest designs of the sensor/source array, which is well-known, is the phased array. The first concept of phased array was known during World War I and they were developed as radio detection and ranging (RADAR) antennas in World War II [Dob68]. A phased array is a low cost array that only uses the phase shifting at each antenna/sensor to form a directivity pattern. Later, they are extensively used in medical imaging with ultrasound. After World War II, phased arrays of hydrophones were used for improving sound navigation and

ranging (SONAR) for the localization of submarines, that is, sound waves are emitted and the echoes are evaluated. Besides, passive sensors are also used by submarines where large line arrays for low frequencies are towed to observe the sound field silently. The principle of underwater sound was summarized by Urick (1983) [Uri83], where he described the traditional techniques and their applications. Further applications are later used in radio-astronomy, such as the electromagnetic waves emitted by celestial sources in the sky are collected by antenna arrays [RTMSJ17; Wia+09; CMW14]. Recently, the applications of the array are extended to the civilian areas such as communication, seismology, smart devices, medical devices and entertainment [VT04].

Early beamforming techniques were developed under the assumption that the channel effect can be modeled by a delay and attenuation only. In actual room acoustics, however, the propagation process is much more complex. Indeed, the propagating signals undergo several reflections before impinging on the microphones [SBA10]. Moreover, in many cases of application, the signal is broadband, e.g. speech and audio signals or sonar and underwater acoustic communication signals. The interest in broadband signals amplifies the complexity of beamforming design, because the characteristics of a beamformer are dependent on the frequency and sometimes they are contradictory each other. Today, in contrast with the broadband's systematic problems, broadband beamforming gains a substantial attention due to their wide applications [VT04]. In early studies, it is common to use dense and large arrays with a large number of sensors to obtain some essential characteristics together, such as frequency-independence, high white noise gain, high directivity factor and high front-to-back ratio, etc.

Recently, adaptive beamforming approaches have been used to deal with interference. From observed data, the weights can be computed adaptively. The most well-known adaptive beamforming technique is the minimum variance distortionless response (MVDR) beamformer [Cap69], whose objective is to minimize the variance of the array output, while assuring the array response to the direction of interest to be distortion-less. Over the years, the extension of MVDR has been studied, that is the so-called linearly constrained minimize variance (LCMV). In LCMV, more constraints are inserted into the optimization problem to increase either the robustness or the flexibility of the beamformer. Normally, both MDVR and LCMV are modeled as a quadratic problem which uses multiple observed frames to estimate the covariance matrix of input signals, then a closed form expression for the solution is derived.

Regarding the frequency-independence in acoustic applications, an attractive concept is differential microphone arrays (DMAs) which refer to arrays that combine closely spaced sensors to respond to the spatial derivatives of the acoustic pressure field. The general principle of DMAs can be traced back to the 1930s when the directional Ribbon microphones were invented [Ols32; Ols46]. Since the sensor spacing of the DMA is much smaller than the acoustic wavelength, then the DMA is small in size which can be easily mounted into small devices. On the other hand, based on short-time Fourier transform, spatial filtering is applied to form a differential beamformer in each subband [TE04; BCC15; ZBC14; CPC15; Hua+20]. The order of differential beamformer could be designed by selecting the number of null-constraints, and the type of differential beamformer could also be obtained by assigning the nulls' position and/or changing the objective function of the optimization problem such as maximum front-to-back ratio for supercardioid beam pattern, maximum directivity index for hypercardioid beam pattern, etc. However, the conventional DMAs or arrays with differential beamformer are still sensitive to

the white noise and array mismatch at low frequencies, the directivity factor is degraded at high frequencies, the higher order systems are somewhat impractical and their beam pattern is restricted to the function of differential beamformer [Elk00].

Recently, the interest in sparse arrays is growing, mainly due to the capacity to reduce the number of sensors but still assure an acceptable performance. Nevertheless, in general, finding a suitable sparse array layout is still a challenging task. Many studies have focused on the convex or stochastic optimization approaches to determine the sparse array deployments. Despite the success of these techniques, the results depend heavily on the strength of the optimization algorithms and it is not guaranteed that the obtained solution is close to the optimal solution. For example, based on the compressed sensing framework, applying the l_1-norm is efficient only if the measurement system (transformation matrix) has a low restricted isometry constant [Boc+15; Don06], but the restricted isometry constant itself is not easy to verify, and it is still an open question in the sparse array design's model. Regarding the stochastic approach, it is inefficient for a large search-space system. More specifically, it does not ensure that its solution is close to the optimal solution if the number of sensors is large. Furthermore, in some circumstances, the optimization or stochastic approaches are not only difficult to tune the parameters, but it is also computationally intensive to solve them with subject to sparse solutions.

1.2 Main Contributions

In light of the drawbacks of the state-of-the-art sparse array designs, this research mainly focuses on the design of sparse arrays with broadband beamforming. Two new approaches are proposed, that is, one for data-independent broadband beamforming and another for adaptive beamforming (data-independent and adaptive beamforming concept were explained in the Abstract of this dissertation). We can briefly summarize the contributions of our work on this topic as follows.

1. For data-independent beamforming, the research proposed a new method to design a sparse array that takes a desired beam pattern and a dense and uniform array (DUA) as the inputs. The new method uses the clustering, classification algorithm in combination with optimization techniques to seek the sensor positions as well as their coefficients. We claim that this method not only outperforms other state-of-the-art methods in some important aspects of beamforming, but it is also efficient in computation time.

2. For adaptive beamforming, we proposed a method to design irregular microphone arrays for the broadband beamforming in conjunction with the compressed sensing framework. The random microphone array having maximum degree-of-freedom (number of distinct distances between microphones) is designed and we argued that this kind of deployment is well-suited to compressed sensing algorithms. First, we use the simulated annealing algorithm to seek a good array deployment, since we know that our optimization problem is an NP-hard problem. Next, we then use the orthogonal matching pursuit algorithm to reconstruct the sparse sound sources, and an interpolated DUA is constructed from the sparse sound sources. Finally, we perform the beamforming for the interpolated DUA. The results of the method are examined with simulations as well as experiments. In the experiments, we constructed a real microphone array and tested in a reverberant room.

3. Both proposed approaches in this research are based on a hypothesized DUA working well with the frequency-independent beam pattern. To do so, we newly proposed an effective method to design the DUA with broadband beamforming. The function inside the new method is named *inverse beam pattern* transform. This transformation transforms an arbitrary desired beam pattern to the sensor coefficients of the DUA. Based on the characteristics of the inverse beam pattern transform, we also proposed new bounds for spatial sampling from broadband beamforming perspective.

4. In addition, alternative approaches to the generalize sidelobe canceler beamformer are proposed to overcome the major drawback of generalize sidelobe canceler that is the subtracting of two different beam patterns. Although this contribution does not help to find the sparse array layout, it provides the methodologies to design a beamformer for the sparse array.

5. Lastly, the research also covers the direction-of-arrival (DOA) estimation topic which normally combines with beamforming algorithms in audio applications. We proposed an extended multiple signal classification (MUSIC) algorithm which takes diffuse noise into account. Also, we proposed the annihilating filter-based method for DOA estimation, that is, the annihilating filter design for multiple data frames. As obtained from simulations results, the proposed annihilating filter-based method outperforms the conventional MUSIC in terms of DOA's accuracy. Furthermore, the proposed method help to reduce the cost of computation compared to subspace-based approaches.

1.3 Organization

These remaining chapters of this dissertation are organized as follows.

Chapter 2 carries out a comprehensive literature review of beamforming, which covers the basic principles of beamforming and several representative beamforming algorithms.

Chapter 3 focuses on uniform array with broadband beamforming. This chapter introduces inverse beam pattern transform and spatial sampling for uniform array from broadband beamforming perspective. The inverse beam pattern transform is applied to the uniform array and it is the fundamental theory to design the sparse arrays in Chapter 4 and 5.

Chapter 4 focuses on data-independent beamforming. This chapter presents a new approach to design the sparse array with frequency-independent beam pattern. This approach is applicable for the linear array (both broadside and endfire configurations) as well as planar array. The proposed method is compared with the state-of-the-art sparse array designs and with the uniform arrays.

Chapter 5 focuses on adaptive beamforming. The sparse array proposed in this chapter aims to reconstruct the sound sources via the compressed sensing framework. From reconstructed sound sources, we propose a simple approach to interpolate a dense and uniform array, and beamforming is deployed in this interpolated dense and uniform array. We also describe the hardware construction of a real microphone array and conduct various experiments to examine the performances of the constructed microphone array in a reverberation room.

Chapter 6 presents some discussions and future works.

Chapter A focuses on both data-independent and adaptive beamforming. This chapter proposes two variants for the generalized sidelobe canceler beamformer. One is for data-independent, another is for adaptive beamforming.

Appendix B presents the direction-of-arrival (DOA) estimation. This appendix proposes a new method for DOA estimation based on the design of annihilating filter for multiple frames. Also, we propose an extension of the MUSIC algorithm that takes diffuse noise into account.

Appendixes C, D provide Matlab programs.

1.4 Published and Submitted Papers

- (Chapter. 4) Phan Le Son, "Using Uniform Microphone Arrays to Design Sparse Microphone Arrays with Frequency-Independent Beam Pattern," published in Deutsche Jahrestagung fur Akustik, DAGA 2020, pp. 126-129, March. 2020.
- (Chapters. 3 and 4) Phan Le Son, "On the design of sparse arrays with frequency-invariant beam pattern," published in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 226–238, 2021.
- (Appendix. A) Phan Le Son, "Alternative approaches to generalized sidelobe canceller beamformer," published in Deutsche Jahrestagung fur Akustik, DAGA 2021.
- (Chapter. 3) Phan Le Son, "Inverse Beam Pattern Transform and Spatial Sampling for Uniform Array from Broadband Beamforming Perspective," submitted in IEEE Transactions on Signal Processing, 2021.
- (Chapter. 5) Phan Le Son, "Irregular Microphone Array Design for Broadband Beamforming," accepted for revision in Elsevier Signal Processing, 2021.

Chapter 2

Scientific Background

In this chapter, the author briefly presents the background of beamforming and review some research methods which are related to this dissertation. The chapter starts with the model of sound propagation and basic principles of microphone. The chapter then introduces some important aspects of beamforming, such as beam pattern, frequency-independence, beam-width, steering, directivity factor, white noise gain, etc. Finally, the chapter explains algorithms of several beamforming techniques.

2.1 Fundamentals of Microphone Array

2.1.1 Sound Propagation

Sound waves propagate through fluids as longitudinal waves. The molecules in the fluid move back and forth in the direction of propagation, producing regions of compression and expansion. By using Newton's equations of motion to consider an infinitesimal volume of fluid, an equation governing the wave's propagation can be developed. A generalized wave equation for acoustic waves is quite complex as it depends upon properties of the fluid. However, assuming an idea fluid with zero viscosity and let $p(x, y, z, t)$ be an infinitesimal variation of acoustic pressure from equilibrium value which satisfies the acoustic wave equation [Wil99]

$$\nabla^2 p - \frac{1}{c^2}\frac{\partial^2 p}{\partial t^2} = 0 \tag{2.1}$$

where c is the sound speed depending upon the pressure and density of the fluid and ∇^2 is the Laplace operator

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}.$$

Let $\bar{p}(\omega)$ be a Fourier transform of $p(t)$ (for simplicity of notation, we omit the dependence of p and $\bar{p}$ on (x, y, z), that is

$$\bar{p}(\omega) = \mathcal{F}\{p(t)\} = \int_{-\infty}^{+\infty} p(t)e^{-j\omega t}dt$$

where ω is the angular frequency and $\mathcal{F}$ is the Fourier transform, then

$$p(t) = \int_{-\infty}^{+\infty} \bar{p}(\omega)e^{j\omega t}d\omega \tag{2.2}$$

Equation (2.2) can be differentiated with respect to time, that is

$$\frac{\partial^2 p}{\partial t^2} = -\int_{-\infty}^{+\infty} \omega^2 \bar{p}(\omega) e^{j\omega t} d\omega$$

Then

$$\mathcal{F}\{\frac{\partial^2 p}{\partial t^2}\} = \int_{-\infty}^{+\infty} \big(-\int_{-\infty}^{+\infty} \omega^2 \bar{p}(\omega) e^{j\omega t} d\omega \big) e^{-j\omega t} dt = -\omega^2 \bar{p}(\omega).$$

The Fourier transform of (2.1) is the Helmholtz equation

$$\nabla^2 \bar{p} + k^2 \bar{p} = 0 \tag{2.3}$$

where the acoustic wavenumber is $k = \omega / c$.

Far Field

If the distance from sound source is much farther than the distance between two examined points, then the far field assumption is applied. For the far field assumption, the signal is modeled as the plane wave. The general solution of the Helmholtz equation (2.3) is

$$\bar{p}(\omega) = \frac{A(\omega)}{4\pi r_0} e^{j(k_x x + k_y y + k_z z)} \tag{2.4}$$

where $A(\omega)$ is an constant presenting for the strength and phase of a sound source at rotation frequency ω, $k_x = k \cos\phi \sin\theta$, $k_y = k \sin\phi \sin\theta$, $k_z = k \cos\theta$ (ϕ, θ is the azimuth and elevation angle in the spherical coordinate, respectively) and r_0 is the radial distance from a source to the sensor (r_0 is much bigger than the distance between sensors in the array and r_0 is approximately constant for all sensors in the array). This yields the final result, a plane wave at a rotation frequency

$$p(x, y, z, t) = \frac{A(\omega)}{4\pi r_0} e^{j(k_x x + k_y y + k_z z - \omega t)} \tag{2.5}$$

Near Field

The solution for a spherical wave can be derived as [Zio20]

$$p(x, y, z, t) = \frac{A}{4\pi r} e^{j(rk - \omega t)} \tag{2.6}$$

where $r = \sqrt{x^2 + y^2 + z^2}$ is the radial distance from a source at $(0, 0, 0)$ to (x, y, z). The spherical wave solution shows that the signal amplitude decays at a rate proportional to the distance from the source.

Superposition Property

Due to the linearity of the wave equation, the monochromatic solution can be expanded to the more general polychromatic case by considering the solution as a sum or integral of such complex exponentials. In other words, the solution in (2.5) and (2.6) are for a stimulating source, if multiple sources stimulate simultaneously, then sum of single solutions is the solution. In array processing, this principle is one

of fundamental principles. Example for far field sources,

$$p(t) = \sum_{i=1}^{N} \frac{A}{4\pi r_0} e^{j(\vec{k}_i \cdot \vec{r}_i - \omega t)} \tag{2.7}$$

where $\vec{k}_i = (k_{xi}, k_{yi}, k_{zi})$ give the direction of stimulating source i, $\vec{r}_i = (x_i, y_i, z_i)$ is the position vector, N is the number of stimulating sources and "." is the dot product.

2.1.2 Microphone

A microphone is a device that converts sound pressure into an electrical signal. Different types of microphones are used today, which employ different methods to convert the air pressure variations of a sound wave to an electrical signal, such as condenser microphone, dynamic microphone, contact microphone, MEMS microphones, .etc.

Condenser Microphone

A condenser microphone is also called a capacitor microphone or electrostatic microphone, which uses the vibrating diaphragm as a capacitor plate. The sound hits the diaphragm, then the value of capacitor is changed. This changing can be recognized by the electronic circuits. Fig. 2.1 [1] plots a construction of a condenser microphone.

FIGURE 2.1: Condenser microphone.

Dynamic Microphone

The dynamic microphone is also known as the moving-coil microphone. It uses the inverse mechanical principle with a loudspeaker where an electrical conductor moves in a magnetic field, an electric current is induced. Fig. 2.2 [2] plots a construction of dynamic microphone.

[1] from https://www.neumann.com/homestudio/
[2] from https://www.neumann.com/homestudio/

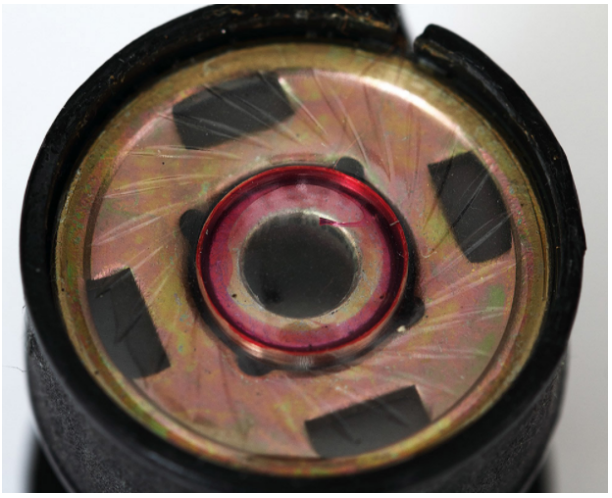


FIGURE 2.2: Dynamic microphone.

MEMS Microphone

A MEMS microphone is also called a microphone chip or silicon microphone. Normally, it is a variant of inexpensive condenser microphone (polarized condenser microphone) where the pressure-sensitive diaphragm is etched directly into a silicon wafer [Pfl17]. Fig. 2.3 plots a construction of MEMS microphone.

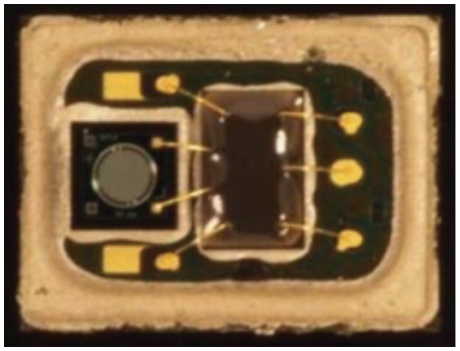


FIGURE 2.3: MEMS microphone [Pfl17].

2.2 Review of Beamforming Techniques

In this section, the thesis represents some representative techniques of the data-independent and adaptive beamforming. All the techniques are modeled in the Fourier domain, except the generalized sidelobe canceler that is presented in time-domain. The spatial filtering of beamforming is constructed in the short-time Fourier transform (STFT) domain, as illustrated in Fig. 2.4.

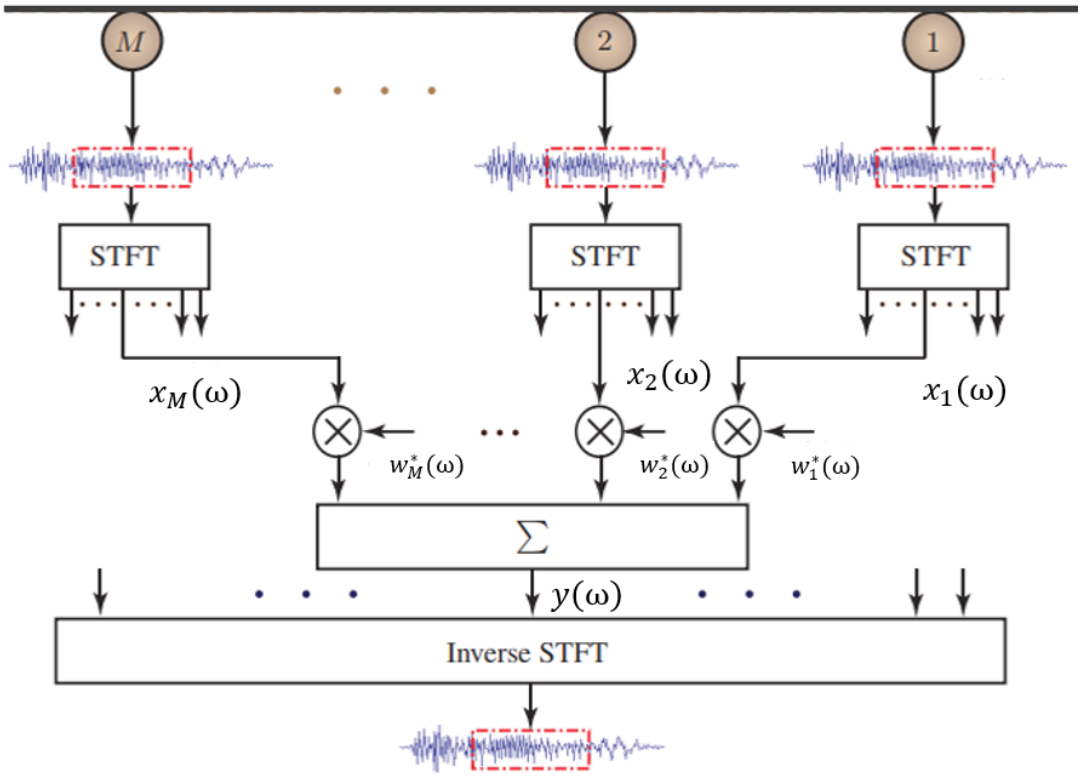


FIGURE 2.4: Spatial filtering of beamforming [BJ12]: Blue is the audio signal, a dash-red box is a frame of the STFT.

In general, we assume that both interference and ambient noise coexist with the target source. Consider an array with M microphones, in the frequency-domain, the array signal at angular frequency ω and frame index n is presented by $\mathbf{x}(\omega, n) = [x_1(\omega, n), \ldots, x_M(\omega, n)]^T$ and it can be decomposed as

$$\mathbf{x}(\omega, n) = s(\omega, n)\mathbf{d}_s(\omega) + \mathbf{i}(\omega, n) + \mathbf{n}(\omega, n) \tag{2.8}$$

where $s(\omega, n)$ is the source of interest (SOI), $\mathbf{d}_s(\omega)$ is the steering vector of SOI, $\mathbf{i}(\omega, n)$ is the incident noise (such as interfering sources), and $\mathbf{n}(\omega, n)$ is the statistical noise (such as white noise, diffuse noise).

For the sake of conciseness, we omit the variable ω in the remainder of this chapter wherever possible. Then, the output of the beamformer can be written as

$$y(n) = \mathbf{w}^H \mathbf{x}(n) \tag{2.9}$$

where $\mathbf{w} = [w_1, \ldots, w_M]$ is the weight vector of the beamformer.

The target of beamforming is to recover the SOI $s(n)$ from the measurement signals $\mathbf{x}(n)$ by using the spatial filter $\mathbf{w}$. According to our signal model, the measure

signals in (2.8) contains two type of noises: Incident noise and statistical noise. For the statistical noise, we assume to know their statistical model in advance, e.g., normal distribution, Gausian distribution, Laplace distribution, etc. then we can find a data-independent weight vector to suppress that noise in the statistical sense. That is the data-independent beamforming, where the weight vector is fixed over the time. For the incident noises, we need to observe the measurement signals to estimate the noises at every frame, therefore if the beamforming wants to suppress the incident noises, its weight vector needs to adapt versus the data, that is the so-called adaptive beamforming.

2.2.1 Characteristics of Beamforming

In order to get a better understanding of the characteristic of beamforming, the chapter first introduces some important measures to analyze their performance.

Beam Pattern

That is the response function of the array to a wavefront coming from a specific angle at a specific frequency, depending on the azimuth ϕ and elevation θ in the spherical coordinate system [BS01]. The spatial-temporal transfer function is given by

$$\mathcal{B}(\phi, \theta, \omega) = \mathbf{w}(\omega)^H \mathbf{d}(\phi, \theta, \omega) \tag{2.10}$$

where $\mathbf{d}(\phi, \theta, \omega)$ is the steering vector at the azimuth ϕ and elevation θ. That is also called the beam pattern, which is usually presented on a logarithmic scale. Since the beam pattern depends on three variables, it is not possible to display in a single plot. It is often plotted in the polar coordinate (for 2-dimensional) or the spherical coordinate (for 3-dimensional) for a single frequency. An example of cardioid beam pattern is plotted in Fig. 2.5, where the look direction at 0^o degree and the null at 180^o degree.

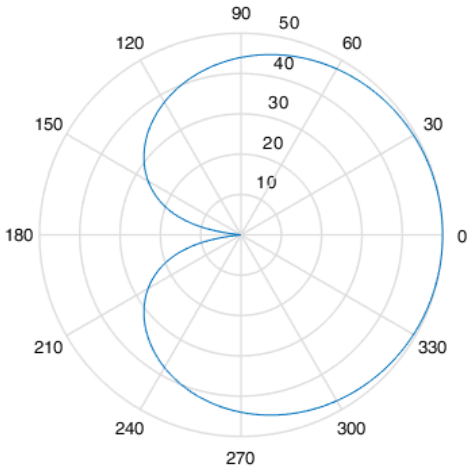


FIGURE 2.5: Cardioid beam pattern.

Beam-width

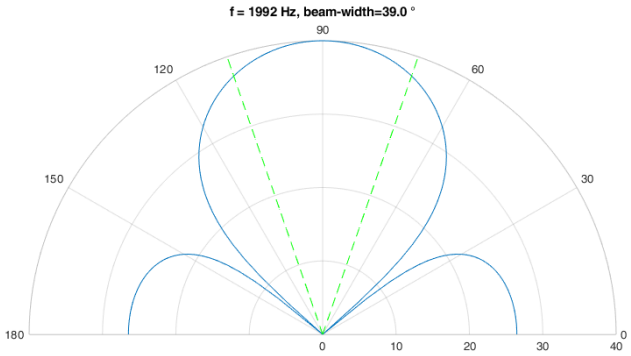


FIGURE 2.6: Beam-width of a beam pattern.

The beam-width (open angle) can be defined in many ways, we will use three dB criteria in this chapter, that is the continuous angle space around the look direction where the gains of beam pattern not less than 3 dB of the gain at look direction

$$BW(\omega) := \{(\phi,\theta) : |20\log_{10}\mathcal{B}(\phi,\theta,\omega)| \geq 20\log_{10}|\mathcal{B}(\phi_0,\theta_0,\omega)| - 3 \text{ and } (\phi,\theta) \in \mathcal{M}\} \tag{2.11}$$

where $\mathcal{M}$ is the main-lobe space and (ϕ_0,θ_0) are the angle of look direction. Fig. 2.6 illustrates the beam-width (open angle is 39^o) of a beam pattern, that is space covered by the dash green lines.

Besides the beam-width, other important characteristics of beam pattern are also illustrated in Fig. 2.7, such as mainlobe, sidelobes, nulls.

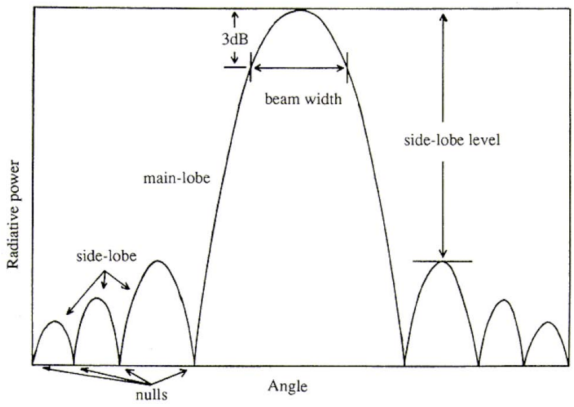
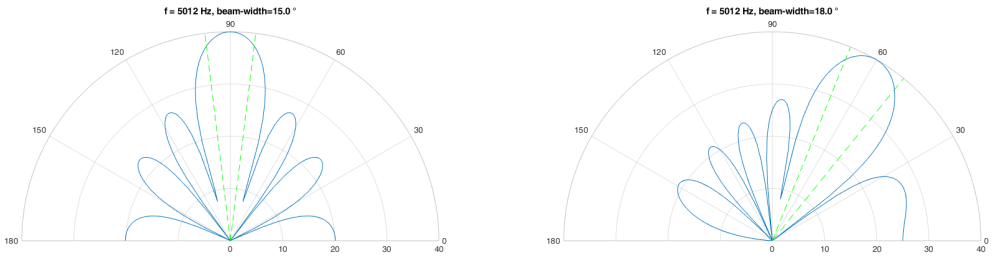


FIGURE 2.7: Null, mainlobe, sidelobe of beam pattern.

Steering

In general, the beamforming is possible to rotate toward a direction of interest by changing the beamforming coefficients. This is termed electronic steering. Although the electronic steering is the low cost solution, quick response and high flexible, the steering beam pattern is suffering more from the spatial aliasing effect. Due to this

drawback, it mainly applies for the linear array with broadside beamforming or for the planar array. Fig. 2.8 presents the beam patterns for different steering angles.



(A) 90^o. (B) 60^o.

FIGURE 2.8: Electronic steering ability of a linear array.

Directivity Factor

A common quantity to evaluate the beam pattern is the directivity factor (DF), it measures the ability to preserve the source of interest while suppressing the signal coming from other directions.

$$\mathcal{D}(\mathbf{w}(\omega)) = \frac{\left|\mathbf{w}(\omega)^H \mathbf{d}_s(\omega)\right|^2}{\mathbf{w}(\omega)^H \mathbf{\Gamma}(\omega) \mathbf{w}(\omega)^H} \tag{2.12}$$

where $\mathbf{\Gamma}(\omega)$ is the pseudo-coherence matrix of the diffuse noise field,

$$\left[\mathbf{\Gamma}\left(\omega\right)\right]_{i,j} = \operatorname{sinc}\left(\frac{\omega d_{ij}}{c}\right)$$

where d_{ij} is the distance between sensor i and sensor j.

White Noise Gain

The white noise gain (WNG) shows the ability of the array to suppress the incoherence noise, such as self-noise, array imperfection, etc. That is given by [BS01],

$$\mathcal{W}\left(\mathbf{w}(\omega)\right) = \frac{\left|\mathbf{w}^H\left(\omega\right) \mathbf{d}_s(\omega)\right|^2}{\mathbf{w}^H\left(\omega\right) \mathbf{w}\left(\omega\right)}. \tag{2.13}$$

Front-to-Back Ratio

In many applications, DF is not the best index to evaluate the array, such as video conferences or the recordings of orchestras. In such applications, a front-to-back ratio (FBR) is a better choice, since in most cases all desired sources are in front of the array and all unwanted sources are behind the array [BS01; Elk00; MH41],

$$\mathcal{F}\left(\omega, \phi_0 = 0, \theta_0 = 0\right) = \frac{\int_0^{\pi}\int_0^{\pi} |\mathcal{B}(\phi,\theta,\omega)|^2 \sin\theta d\phi d\theta}{\int_{\pi}^{2\pi}\int_0^{\pi} |\mathcal{B}(\phi,\theta,\omega)|^2 \sin\theta d\phi d\theta} \tag{2.14}$$

or

$$\mathcal{F}\left(\omega, \phi_0, \theta_0\right) = \frac{\mathbf{w}^H \mathbf{\Gamma}_f \mathbf{w}}{\mathbf{w}^H \mathbf{\Gamma}_b \mathbf{w}}. \tag{2.15}$$

where

$$\mathbf{\Gamma}_f = \int_0^{\pi} \int_0^{\pi} \mathbf{d}(\phi, \theta, \omega)\mathbf{d}^H(\phi, \theta, \omega) \sin\theta d\phi d\theta$$
$$\mathbf{\Gamma}_b = \int_{\pi}^{2\pi} \int_0^{\pi} \mathbf{d}(\phi, \theta, \omega)\mathbf{d}^H(\phi, \theta, \omega) \sin\theta d\phi d\theta.$$

2.2.2 Data-independent Beamforming

For instance, the noise fields are stationary and well-defined, the beamforming coefficients can be presented by a closed-form formula whose inputs are the noise model, array layout and look direction.

Delay and Sum Beamforming

Delay and sum beamforming (DSB) is the simplest beamforming technique where the signal of sensors are delayed to align in phase and then be summed. That is

$$\mathbf{w} = \frac{\mathbf{d}_s}{M}.$$

Fig. 2.10 plots the DSB beam patterns of an uniform linear array (ULA) in Fig. 2.9, which contains nine microphones having the inter-distance of 0.025 m.

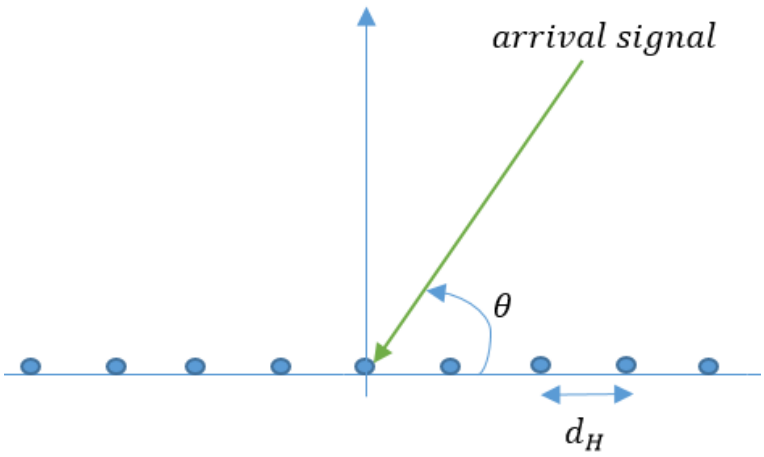


FIGURE 2.9: Uniform linear array configuration.

The weight of DSB is actually the optimal beamforming for optimizing the WNG where the amplitude of the coefficients are equal. The DSB is a well-known method since it is simple and robust. However, its directivity is not optimal. The term directivity describes the ability of a beamformer to suppress noise coming from all directions other than the source of interest. Therefore, for diffuse noise, the DSB's performance is degraded at low frequencies.

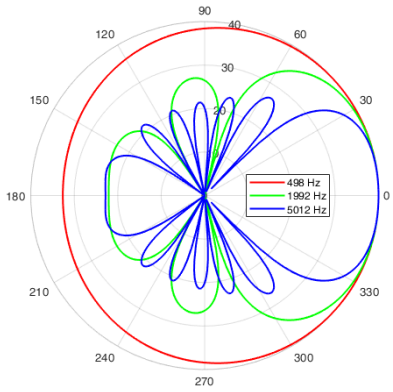


FIGURE 2.10: DSB Beam patterns of the ULA in Fig. 2.9 at different frequencies.

Superdirective Beamforming

Superdirective beamforming (SD) [Elk00] is the optimal beamforming for optimizing the directivity, that is

$$\begin{aligned} &\underset{\mathbf{w}}{\text{minimize}} \quad \mathbf{w}^H \boldsymbol{\Gamma} \mathbf{w} \\ &\text{subject to} \\ &\quad \mathbf{w}^H \mathbf{d}_s = 1 \end{aligned} \tag{2.16}$$

where $\mathbf{w}^H \boldsymbol{\Gamma} \mathbf{w}$ is the diffuse noise energy and $\mathbf{w}^H \mathbf{d}_s = 1$ is distortion-less constraint at the look direction. By using the Langrange-multiplier [Fro72], the closed-form solution of (2.16) is

$$\mathbf{w} = \frac{\boldsymbol{\Gamma}^{-1} \mathbf{d}_s}{\mathbf{d}_s^H \boldsymbol{\Gamma}^{-1} \mathbf{d}_s}. \tag{2.17}$$

However, at low frequency, the SD beamforming is worse at WNG performance since the amplitude of coefficients are very large at some sensors. In practical applications, the noise signals normally contain the white noise as well as diffuse noise, therefore a beamforming compromising between WNG and DF often be used by introducing a parameter μ that is adjusted over frequency range to control the WNG, DF within the acceptable ranges. Gilbert and Morgan [GM55] suggested that μ should be small

$$\mathbf{w} = \frac{(\boldsymbol{\Gamma} + \mu \mathbf{I})^{-1} \mathbf{d}_s}{\mathbf{d}_s^H (\boldsymbol{\Gamma} + \mu \mathbf{I})^{-1} \mathbf{d}_s}. \tag{2.18}$$

Fig. 2.11 plots the SD beam patterns ($\mu = 0.01$) of an ULA with nine microphones having the inter-distance of 0.025 m.

Since the problems in this section contains only one constraint (distortion-less at the look direction), they belong to the minimum variance distortionless response (MVDR) beamforming class.

Differential Beamforming

Differential microphone arrays (DMAs) refer to the arrays that are responsive to the spatial derivatives of the acoustic pressure field [TE01]. As shown in Fig. 2.12, the third-order DMA is implemented by three stages of delay and subtraction. If the number of sensors is $M = 4$, the first stage yields three first-order DMA, the second

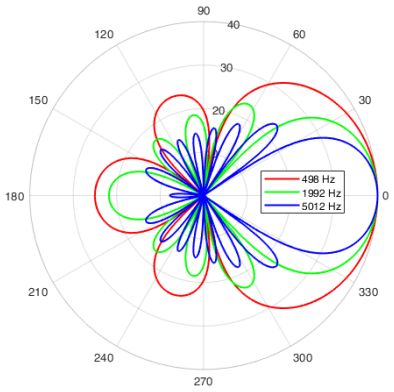


FIGURE 2.11: SD Beam patterns of the ULA in Fig. 2.9 at different frequencies.

stage yields two second-order DMA and the last stage yields a third-order DMA. The higher-order DMA is normally proportional to the higher directivity factor and the delays at a stage decide the nulls' position of beam patterns. Since the position of nulls are invariant with frequency, the beam pattern of DMA is almost frequency-independent. Of course, spatial aliasing may produce some more nulls, it leads to degradation of the frequency-independent beam pattern at high frequencies. To reduce the spatial aliasing effect, the inter-distance of sensors in DMA should be small, a small inter-distance also mitigates the amplitude differences of measurement positions.

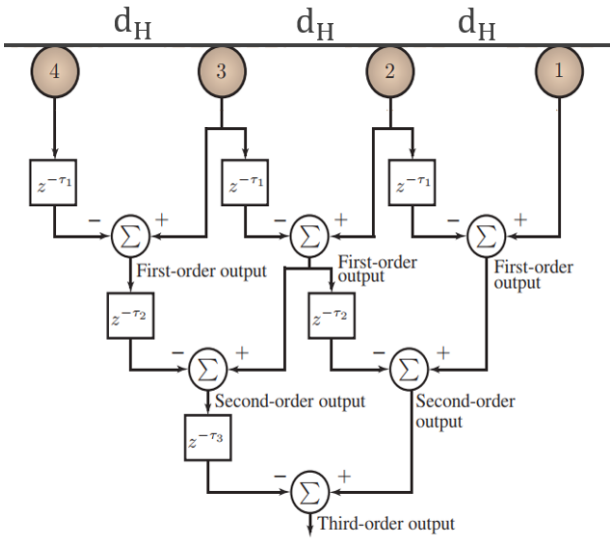


FIGURE 2.12: The implementation of third-order DMA in time-domain [BJ12].

In Fig. 2.13, we design the conventional DMA with $M = 4$ microphones, $d_H = 0.01$ m, the look direction at 0^o and the nulls at $60^o, 120^o, 180^o$.

The conventional DMA deals with the uniform linear array (ULA) in the time-domain. Later, they extended the method of DMA to the Fourier domain, which increase the flexible of DMA [BJ12], e.g., the DMA designs for the uniform circle

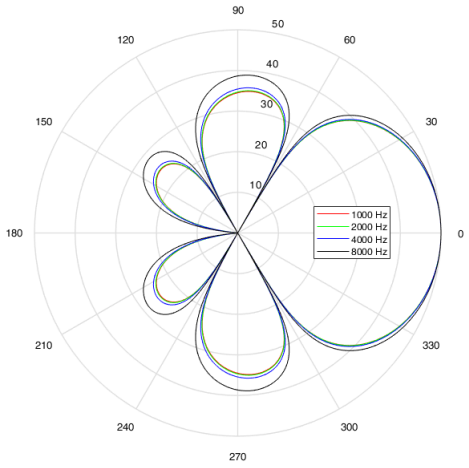


FIGURE 2.13: Beam patterns of the third-order DMA.

array, the DMA with maximizing the DF, maximizing the FBR, or the DMA with fractional order, etc.

Herein, we only consider the ULA and the processing ULA in Fourier domain with short-time Fourier transform (STFT). The steering vector of ULA at the incident angle θ is

$$\mathbf{d}(\theta) = [1, e^{-j\omega d_H \cos\theta/c}, \ldots, e^{-j\omega(M-1)d_H \cos\theta/c}]^T$$

where d_H is the inter-distance of sensors and c is the speed of sound. As mentioned earlier, it is assumed that the inter-distance is much less than the wavelength or $d_H << \frac{2\pi c}{\omega}$ [EWT03].

The function of beam pattern is

$$\mathcal{B}(\theta) = \mathbf{w}^H \mathbf{d} = \sum_{m=1}^{M} w(i) e^{-j\omega(m-1)d_H \cos\theta/c}. \tag{2.19}$$

The frequency-independent beam pattern of an Nth-order DMA is defined as [Elk00]

$$\mathcal{B}_N(\theta) = \sum_{n=0}^{N} a_{N,n} \cos^n \theta \tag{2.20}$$

where $a_{N,n},\ n = 0, 1, \ldots, N$ are real coefficients. The different values of these coefficients determine the different directional patterns of the Nth-order DMA. In look direction, e.g. $\theta = 0^o$, the array gain is 1, that is $\mathcal{B}_N(0^o) = 1$, then

$$\sum_{n=0}^{N} a_{N,n} = 1.$$

Similarly for the nulls of beam pattern, we have

$$\sum_{n=0}^{N} a_{N,n} \cos^n \theta_i = 0$$

where θ_i is the null angle of beam pattern. To achieve $N+1$ coefficients of Nth-order DMA, the beam pattern has at most N nulls, that can be presented by the product of a Vandermonde matrix and coefficient vector as

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & \cos^1\theta_1 & \dots & \cos^N\theta_1 \\ \dots & \dots & \dots & \dots \\ 1 & \cos^1\theta_N & \dots & \cos^N\theta_N \end{bmatrix} \begin{bmatrix} a_{N,0} \\ a_{N,1} \\ \dots \\ a_{N,N} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \dots \\ 0 \end{bmatrix} \tag{2.21}$$

where $\theta_1, \dots, \theta_N$ are the null directions. (2.21) always has a unique solution if the null directions are distinct ($\theta_i \neq \theta_j, \forall i \neq j$) [Tur66]. Similarly, if number of sensors is equal to the order of DMA plus one $M = N+1$, we can find $\mathbf{w}(\omega)$ at every frequency of interest. That allows us to design the DMA in the Fourier-domain by STFT.

More importantly, if the number of sensors is greater than the order of DMA plus one $M > N+1$, there is room to find the optimum $\mathbf{w}(\omega)$ towards different beamformers' characteristics, such as maximizing the WNG, DF or FBR, while the beam pattern still assures the Nth order DMA. For examples, the hypercardioid DMA is achieved by maximizing the DF in (2.16) with subject to the distortion-less and nulls, that is

$$\mathbf{w} = \frac{\mathbf{\Gamma}^{-1}\mathbf{D}}{\mathbf{D}^H\mathbf{\Gamma}^{-1}\mathbf{D}}\mathbf{i} \tag{2.22}$$

where $\mathbf{D} = [\mathbf{d}(0^o), \mathbf{d}(\theta_1), \dots, \mathbf{d}(\theta_N)]$ is the matrix size of $M \times (N+1)$ and $\mathbf{i} = [1, 0, \dots, 0]^T$ is a vector size of $N+1$.

For example, a similar differential beamforming with conventional third-order DMA designed via STFT for ULA with $M = 7$ microphones and $d_H = 0.01$ m is illustrated in Fig. 2.14.

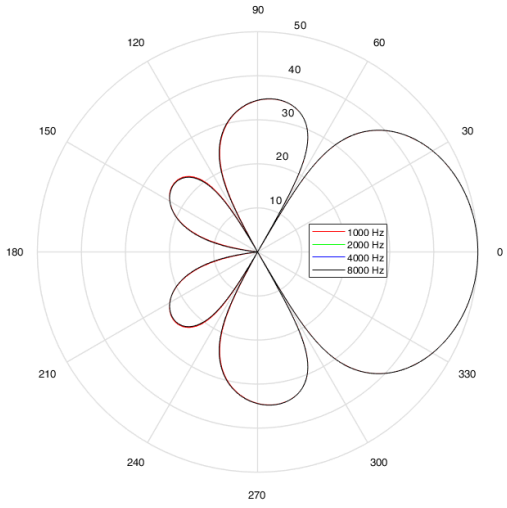


FIGURE 2.14: Beam patterns of the third-order DMA are designed in Fourier domain.

2.2.3 Adaptive Beamforming

In this section, we present representative methods of adaptive beamforming.

Linearly Constrained Adaptive Beamforming

The most well-known adaptive beamforming techniques are the minimum variance distortionless response (MVDR) and linearly constrained minimum variance (LCMV). MVDR minimizes the variance of the output while assuring the distortionless at a look direction. LCMV is the extension of MVDR where more constraints are added to the optimization problem.

Back to history, a classical adaptive beamforming technique is the Capon beamformer [Cap69], which belongs to the MVDR beamforming class where the covariance matrix is updated online. The following optimization problem could present for the Capon beamformer

$$\begin{aligned} &\underset{\mathbf{w}}{\text{minimize}} \quad E[|\mathbf{w}^H\mathbf{x}|^2] \\ &\text{subject to} \\ &\quad \mathbf{w}^H\mathbf{d}_s = 1 \end{aligned} \tag{2.23}$$

where $E[.]$ is the expectation operation. A solution of (2.23) for the optimal weighting vector is given by

$$\mathbf{w} = \frac{\mathbf{R}_x^{-1}\mathbf{d}_S}{\mathbf{d}_s^H\mathbf{R}_x^{-1}\mathbf{d}_s} \tag{2.24}$$

where $\mathbf{R}_x$ is the covariance matrix of the received signals, that matrix can be estimated from multiple snapshots (data frames). In Fig. 2.15, the beam pattern of linear array with $M = 9$ is plotted, which focuses on the the broadside direction and the interferences present at 45^o and 180^0. It is clearly seen that distortion-less property at the look direction is assured, while nulls are automatically inserted at the directions of interference. In Fig. 2.16, the MVDR beamforming for frequency range from

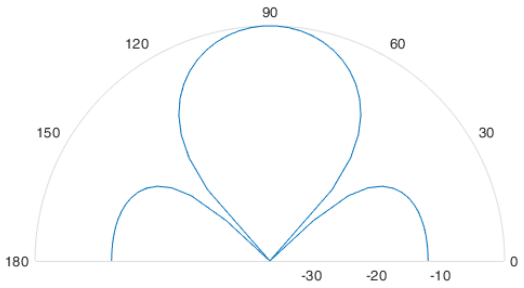


FIGURE 2.15: MVDR beamforming at 1.6 kHz focuses on the broadside direction and the interference presents at 45^o and 135^o.

1 kHz to 7 kHz is plotted. Although the beam pattern is not frequency-independent, the distortion-less property and nulls of the beam pattern are always ensured.

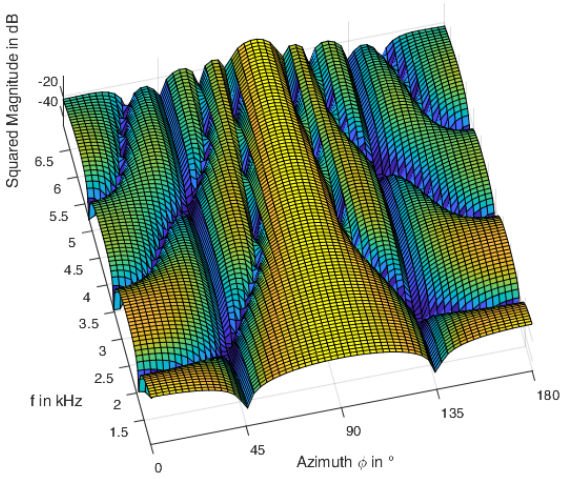


FIGURE 2.16: MVDR beamforming with broadband focus on the broadside direction and interference present at 45^o and 135^o.

A problem of the MVDR beamforming is that it is sensitive to steering vector mismatch or it needs to know the exact look direction in advance, which is somewhat impractical. In order to improve the robustness, some other robust beamforming techniques are proposed such as LCMV beamformer, general steering vector mismatch, etc. The common idea is to construct the uncertain steering vector within a specific bound as the additional constraints and combining with the minimum variance (MV) or minimum dispersion (MD) criterion, which can be applied for Gaussian signals as well as non-Gaussian signals [Vor13].

The objective function and target of MD criterion are presented by

$$\underset{\mathbf{w}}{\text{minimize}} \quad E[|\mathbf{w}^H\mathbf{x}|^p] \tag{2.25}$$

where $E[|\mathbf{w}^H\mathbf{x}|^p]$ is the dispersion. The parameter-p is presented for the shape of the probability density function (PDF) of the signal, e.g., $p = 1$ and $p = 2$ are for Laplacian and Gaussian signal, respectively. It is interesting to explain the formula (2.25) from a statistic standpoint as follows. The PDF of the generalized Gaussian distribution of the output is given by [NAR09]

$$p_s(y(n)) = \frac{p}{2\sigma\Gamma(1/p)} e^{-(\frac{|y(n)-\mu|}{\sigma})^p} \tag{2.26}$$

where $\Gamma(.)$ is the gamma function that ensures (2.26) is correctly normalized, σ and μ are the standard deviation and expectation value, respectively, and p is the shape parameter. With beamforming, we hope to recover the SOI by $y(n) = \mathbf{w}^H\mathbf{x}(n) \approx s(n)$, then the PDF of the output should have a similar type with that of the SOI (decided by the shape parameter p). In other words, the output of beamforming $y(n)$ is robust against noises if its statistical model fits to the SOI's statistical model. Therefore, we need to find the weight vector that maximizes the likelihood function. Since the logarithm is a monotonic function, we would consider the logarithmic

likelihood function, which is given by [Bis06]

$$\mathcal{L}(\mathbf{w}) = N\log(p) - N\log(2\sigma\Gamma(1/p)) - \frac{1}{\sigma^p}\sum_{i=1}^{N}|\mathbf{w}^H\mathbf{x}(n) - \mu|^p, \tag{2.27}$$

where N is the number of snapshots (measurement frames). For the case $\mu = 0$, it is clearly seen that the maximized likelihood function is equivalent to minimize $E[|\mathbf{w}^H\mathbf{x}|^p]$, that explains the expression in (2.25).

Here, we present an example of broadband beamforming design via LCMV. The simulation conducts for a ULA with $N = 8$ sensors, with the inter-distance of sensor of 3 cm, a look direction of 0^o, and a frequency band from 400 Hz to 3200 Hz with 8 kHz sampling frequency. The broadband beamforming of LCMV is plotted in Fig. 2.17.

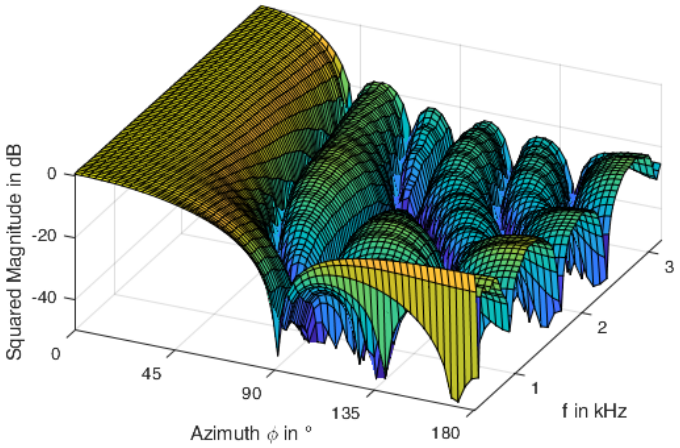


FIGURE 2.17: Beam pattern with LCMV over frequency.

Generalized Sidelobe Canceler Beamformer

For broandband signals, an alternative approach to LCMV beamforming processing the signal in time-domain was proposed by Griffith and Jim [GJ82], that is the generalized sidelobe canceler (GSC) where the beamforming can be decomposed into two orthogonal paths, as plotted in Fig. 2.18. The upper path is the conventional beamformer consisting of a set of fixed beamformer coefficients aims to control the main-lobe. The lower path is the side-lobe canceling path, which uses the combination of a blocking matrix and adaptive FIR filters to reproduce the sidelobes of the upper path. The output signal is the subtraction of signals between the upper path and the lower path.

From the LCMV perspective, one part represents the distortion-less constraint and the other part represents the minimum noise power. The original GSC structure is connected with Frost's procedure [Fro72] (constrained least-mean-square algorithm), that is, the DSB is applied for the upper path, meanwhile, the input signal is projected onto the noise subspace by multiplying with an orthogonal blocking matrix $\mathbf{B}$. Although a similar result is obtained, the optimization problem of GSC is reduced to the unconstrained problem which is simpler than that of Frost's procedure. Moreover, the GSC can be extended to other structures whose learning curves of adaptive filter have different trajectories.

The GSC algorithm was proposed in the time-domain where the signal of sensor m at sample n is defined by

$$x_m(n) = s_m(n) + n_m(n),\ m = 1, \ldots, M$$

where $s_m(n)$ and $n_m(n)$ are the desired signal and noise, respectively.

Let $\mathbf{x}_A$ be the time-alignment of $[x_1, \ldots, x_M]^T$ towards the look direction. In the lower path, the desired signal inside the time-aligned signals are removed via a blocking matrix

$$\mathbf{x}_B = \mathbf{B}\mathbf{x}_A$$

where $\mathbf{x}_B$ is the signal vector after removing the desired signal. To do so, the blocking matrix $\mathbf{B}$ has to fulfill the following properties [BS01; GJ82]:

- The size of the matrix is $(M-1) \times M$
- The sum of all values in one row is zero
- The matrix has to be of rank $M-1$.

An example of a blocking matrix for the case $M = 4$ is

$$\mathbf{B} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}.$$

The vector $\mathbf{x}_B$ is processed with adaptive FIR filters $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_{M-1}]^T$ and then be subtracted from the output of DSB to get the noise-reduced output signal, that is

$$y = y_{DSB} * \mathbf{g}_f - \sum_{i=1}^{M-1} [\mathbf{x}_B]_i * \mathbf{H}(i, :)$$

where $*$ is the convolution operation, y_{DSB} is the output of DSB, $\mathbf{g}_f$ is a fixed FIR filter which ensures a specified gain and phase response for the output signal.

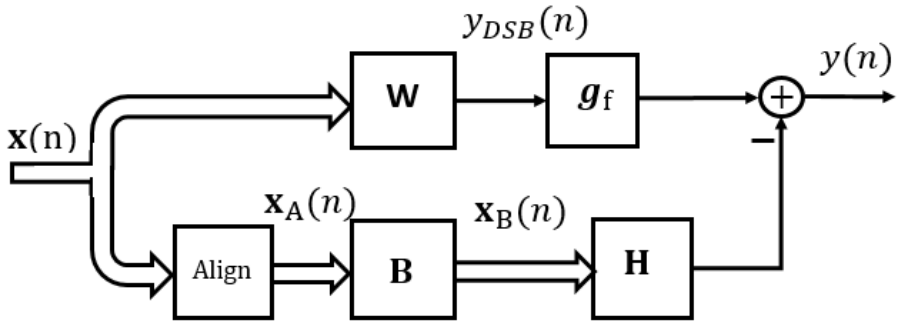


FIGURE 2.18: Schematic description of the decomposition of the optimal weight vector into two orthogonal parts.

The early paper of GSC [GJ82] used an iterative procedure to adaptive updating $\mathbf{H}$ in the least-mean-square sense

$$\mathbf{H}(:,k)_{n+1} = \mathbf{H}(:,k)_n + \mu y(n)\mathbf{x}_B(n-k),\ k = 1,\ldots,L$$

where L is the length of FIR filters and μ is the normalized step size.

In Fig. 2.19 and Fig. 2.20, we simulated for the case where signals from different directions impinge to the ULA with $M = 7$ microphones (the inter-distance $d_H = 0.02$ m): The interferences containing the narrow band signal centered at 1 kHz come from the directions $\phi_{r1} = 70^o$ and $\phi_{r2} = 150^o$, the SOI containning the impulse response comes from the direction $\phi_o = 0^o$. After processed with GSB, the output signal mainly contains the impulse response from the SOI. A small difference in amplitude of the desired signal and the beam's signal was studied by Griffith and Jim [GJ82], that is mainly due to the presence of the white noise component and the 'signal leak through' [Wid+75] which are negligible compared to other methods.

The GSC is a flexible structure due to the separation of the beamformer into a fixed and adaptive path. However, the GSC suffers from 'signal leak through' [Wid+75] and partially interference signal block in the lower path which may result in desired signal's distortion. The 'signal leak through' occurs when the blocking matrix does not completely suppress the desired signal and this can be problematic for broadband beamforming where it is difficult to ensure perfect signal cancellation across the frequencies of interest [McC01]. Also, the interference signal is partially blocked in the lower path, that happens when the interference signal is not completely orthogonal to the desired signal, which leads to the interference in the upper path leaks to the final output.

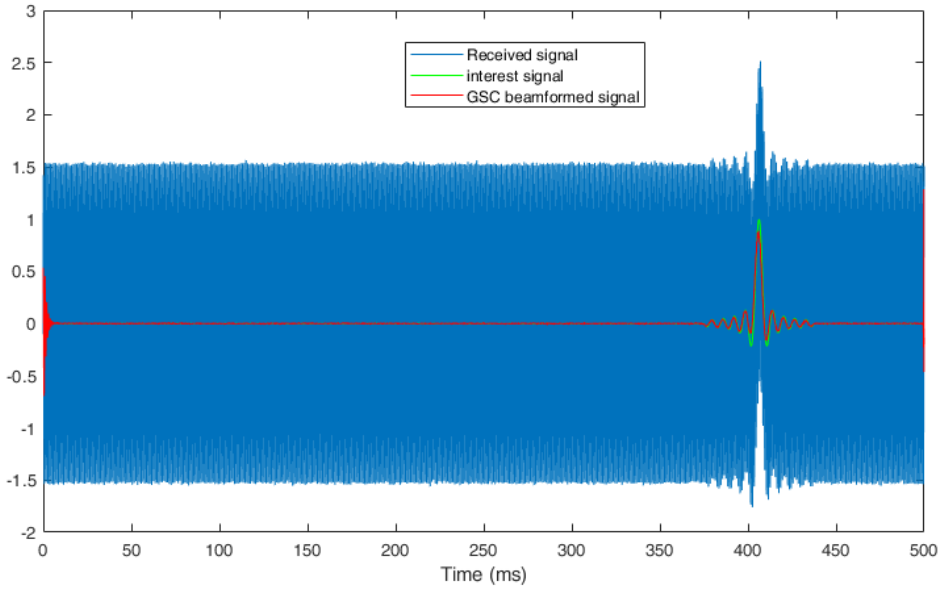


FIGURE 2.19: 0.5 second waveform: Microphone (blue line) contains the interference, white noise and interested signal; the SOI (green line) contains a impulse response; GSC (red line) beams to the SOI.

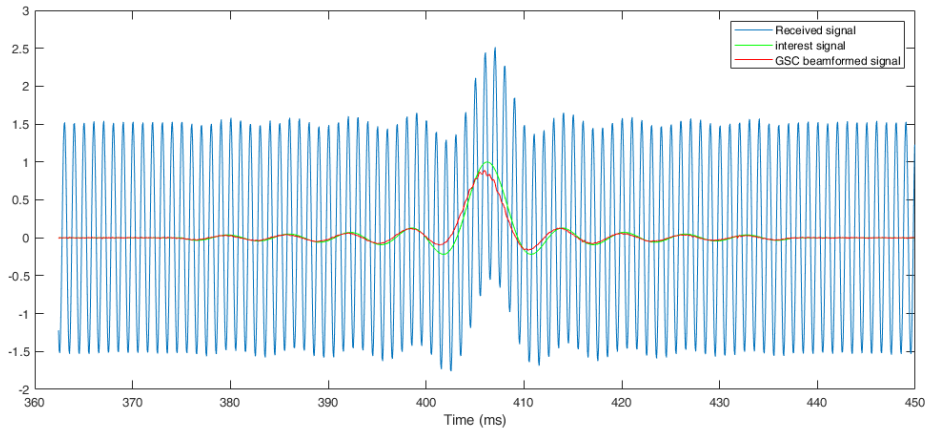


FIGURE 2.20: Zoom-in the waveform: Microphone (blue line) contains the interference, white noise and interested signal; the SOI (green line) contains a impulse response; GSC (red line) beams to the SOI.

2.3 Sparse Array and Broadband Beamforming

The sparse array is the array in that the inter-distances between sensors are not homogeneous. The beamforming applied for the broadband signals is called broadband beamforming. It is well-known that if the characteristics of beamforming are dependent with frequency, then the uniform array needs to be dense and large to cover high and low frequencies of the broadband signals. As such, the number of sensors for broandband beamforming is sometimes too big for some applications.

On the other hand, an irregular layout of sparse array could be an optimal array deployment for broadband beamforming. Normally, a sparse array contains sensors with small distances as well as big distances but the total number of sensors is less than that of uniform arrays.

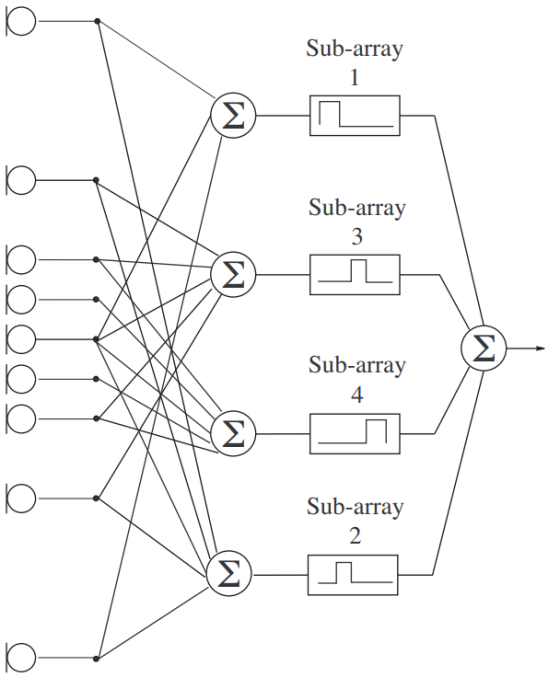
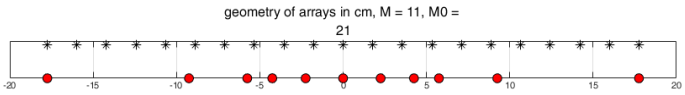


FIGURE 2.21: An example of nested array.

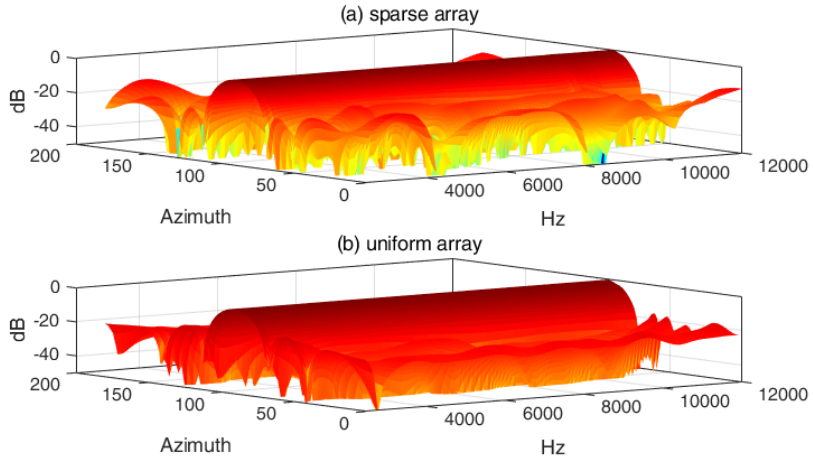
A straightforward array structure of broadband beamforming is based on the concept of harmonic nested arrays [Cho95], [ZGET04], where the final array is the union of several uniform arrays, each one matches to a different frequency subband, as shown in Fig. 2.21 [McC01].

Later, they proposed further improvements for sparse array layouts where advanced algorithms are applied to search the optimal position of sensors as well as their beamforming coefficients. Generally speaking, many studies have focused on the convex [HL13; HL14; CWB08; Liu+15b] or stochastic [Dob08; CT12; CT14; LYF13; ES+18] optimization approaches to determine sparse array deployments. Despite the success of these techniques, the results depend heavily on the strength of the optimization algorithms and it is hard to decide if the obtained solution is close to the optimal solution. For example, as based on compressed sensing framework, applying l_1-norm is efficient only if the measurement system (measurement matrix) has a low restricted isometry property (RIP) [CRT06; Don06], but the RIP itself is not easy to verify, and it is still an open question in the sparse array design's model. For the stochastic approach, it is not so efficient for a large search-space system or it does not guarantee that its solution is close to the optimal solution if the number of sensors is large. Furthermore, in some circumstances, the optimization or stochastic approaches are not only difficult to tune the parameters, but it is also computationally intensive to solve them with subject to the sparse solution.

Another example is shown in Fig. 2.22 [LS20], that is a sparse array with eleven microphones whose performance is equivalent to that of an uniform array with 21 microphones in the frequency range from 700 Hz to 12 kHz.



(A) Array deployments.



(B) Frequency-independent beamforming.

FIGURE 2.22: Sparse array vs uniform array.

Chapter 3

Using Inverse Fourier Transform to Obtain the Beamformer Weights

A sensor array collects spatial samples of propagating wave fields, and a beamformer performs spatial filtering to preserve the desired signal while suppressing interfering signals and noise arriving from directions other than the direction of interest. Given a signal with wideband frequency, using a uniform array is one of the most common approaches to obtain broadband beamforming. In this work, a function formulating the relations between the sensor coefficients and its beam pattern over frequency is introduced. The new function mainly contains the coordinate transform and inverse Fourier transform. From the view of spatial aliasing, the inter-distance of the sensors should be less than half of the minimum wavelength of the signal. However, from the bijection of the new function and broadband beamforming perspective, this chapter proposes the other lower and upper bounds for the inter-distance. Within these bounds, the new function is the bijective function which can be applied to design the uniform array with broadband beamforming.

3.1 Introduction

Spatial sampling is the bridge between a continuous sensor and a discrete sensor. It is similar to temporal sampling, other than sampling over the space. The Nyquist-Shannon sampling theorem [Sha49] is widely used in digital signal processing, it establishes a sufficient condition for a sample rate that permits a discrete sequence of samples to capture all the information from a continuous-time signal of finite bandwidth. To avoid the ambiguities resulting from aliasing, we must select the sampling rate f_s to be greater than two times the largest frequency of the signal f_{max} [JG07],

$$f_s > 2f_{max}$$

or, equivalently in the spatial sampling, the inter-distance of sensors Δ_d should be less than half of the smallest wavelength λ_{min},

$$\Delta_d < \frac{\lambda_{min}}{2}. \tag{3.1}$$

However, temporal sampling is developed for the signal reconstruction and the number of samples could be very large, while spatial sampling is normally built for beamforming, sound source localization, sound field reconstruction, etc., and the number of samples is restricted. A common justification for spatial sampling in

(3.1) is the spatial aliasing problem, but normally an array with a large aperture suffers from spatial aliasing when perform beamforming since beamforming is often computed based on the modeling of measurement points with respect to a reference point and the computed distances are much bigger than the inter-distance. From the spatial sampling constraint in (3.1), some questions arise as the following.

- How many samples (sensors in the array) are enough?
- How small must the inter-distance be?

In this study, the chapter provides some new insight into these questions from broadband beamforming perspective. Broadband beamforming is a beamforming technique used for the signal with a wide range of frequency and assuring certain requirements, e.g. the beam pattern is frequency-independent [P.E88; Syd94]. Many studies proposed good uniform array designs for broadband beamforming in the literature [VB88; LW08; Nor+14; DBC01; Yan06; DBWW95; SC07; Li+13; YZL11; WWM20]. Herein, we review some representative approaches. Van Veen and Buckley in [VB88] applied the least square error minimization between the desired and actual response of an array to find the beamformer weights. Later, the method proposed by Wei Liu and Stephan Weiss in [LW08] utilized the relationship between the beam pattern and Fourier transform to find the beamformer weights over the frequency which assure that the synthesized beam pattern is frequency-independent. The desired beam pattern of this approach is restricted to a function of variables: $\cos\phi \sin\theta$ and $\sin\phi \sin\theta$, where ϕ and θ are the azimuth and elevation angle of a direction, respectively. Recently, Sven *at al.* in [Nor+14] used convex optimization to design broadband beamforming for an arbitrary desired beam pattern, and Wenxia Wang *at al.* in [WWM20] also used optimization techniques to synthesize the frequency-independent beam pattern with respect to multiple constraints. Besides, the concept of line source with broadband beam pattern also attracts an amount of interest [DVMP01; DVMP03; Vil04]. De Villiers in [Vil04] (this is the extended version of methods in [DVMP01; DVMP03] for narrow bands) utilized the singular function for line sources to design the broadband beamforming.

To the best of the author's knowledge, the present approaches do not clearly explain the reasons behind the discrepancy between the desired beam pattern and the synthesized beam patterns. In order to analyze the deviation between the desired and synthesized beam pattern over frequency, the chapter formulates the transform from an arbitrary desired beam pattern to sensor weights. First, the chapter presents a geometric visualization to explain the relation between the desired beam pattern and functions in the spherical coordinates at different frequencies. A function in the spherical/Cartesian coordinate that presents the gain of array response with respect to arrival direction is called a *gain function*. It is similar to the beam pattern, but it could be presented in either the spherical or Cartesian coordinate. Next, the gain function in the spherical coordinate is transformed to another gain function in the Cartesian coordinate. Finally, the inverse Fourier transform is applied for the gain function in the Cartesian coordinate to achieve the beamformer weights. The overview of the procedure is presented in Fig. 3.1.

After having successfully formulated the synthesized beam pattern of the sensor array, we deduce new bounds for the inter-distance of sensors. The bounds assure that the beam pattern is frequency-independent with regard to the bandwidth of interest. Hence, the designers gain more information to adjust the sensor array configurations. In order to examine the function, that transforms an arbitrary desired beam pattern to beamformer weights, some examples with different configurations (the planar array and linear array) are given.

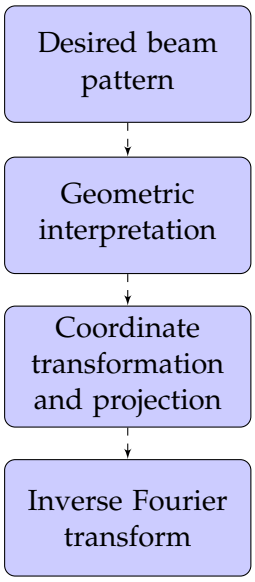


FIGURE 3.1: The steps inside the proposed method to obtain the beamformer coefficients from an arbitrary beam pattern.

The chapter is organized as follows. In Section 3.2, the chapter addresses the problem as the continuous sensor. In Section 3.3, the chapter formulates for the discrete sensor and connects the problem with the continuous sensor. In Section 3.4, the chapter analyzes imperfections of the transforming function. Numerical simulations are given in Section 3.5. Finally, conclusions are drawn in Section 3.6.

3.2 Continuous Sensor

Without loss of generality, we start with a planar array, similar results could be applied for a linear array. A sensor at position (x, y) in the Cartesian coordinate has the measurement value $p(x, y, f)$ at the frequency f, and a complex-valued $w(x, y, f)$ is a filter element for that sensor. The output of the beamformer is

$$y(f) = \int \int_{-\infty}^{\infty} w(x, y, f) p(x, y, f) dx dy.$$

Consider a far-field wave, then the wave reaching the sensor is planar. c is the speed of wave propagation. The spatial response of the array for a source at a direction indicated by the azimuth angle ϕ and elevation angle θ is

$$b(\theta, \phi, f) =$$

$$\int \int_{-\infty}^{\infty} w(x, y, f) e^{-j\frac{2\pi f}{c}(x \sin\theta \cos\phi + y \sin\theta \sin\phi)} dx dy \tag{3.2}$$

where $j = \sqrt{-1}$ is the imaginary unit. Substitutions:

$$R = \frac{f}{c} \tag{3.3}$$

$$u = R \sin\theta \cos\phi, \quad v = R \sin\theta \sin\phi. \tag{3.4}$$

$b(\theta, \phi, f)$ could be presented by $b(u, v, f)$,

$$b(u, v, f) = \int \int_{-\infty}^{\infty} w(x, y, f) e^{-j2\pi(ux+vy)} dxdy. \tag{3.5}$$

From Eq. (3.5), it is clearly seen that $b(u, v, f)$ is the 2-Dimensional (2D) Fourier transform of $w(x, y, f)$ with respect to x, y or $w(x, y, f)$ is the 2D inverse Fourier transform of $b(u, v, f)$.

If $R = f/c$ is a constant, Eq. (3.4) is the formula transforming a point (θ, ϕ, R) in the spherical coordinate to a point (u, v) in the Cartesian coordinate (we are only interested in the u-v plane where $z = 0$). From this observation, a method to obtain the weights from a desired beam pattern is proposed. First, a desired beam pattern is interpreted as the functions in spherical coordinates, and then functions in the spherical coordinates are transformed to Cartesian coordinates. Finally, applying inverse Fourier transform to obtain the complex-valued weights.

Let $B(\theta, \phi)$ be a desired beam pattern, below are some highlights:

- In the spherical coordinate, a single value R is mapping with a single value f and vice versa.
- The desired beam pattern $B(\theta, \phi)$ is interpreted as gain functions in spherical coordinates $b_R(\theta, \phi)$: Different frequency f (radius R) associated with different gain functions.
- A gain function in the spherical coordinate $b_R(\theta, \phi)$ is located in a hemisphere with radius R (for visualization, the gain values could be presented by the color on the surface of the hemisphere, as shown in Fig. 3.2).

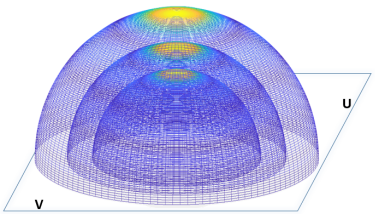


FIGURE 3.2: Example of gain functions associated to different frequencies in the spherical coordinate.

- $b_f(u, v)$ is the gain function at frequency f, as shown in Fig. 3.3. $b_f(u, v)$ is obtained by transforming (Eq. (3.4)) $b_R(\theta, \phi)$ to the Cartesian coordinate and projection (assign $z = 0$).

Below, the chapter suggests the steps to obtain the weights of the uniform array from a desired beam pattern,

- Step 1: Define a desired beam pattern $B(\theta, \phi)$.
- Step 2: At a single radius R, presenting the desired beam pattern to a gain function in the spherical coordinate,

$$b_R(\theta, \phi) := b(\theta, \phi, R) = \mathcal{C}\{B(\theta, \phi), f\}$$

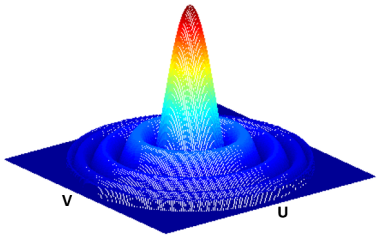


FIGURE 3.3: An example of the gain function in the Cartesian coordinate at one frequency.

where $\mathcal{C}\{.\}$ is a *copy/interpreting* function and R is constant. A copy/interpreting function is the function that copies/interprets gain values of the desired beam pattern to the gain function at a frequency in the spherical coordinate, that is the function with inputs located in the hemisphere of radius R.

- Step 3: At a single frequency $f = Rc$, $b_f(u,v)$ is achieved by transforming and projecting the gain function $b_R(\theta,\phi)$ in the spherical coordinator to the u-v plane of the Cartesian coordinate,

$$b_f(u,v) := b(u,v,f) = \mathcal{T}\{b_R(\theta,\phi)\}$$

where $\mathcal{T}\{.\}$ is the coordinate transform and projection (assign $z = 0$).

- Step 4: Applying the 2D inverse Fourier transform of $b_f(u,v)$ to achieve $w(x,y,f)$,

$$w(x,y,f) = \mathcal{F}^{-1}\{b_f(u,v) \cup \mathbb{A}\}$$

where $\mathcal{F}^{-1}\{.\}$ is the inverse Fourier transform, $\mathbb{A}$ is the set of arbitrary values associating with the points in plane u-v which are outside the hemisphere (in Fig. 3.3, $\mathbb{A}$ is the set of zeros).

For a linear array, the desired beam pattern is interpreted as the gain functions in the polar coordinate.

In summary, the complex-valued weight at a frequency is formulated as below,

$$w(x,y,f) =$$

$$\mathcal{W}\{B(\theta,\phi),f\} := \mathcal{F}^{-1}\{\mathcal{T}\{\mathcal{C}\{B(\theta,\phi),f\}\} \cup \mathbb{A}\} \tag{3.6}$$

where $\mathcal{W}\{.\}$ is the inverse beam pattern function (inverse beam pattern transform), as illustrated in Fig. 3.4. Note that $\mathbb{A}$ is a set containing arbitrary padding values, and $\mathbb{A}$ also a input of inverse beam pattern transform. However, the space of $\mathbb{A}$ is dependent on the frequency f, then we omitted it in the input argument of $\mathcal{W}\{.\}$.

Theorem 1. *$\mathcal{W}\{B(\theta,\phi),f\}$ in (3.2) is a bijective function for $\theta \in (0, \frac{\pi}{2}], \phi \in [0, 2\pi)$ and $f > 0$.*

Proof. If all the functions inside $\mathcal{W}\{B(\theta,\phi),f\}$ are bijective functions, then $\mathcal{W}\{B(\theta,\phi),f\}$ is a bijection [Ham13; Pas15].
$\mathcal{F}^{-1}\{.\}$ is the inverse Fourier transform, then it is a bijection [TT12].
$\mathcal{C}\{.\}$ does the copy operators, it is a bijection.
Consider R is constant and $R > 0$, (θ,ϕ,R) belongs to the surface of a hemisphere, then $\mathcal{T}\{b_R(\theta,\phi)\}$ is a bijection for $\theta \in (0, \frac{\pi}{2}]$. □

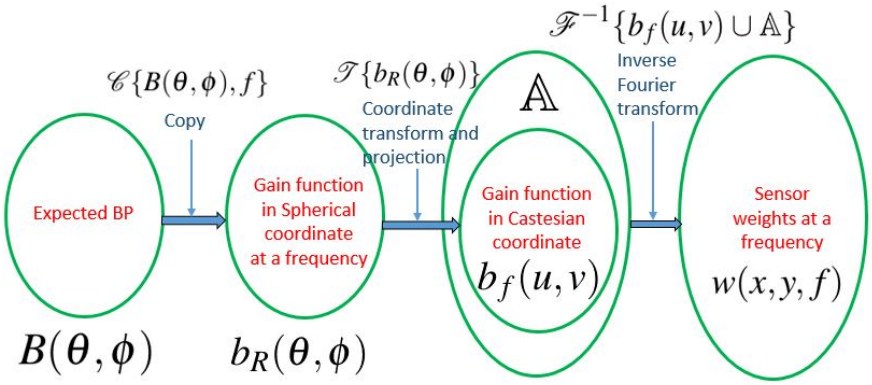


FIGURE 3.4: Inside inverse beam pattern transform at a frequency.

The weights derived from Eq. (3.2) could form the beam pattern which is identical with the desired beam pattern, that means a beamforming function in (3.2) is invertible via $\mathcal{W}\{B(\theta,\phi),f\}$.

3.3 Discrete Sensor

Consider the planar uniform array: the inter-distance of sensor array Δ_d in X-direction is equal to the inter-distance in Y-direction, the number of sensors in each direction N is an odd number. The origin of the coordinate is at the center of the planar array and coincides with a sensor, as illustrated in Fig. 3.5.

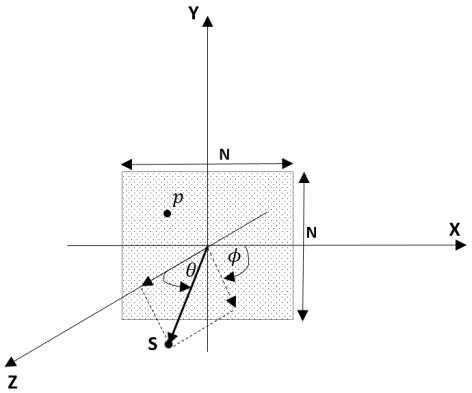


FIGURE 3.5: Discrete sensor array in the Cartesian coordinate.

Let $w(n_1,n_2,f)$ be a weight factor of a sensor at a position with index (n_1,n_2) ($n_1,n_2 \in \mathbb{Z}$), Eq. (3.2) becomes,

$$b(\theta,\phi,f) = \sum_{n_1,n_2=-\frac{(N-1)}{2}}^{\frac{(N-1)}{2}} w(n_1,n_2,f)$$

$$\times e^{-j\frac{2\pi f}{c}(n_1\Delta_d \sin\theta\cos\phi + n_2\Delta_d \sin\theta\sin\phi)}.$$

The radius in the spherical Coordinate is defined by

$$R = \frac{fN\Delta_d}{c}. \tag{3.7}$$

We also define for u, v

$$u = \frac{fN\Delta_d}{c}\sin\theta\cos\phi, \quad v = \frac{fN\Delta_d}{c}\sin\theta\sin\phi. \tag{3.8}$$

Eq. (3.5) becomes:

$$b(u,v,f) = \sum_{n_1,n_2=-\frac{(N-1)}{2}}^{\frac{(N-1)}{2}} w(n_1,n_2,f)e^{-j\frac{2\pi}{N}(n_1u+n_2v)}. \tag{3.9}$$

If u, v are restricted to integer numbers, Eq. (3.9) is a 2D discrete Fourier transform with respect to the variables n_1, n_2 [Rab75]. More specifically, u, v are frequency indexes in the 2D frequency-domain, therefore we can imagine that the sensor indexes n_1, n_2 of the planar array are associated with grid points u, v in Cartesian coordinates, as shown in Fig. 3.6 (every grid point in u-v plane is associated with a sensor).

From Eq. (3.8), one may confuse the integer values of u, v, however, as illustrated in Fig. 3.7, the integer value of u is mapping with the non-uniform elevation angle. For the fractional value u or v, it can not be realized in the inverse discrete Fourier transform (IDFT) of Eq. (3.9), therefore the fractional value u or v is discarded in the discrete domain. Let us define a discrete desired beam pattern

$$B(\mathbf{r})|_{\mathbb{P}_f} := \sum_{\mathbf{r}_0\in\mathbb{P}_f}\int B(\mathbf{r})\delta(\mathbf{r}-\mathbf{r}_0)d\mathbf{r} \tag{3.10}$$

where $\mathbf{r} = (\theta, \phi)$ is a direction in the spherical coordinate, $\mathbb{P}_f$ is the set of discrete directions at a frequency f and $\delta(.)$ is a Dirac delta function. From (3.8), we obtain

$$\theta = \arcsin(\frac{\sqrt{u^2+v^2}}{R}) \tag{3.11}$$

and

$$\phi = \arctan(\frac{v}{u}). \tag{3.12}$$

In order to apply the IDFT in Eq. (3.9), we need to define a set of integer points (u, v) inside the hemisphere with radius R in the following way:

$$\mathbb{E}_R := \{(u,v) : u,v \in \mathbb{Z}, \sqrt{u^2+v^2} \leq R\} \tag{3.13}$$

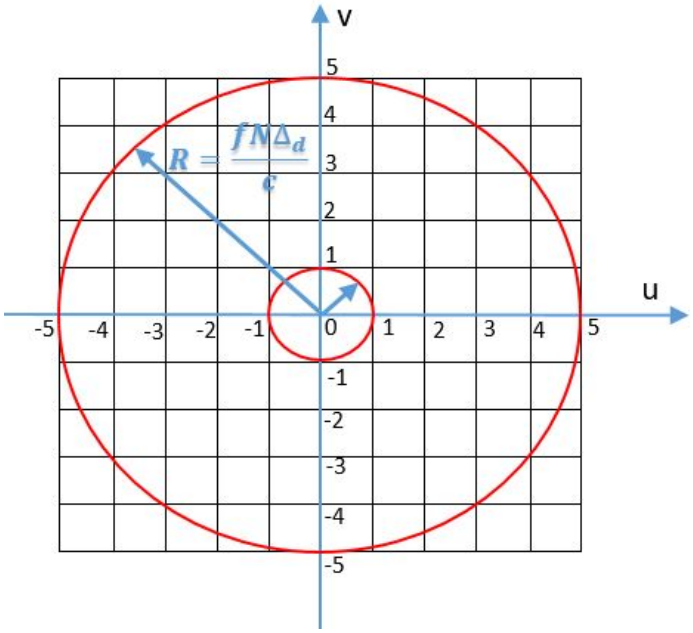
and $\mathbb{P}_f$ is obtained by applying Eqs. (3.11), (3.12) for every (u, v) in the set $\mathbb{E}_R$. The cardinality of the set $\mathbb{P}_f$ is

$$|\mathbb{P}_f| = |\mathbb{E}_R| = 1 + 4\sum_{k=0}^{\lfloor R\rfloor}\lfloor\sqrt{R^2-k^2}\rfloor, \tag{3.14}$$

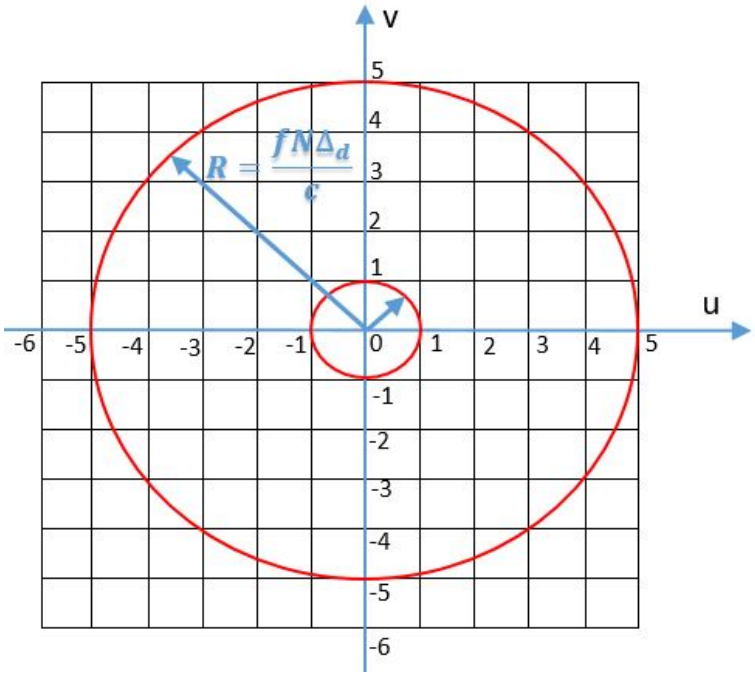
where $\lfloor.\rfloor$ is the floor function (see Appendix 3.7: The Number of Encoding Points).

For a discrete sensor, we adopt the design steps from a continuous sensor by replacing Eqs. (3.3), (3.4) by Eqs. (3.7), (3.8). An inverse Fourier transform is replaced

by an IDFT [1]. The input of inverse beam pattern function is $B(\mathbf{r})|_{\mathbb{P}_f}$ and the frequency index is restricted to the set $\mathbb{I}$ in (3.15). Note that $\mathbb{I} \setminus \mathbb{E}_R$ is the set of index of $\mathbb{A}$.



(A) $N = 11$ (N is an odd number)



(B) $N = 12$ (N is an even number)

FIGURE 3.6: Sensor indices in the u-v plane of the Cartesian coordinate and the radius of the hemispheres

[1] Since $\frac{-(N-1)}{2} \leq n_1, n_2 \leq \frac{(N-1)}{2}, n_1, n_2 \in \mathbb{Z}$, before and after doing inverse Fourier transform, shifting zero-frequency component to begin and center of the array are required, respectively (the function of IDFT in the programming languages is normally applied for $n_1, n_2 \in \mathbb{Z}, 0 \leq n_1, n_2 \leq N$).

Theorem 2. *In discrete domain, suppose that N is an odd number, $\mathcal{W}\{B(\mathbf{r})|_{\mathbb{P}_f}, f\}$ in (3.2) is a bijection if*

$$f < \frac{(N+1)c}{2N\Delta_d}.$$

Conversely, if

$$f \geq \frac{(N+1)c}{2N\Delta_d},$$

then $\mathcal{W}\{B(\mathbf{r})|_{\mathbb{P}_f}, f\}$ is not a bijection.

Proof. For discrete sensor, the proof is similar to the proof of Theorem 1 except that the frequencies indexes are restricted to the set

$$\mathbb{I} = \{(u,v) : \frac{-(N-1)}{2} \leq u,v \leq \frac{(N-1)}{2}, u,v \in \mathbb{Z}\}. \tag{3.15}$$

In other words, the projection (assign $z = 0$) inside $\mathcal{T}\{b_R(\theta,\phi)\}$ is limited to an area in (3.15).
In case of $f < \frac{(N+1)c}{2N\Delta_d}$, from (3.8) we obtain

$$-\frac{N+1}{2}\sin\theta\cos\phi < u < \frac{N+1}{2}\sin\theta\cos\phi,$$

$$-\frac{N+1}{2}\sin\theta\sin\phi < v < \frac{N+1}{2}\sin\theta\sin\phi.$$

Since $-1 \leq \sin\theta\cos\phi,\ \sin\theta\sin\phi \leq 1$, then the bound of the projection of the coordinate transform $\mathcal{T}\{b_R(\theta,\phi)\}$ is

$$-\frac{N+1}{2} < u,v < \frac{N+1}{2},\ \theta \in (0,\frac{\pi}{2}], \phi \in [0,2\pi). \tag{3.16}$$

Consider $u,v \in \mathbb{Z}$, the set in (3.16) equals the set $\mathbb{I}$ in (3.15), then the projection inside $\mathcal{T}\{b_R(\theta,\phi)\}$ is a bijection which prove the claim of the bijection.

We now examine the converse claim. We need to find a direction $(\theta,\phi) \in \mathbb{P}_f$ which does not associate with any element in $\mathbb{I}$. It means the projection inside $\mathcal{T}\{b_R(\theta,\phi)\}$ is not an injection, which implies that $\mathcal{W}\{B(\mathbf{r})|_{\mathbb{P}_f}, f\}$ is not a bijection. Let us select $(\theta,\phi) = (\frac{\pi}{2},0) \in \mathbb{P}_f$, then, from (3.8), we obtain $u,v \geq \frac{N+1}{2}$ for all $f \geq \frac{(N+1)c}{2N\Delta_d}$. It is easy to select a f that satisfies $u,v \in \mathbb{Z}$. An obvious selection is $f = \frac{(N+1)c}{2N\Delta_d}$ corresponding to $u = v = \frac{N+1}{2}$, $(u,v) \notin \mathbb{I}$.

□

When $R < \frac{N+1}{2}$ (or $f < \frac{(N+1)c}{2N\Delta_d}$), all the information of the gain function in the spherical coordinate is encoded by the integer grid points in the u-v plane. The hemisphere with the bigger radius contains more u-v grid points inside it, then the gain function is encoded by more grid points, it means the resolution of the synthesized beam pattern is higher. If the radius of the hemisphere is too small, less u-v grid points are inside the hemisphere and the resolution of synthesized beam pattern is reduced.

From above analysis, to assure the frequency-independent beam pattern, the chapter suggests that at least five grid points are inside the smallest hemisphere for the lowest frequency, as shown in Fig. 3.6a. Note that five grid points for encoding the beam pattern at the lowest frequency is a weak assumption, but it is a good

value to start with and the chapter will examine the effects in the simulation section.

$$1 \leq R,$$
$$1 \leq \frac{fN\Delta_d}{c},$$
$$\frac{c}{Nf} \leq \Delta_d.$$

Substituting the wavelength $\lambda = \frac{c}{f}$, we obtain

$$\frac{\lambda}{N} \leq \Delta_d.$$

The same approach for N even, we could have the following constraints for the inter-distance:

$$\begin{cases} \frac{\lambda}{N} \leq \Delta_d < \frac{\lambda}{2}, & N \text{ is even}, \\ \frac{\lambda}{N} \leq \Delta_d < \frac{(N+1)\lambda}{2N}, & N \text{ is odd}. \end{cases} \tag{3.17}$$

The origin of the Cartesian coordinate is always consolidated with a sensor's position is the best choice. In the case of N even, the origin is not at the center of the array, therefore the array is not symmetric to the origin, as shown in Fig. 3.6b. Thus, the maximum of radius is less than $\frac{N}{2}$.

In summary, inequality (3.17) can be used as the bound constraints for inter-distance of sensors.

3.4 Imperfections of Applying Inverse Beam Pattern Transform

For the sake of simplicity, in this section, we analyze the linear array with the polar coordinate system.

3.4.1 Non-uniform Discrete Direction Angle

Assuming that the inverse beam pattern transform is a bijection in the bandwidth of interest, the discretized space by (3.11), (3.12), (3.13) affects the quality of the synthesized beam pattern. That is the resolution of the beam pattern at different angle ϕ is not equal, as shown in Fig. 3.7. The resolution of beam pattern at broad-side directions ($\phi \approx 0^o$) is higher, while it is lower at end-fire directions ($\phi \approx 90^o$ or $\phi \approx -90^o$). It is caused by the non-uniform discretized direction angle of $B(\phi)$ in order to obtain the integer value u for the IDFT.
The resolution of direction angle corresponding to two adjacent sensors n_0, n_1 (where $n_1 - n_0 = 1$) is

$$\Delta\phi_{01} = \arcsin(\frac{n_1}{R}) - \arcsin(\frac{n_0}{R}).$$

The resolution of direction angle corresponding to two adjacent sensors n_4, n_5 (where $n_5 - n_4 = 1$) is

$$\Delta\phi_{45} = \arcsin(\frac{n_5}{R}) - \arcsin(\frac{n_4}{R}).$$

If $n_5, n_4 > n_1, n_0$, then $\Delta\phi_{45} > \Delta\phi_{01}$, since the arcsin function is a non-linear function.

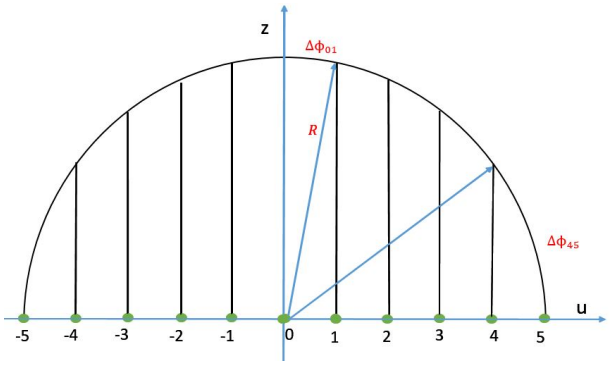


FIGURE 3.7: Transform from the polar coordinate to the Cartesian coordinate and the non-linear resolution of direction angles.

3.4.2 The Number of Encoding Points at Low Frequencies

As mentioned earlier, the number of u-v grid points used to encode the desired beam pattern is lower at the lower frequency. Herein, we illustrate this drawback by an example. The list of encoding points for the desired beam pattern at frequency f,

$$\mathbb{E} = \{u : u \leq R,\ R = \frac{fN\Delta_d}{c}, u \in \mathbb{Z}\}.$$

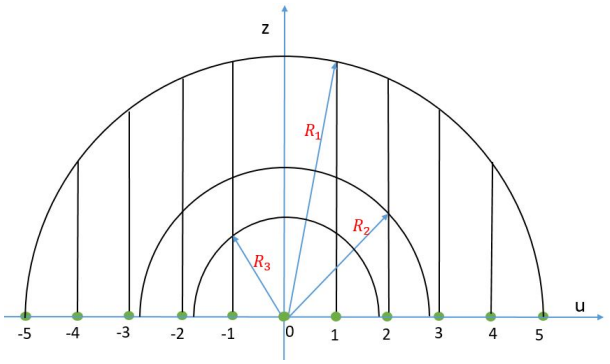


FIGURE 3.8: The number of encoding points is changed corresponding to the frequency.

As shown in Fig. 3.8, when $R_1 > R_2 > R_3$ (corresponding to $f_1 > f_2 > f_3$), then $|\mathbb{E}_1| > |\mathbb{E}_2| > |\mathbb{E}_3|$ $(11 > 5 > 3)$ where $|\mathbb{E}_x|$ is the cardinality of set $\mathbb{E}_x$.

In the constraint (3.17), the lower bound $\frac{\lambda}{N}$ is to assure at least five grid points (three grid points) of the u-v plane (u axis) are inside the smallest hemisphere (half circle) containing the gain function of the lowest frequency. In some applications, the lower bound should be set more strictly $\Delta_d >> \frac{\lambda}{N}$ to assure enough encoding points for the beam pattern's shape.

In Section 3.5, the chapter will provide numerical simulations to illustrate the inverse beam pattern transform in Eq. (3.2) and the bound constraints in Eq. (3.17). However, instead of changing the inter-distance, the simulation changes frequencies and examines the change of beam patterns.

3.5 Numerical Simulations

The chapter provides three examples: Two examples for planar arrays and one example for a linear array. The MATLAB code for the examples is provided in Github [2].

Besides frequency-independence, some other beamformer performances are also important, such as white noise gain (WNG), directivity factor (DF), steering ability, front-to-back ratio, etc [BJ01]. However, if beam patterns are almost identical then the DFs or front-to-back ratio are also almost identical, therefore we only measure the WNG and analyze the steering in simulations. The WNG is a useful measure for the robustness against random errors. That is given by [BJ01],

$$\mathcal{N}(\omega) = \frac{1}{\sum_{n_1,n_2} w^*(n_1,n_2,f)w(n_1,n_2,f)} \tag{3.18}$$

where $(.)^*$ denotes the conjugate operator. Note that the Eq. (3.18) is only applied for the beamforming with the distortionless at the look direction.

Example 1. *Example for planar array: Desired beam pattern has 2D-Sinc shape.*

- Step 1: Define the desired beam pattern, as shown in Fig. 3.9.

$$B(\theta,\phi) = \begin{cases} |\frac{\sin(5\pi\theta)}{5\pi\theta}|, & \theta > 0, \\ 1, & \theta = 0. \end{cases}$$

- Step 2: Interpret the desired beam pattern to a gain function in the spherical coordinate system, the different gain functions are shown in Fig. 3.2,

$$b_R(\theta,\phi) = \begin{cases} |\frac{\sin(5\pi\theta)}{5\pi\theta}|, & \theta \in (0,\frac{\pi}{2}], \\ 1, & \theta = 0. \end{cases}$$

- Step 3: Transform $b_R(\theta,\phi)$ to $b_f(u,v)$. The relationship between the spherical and the Cartesian coordinates system is $\theta = \arcsin(\frac{\sqrt{u^2+v^2}}{R})$ or $\theta = \arcsin(\frac{c\sqrt{u^2+v^2}}{fN\Delta_d})$. Then,

$$b_f(u,v) = \begin{cases} \left|\frac{\sin(5\pi \arcsin(\frac{c\sqrt{u^2+v^2}}{fN\Delta_d}))}{5\pi \arcsin(\frac{c\sqrt{u^2+v^2}}{fN\Delta_d})}\right|, u,v \neq 0, \sqrt{u^2+v^2} \leq \frac{fN\Delta_d}{c}, \\ 1,\ u = v = 0, \\ 0,\ \text{Otherwise.} \end{cases}$$

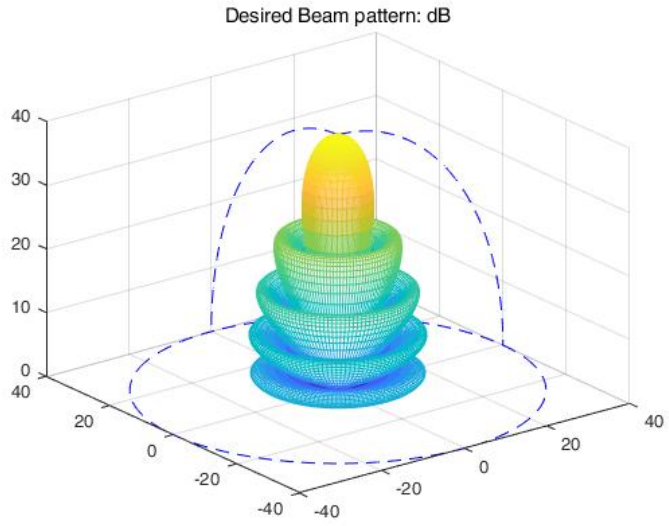
  Fig. 3.3 shows a gain function in the Cartesian coordinate system at 16 kHz, it is resulted from one of the gain functions in Fig. 3.2.

- Step 4: Sensor weights are achieved by applying IDFT, e.g., the beam pattern at 16 kHz is showed in Fig. 3.10a.

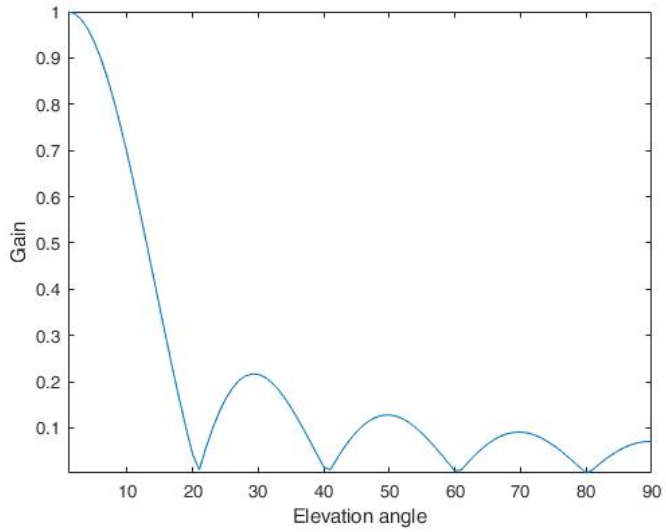
With $N = 200$, $\Delta_d = 0.01$ m and $c = 343.2$ m/s, from Eq. (3.17), we have the range of frequencies,

$$\frac{c}{N\Delta_d} \leq f < \frac{c}{2\Delta_d},$$

[2] https://github.com/PhanLeSon03/inverse_beam



(A)



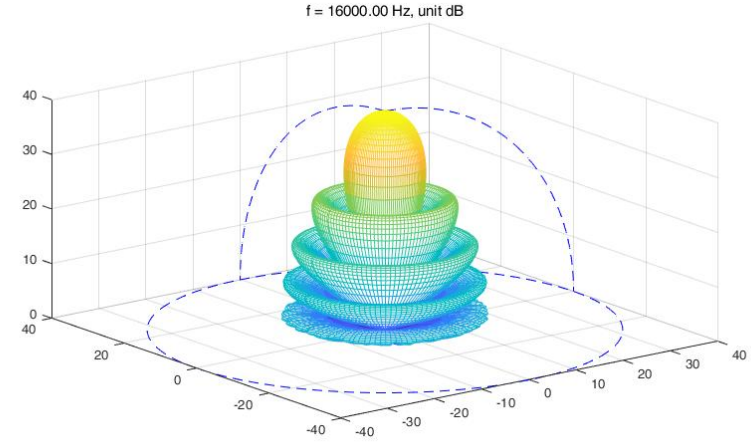
(B)

FIGURE 3.9: (a) Desired beam pattern (in dB scale). (b) Cross-section of beam pattern.

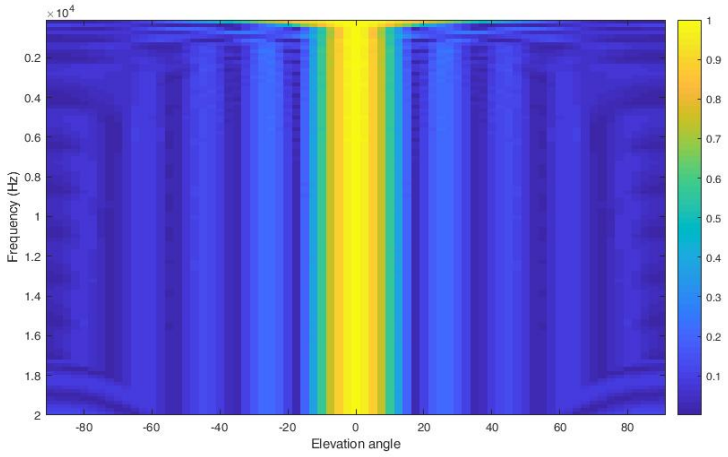
$$176.1 \text{ Hz} \leq f < 17160 \text{ Hz}.$$

The cross-section of beam patterns and WNGs over frequency from 200 Hz to 20 kHz are presented in Fig. 3.10b and Fig. 3.10c, respectively. At $f > 17160$ Hz, the side-lobes of beam patterns are distorted, since the radius of hemispheres are bigger than the boundary of the u-v plane. At low frequencies, the beam patterns are also distorted, since the shape of the desired beam pattern has higher-order and the desired beam pattern needs more grid points to encode its shape, so we have to increase the lower bound of frequency in order to maintain the shape of the desired beam pattern.

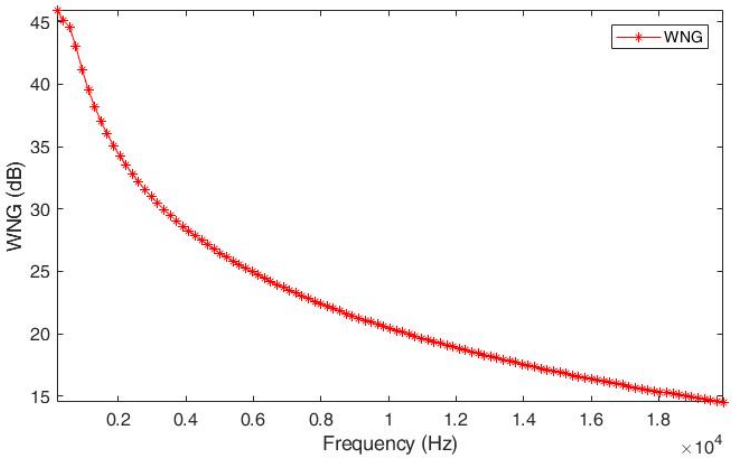
Example 2. *Example for planar array: Desired beam pattern has two levels of gain and no side-lobe.*



(A)



(B)



(C)

FIGURE 3.10: (a) Beam pattern at 16 kHz, (b) Beam patterns over frequency from 200 Hz to 20 kHz, (c) WNGs over frequency ($N = 200$, $\Delta_d = 0.01$ m and $c = 343.2$ m/s).

- Step 1: Define the desired beam pattern which the gain is the function of elevation angle θ shown in Fig. 3.11,

$\theta \leq \theta_{C1}$: Gain is 1.
$\theta_{C1} \leq \theta \leq \theta_{C2}$: Gain is 1/10 (reduce 20 dB).
Others: Gain is 0.

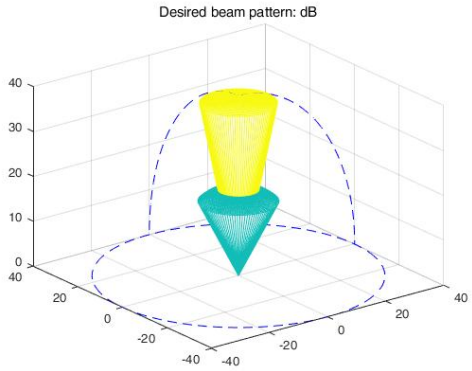


FIGURE 3.11: Desired Beam-Pattern in dB scale (maximum dB scale is 40dB).

- Step 2: Interpret the desired beam pattern to a gain function in the spherical coordinate,

$$b_R(\theta,\phi) = \begin{cases} 1, & \theta \leq \theta_{C1}, \\ 1/10, & \theta_{C1} < \theta \leq \theta_{C2}, \\ 0, & \text{Otherwise.} \end{cases}$$

  where $\theta_{C1} = \frac{\pi}{16}, \theta_{C2} = \frac{\pi}{8}$.

- Step 3: Transform the gain function in the spherical coordinate $b_R(\theta,\phi)$ to a gain function in the Cartesian coordinate ($Rsin\theta = \sqrt{u^2+v^2}, R = \frac{N\Delta_d f}{c}$),

$$b_f(u,v) = \begin{cases} 1, & \sqrt{u^2+v^2} \leq R\sin\theta_{C1}, \\ 1/10, & R\sin\theta_{C1} < \sqrt{u^2+v^2} \leq R\sin\theta_{C2}, \\ 0, & \text{Otherwise.} \end{cases}$$

  Fig. 3.12 illustrates a gain function in the Cartesian coordinate at 16 kHz.

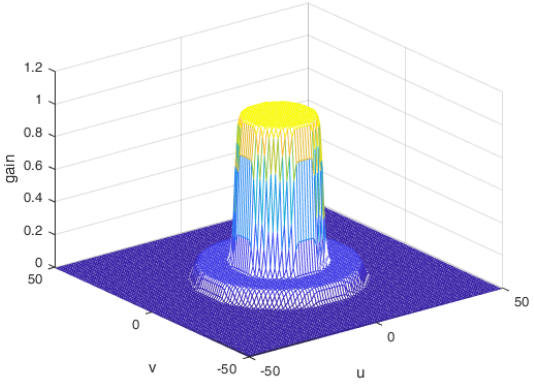


FIGURE 3.12: A gain function in the Cartesian coordinate at 16 kHz.

- Step 4: Sensor weights of the sensor array are obtained by applying IDFT, e.g., the synthesized beam-pattern at 16 kHz is depicted in Fig. 3.13a. The cross-section of beam patterns versus frequency is depicted in Fig. 3.13b.

Selecting $N = 100$, $\Delta_d = 0.015$ m and $c = 343.2$ m/s. From Eq. (3.17), we obtain a frequency range,

$$\frac{c}{N\Delta_d} \leq f < \frac{c}{2\Delta_d},$$
$$228.8 \text{ Hz} \leq f < 11440 \text{ Hz}.$$

We simulate for the frequency range from 250 Hz to 16 kHz. At $f > 11440$ Hz, the two-steps cylinder shape (the main-lobe is shown in Fig. 3.11) of the gain functions are still inside the boundary of u-v plane, although the hemispheres are bigger than the boundary of u-v plane, so the synthesized beam patterns still preserve the main-lobe of the desired beam pattern, as shown in Fig. 3.13a and Fig. 3.13b. Fig. 3.13c plots the WNGs over frequency. For this special beam pattern (there is no side-lobe), the inverse beam pattern function is surjection for some frequencies but this surjection does not affect the main-lobe of beam pattern.

The lower bound 228.8 Hz assures five grid points are inside the smallest hemisphere, but only one grid point is inside the smallest two-steps cylinder or the shape of two-steps cylinder is encoded by only one point, therefore the beam patterns are distorted significantly at low frequencies.

Example 3. *Example for linear array: Desired beam pattern only has a main-lobe.*

- Step 1: A similar desired beam pattern used in [LW08] is taken and removed the side-lobes,

$$B(\theta) = \sum_{m=-3}^{3} h_m e^{-jm\pi \sin\theta} \tag{3.19}$$

where the coefficients

$$\{h_m : m = -3, ..., 3\} =$$
$$[0.0307, 0.2028, 0.1663, 0.2004, 0.1663, 0.2028, 0.0307].$$

In Fig. 3.14a, red "*" plotted the desired beam pattern without side-lobe.

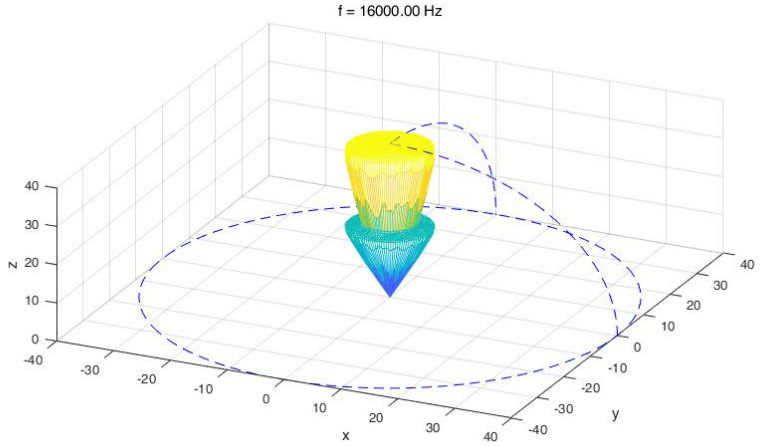
- Step 2: Interpret the desired beam-pattern to a gain function in the polar coordinate,

$$b_R(\theta) = \begin{cases} B(\theta), & -20^o \leq \theta \leq 20^0, \\ 0, & \text{Otherwise.} \end{cases}$$

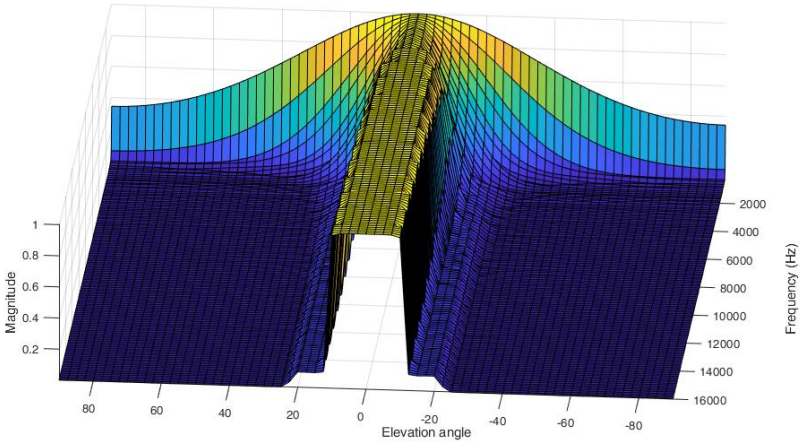
- Step 3: Transform the gain function in the polar coordinate $b_R(\theta)$ to a gain function in the Cartesian coordinate ($u = Rsin\theta, R = \frac{N\Delta_d f}{c}$),

$$b_f(u) = \begin{cases} B(\arcsin(\frac{u}{R})), & R\sin(-20^0) \leq u \leq R\sin(20^0), \\ 0, & \text{Otherwise.} \end{cases}$$

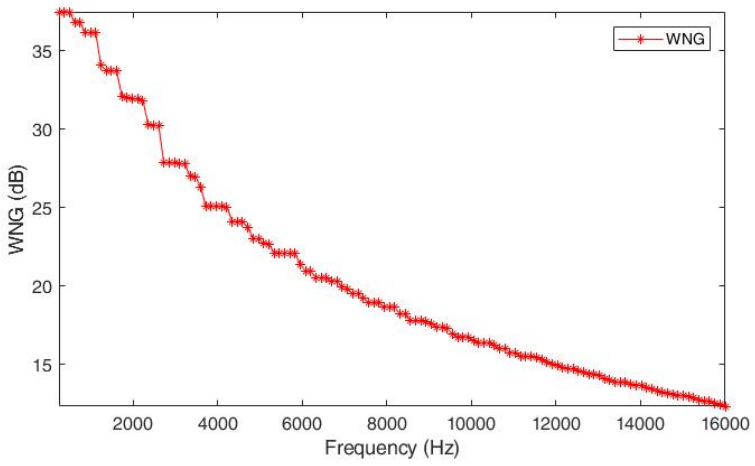
- Step 4: Sensor weights are obtained by applying IDFT. The beam pattern at 16 kHz, beam patterns over frequency and WNGs over frequency are plotted in Fig. 3.14b, Fig. 3.14c and Fig. 3.14d, respectively.



(A)



(B)

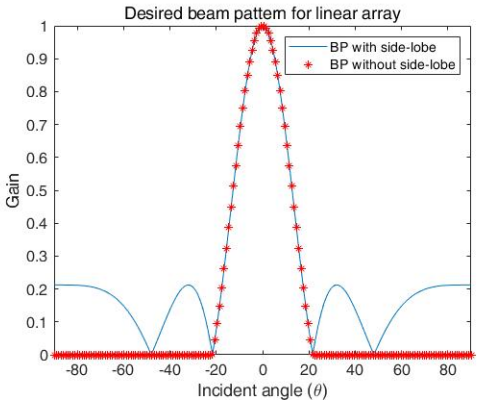


(C)

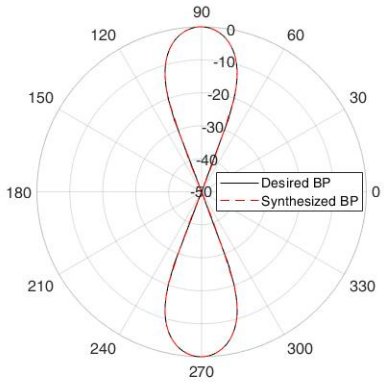
FIGURE 3.13: (a) Beam pattern at 16 kHz (in dB scale). (b) Beam patterns over frequency from 250 Hz to 16 kHz. (c) WNGs over frequency ($N = 100$, $\Delta_d = 0.015$ m and $c = 343.2$ m/s).

Selecting $N = 101$, $\Delta_d = 0.01$ m and $c = 343.2$ m/s. From (3.17), we obtain a frequency range,

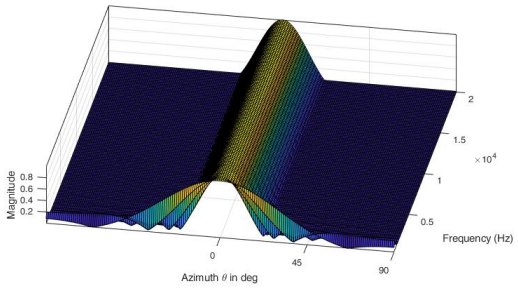
$$\frac{c}{N\Delta_d} \leq f < \frac{c(N+1)}{2N\Delta_d},$$



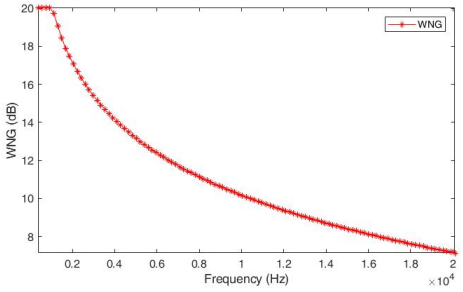
(A)



(B)

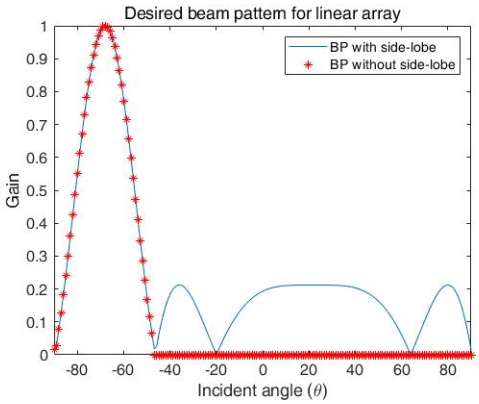


(C)

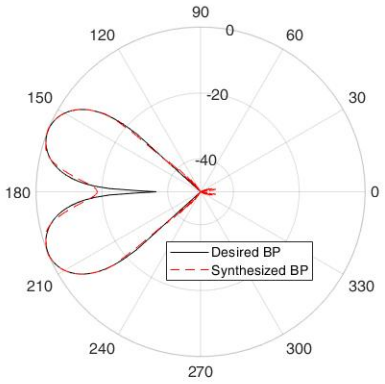


(D)

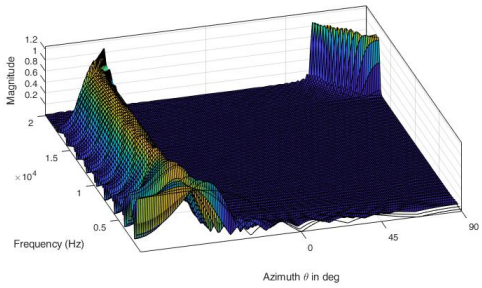
FIGURE 3.14: (a) Blue line is the beam pattern (BP) in Eq. (3.19), red "*" is the desired beam pattern without side-lobe. (b) Beam pattern at 16 kHz (in dB scale). (c) Beam patterns over frequency from 350 Hz to 20 kHz. (d) WNGs over frequency ($N = 101$, $\Delta_d = 0.01$ m and $c = 343.2$ m/s).



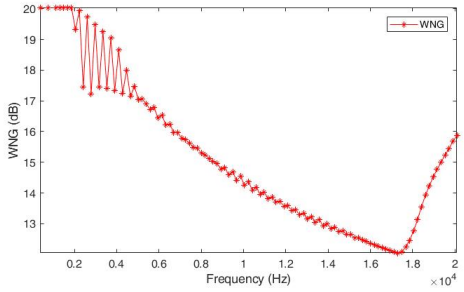
(A)



(B)



(C)



(D)

FIGURE 3.15: (a) Blue line is the beam pattern in Eq. (3.19) steering to $\theta = -70^o$, red "*" is the desired beam pattern without side-lobe. (b) Beam pattern at 16 kHz (in dB scale). (c) Beam patterns over frequency from 350 Hz to 20 kHz. (d) WNGs over frequency ($N = 101$, $\Delta_d = 0.01$ m and $c = 343.2$ m/s).

$$339.8 \text{ Hz} \leq f < 17329.9 \text{ Hz}.$$

We simulate for the frequency range from 250 Hz to 20 kHz. For this example, similar results with Example 2 (the beam patterns are distorted significantly at the low frequencies) could be observed in Fig. 3.14.

In order to observe the non-bijection effect of the inverse beam pattern transform at high frequencies, we examine for the steering case. The linear array steers to the incident angle $\theta = -70^o$. The desired beam pattern is shown in Fig. 3.15a where the red "*" is the desired beam pattern. The beam pattern at 16 kHz, beam patterns over frequency and WNGs over frequency are shown in Fig. 3.15b, Fig. 3.15c and Fig. 3.15d, respectively.

3.6 Conclusions

The chapter has proposed the inverse beam pattern transform for the uniform array, which deduces the sensor coefficients from an arbitrary desired beam pattern $B(\theta, \phi)$,

$$w(x, y, f) =$$
$$\mathcal{W}\{B(\theta, \phi), f\} := \mathcal{F}^{-1}\{\mathcal{T}\{\mathcal{C}\{B(\theta, \phi), f\}\} \cup \mathbb{A}\}.$$

It is worthwhile to note that the computation of the inverse beam pattern function mainly contains the coordinate transform and inverse Fourier transform, it has high potential to be realized in real-time applications. As based on the bijection of the inverse beam pattern functions and broadband beamforming perspective, the bound constraints for spatial sampling are proposed

$$\begin{cases} \Delta_d < \frac{(N+1)\lambda_{min}}{2N}, & N \text{ is odd}, \\ \Delta_d < \frac{\lambda_{min}}{2}, & N \text{ is even}. \end{cases} \tag{3.20}$$

For the lower bound, from the inequality (3.17) we have,

$$\Delta_d \geq \frac{\lambda_{max}}{N}$$

the inter-distance of sensors should be greater than or equal to the largest wavelength divided by the number of sensors in the vertical or horizontal axis. However, from simulation results the chapter suggests that the planar array (linear array) needs more than five points (three points) inside the smallest hemisphere (half circle), so

$$\begin{cases} \Delta_d \geq \frac{m\lambda_{max}}{N}, \frac{N+1}{2} > m >> 1, & N \text{ is odd}, \\ \Delta_d \geq \frac{m\lambda_{max}}{N}, \frac{N}{2} > m >> 1, & N \text{ is even}. \end{cases} \tag{3.21}$$

the parameter m determines the number of encoding points $|\mathbb{P}_{f_{min}}|$ for the beam pattern at the lowest frequency. For the linear array

$$|\mathbb{P}_{f_{min}}| = 2\lfloor m \rfloor + 1, \tag{3.22}$$

and for the planar array

$$|\mathbb{P}_{f_{min}}| = 1 + 4\sum_{k=0}^{\lfloor m \rfloor} \lfloor \sqrt{m^2 - k^2} \rfloor \tag{3.23}$$

(see Appendix 3.7:The Number of Encoding Points).

3.7 Appendix: The Number of Encoding Points

For the linear array, it is obvious (see Fig. 3.8).
For the planar array, let us consider the right half circle with radius m, as shown in Fig. 3.16.

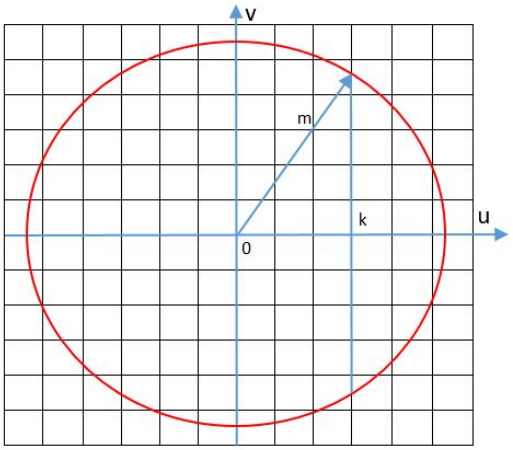


FIGURE 3.16: The number of integer points on the line $u = k$.

Let k be an integer number and $0 \leq k \leq \lfloor m \rfloor$, the number of integer points on the line $u = k$ is

$$N_k = 2\lfloor \sqrt{m^2 - k^2} \rfloor + 1.$$

The number of integer points belong to the right half circle

$$N_r = \sum_{k=0}^{k=\lfloor m \rfloor} N_k = \sum_{k=0}^{k=\lfloor m \rfloor} (2\lfloor \sqrt{m^2 - k^2} \rfloor + 1).$$

The circle is symmetric, so the number of integer points belongs to the left half circle $N_l = N_r$. The total integer points belong to full circle is $2N_r - N_0$, where N_0 is the common points of two halves, thus we achieve the result in (3.23).

Chapter 4

Sparse Arrays with Frequency-Independent Beam Pattern

Beamformer performs spatial filtering to preserve the desired signal while suppressing interfering signals and noise arriving from directions other than the direction of interest. However, the beam pattern of the conventional beamformer is dependent on the frequency of the signal. It is common to use dense or uniform arrays for a broadband signal to achieve some essential performances together, such as frequency-independence, high white noise gain, high directivity factor, high front-to-back ratio, etc. Recently, the interest in sparse arrays is growing, mainly due to the capacity to reduce the number of sensors with acceptable deteriorated performance. Nevertheless, in general, finding a suitable sparse array layout is still a challenging task. Many studies have focused on optimization procedures to seek a sparse array deployment. This chapter presents an alternative approach to determine the location of sensors. Starting with a weight spectrum of a virtual uniform array, some techniques are used, such as analyzing the weight spectrum to determine the critical sensors, applying the clustering technique to group the sensors into different groups, and selecting the representative sensors for each group. After the sparse array deployment is specified, the optimization technique is applied to find beamformer coefficients. The proposed method helps to save the computation time in the design phase, and its beamformer performance outperforms other state-of-the-art methods.

4.1 Introduction

Frequency-independent (FI) beamforming is a technique used to obtain signals with a wide frequency band. It maintains signal integrity and spatial selectivity over the band of interest. It has been widely used in many applications such as radar, sonar, communication and acoustic systems where the bandwidth of signals is several octaves. In the past few decades, many design methodologies for a dense and uniform array (DUA) with FI beamforming have been proposed and widely applied in many practical applications [Nor+14; LW08; WKW01; GHB18; LW08; DBWW95; Yan06; SYH07; SC07; EMK09; Li+13; YZL11]. Besides FI beamforming, other performances of the array such as directivity factor (DF), white noise gain (WNG) are also important. Sometimes these performances contradict each other, e.g., a conventional

beamformer with high DF has a large amplification of sensor noise and sensor position errors (low WNG). Given a reference beam pattern, a uniform array with a small equidistance ensures the FI beam pattern at high frequencies, while a uniform array with a big equidistance ensures the FI beam pattern at low frequencies. In case of a wide frequency range, the uniform array needs to be dense and large to cover both the near and far distances of sensors; therefore, the traditional techniques sometimes require a large number of sensors. For FI in acoustic applications, another concept is the differential microphone array (DMA) which is widely used in practical applications. The conventional DMA is based on the spatial derivatives of the acoustic pressure field [Elk00; Elk00; TE01; Buc02]. Since the sensor spacing of the DMA is much smaller than the acoustic wavelength, then the DMA is small in size which can be easily mounted into other devices. On the other hand, based on short-time Fourier transform, spatial filtering is applied to form a differential beamformer in each subband [BJ12; BCC15; ZBC14; CPC15; Hua+20]. The order of the differential beamformer could be designed by selecting the number of beam patterns' nulls and the type of differential beamformer could also be obtained by assigning the null positions and/or changing the optimization objective function such as maximum front-to-back ratio for supercardioid microphone, maximum directivity index for hypercardioid, etc. Besides, some studies show that, for the circular or arbitrary array, the differential beamformer has ability to steer to any direction of the sensor plane [EP97; HBC17; HCB18; Bor+20]. However, conventional DMAs or arrays with differential beamformers are still sensitive to the white noise and array mismatch at low frequencies, the DF is degraded at high frequencies, the higher order systems are somewhat impractical and their beam pattern is restricted to the function of a differential beamformer [Elk00].

Over the past years, the interest in the sparse arrays is growing. Many sparse array designs for broadband beamformers have been developed previously [Cho95; ZGET04; Dob08; CT12; CT14; LYF13; Liu+15a; HL13; HL14; Liu+15b; Buc+18], which are briefly mentioned as follows. A straightforward array structure of FI beamforming is based on the concept of harmonic nested arrays [Cho95], [ZGET04], where the final array is the union of several uniform arrays, each one matches to a different frequency subband. With the goal of reducing the number of sensors, Doblinger [Dob08] proposed a sparse array based on a superdirective beamformer [Elk00] and simulated annealing [PWV08] where the cost function achieved from the superdirectivity beamformer of a DUA. A similar idea to Doblinger's idea was also presented by Marco Crocco and Andrea Trucco in [CT12], [CT14], they presented more parameters in the cost function of the simulated annealing to adjust the beamformer performances toward different objectives such as FI, DF, WNG, sidelobe and main-lobe. Later, in [LYF13], a genetic algorithm [ES+18] together with a gradient-based method was applied for microphone array placement, this approach compromises between searching the optimal solution rapidly and jumping out from the local minima. A design of FI beamforming for linear arrays based on the generalized matrix pencil method was proposed in [Liu+15a]; this approach performs the singular value decomposition of a reference beam pattern to determine how many array elements are required, and then applying the generalized matrix pencil method to determine the positions and weight-valued of the sensors. Recently, as based on the compressed sensing [Boc+15], [Mig14] concept, Hawes and Liu [HL13] proposed the l_1-norm to find the position of the sensors and their coefficients together. To extend the sparsity of the conventional l_1-norm, Hawes and Liu [HL14] and Liu *et al.* [Liu+15b] applied a sequence l_1-norm method [CWB08] which is iteratively reweighting the conventional l_1-norm in order to achieve the solutions that

are closer to the sparse solution. In this chapter, the method in [Liu+15b] is called the *coherence sparse array design*. More recently, Yaakov *et al.* [Buc+18] also applied the sequence l_1-norm method to find the positions of sensors at different subbands and then merging these SAs by a clustering algorithm to obtain the final sparse array. In this chapter, this approach is called *incoherence sparse array design*.

Broadly speaking, many studies have focused on the convex or stochastic optimization approaches to determine the sparse array deployments. Despite the success of these techniques, the results depend heavily on the strength of the optimization algorithms and it is hard to say that the obtained solution is close to the optimal solution. For example, based on a compressed sensing framework, applying the l_1-norm is efficient only if the measurement system (measurement matrix) has a low restricted isometry property [CRT06; Don06], but the restricted isometry property itself is not easy to verify, and it is still an open question in the sparse array design's model. For the stochastic approach, it is not so efficient for a large search-space system or it does not guarantee that its solution is close to the optimal solution if the number of sensors is large. Furthermore, in some circumstances, the optimization or stochastic approaches are not only difficult to tune the parameters, but it is also computationally intensive to solve them with subject to the sparse solution. To the best of the author's knowledge, although the DUAs work well with the FI beam pattern, few approaches use the information of DUAs as the input for the sparse array design. In this work, a new method that takes the information of the DUA as the input to design the sparse array is proposed. The general idea is shown in Fig. 4.1.

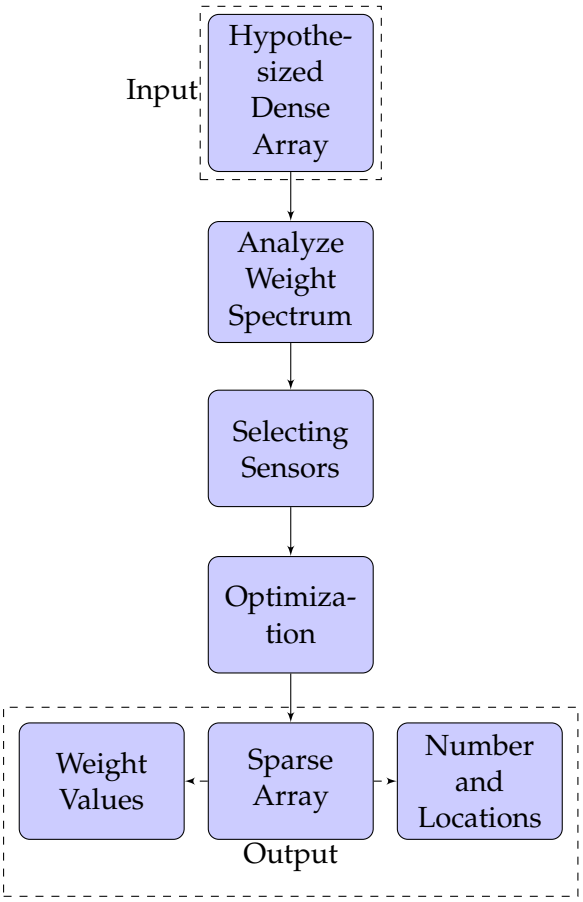


FIGURE 4.1: Flowchart for the proposed sparse array design.

- First, a DUA with the broadband beamforming is designed. Spatial filters of the DUA at different frequencies constitute a weight matrix.
- Then, we analyze the spectrum of the weight matrix to identify the critical sensors, .i.e, the "Analyze Weight Spectrum" block in Fig. 4.1 can be detailed as follows. Principal component analysis (PCA) [WEG87] is applied to reduce the size of the weight matrix because this matrix does not have a full-rank. After compressing the weight matrix, we identify the rows in the compressed matrix which was superior energy corresponding with the critical sensors that need to be kept in the sparse array.
- Next, K-means clustering algorithm [HW79] is applied to categorize the remaining sensors in the DUA into different groups [Buc+18]. For each group, the sensors which have the closest distances to the group's centroid are also added to the sparse array. Therefore, the number of sensors in the sparse array is proportional to the number of groups. Moreover, if the number of groups increases, the total distance from the points to the centroid points decreases. This decrease is proportional to the error-tolerances of the FI beam pattern.
- Once the positions of the sensors in the sparse array are identified, optimization approaches are used to find the sensor coefficients at every frequency. In order to examine the proposed method, some examples with different configurations (broad-side, end-fire and planar array) are given.

The layout of this chapter is organized as follows. Section 4.2 presents the signal model. Section 4.3 describes the design method. Section 4.4 presents some evaluation metrics. The simulations and their results are provided in Section 4.5. The conclusion is drawn in Section 4.6.

Notations: The vectors (matrices) are represented by lower-case (upper-case) bold characters. $(.)^T$ and $(.)^H$ denote the transpose and Hermitian transpose, respectively. $j = \sqrt{-1}$ is the imaginary unit. $[\mathbf{A}]_{i,j}$ denotes the (i,j) entry of $\mathbf{A}$. $\mathbf{A}(i,:)$ is the row i^{th} of $\mathbf{A}$. $\mathbf{A}(:,i)$ is the column i^{th} of $\mathbf{A}$. $\text{diag}(\mathbf{A})$ returns a column vector formed from the main diagonal of $\mathbf{A}$. $\|.\|_2$ is the l_2-norm. $|.|$ is the absolute operator. $\text{mean}(\mathbf{v})$ is mean value of the elements in vector $\mathbf{v}$.

4.2 Signal Model

Consider the far-field wave, then the wave reaching the sensors is planar. Without loss of generality, we assume that the plane of array consolidates with X-Y plane in the Cartesian coordinate. A look direction described by $S = (\phi_0, \theta_0)$ is specified by the azimuth angle ϕ_0 and the elevation angle θ_0. The beam pattern in an interested bandwidth Ω is given by

$$b(\phi, \theta, \omega) = \mathbf{w}^H(S, \omega)\mathbf{d}(\phi, \theta, \omega), \forall \omega \in \Omega, \tag{4.1}$$

where

$$\mathbf{w}(S, \omega) = [w_1(\phi_0, \theta_0, \omega), ..., w_M(\phi_0, \theta_0, \omega)]^T$$

is a weight vector (from now on, the manuscript drops the explicit S from the expression $\mathbf{w}(S, \omega)$ in order to keep the notation uncluttered), M is the number of sensors.

$$\mathbf{d}(\phi, \theta, \omega) = [e^{-j\omega t_1}, .., e^{-j\omega t_M}]^T$$

is a steering vector at the direction specified by the azimuth and elevation angle $\rho = (\phi, \theta)$, $\rho \in \Theta$ (Θ is the operating space of beamforming).

$$t_k = \frac{x_k sin\theta cos\phi + y_k sin\theta sin\phi}{c}, k = 1, ..., M$$

is the delay of wave propagation from the origin to the sensor k^{th} at the position (x_k, y_k), c is the wave speed.
From M sensors with equidistance in the DUA, we select $K(<< M)$ sensors whose indices belong to the set $\mathbb{S}_K$,

$$\mathbb{S}_K := \{i_k : i_k \in [1, .., M], k = 1, ..., K\}.$$

The beam pattern at the angular frequency ω is formed by the sensors in $\mathbb{S}_K$,

$$b_S(\phi, \theta, \omega) = \mathbf{w}_S^H(\omega)\mathbf{d}(\phi, \theta, \omega) \tag{4.2}$$

where $\mathbf{w}_S$ having K non-zero elements and $M - K$ zero elements is weight vector of the sparse array.
$\mathbf{W}_A$ size of $M \times J$ is the weight matrix of the uniform array that contains the weight vectors $\mathbf{w}(\omega_i), \omega_i \in \Omega, i = 1, ..., J$,

$$\mathbf{W}_A = [\mathbf{w}(\omega_1), \mathbf{w}(\omega_2), ..., \mathbf{w}(\omega_J)].$$

$\mathbf{W_S}$ size of $M \times J$ is the weight matrix of the sparse array that contains the weight vectors $\mathbf{w_S}(\omega_i), \omega_i \in \Omega, i = 1, ..., J$,

$$\mathbf{W}_S = [\mathbf{w}_S(\omega_1), \mathbf{w_S}(\omega_2), ..., \mathbf{w}_S(\omega_J)].$$

Let $\mathbf{b}(\rho, \Omega)$ ($\mathbf{b}_S(\rho, \Omega)$) be the beam pattern vector of a DUA (sparse array) at single direction $\rho = (\phi, \theta)$ across the bandwidth of interest. From (4.1) and (4.2), we have

$$\mathbf{b}(\rho, \Omega) = \text{diag}(\mathbf{W}_A^H \mathbf{D}_\rho(\Omega)), \tag{4.3}$$

$$\mathbf{b}_S(\rho, \Omega) = \text{diag}(\mathbf{W}_S^H \mathbf{D}_\rho(\Omega)), \tag{4.4}$$

where $\mathbf{D}_\rho(\Omega) = [\mathbf{d}(\rho, \omega_1), \mathbf{d}(\rho, \omega_2), ..., \mathbf{d}(\rho, \omega_J)]$.

Suppose a reference beam pattern is given, then it is possible to design a DUA with M sensors, equidistance δ and the weight matrix $\mathbf{W_A}$ to assure that $\mathbf{b}(\rho, \Omega)$ is almost unchanged in Ω. The next section will present a method to find the set $\mathbb{S}_K = [i_1, i_2, ..., i_K]$ so that $\mathbf{b}_S(\rho, \Omega)$ in (4.4) is close to $\mathbf{b}(\rho, \Omega)$ in (4.3) for all $\rho \in \Theta$.

4.3 Design Method

A three-step method is proposed to design the sparse array: Analysis, Selecting and Optimization. This method finds the $\mathbb{S}_K$ and $\mathbf{w}_S$ separately. In the Analysis step, a DUA performing well with the FI beam pattern is designed and analyzed. In the Selecting step, the weight matrix $\mathbf{W}_A$ of the DUA is used as a basis to select the set $\mathbb{S}_K$. In the last step, the optimization technique is deployed to find the weight vector $\mathbf{w}_S$. It is worth noting that the optimization efforts to find the weight vector is negligible compared with that to find the sparse solution (position of sensors).

4.3.1 Analysis

In this section, a design method for the planar DUA with an arbitrary FI beam pattern is briefly presented (more details could be found in Chapter 3. Let $w(n_1, n_2, \omega)$ be an element of the weight vector $\mathbf{w}(\omega)$, which associates with the sensor at position $(n_1\delta, n_2\delta)$, where (n_1, n_2) are the indices in X and Y axis,

$$-\frac{(N-1)}{2} \leq n_1, n_2 \leq \frac{(N-1)}{2}, n_1, n_2 \in \mathbb{Z}.$$

Assuming that $N = \sqrt{M}$ is an odd number, N is the number of sensors in X or Y axis (note that for the linear array $N = M$). (4.1) is presented by,

$$b(\theta, \phi, \omega) = \sum_{n_1, n_2 = -\frac{(N-1)}{2}}^{\frac{(N-1)}{2}} w(n_1, n_2, \omega)$$

$$\times e^{-j\frac{\omega}{c}(n_1\delta \sin\theta\cos\phi + n_2\delta \sin\theta \sin\phi)}. \tag{4.5}$$

Let us define the spherical coordinate where a point is presented by ϕ, θ and radius R,

$$R = \frac{fN\delta}{c}, \tag{4.6}$$

where $f = \frac{\omega}{2\pi}$. Let us define

$$u = R\sin\theta\cos\phi, \quad v = R\sin\theta\sin\phi. \tag{4.7}$$

(4.3.1) is equivalent to

$$b(u, v, f) = \sum_{n_1, n_2 = -\frac{(N-1)}{2}}^{\frac{(N-1)}{2}} w(n_1, n_2, f) e^{-j\frac{2\pi}{N}(n_1 u + n_2 v)} \tag{4.8}$$

It is clearly seen that $b(u, v, f)$ is a 2-Dimensional (2D) Discrete Fourier transform of $w(n_1, n_2, f)$ with respect to the variables n_1, n_2. Therefore, $w(n_1, n_2, f)$ is achieved by applying inverse Fourier transform of $b(u, v, f)$. From now on, we use notation $b_f(u, v)$ instead of using $b(u, v, f)$.

As presented in the equation (4.8), u, v are frequency indices in 2D frequency-domain, therefore they are integer numbers. Intuitively, one can imagine that the variables n_1, n_2 build planar grid points associated with grid points u, v in the Cartesian coordinate and every grid point in the u-v plane is associated with a sensor.

At a single frequency, R is constant, (4.7) is the formula transforming a surface $b_R(\theta, \phi)$ in the spherical coordinate to a surface $b_f(u, v)$ in the Cartesian coordinate. From this observation, we could propose a method to obtain the weights from a reference beam pattern. First, the reference beam pattern is translated to gain functions in the spherical coordinate, and then the gain functions in the spherical coordinate are transformed to the Cartesian coordinate. Finally, the inverse Fourier transform is applied to achieve the weight vector of the uniform array.
Let $B(\theta, \phi)$ be a reference beam pattern, below are some properties:

- In the spherical coordinate, radius R is proportional to frequency f (a single value R is mapping with a single value f and vice versa).

- A reference beam pattern $B(\theta, \phi)$ is presented by different gain functions in the spherical coordinate $b_R(\theta, \phi)$: Different radii R are associated with different gain functions.
- A gain function in the spherical coordinate $b_R(\theta, \phi)$ is located in a hemisphere with radius R (the gain values could be presented by the color on the surface of the hemisphere, as shown in Fig. 4.2).

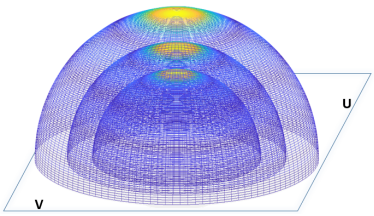


FIGURE 4.2: Example of gain functions associate with different frequencies in the spherical coordinate: Concentric hemispheres with different radii.

- $b_f(u, v)$ is the gain function in the Cartesian coordinate at frequency f, as shown in Fig. 4.3. $b_f(u, v)$ is $b_R(\theta, \phi)$ after applying the coordinate transform.

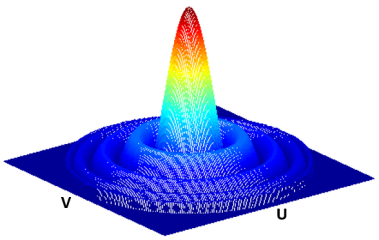


FIGURE 4.3: A gain function in the Cartesian coordinate at a frequency.

Note that, for the linear sensor array, the reference beam pattern is translated to the polar coordinate system, and then, it is transformed to the Cartesian coordinate system.

The FI beam pattern is assured if the coordinate transforms do not change the gain function over frequency/radius of the hemisphere. Therefore, the radius in the spherical coordinate should not be greater than the boundary of grid points in the plan u-v and at least more than five grid points are inside the smallest hemisphere (the smallest hemisphere associates with the lowest frequency), as shown in Fig. 4.4,

$$1 \leq R \leq \frac{(N-1)}{2}.$$

From (4.6), we have

$$1 \leq \frac{fN\delta}{c} \leq \frac{(N-1)}{2},$$

$$\frac{c}{N\delta} \leq f \leq \frac{c(N-1)}{2N\delta}.$$

The origin of the Cartesian coordinate is always consolidated with a sensor's posi-

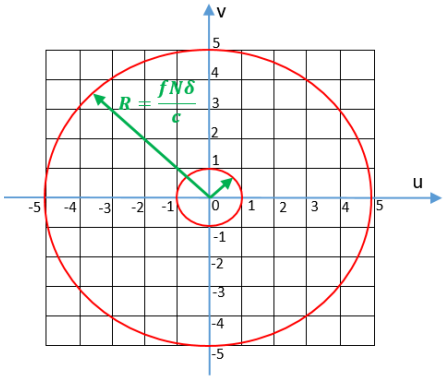


FIGURE 4.4: Sensor indices in the plan u-v of the Cartesian coordinate and the radius of the hemispheres.

tion. For the case of N odd, the origin is at the center of the array. For the case of N even, the origin is not at the center of the array, therefore the maximum radius is $\frac{(N-2)}{2}$. We yield the following constraints for both cases:

$$\begin{cases} \frac{mc}{N\delta} \leq f \leq \frac{c(N-2)}{2N\delta}, & N \text{ is even}, \\ \frac{mc}{N\delta} \leq f \leq \frac{c(N-1)}{2N\delta}, & N \text{ is odd}. \end{cases} \tag{4.9}$$

A parameter $m \geq 1$ is introduced. If $m = 1$, the lower bound $\frac{c}{N\delta} \leq f$ is to assure at least five u-v grid points inside the smallest hemisphere. If the reference beam pattern has higher-order, it needs more than five values to encode its shape. Therefore, the lower bound constraint could be set more strictly $\frac{mc}{N\delta} \leq f$ with $m >> 1$. A simple way to express this is that the bigger m, the more grid points inside the smallest hemisphere or the higher resolution of synthesized beam pattern at the low frequencies.

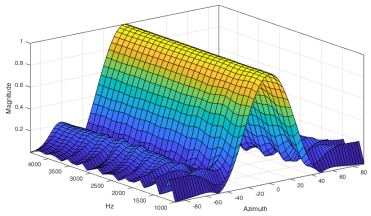
In summary, the CT method consists of four steps that apply to all $\omega_i \in \Omega$ (some examples of this method are given in [Son20]):

- Step 1: Define a reference beam pattern $B(\phi, \theta)$.
- Step 2: At a single R, presenting the reference beam pattern $B(\phi, \theta)$ to a gain function $b_R(\phi, \theta)$ in the spherical coordinate.
- Step 3: Gain function in the Cartesian coordinate $b_f(u, v)$ is achieved by transforming the surface $b_R(\phi, \theta)$ in the spherical coordinate to the Cartesian coordinate.
- Step 4: Apply the inverse Fourier Transform for $b_f(u, v)$ to achieve $\mathbf{w}(\omega)$.

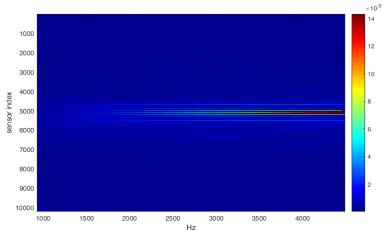
After J iteratives for J frequencies in Ω, a weight matrix is obtained

$$\mathbf{W}_A = [\mathbf{w}(\omega_1), \mathbf{w}(\omega_2), ..., \mathbf{w}(\omega_J)].$$

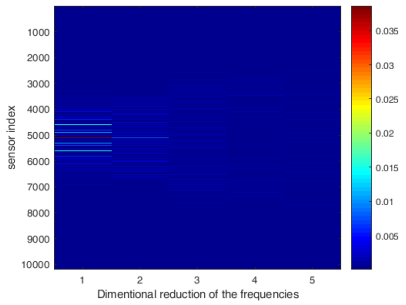
The weight matrix achieved from this step decides the possibility of reducing the amount of the sensors in the sparse array. Fig. 4.5(b) presents the weight spectrum $\mathbf{W}_A$, which are discussed later in Section 4.5.3 (101×101 sensors are rearranged into the vertical of the matrix). If there are only some rows having energy dominate other rows in the weight matrix, the reference beam pattern has a high potential to reduce a lot of sensors in the sparse array. If the energy spectrum of the weight matrix spreads



(A) Cross-cut of beam patterns across frequency.



(B) Weight matrix.



(C) Dimensional reduction matrix ($\alpha = 0.98$).

FIGURE 4.5: Planar array: Analyzing the DUA.

across the rows and columns, there is less chance to reduce the number of sensors in the sparse array. Therefore, in general, using optimization techniques to find the active sensors in this step may be inefficient. Note that designing a DUA in this step uses less computation time than solving the optimization with sparse solutions, since this coordinate transform (CT) method mainly uses coordinate transform and inverse Fourier transform for its computation.

4.3.2 Selecting

Before choosing important sensors, one could apply the dimensional reduction algorithm for the weight matrix to save the computation time of the selecting algorithm. If $\mathbf{W}_A$ is a rank-deficient matrix with decaying singular-values, using PCA is a good option for dimensional reduction in terms of keeping the essential information. In order to show that $\mathbf{W}_A$ is not a full-rank matrix, one may look at the correlation matrix of $\mathbf{R}_A$ [Buc+18]:

$$\mathbf{R}_A = \mathbf{W}_A^H \mathbf{W}_A$$

of size $J \times J$. If $\mathbf{R}_A$ is low rank with decaying eigenvalues. Let $\sigma_i, i = 1, ..., r$ be the singular-values of matrix W_A, where r denotes its rank, then $\lambda_i = \sigma_i^2, i = 1, ..., r$ are the eigenvalues of matrix $\mathbf{R}_A$.
The input of the PCA algorithm is the weight matrix $\mathbf{W}_A$ of size $M \times J$. Define matrix

$\overline{\mathbf{W}}_A$ to be the central version of $\mathbf{W}_A$:

$$\overline{\mathbf{W}}_A(:,j) = \mathbf{W}_A(:,j) - \text{mean}(\mathbf{W}_A(:,j)), \forall j \in J$$

The output of the PCA algorithm is the dimensional reduction matrix,

$$\mathbf{W}_R = \overline{\mathbf{W}}_A \mathbf{U},$$

where $\mathbf{U}$ is a matrix size $J \times L$, containing L eigenvectors corresponding to L largest eigenvalues, $L << J$.
The column dimension L of the matrix $\mathbf{U}$ is set according to the following criteria: $i^{th}(i \leq L)$ first eigenvalues of $\mathbf{R}_A$ in decreasing order $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_L$ (if $i > r$, then $\lambda_i = 0$) are selected which satisfy [Buc+18]:$0.6 < \alpha < 1$,

$$\alpha = \frac{\sum_{j=1}^{L} \lambda_j}{\sum_{j=1}^{J} \lambda_j}.$$

After having dimensional reduction of the weight matrix $\mathbf{W}_R$, as shown in Fig. 4.5(c), we present the way to select the critical sensors. The selection follows two criteria: Energy contribution in the weight matrix's spectrum and minimize the error-tolerance of the FI beam pattern.
Let $\mathbb{S}_{K_1}$ be a subset containing indices of K_1 sensors in the DUA which have the strongest energy span over row of $\mathbf{W}_R$ and let subset $\overline{\mathbb{S}}_{K_1} := \{1, ..., M\} \setminus \mathbb{S}_{K_1}$ be a subset which contains the sensor indices in the DUA other than the sensor indices in $\mathbb{S}_{K_1}$.
Let a spanning energy for a sensor be a sum of that sensor's energy across the frequency range, herein we use the dimensional reduction matrix as the input for the spanning energy's calculating,

$$\epsilon_i = \|\mathbf{W}_R(i,:)\|_2, i = 1, ..., M$$

and the maximum value of the spanning energy,

$$\epsilon_{\max} = \max(\epsilon_i), i = 1, ..., M.$$

The set of sensors having spanning energy is greater than a threshold,

$$\mathbb{S}_{K_1} := \{i : (\epsilon_i > \beta \epsilon_{\max}), i = 1, ..., M\}, \tag{4.10}$$

and the remaining sensors,

$$\overline{\mathbb{S}}_{K_1} := \{i : (\epsilon_i \leq \beta \epsilon_{\max}), i = 1, ..., M\},$$

where $0 < \beta < 1$ is a spanning energy factor for the sparse array. If β is close to 0, the sparse array is close to the DUA, it means more sensors from the DUA are taken. If β is close to 1, the sparse array only takes a few most important sensors from the DUA.

From $\overline{\mathbb{S}}_{K_1}$, we then apply a clustering algorithm over the rows of $\mathbf{W}_R(\overline{\mathbb{S}}_{K_1}, :)$ to categorize sensors in $\overline{\mathbb{S}}_{K_1}$ into K_2 groups. Each group contains the sensors having similar characteristics (similar pattern of the energy distributing over frequency). K-means clustering algorithm [HW79], which minimizes the within-cluster sums of point-to-centroid distances, is used. The sums of point-to-centroid distances could present the gap between the sparse array and the DUA in terms of FI beam pattern.

K-means algorithm divides $(M - K_1)$ sensors into K_2 groups. The larger K_2 leads to the smaller sum of distances to centroids, which means the larger K_2 is relative to the smaller error between the beam pattern of the sparse array and the beam pattern of the DUA. As will be mentioned again in Section 4.3.3, the setting of K_2 is as big as possible to approximate the reference beam pattern within a desired tolerance. For each group, one or several representative sensors which are closest to their group's centroid are chosen [Buc+18] to add to the subset $\mathbb{S}_{K_2}$. If the grids of the DUA is small enough, the discrepancy between the representative sensors and the centroid of the group will be small. It is possible to have a large amount of sensors in the DUA since it is the virtual sensor array. On the other hand, too many sensors in the DUA increases the computation time in the Analysis and Selecting step.
Finally, the set $\mathbb{S}_K$ is the union of the subset $\mathbb{S}_{K_1}$ and the subset $\mathbb{S}_{K_2}$,

$$\mathbb{S}_K = \mathbb{S}_{K_1} \cup \mathbb{S}_{K_2}. \tag{4.11}$$

K-means clustering algorithm does not guarantee to obtain the global minimum, its solution depends on the initial groups. Hence, it is required to run the algorithm multiple times with different initial conditions to find a good solution. However, it is time-consuming to find a good solution with a large number of sensors. Therefore, applying the stochastic techniques, such as the simulated annealing [PWV08] or genetic algorithm [ES+18], for initial values of K-means clustering algorithm is a good approach to reduce the running time and increase the chance to find the global optimal solution.

4.3.3 Optimization

In the previous step, the K indices of the sparse array are determined by (4.11), but the weights of these sensors are not determined. Now, the optimization techniques are used to find their complex-valued weights.

The discretized angle spaces are made uniformly, and P directions $\rho_i = (\phi_i \theta_i) \in \Theta$ are introduced that cover the entire direction of the beam pattern. U out of P directions are taken that cover the main-lobe region Θ_m, and let $\mathcal{K}^m$ be a set containing these directions,

$$\mathcal{K}^m = \{\rho_1, \rho_2, ..., \rho_U)\}$$

where superscript m stands for main-lobe. Similarly, $\mathcal{K}^s$ contains $P - U$ directions that cover the side-lobe region Θ_s is defined,

$$\mathcal{K}^s = \{\rho_{U+1}, \rho_{U+2}, ..., \rho_P\}$$

where superscript s stands for side-lobe. A steering matrix over direction is the matrix that contains the steering vectors over the angle space at an angular frequency ω_i is defined,

$$\mathbf{D}_{\omega_i}(\Theta) = [\mathbf{d}(\rho_1, \omega_i), \mathbf{d}(\rho_2, \omega_i), ..., \mathbf{d}(\rho_P, \omega_i)].$$

The steering matrix over direction for the main-lobe region Θ_m at the angular frequency ω_i: $\mathbf{D}_{\omega_i}(\mathcal{K}_i^m)$.
The steering matrix over direction for the side-lobe region Θ_s at the angular frequency ω_i: $\mathbf{D}_{\omega_i}(\mathcal{K}_i^s)$.
The reference beam pattern for the main-lobe region Θ_m is a vector: $\mathbf{b}_d^m$.
The reference beam pattern for the side-lobe region Θ_s is a vector: $\mathbf{b}_d^s$.

The main-lobe constraints are given by,

$$\mathcal{C}_1 : \left\| \mathbf{b}_d^m - \mathbf{D}_{\omega_i}^H(\mathcal{K}^m)\mathbf{w}_S(\omega_i) \right\|_2 \leq \epsilon 1(\omega_i), \forall \omega_i \in \Omega.$$

The side-lobe constraints are given by,

$$\mathcal{C}_2 : \left\| \mathbf{b}_d^s - \mathbf{D}_{\omega_i}^H(\mathcal{K}^s)\mathbf{w}_S(\omega_i) \right\|_2 \leq \epsilon 2(\omega_i), \forall \omega_i \in \Omega$$

where $\epsilon 1(\omega_i), \epsilon 2(\omega_i)$ are the error-tolerances for the main-lobe and side-lobe, respectively. As mentioned in the Analysis step, the sums of point-to-centroid distances is correlated with $\epsilon 1(\omega_i), \epsilon 2(\omega_i)$. The smaller the sums of point-to-centroid distances is, the smaller $\epsilon 1(\omega_i), \epsilon 2(\omega_i)$ could be set.
The distortion-less response constraint for the look direction $S = (\phi_0, \theta_0)$ is given by,

$$\mathcal{C}_3 : \mathbf{w}_S^H(\omega_i)\mathbf{d}(\phi_0, \theta_0, \omega_i) = 1, \forall \omega_i \in \Omega.$$

The optimization may achieve the vector $\mathbf{w}_S(\omega_i)$ containing the elements with a big number. In such a scenario, the array is sensitive with white noise, therefore the WNG constraint is also needed,

$$\mathcal{C}_4 : \mathbf{w}_S^H(\omega_i)\mathbf{w}_S(\omega_i) \leq \gamma(\omega_i), \forall \omega_i \in \Omega.$$

A reasonable choice is to minimize the white noise's power $\mathcal{C}_4$, subject to the remaining constraints $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3$ [Buc+18].

$$\begin{aligned} &\underset{\mathbf{w}_S(\omega_i)}{\text{minimize}} \quad : \mathbf{w}_S^H(\omega_i)\mathbf{w}_S(\omega_i) \\ &\text{subject to} \\ &\left\| \mathbf{b}_d^m - \mathbf{D}_{\omega_i}^H(\mathcal{K}^m)\mathbf{w}_S(\omega_i) \right\|_2 \leq \epsilon 1(\omega_i) \\ &\left\| \mathbf{b}_d^s - \mathbf{D}_{\omega_i}^H(\mathcal{K}_i^s)\mathbf{w}_S(\omega_i) \right\|_2 \leq \epsilon 2(\omega_i) \\ &\mathbf{w}_S^H(\omega_i)\mathbf{d}(\phi_0, \theta_0, \omega_i) = 1 \end{aligned} \tag{4.12}$$

This is a convex optimization problem, it can be efficiently solved by using a MATLAB toolbox, such as CVX [GBY09], [GB08].

4.4 Evaluation metrics

The synthesized pattern would be affected by some random errors in practical arrays, such as position errors, excitation mismatch, non-ideal sensor characteristics. As is well known, the WNG is a useful measure for the robustness against the random errors. That is given by [Elk00],

$$\mathcal{W}\left(\mathbf{w}_S(\omega)\right) = \frac{\left|\mathbf{w}_S^H\left(\omega\right)\mathbf{d}(\phi_0, \theta_0, \omega)\right|^2}{\mathbf{w}_S^H\left(\omega\right)\mathbf{w}_S\left(\omega\right)}.$$

Another important measure which qualifies the ability of the beamformer to suppress the diffuse noise is the DF [Elk00],

$$\mathcal{D}(\mathbf{w}_S(\omega)) = \frac{\left|\mathbf{w}_S(\omega)^H\mathbf{d}(\phi_0, \theta_0, \omega)\right|^2}{\mathbf{w}_S(\omega)^H\mathbf{\Gamma}(\omega)\mathbf{w}_S(\omega)^H},$$

where $\mathbf{\Gamma}(\omega)$ is the pseudo-coherence matrix of the diffuse noise field,

$$[\mathbf{\Gamma}(\omega)]_{i,j} = \text{sinc}\left(\frac{\omega d_{ij}}{c}\right)$$

where d_{ij} is the distance between sensor i^{th} and sensor j^{th}.

In the case of diffuse noise, the noise energy is depended on the shape of the beam pattern, if the beam patterns are almost identical then the DFs are also almost identical. If the target of the design method is to approximate the synthesized beam pattern to the reference beam pattern, so it is better to present a new measure which can consider this objective. In this work, beam pattern error average (BPE) indicates the average of the difference between the synthesized beam pattern $b_r(\rho_i, \omega)$ and the desired beam pattern $b_d(\rho_i, \omega)$ at a single angular frequency ω,

$$\mathcal{E}(\omega) = \frac{\sum_{\rho_i \in \Theta} |b_r(\rho_i, \omega) - b_d(\rho_i, \omega)|}{P},$$

where P is the size of Θ. The ideal value of $\mathcal{E}(\omega)$ is zero.

4.5 Numerical Simulations

In this section, three examples are provided. *Example A* is for the broad-side linear array, the performances of the proposed method are compared with the performances of the uniform arrays and coherence sparse array design in [Liu+15b]. *Example B* is for the end-fire linear array, the performances of the proposed method are compared with the performances of the uniform arrays and coherence sparse array design in [Buc+18]. *Example C* is for the planar array, the performances of the proposed method are compared with the performances of the uniform arrays with a higher number of sensors. In the remainder of this chapter, the blank area or disconnection line in the figure means that the optimizations are infeasible.

4.5.1 Sparse Broad-side Linear Array

For comparison, a similar reference beam pattern used in [LW08], [Liu+15b] is taken.

Analysis Step

The reference beam pattern is defined,

$$b(\theta) = \sum_{m=-3}^{3} h_m e^{-jm\pi \sin\theta},$$

where the coefficients

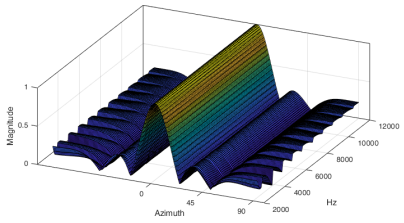
$$\{h_m : m = -3, ..., 3\} = [0.0307, 0.2028, 0.1663, 0.2004, 0.1663, 0.2028, 0.0307].$$

From (4.9), configuration of the DUA is selected: $N = M = 101$ sensors, $\delta = 0.01$ meter, $c = 340$ m/s. The frequency range of the DUA which ensures FI beamformer is given by,

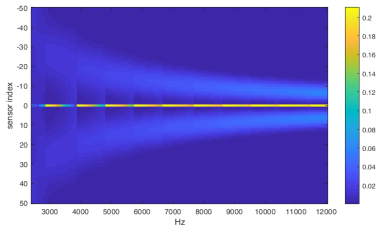
$$\frac{mc}{N\delta} \leq f \leq \frac{c(N-1)}{2\delta N}, m \geq 1$$
$$337m \text{ Hz} \leq f \leq 16832 \text{ Hz}, m \geq 1.$$

Herein, the frequency range $\Omega = \{2.4\,\text{kHz}, \ldots, 12\,\text{kHz}\}$ of the coherence sparse array design in [Liu+15b] is also used to select the important sensors. The beam pattern of the DUA across frequency are presented in Fig. 4.6(a) and the corresponding weight matrix spectrum is shown in Fig. 4.6(b). With $\alpha = 0.99$, the dimensional reduction of the weight matrix is presented in Fig. 4.6(c). In this figure, the vertical axis is the sensor's indices $\{1, 2, \ldots, 100, 101\}$.

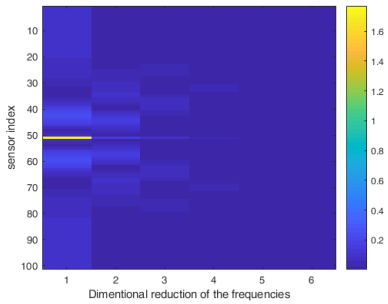
Selecting Step



(A) Cross-cut of beam patterns across frequency.



(B) Weight matrix.



(C) Dimensional reduction matrix ($\alpha = 0.99$).

FIGURE 4.6: Broad-side linear array: Analyzing the DUA.

Set $\beta = 0.7$, from (4.10) one sensor with index $\mathbb{S}_{K_1} = \{51\}$ is chosen. The remaining sensors in the DUA are divided into $K_2 = 5$ groups by K-means clustering algorithm. In each group, two representative sensor which are closest to the centroid of its group are chosen because we aim to design a symmetric array layout, that is, the linear sparse array contains two symmetric halves. Then. we obtain the subset $\mathbb{S}_{K_2} = \{12, 26, 31, 40, 45, 57, 62, 71, 76, 90\}$. Finally, the total number of sensors in new sparse array is 11. For comparison, four array layouts having the same amount of sensor $M = 11$ are considered, as shown in Fig. 4.7: The new sparse array (red circles) , small size uniform array (SUA) containing sensors located at close distances (green circles), big size uniform array (BUA) containing sensors located at far distances (blue circle) and coherence sparse array design design (black *).

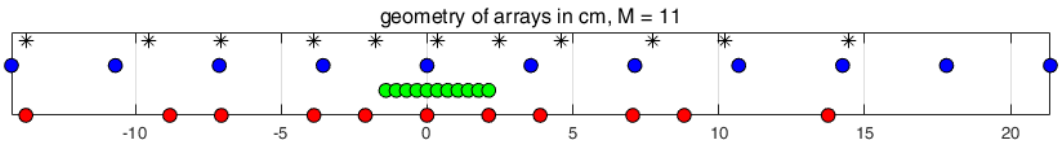


FIGURE 4.7: The broad-side linear array deployments: proposed symmetric sparse array (red circle), SUA (green circle) and BUA (blue circle) and coherence sparse array design (black *).

Optimization Step

We run the simulation for two cases.

Case 1. *beam pattern error-tolerances are relaxed.*

The error-tolerances are selected to assure the acceptable FI beam pattern and the optimizations are feasible for the proposed sparse array and coherence sparse array design in the bandwidth of interest.
The main-lobe error-tolerances for $\theta \in [70^o, 110^o]$, $\epsilon 1(\omega_i) = 0.006U, \quad \forall \omega_i \in \Omega$. The side-lobe error-tolerances for $\theta \in [0^o, 70^o) \cup (120^o, 180^o]$, $\epsilon 2(\omega_i) = 0.016(P - U), \quad \forall \omega_i \in \Omega$ where $P = 180$ (the resolution of azimuth angle $\Delta\theta = 1^o$), $U = 41$.
For comparison, the same error-tolerances apply to all sensor arrays. The synthe-

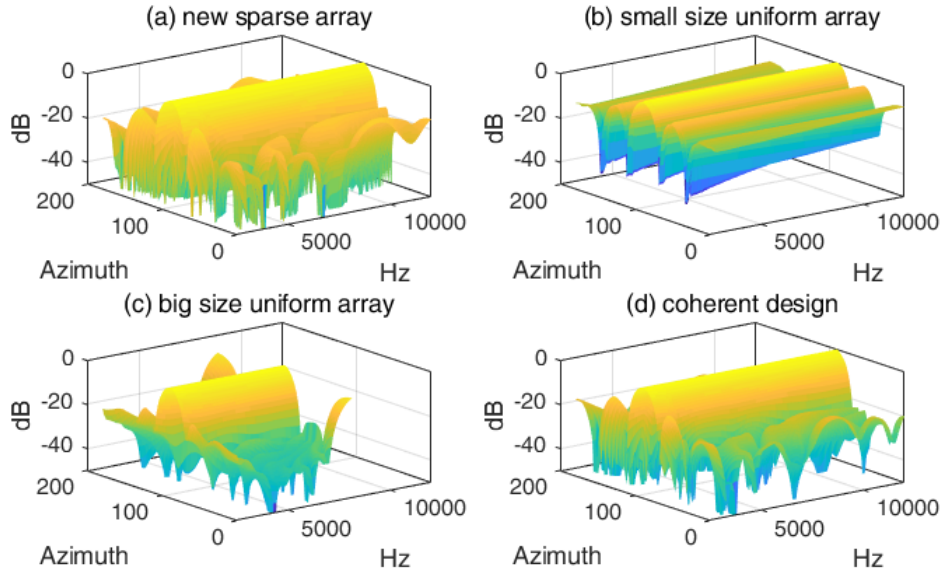


FIGURE 4.8: Broad-side linear array: The beam patterns versus frequency.

sized beam patterns versus frequency are shown in Fig. 4.8. It is clearly seen that the proposed sparse array, SUA and coherence sparse array design satisfy the FI beam pattern constraints, while the BUA does not satisfy them (its optimization is not feasible for some $\omega_i \in \Omega$). There are the blank areas in Fig. 4.8(c) where the optimizations in (4.12) are infeasible.

Fig. 4.9 depicts the synthesized beam patterns and the reference beam pattern at 10 kHz. Note that the synthesized beam pattern in Fig. 4.9(c) is blank since the optimization in (4.12) of the BUA is infeasible.

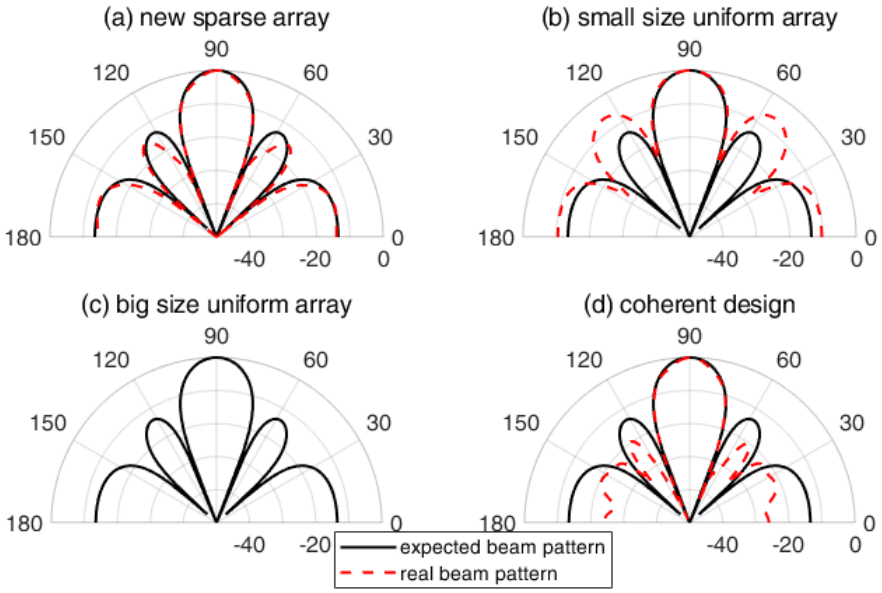


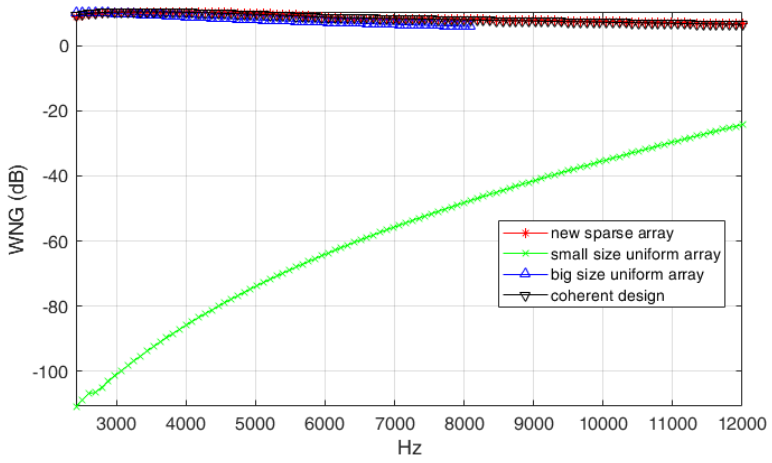
FIGURE 4.9: Broad-side linear array: Beam patterns (dB) at 10 kHz for different broad-band linear arrays, the optimization of the BUA (c) is infeasible.

Fig. 4.10 shows the WNGs, DFs and BPEs of four array layouts (proposed sparse array, SUA, BUA and incoherence sparse array design). The disconnection of the blue line presents for the empty solution. The performances of the proposed sparse array and coherence sparse array design are comparable in almost all aspects. More specifically, the proposed sparse array is slightly better than the coherence sparse array design in terms of the BPE.

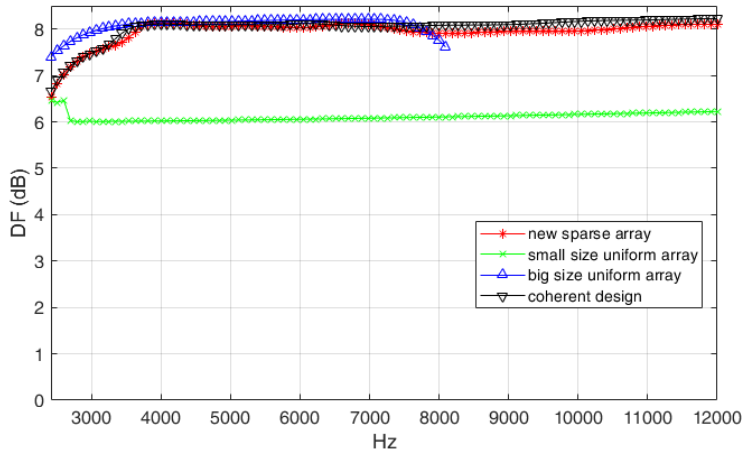
Case 2. *Beam pattern error-tolerances are strictly set.*

If we set the side-lobe error-tolerances in (4.12) more strictly, $\epsilon 2(\omega_i) = 0.01(P - U), \quad \forall \omega_i \in \Omega$ which means the synthesized beam patterns and the reference beam pattern is less (the nulls and side-lobes are preserved). For the coherence sparse array design and SUA, the optimization in (4.12) becomes infeasible at some frequencies, while for the proposed sparse array the optimization in (4.12) is still feasible for the bandwidth of interest Ω, as shown in Fig. 4.11.

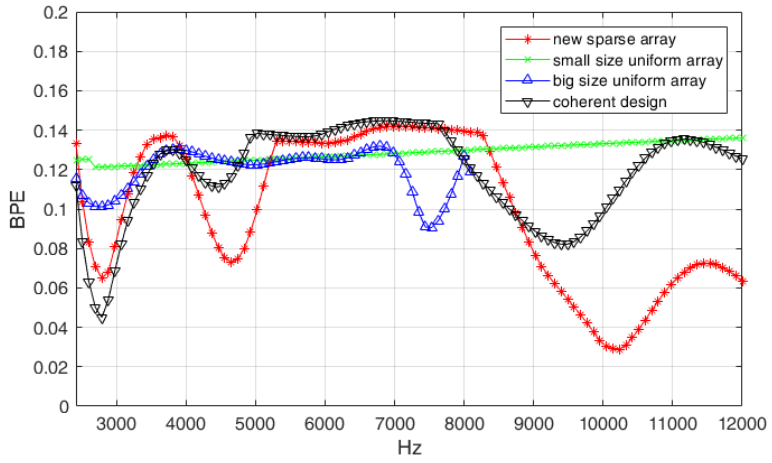
Although the proposed approach does not take the steering ability into account, but it is worthy to observe how the sparse array responses to different look directions. Fig. 4.12 plots the beam patterns of new sparse array pointing to different directions: (a) $\theta = 90^o$, (b) $\theta = 95^o$, (c) $\theta = 100^o$, (d) $\theta = 105^o$, (e) $\theta = 110^o$ and (f) $\theta = 115^o$. It is clearly seen that the shape of the main-lobe is always preserved. However, the side-lobes suffer from the deformation, the side-lobe degradation is more when the steering direction goes far away the broad-band direction.



(A)

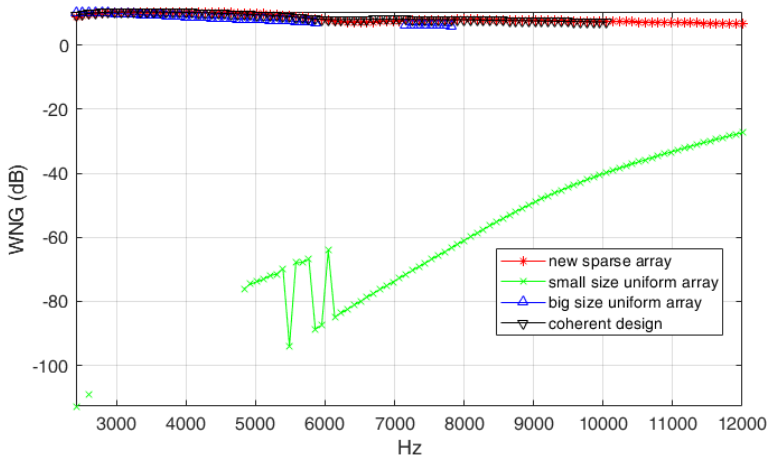


(B)

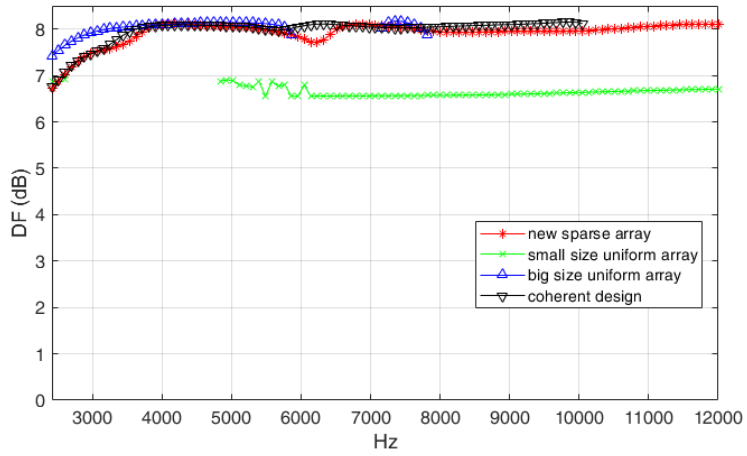


(C)

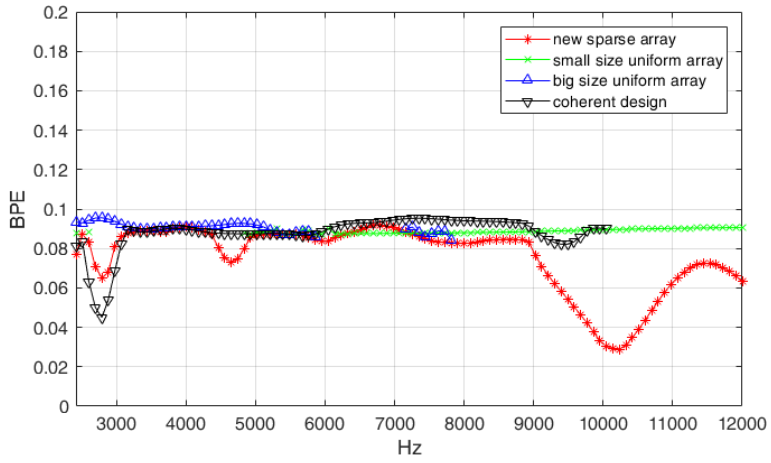
FIGURE 4.10: Side-lobe constraints are relaxed, the performance comparison of different broad-side linear array layouts (the disconnection lines mean that the optimizations are infeasible).



(A)



(B)



(C)

FIGURE 4.11: The side-lobe constraints are strictly set, the performance comparison of different Broad-side linear array layouts.

4.5.2 Sparse End-fire Linear Array

For comparison, a similar reference beam pattern with the example in [Buc+18] is used . The beam pattern of an Q^{th}-order DMA is given by [Elk00]

$$b(\theta) = \sum_{n=0}^{Q} a_n \cos^n \theta,$$

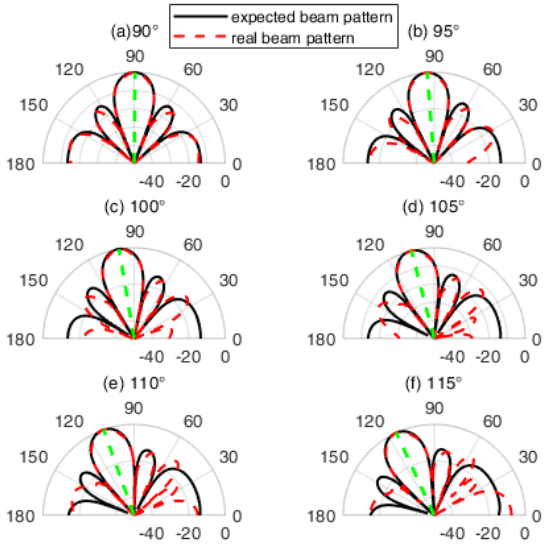


FIGURE 4.12: Broad-side linear array: beam patterns (dB) at 10 kHz for different look directions: (a) 90^o, (b) 95^o, (c) 100^o, (d) 105^o, (e) 110^o, (f) 115^o.

where $a_n, n = 0, ..., Q$ are coefficients.

Analysis Step

Let us define a third-order hypercardioid beam pattern

$$b(\theta) = -0.14 - 0.57cos\theta + 0.57cos^2\theta + 1.14cos^3\theta.$$

From (4.9), the configuration for the DUA is selected, $M = N = 141$ sensors, $\delta = 0.01$ meter, $c = 340$ m/s. The frequency range of the DUA is given by

$$\frac{mc}{N\delta} \leq f \leq \frac{c(N-1)}{2\delta N}, m \geq 1,$$

$$241m \text{ Hz} \leq f \leq 16879 \text{ Hz}, m \geq 1.$$

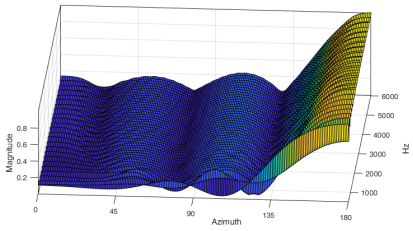
Let us design the sparse array with FI beam pattern at the bandwidth

$$\Omega = \{500 \text{ Hz}, \ldots, 6 \text{ kHz}\}.$$

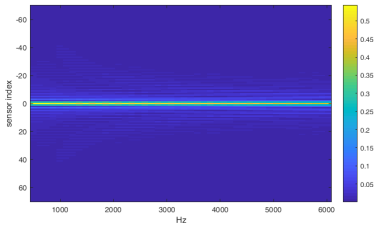
We apply the CT method to get a weight matrix for DUA, as shown in Fig. 4.13(b). With $\alpha = 0.99$, Fig. 4.13(c) depicts the spectrum of dimensional reduced weight matrix.

Selecting Step

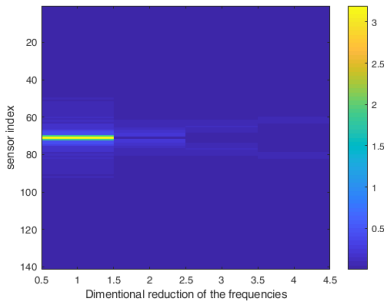
For comparison, the setting parameters this step intends to obtain the same amount of sensors with the incoherence sparse array design in [Buc+18]. The spanning energy threshold is set $\beta = 0.5$, then three sensors with indices $S_{K_1} = \{70, 71, 72\}$ are chosen, the indices of sensor are shown in the vertical axis of Fig. 4.13(c). The



(A) Cross-cut of beam patterns across frequency.



(B) Weight matrix.



(C) Dimensional reduction matrix ($\alpha = 0.99$).

FIGURE 4.13: End-fire linear array: Analyzing the DUA with broadband beamforming.

remaining sensors in the DUA are divided into $K_2 = 7$ groups by the K-means clustering algorithm. In each group, a representative sensor, which is closest with the centroid of its group, is chosen. Then, the subset $\mathbb{S}_{K_2} = \{28, 53, 63, 67, 73, 77, 83\}$ is obtained. Finally, the proposed sparse array contains ten sensors. Fig. 4.14 presents different array layouts: The proposed sparse array (red circle), SUA (green circle), BUA (blue circle) and incoherence sparse array design (black *).

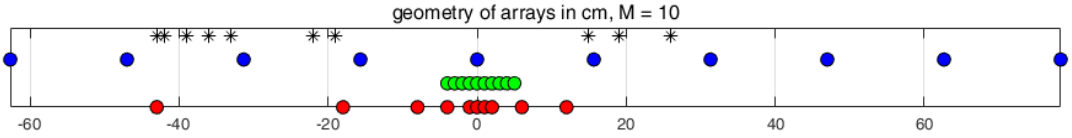


FIGURE 4.14: The end-fire linear array deployments: New sparse array (red circle), SUA (green circle), BUA (blue circle) and incoherence sparse array design (black *).

Optimization Step

We run the simulation for two cases.

Case 1. *Beam pattern error-tolerances are strictly set.*

The constraint parameters are selected to assure the FI beam pattern and the $\mathcal{W} \geq -15$ dB, $\forall \omega_i \in \Omega$. The main-lobe error-tolerances for $\theta \in (120^o, 180^o]$, $\epsilon 1(\omega_i) = 0.003U, \quad \forall \omega_i \in \Omega$. The side-lobe error-tolerances for $\theta \in [0^o, 120^o]$, $\epsilon 2(\omega_i) = 0.004(P - U), \quad \forall \omega_i \in \Omega$, where $P = 180$ (the resolution of the azimuth angle $\Delta\theta = 1^o$), $U = 60$.

The minimization problem in (4.12) is performed with the same error-tolerances for the proposed sparse array, SUA, BUA and incoherence sparse array design. The synthesized beam patterns versus frequency are shown in Fig. 4.15, and the beam patterns at 1 kHz are shown in Fig. 4.16. Fig. 4.17 describes the WNGs, DFs, BPEs over frequency. For the incoherence sparse array design and BUA, the optimizations are infeasible at high frequencies. Both the proposed SA and SUA perform well in terms of BPE, but the proposed sparse array achieves better WNGs at the low frequencies, as shown in Fig. 4.17.

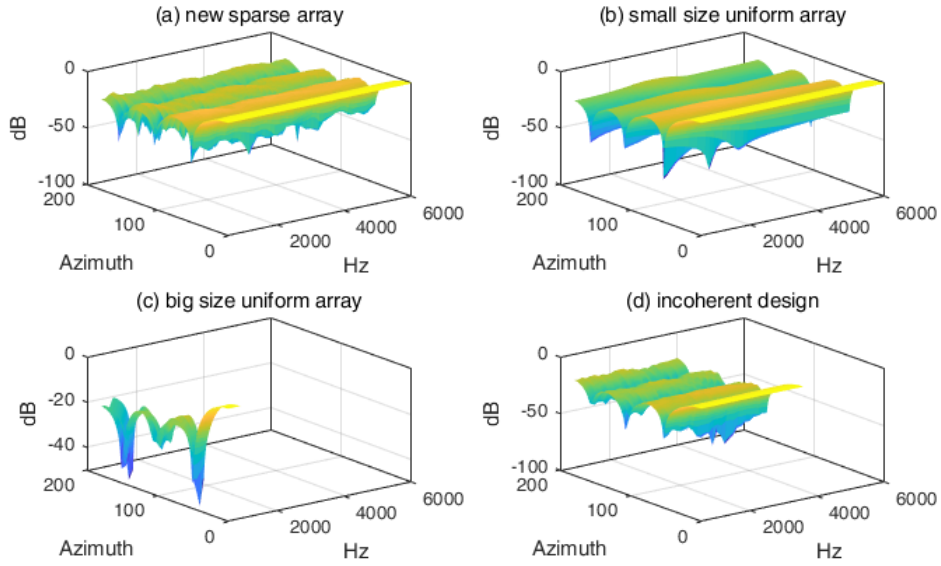


FIGURE 4.15: End-fire linear array: Beam patterns versus frequency.

Case 2. *Beam pattern error-tolerances are relaxed.*

In order to make the optimization for the incoherence sparse array design feasible for the bandwidth of interest, we relax the side-lobe, main-lobe error-tolerances. The new main-lobe error-tolerances are set $\epsilon 1(\omega_i) = 0.01U, \quad \omega_i \leq 2\pi 5000, \omega_i \in \Omega$ and $\epsilon 1(\omega_i) = 0.015U, \quad \omega_i > 2\pi 5000, \omega_i \in \Omega$. The new side-lobe error-tolerances are set $\epsilon 2(\omega_i) = 0.02(P - U), \quad \omega_i \leq 2\pi 5000, \omega_i \in \Omega$ and $\epsilon 2(\omega_i) = 0.03(P - U), \quad \omega_i > 2\pi 5000, \omega_i \in \Omega$. The new performances are shown in Fig. 4.18. The proposed sparse array and incoherence sparse array design have the same performances in terms of DF and BPE, however the proposed sparse array is better in terms of WNG.

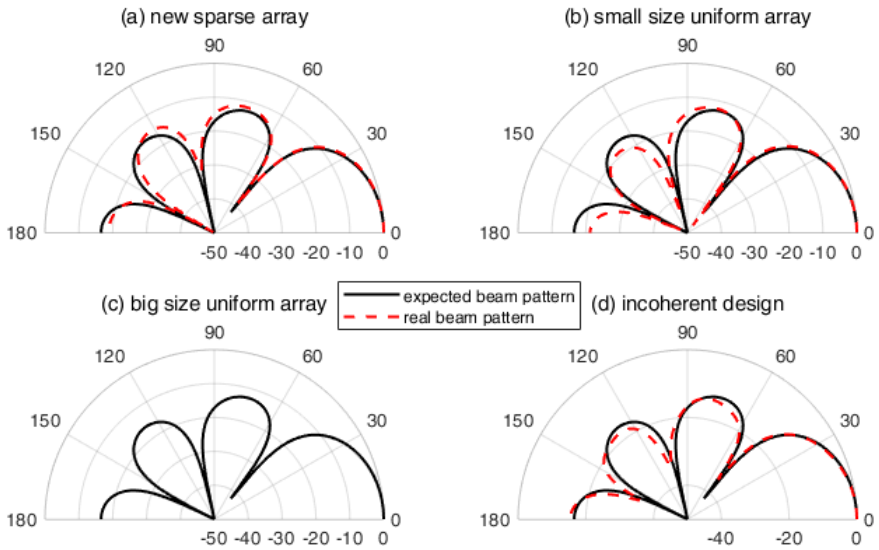
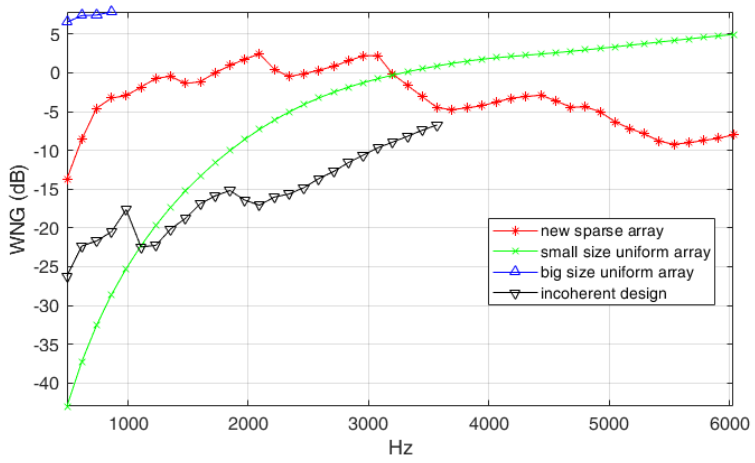


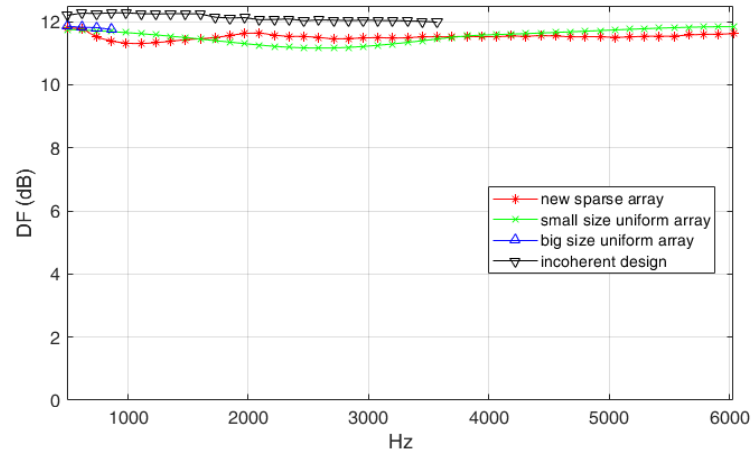
FIGURE 4.16: End-fire linear array: Beam patterns at 1 kHz (for the case the error tolerance of the BPE is strictly set). The optimization of the BUA (c) is infeasible.

4.5.3 Planar Symmetric Sparse Array

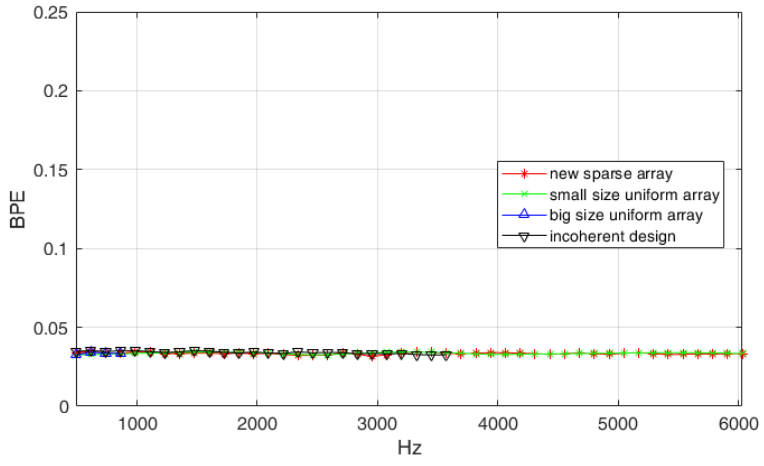
This example focuses only on the proposed sparse array since for such a high number of potential sensors, other approaches are computationally intensive and hard to find a feasible solution. The proposed approach ran on an i7-6700U CPU @ 3.4 GHz of INTEL with 32 GB ram and it took less than one second to find the sparse array deployment as well as their beamformer coefficients. We compare the performance of the sparse array with the other uniform arrays having higher the number of sensors (81 sensors in the sparse array compares with 121 sensors in the SUA and BUA).



(A)



(B)



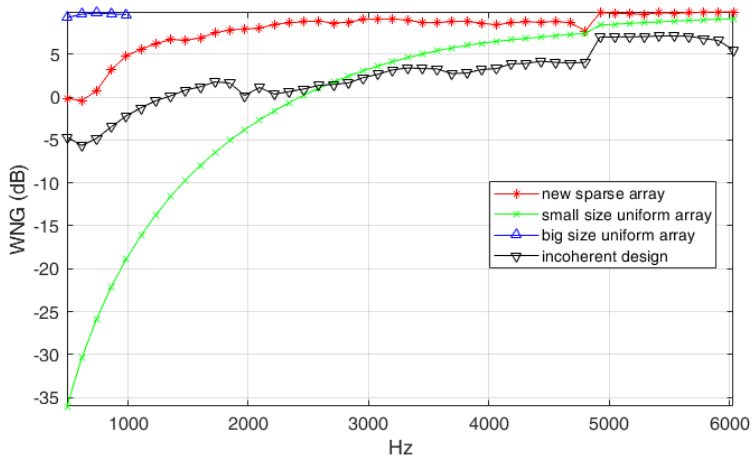
(C)

FIGURE 4.17: Beam pattern error-tolerances are strictly set, the performance comparison of different end-fire linear array layouts.

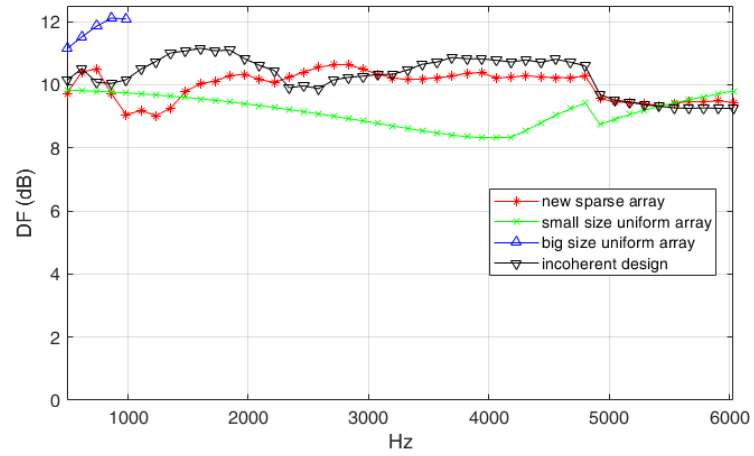
Analysis Step

Define the reference beam pattern:

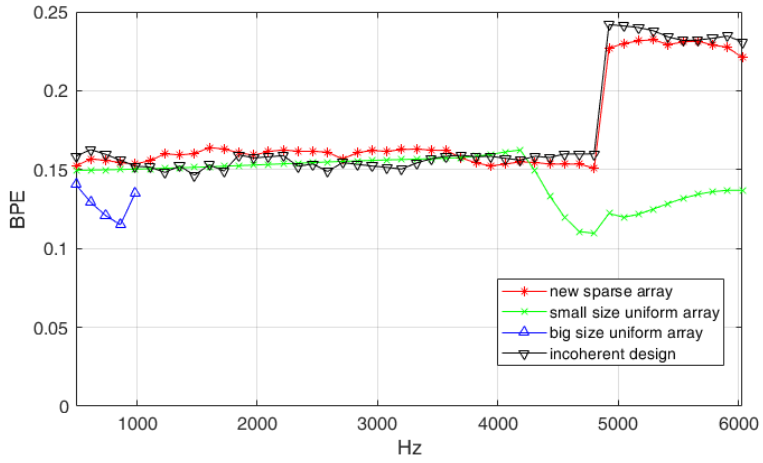
$$b(\phi,\theta) = \begin{cases} |\frac{\sin(4.4\theta)}{4.4\theta}|, & \theta \in (0, \frac{\pi}{2}], \\ 1, & \theta = 0. \end{cases}$$



(A)



(B)



(C)

FIGURE 4.18: The beam pattern error-tolerances are relaxed, the performance comparison of different end-fire linear array layouts.

From (4.9), the configuration of a DUA is selected: $N = 101$ sensors, $M = 101 \times 101$ sensors, $\delta = 0.01$ meter, $c = 340$ m/s. The frequency range of the DUA is given

by,

$$\frac{mc}{N\delta} \leq f \leq \frac{c(N-1)}{2\delta N}, m \geq 1,$$
$$337m \text{ Hz} \leq f \leq 16832 \text{ Hz}, m \geq 1.$$

Let us select the frequency range for the sparse array, $\Omega = \{1 \text{ kHz}, \ldots, 4.5 \text{ kHz}\}$. Applying the CT method and set $\alpha = 0.98$, we yield the weight matrix and its dimensional reduction showed in Fig. 4.5(b) and Fig. 4.5(c) in Section 4.3.1, respectively.

Selecting Step

Set $\beta = 0.8$ then we obtain $K_1 = 21$ sensors. The remaining sensors in the DUA are divided into 15 groups by K-means clustering algorithm. The iterative of K-means clustering is 500 times for this example to find the acceptable solution. For the planar sparse array, let us set the number of the representative sensors for one group to four because we aim to design a symmetric array layout, that is, the planar sparse array contains four quarters and the layout pattern in a quarter is symmetric with each others. Finally, 60 sensors in S_{K_2} are obtained and the total sensor in the sparse array is 81, as shown by the red circle in Fig. 4.19.

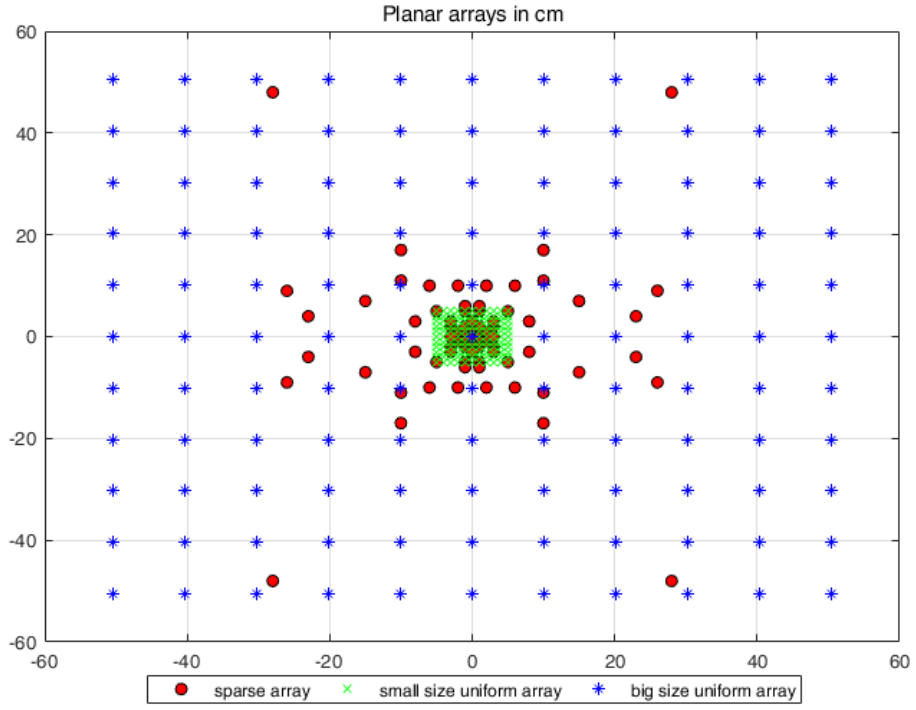
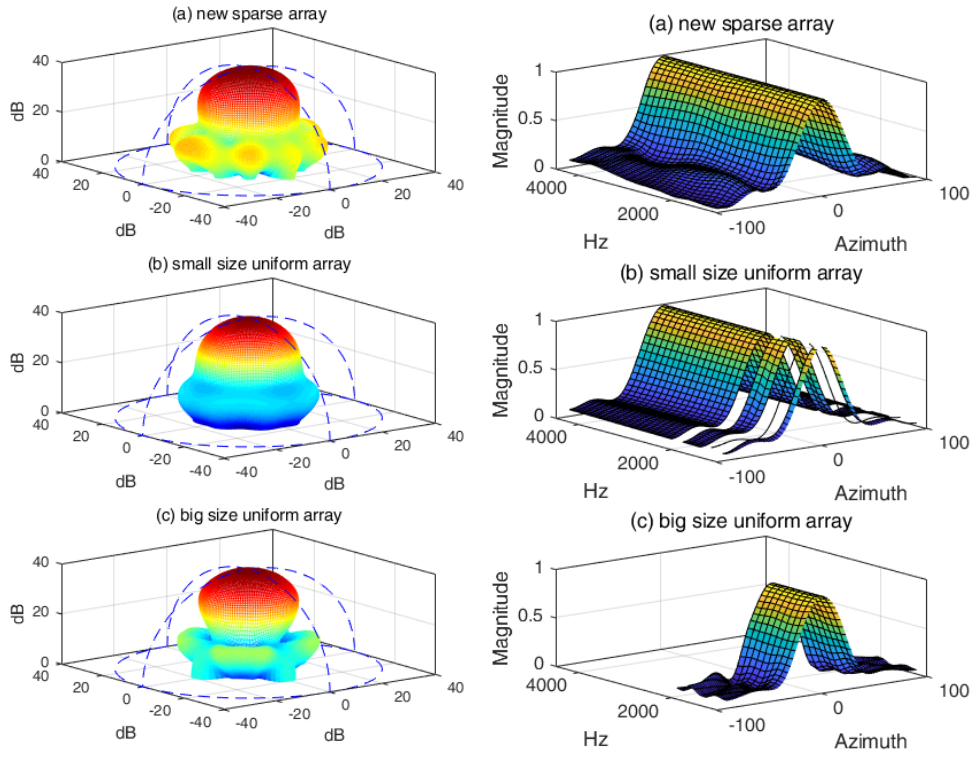


FIGURE 4.19: Symmetric sparse array (81 sensors constitute an irregular layout but having four symmetric quarters), SUV (121 sensors) and BUA (121 sensors).

Optimization Step

The main-lobe error-tolerances for the elevation angle space $\theta \in [0^o, 30^o]$, $\epsilon 1(\omega_i) = 0.0002U, \quad \forall \omega_i \in \Omega$ and the side-lobe error-tolerances for the elevation angle space $\theta \in (30^o, 90^o]$, $\epsilon 2(\omega_i) = 0.0006(P - U), \quad \forall \omega_i \in \Omega$ where $P = 90 \times 120$ (the resolution of the elevation, azimuth angle are $\Delta\theta = 1^o, \Delta\phi = 3^o$, respectively), $U =$

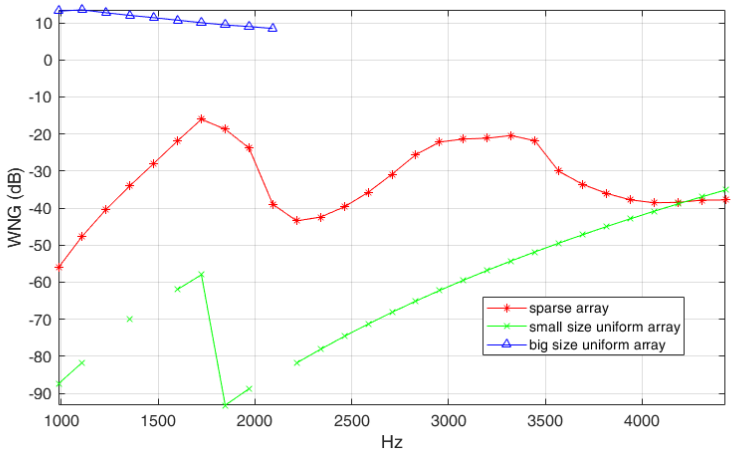
30×120.
The beam patterns at 1.7 kHz and beam patterns versus frequency are shown in Fig. 4.20(a) and 4.20(b), respectively. Fig. 4.21 depicts the WNGs, DFs, BPEs over the



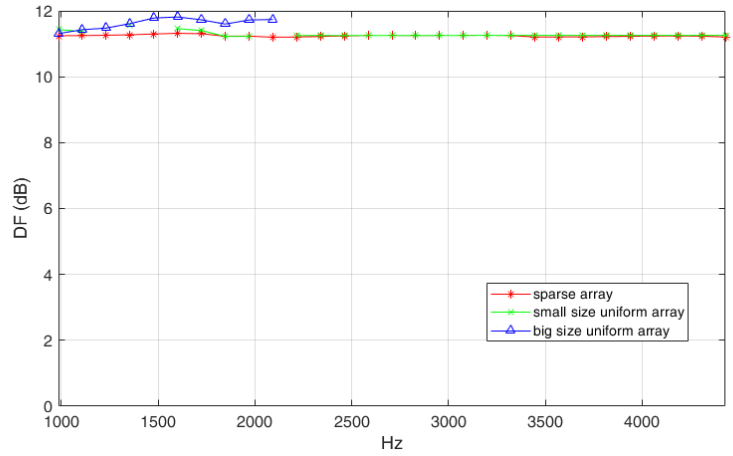
(A) Beam patterns. (B) Cross-cuts.

FIGURE 4.20: Planar array: (left column) beam patterns at 1.7 kHz and (right column) their cross-sections over frequency.

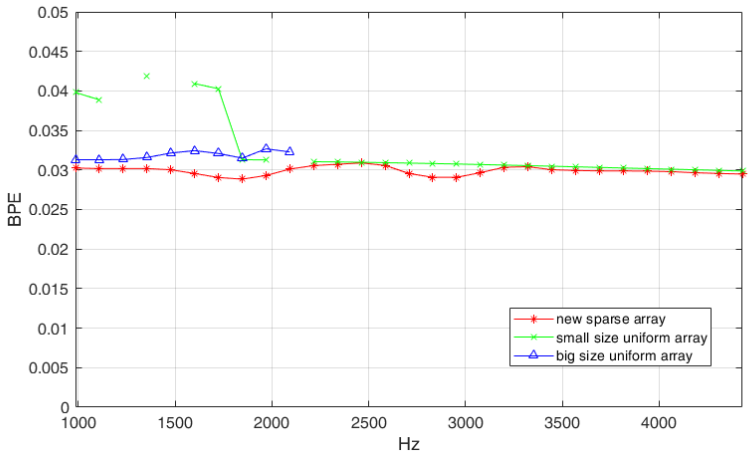
frequency range. The optimization in (4.12) for the sparse array always gives the solution in the bandwidth of interest, while the optimization for the SUA and the BUA is infeasible at some frequencies, despite they have 40 sensors more.
Herein, a similar simulation applying to the broad-side linear array is performed to examine the steering ability of the planar sparse array. It is noted that the spatial aliasing problem of the planar array is more severe with the elevation angle beam steering. Therefore, in this simulation, the look direction is fixed the azimuth angle and only varied the elevation angle. Fig. 4.22 plots the beam patterns at 1.7 kHz. Each beam pattern points to different look directions: (a) $(\theta, \phi) = (0^o, 180^o)$, (b) $(\theta, \phi) = (5^o, 180^o)$, (c) $(\theta, \phi) = (10^o, 180^o)$, (d) $(\theta, \phi) = (15^o, 180^o)$, (e) $(\theta, \phi) = (20^o, 180^o)$ and (f) $(\theta, \phi) = (25^o, 180^o)$. At look directions $\theta > 20^o$, the shape of beam pattern has much deformation, especially for the side-lobes.



(A)



(B)



(C)

FIGURE 4.21: Comparison of the performances of different planar array layouts.

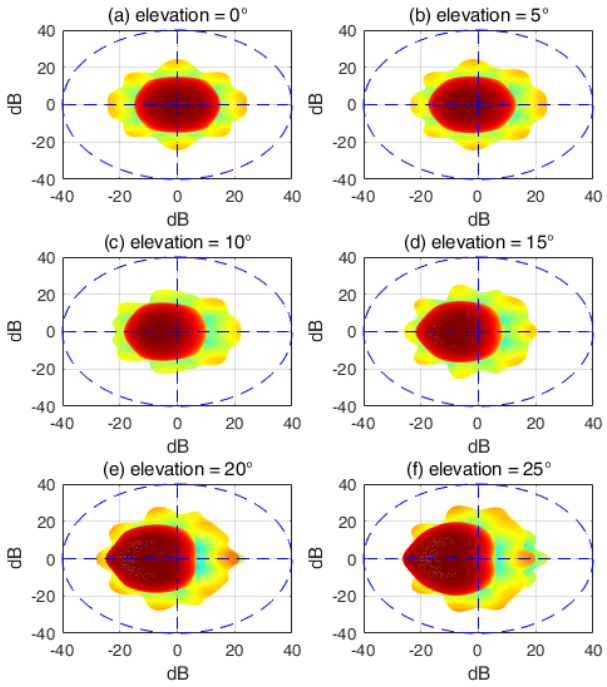


FIGURE 4.22: Beam patterns (top view) at 1.7 kHz for different look directions.

4.6 Conclusions

In this chapter, a new approach for the sparse array design via a hypothesized dense and uniform array has been developed. The proposed method not only outperforms other state-of-the-art methods in some aspects of beamforming, but it is also efficient in computation time. Given a reference beam pattern, the chapter has also presented a new method to design the broadband beamforming for a dense and uniform array, and then the sparse array design analyses the weight matrix of the dense and uniform array to identify the critical sensors. However, some parameters are empirically set in the proposed approach, such as the number of the groups or the number of representative sensors for each group. These settings need to be addressed in a mathematical sense in future work. Furthermore, other common classification algorithms such as the support vector machine or relevance vector machine could be applied to identify the critical sensors instead of taking the closest sensors to the group's centroid of K-means clustering.

Chapter 5

Irregular Microphone Array Design for Broadband Beamforming

Sparse microphone array design is receiving a great deal of attention, largely due to the capacity to reduce the number of microphones without sacrificing much of the array performance. However, in general, using a sparse array for broadband beamforming still requires a large number of microphones for a complex/higher-order beam pattern, while a sparse array deployment is normally designed for a specific beam pattern. This chapter aims to design a versatile sparse microphone array that can be used for different beam patterns. Furthermore, we aim to reduce the number of microphones in the sparse array while ensuring that its performance can continue to compete with a highly dense and uniform array in terms of broadband beamforming. An irregular microphone array in a planar surface with the maximum number of distinct distances between the microphones is proposed. It is demonstrated that the irregular microphone array is well-suited to the sparse recovery algorithms that are used to reconstruct the sound sources. From the reconstructed sound sources, the array interpolation method is presented to obtain an interpolated dense and uniform microphone array that performs well with broadband beamforming. This chapter covers the theoretical background of the method as well as the experimental results.

5.1 Introduction

Broadband beamforming techniques are important for numerous applications, including microphone arrays, sonar arrays, radio astronomy and broadband radar. The techniques allow for maintaining signal integrity and spatial selectivity over a given frequency range and are widely used for audio signals where the bandwidth of the signals extends to several octaves. A conventional approach involves using a dense and uniform array (DUA) for broadband beamforming where the sensors exhibit a close distance response for beamforming at high frequencies and a far distance response for beamforming at low frequencies [DBWW95; Yan06; LW08; WKW01; Nor+14; SC07; EMK09; Yan06; MSK09; LW10; VVB88]. In terms of acoustic applications, the differential microphone array (DMA) is also widely used, largely due to its compact size [Elk00; TE01; Buc02; Elk04]. Given that the principle of DMA is based on the spatial derivatives of the acoustic pressure field, the inter-distance of the sensors is much smaller than the acoustic wavelength.

New approaches for broadband beamforming based on sparse arrays have been proposed in recent studies. For a sparse array design, the target is to determine the array layout as well as beamforming coefficients. Doblinger [Dob08] applied the simulated annealing (SA) algorithm to determine an array layout where the cost function of SA and the sensor coefficients were achieved through a superdirectivity beamformer [Elk00]. Meanwhile, Marco Crocco[CT12] and Andrea Trucco [CT14] also used the SA algorithm, but introduced more parameters in the cost function to adjust the different beamformer characteristics, including the directivity factor, the white noise gain, the side-lobe and the main-lobe. More recently, based on the concept of compressed sensing (CS) [Boc+15; Mig14], Hawes and Liu [HL13] proposed the application of the l_1-norm to simultaneously ascertain the position of the sensors and their coefficients. To extend the sparsity of the conventional l_1-norm, both Hawes and Liu [HL14] and Liu *et al.* [Liu+15b] applied a sequential l_1-norm method [CWB08] that iteratively reweights the conventional l_1-norm to achieve solutions that are closer to the sparse solution. Similarly, Yaakov *et al.* [Buc+18] also applied the sequence l_1-norm method to determine the positions of the sensors at different narrow bands before merging these sub-arrays via a clustering algorithm to obtain the final sparse array. Meanwhile, Phan [LS20] analyzed a hypothesized DUA and applied a clustering algorithm to it to design the sparse array. While good results have generally been reported, the state-of-the-art methods still require numerous microphones, especially for the higher-order beam patterns. In other words, the number of microphones is proportional to the complexity of the beam pattern. Moreover, a sparse microphone array is generally designed for a specific beam pattern, which limits the sparse arrays' applications where the beam pattern needs to be changed during the running time.

Suppose a signal is a sparse representation in a certain basis and that the matrix is transforming this signal to another basis, compressed sensing is a method that can be used to invert the transform. The inverse transform is unique if the transform matrix has certain properties. Here, the restricted isometry constant (RIC) of a matrix is an important aspect [Can+08; Bar+08] since it indicates the ability of preserving the Euclidean distance of the transform and then determines the possibility and stability of the inversion. In a given measurement system, a transformation matrix is the matrix describing that system, which is generally a fat matrix (a fat matrix is the matrix that has more columns than rows), since the point of interest is involving fewer measurement points while ensuring there are enough to reconstruct the signal. In such a case, the transformation matrix transfers the sparse signal from a high-dimensional subspace to a lower-dimensional subspace. However, the restricted isometry constants itself is difficult to verify, and we are generally interested in analyzing the column coherence of the transformation matrix [Boc+15].

In terms of microphone arrays, if we wish to reconstruct a sparse sound source, then the RIC will be related to the positions of the microphones. Therefore, in this chapter, a rule for microphone array deployment is proposed, that is, maximizing the number of distinct distances between the microphones, while assuring that the distribution of the microphones in a plane is almost equal. Since the target function does not have an explicit derivative, we must use the stochastic method to solve this problem. Here, the simulated annealing algorithm [Tru01; MTR96] is selected. The simulated annealing is an algorithm of approximating the global optimum, it simulates the natural process of annealing in metallurgy, assuring that the searching process is not trapped in a local minimum at the high temperature. As such, an approximate global solution is obtained in reasonable running time, even in a large search space.

Meanwhile, due to physical ill-conditions of the system, the measurement matrices at certain frequencies are ill-suited for the recovery algorithms, which means determining the suitable sparse recovery algorithms for the transformation matrix is also important. Among the numerous available algorithms such as iterative hard threshold [BD08], subspace pursuit [DM09], Bayesian compressive sensing [WR04] [JXC08], compressive sampling matching pursuit [NT09], basic pursuit [CDS01], and orthogonal matching pursuit [TG07], we had to select the most suitable by verifying them with the transformation matrix of the microphone array. Following this, we present a method for obtaining the interpolated DUA based on the sparse sound source. Finally, the broadband beamforming is performed on the interpolated DUA. An overview of the method is presented in Fig. 5.1.

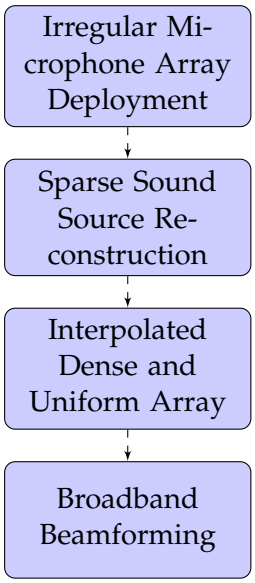


FIGURE 5.1: The steps of the proposed method.

We can briefly summarize the contributions of our work on broadband beamforming as follows.

1. We present a new perspective for design of the sparse array layout that is well-posed for sparse recovery algorithms.

2. We verify different sparse recovery algorithms using our sparse array to determine the most suitable algorithm, while we also present certain properties of the selected algorithm to explain its fitting.

3. We propose a simple approach to interpolate a DUA. This interpolation approach is compared to the least-square error (LSE) minimization approach in the simulation and experiment sections.

4. In addition, we construct a real microphone array and conduct various experiments to examine the performances of the constructed microphone array in a reverberant room.

The remainder of this chapter is organized as follows. Section 5.2 presents the signal model before section 5.3 describes the algorithms for the sparse array deployment, the sound source reconstruction, the DUA interpolation and the broadband

beamforming. The numerical simulations and the experimental results are then presented in section 5.4 and section 5.5, respectively. Finally, conclusions are drawn in section 5.6.

Notations: The bold uppercase and lower-case letters denote matrices and column vectors, respectively. $(.)^T$ and $(.)^H$ denote the transpose and Hermitian transpose, respectively. $j = \sqrt{-1}$ is the imaginary unit. $\|.\|_2$ is the l_2-norm. $|.|$ for a complex-value number is the amplitude operator, $|.|$ for a set is the cardinality of that set. $\mathbb{C}$ is the set of complex numbers.

5.2 Signal Model

In terms of the far-field signal, the sound impinging on the microphone array is a planar wave. The plane of the array coincides with the X-Y plane of the Cartesian coordinate. The position of a microphone in the Cartesian coordinate is given by

$$\mathbf{p}_i = [x_{p_i}, y_{p_i}, z_{p_i}]^T, i = 1, ..., M$$

where M is the number of microphones in the array. For a planar array, $z_{p_i} = 0$ for all $i = 1, ..., M$.

Let us suppose that there are N grid points corresponding to N directions that cover the entire space. The direction of incident sound k is then presented by a vector

$$\mathbf{d}_k = -[\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta]^T$$

where ϕ, θ is the azimuth and elevation angle in the spherical coordinate system. The incident sounds impinge on the array from multiple directions, and then the sound pressure of a microphone in position $\mathbf{p}_i$ at a frequency of f is estimated by

$$x_i(f) := x(\mathbf{p}_i, f) = \sum_{k=1}^{N} s_k(f) e^{-\frac{j2\pi f \mathbf{p}_i^T \mathbf{d}_k}{c}} + n_i(f) \tag{5.1}$$

where $s_k(f)$ is a complex-valued number representing for the strength and phase of an incident sound at the direction $\mathbf{d}_k$, c is the sound speed and $n_i(f)$ is white Gaussian noise with zero mean of the microphone at position $\mathbf{p}_i$, e.g., electrical noise, mechanical imprecise, acoustic environment, .etc. For conciseness, we omit the variable f in the remainder of the chapter wherever possible.

In three-dimensional space, we can uniformly discretize θ, ϕ to the grid of directions. Assuming the grid size is $N = m_1 \times m_2$ (m_1 grid points for θ and m_2 grid points for ϕ), we could assign the complex-valued numbers to all points in the grids to form a sound source matrix (SSM) $\mathbf{S}$ with dimension of $m_1 \times m_2$. Then, $\mathbf{S}$ contains the information of the sound sources in the working space.

If there are S sound sources impinging on the array, the SSM will contain S non-zero elements and the remaining elements will be zero. A non-zero element is associated with an active incident sound and the information of that sound source is encoded by a complex number. It is highly unlikely that at a given frequency the array is impinged on by the incident sounds coming from all directions of the space simultaneously. Hence, in an environment without noise/reverberation, the SSM is generally a sparse matrix or the sparse SSM will stand for the sound sources at a frequency after removing the noises.

In order to solve a optimization problem with a sparse solution constraint, we are interested in working with a sparse vector. Hence, we vectorize the SSM to a sound

source vector (SSV). The row in a SSM stacks on top of another row to construct a vector $\mathbf{s}$ with size of $N = m_1 \times m_2$

$$\mathbf{s} = \text{vec}(\mathbf{S}^T)$$

where vec(.) is the vectorization of a matrix, e.g., it can be implemented by the reshape function in Matlab: $\mathbf{s} = \text{reshape}(\mathbf{S}', [m_1 * m_2, 1])$, where $\mathbf{S}$ is a matrix size of $m_1 \times m_2$.

In the CS framework, a vector is called *S-sparse* if the number of non-zero elements is not larger than S. From a geometrical point of view, an S-sparse vector belongs to the set obtained by the union of all the $\binom{N}{S} = \frac{N!}{S!(N-S)!}$ S-dimensional subspaces in $\mathbb{C}^N$. This set is called Σ_S [Mig14].

From M microphones, the sound pressure at M points in the plane could be measured. From (5.1), we can construct a linear relation between the SSV and the measured signals

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n}, \tag{5.2}$$

where $\mathbf{x} = \begin{bmatrix} x_1(f) \\ x_2(f) \\ \cdots \\ x_M(f) \end{bmatrix}$ ($\mathbf{x} \in \mathbb{C}^M$) is a vector that contains measured signals at the frequency f,

$$\mathbf{A} = \begin{bmatrix} e^{-\frac{j2\pi f \mathbf{p}_1^T \mathbf{d}_1}{c}} & \cdots & e^{-\frac{j2\pi f \mathbf{p}_1^T \mathbf{d}_N}{c}} \\ e^{-\frac{j2\pi f \mathbf{p}_2^T \mathbf{d}_1}{c}} & \cdots & e^{-\frac{j2\pi f \mathbf{p}_2^T \mathbf{d}_N}{c}} \\ \cdots & \cdots & \cdots \\ e^{-\frac{j2\pi f \mathbf{p}_M^T \mathbf{d}_1}{c}} & \cdots & e^{-\frac{j2\pi f \mathbf{p}_M^T \mathbf{d}_N}{c}} \end{bmatrix}$$

($\mathbf{A} \in \mathbb{C}^{M \times N}$) is a transformation matrix at the frequency f, and $\mathbf{n}$ is the noise vector ($\mathbf{n} \in \mathbb{C}^M$).

The problem in (5.2) is a basic form of the CS framework. To recover the sparse solution $\mathbf{s}$, two aspects need to be considered: designing a transformation matrix that is well-suited to the CS algorithms and determining a suitable sparse recovery algorithms for that transformation matrix. Given the matrix entries are a function of the position of the microphones, our method focuses on two aspects: the microphone array deployment and suitable CS algorithms.

5.3 Design Irregular Microphone Array

5.3.1 Irregular Microphone Array Deployment

This section aims to construct a transformation matrix $\mathbf{A}$ that is well-suited to CS-based techniques. As will be addressed in Section 5.3.2, the transformation matrix should have a small column coherence [Boc+15]. It is equivalent to finding the microphone array layout with maximum degrees-of-freedom or maximizing the number of distinct distances between the microphones.

Let us discretize the possible position of the microphones in the planar surface to $M \times M$ rectangular grid points and let us assume that the array has only M microphones allocated in these grid points. If the target function is to maximize the degrees-of-freedom of the array, we need to search on $\binom{M^2}{M}$ possible solutions. Since an algorithm that can identify the best combination in a timely manner is not known,

the probabilistic techniques are used to find the solution. In order to narrow the search area, we propose a rule to restrict the position of a microphone. Here, the grid points are divided into M segments, as shown in Fig. 5.2, each microphone is restricted to be within one of the segments. This deployment manner not only enhances the search efficiency but also results in a reasonable deployment. This means that even though the location of the microphones is somewhat random, the distribution across the plane is almost equal, which avoids both small areas with a large number of microphones and large areas with no microphone (this is the scenario of the sparse array design using the Golumb-ruler method [Ras+11]). We call the new constraint a *distribution constraint*. With the distribution constraint, we could reduce the search space of the algorithm from $\binom{M^2}{M}$ to $M!$ possible solutions.

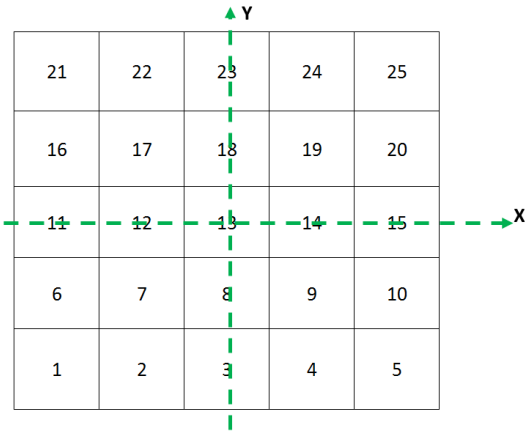


FIGURE 5.2: Segment layout for microphones.

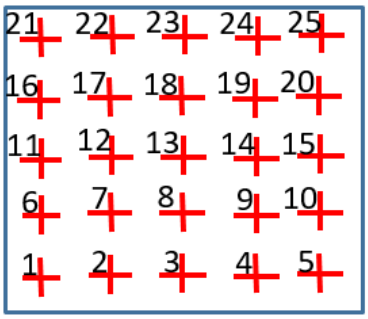


FIGURE 5.3: The possible positions of microphones in a segment.

For example, where $M = 25$, every segment has 25 positions that are uniformly distributed over the X and Y axis, as shown in Fig. 5.3. One out of the 25 positions is randomly selected for the placement of one microphone. The position of the microphone in different segments will be different. We could assign a number between 1 to 25 to indicate the position of the microphone in a given segment. As such, the list containing the sequence of 25 numbers could describe the array deployment with 25 microphones.

We used the simulated annealing algorithm with the cost function of maximizing the number of distinct distances between the microphones in the array subject to the distribution constraint. In each iteration step, the algorithm inherits a good array deployment from previous step and a permutation on the sequence encoding for the

array deployment occurs to generate a new array deployment which is close to the current array deployment. If the permuted sequence is better than the current one, it is always accepted as the good sequence which is used in the next iteration. If not, the permuted sequence is accepted with a probability that is high at the high temperatures and vice versa. Herein, a *permutation* is a random selection of two numbers in the sequence and swapping their positions. For example, a sequence of

$$\{1,2,3,4,5,6,7,8,\mathbf{13},10,11,12,\mathbf{9},14,15,16,17,18,19,20,21,22,23,24,25\}$$

is the permutation of the sequence of

$$\{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25\}.$$

The *number of distinct distances between the microphones* is defined in (5.4).

Algorithm 1 Simulated annealing for array deployment

Input: randomly choose a sequence of number $\mathbb{L}_r$
1: select start temperature T_0
2: $k = 1$
3: **repeat**
4: reduce T at each iteration $T = T_0/k$
5: **repeat**
6: $\mathbb{L}_p$ is the permuted sequence of $\mathbb{L}_r$
7: compute the number of distinct distances between the microphones in the array B_k by using (5.4)
8: compute the cost function $C_{new} = B_{max} - B_k$
9: $\delta C = C_{new} - C_{old}$
10: select random number $x \in [0,1]$
11: **if** $\delta C < 0 \quad or \quad x < e^{-\delta C/T}$ **then**
12: $C_{old} = C_{new}$
13: $\mathbb{L}_r = \mathbb{L}_p$
14: **end if**
15: **until** N_p permutations
16: $k = k + 1$
17: **until** N_i iterations
Output: $\mathbb{L}_r$.

The procedure of simulated annealing is list in Algorithm 1 (the Matlab code is provided in Appendix C). Given the number of microphones M, the maximum possible of the number of distinct distances between the microphones in the array is $B_{max} = \binom{M}{2}$ and the number of distinct distances between the microphones at the k^{th} iteration is B_k. In order to update B_k, we need to calculate the distances between the microphones in the array as follows:

$$\mathbb{D} = \{d_{il} : d_{il} = \sqrt{(x_{p_i} - x_{p_l})^2 + (y_{p_i} - y_{p_l})^2}, \\ \forall i,l \in 1,\ldots,M, i \neq l\} \tag{5.3}$$

where (x_{p_i}, y_{p_i}) is the position of microphone i in the layout $\mathbb{L}_p$. For example, in Fig. 5.4, with $\mathbb{L}_p(5) = 10$, the position of microphone 5 is $(x_{p_5}, y_{p_5}) = (25,1)$. We group the distances in $\mathbb{D}$ which have close values and only select one distance as the

representative distance to form a subset $\mathbb{D}_k$. That can be formulated by

$$\begin{aligned} B_k = \underset{\mathbb{D}_k}{\text{maximize}} \quad & : |\mathbb{D}_k| \\ & \text{subject to} \\ & \mathbb{D}_k \subseteq \mathbb{D} \\ & |d_i - d_l| \geq e_d, \forall d_i, d_l \in \mathbb{D}_k \end{aligned} \tag{5.4}$$

where e_d is a small number used to round the fractional value of the distance. For the sake of simplicity, we solve (5.4) approximately, that is, we round the distances in the set $\mathbb{D}$ to obtain the subset $\mathbb{D}_k$, e.g. e_d = 0.01 and d_H = 0.01 m, then $\mathbb{D}_k$ is obtained by rounding the distance in $\mathbb{D}$ with the resolution of 0.01 cm. Then, the cost function of the simulated annealing at the k^{th} iterative is defined as

$$C_k = B_{max} - B_k. \tag{5.5}$$

By selecting an appropriate starting temperature T_0, the cool down phase of the simulated annealing algorithm is slow enough to potentially achieve a global minimum.

After running the simulated annealing algorithm with the parameters $N_i = 5$ (number of steps associated with reducing the temperature), $N_p = 100000$ (number of steps associated with the permutation), $T_0 = 1$ and e_d = 0.01, we obtain the position of the 25 microphones in Fig. 5.4 corresponding the sequence of numbers

$$\mathbb{L} = \{12, 5, 2, 13, 10, 17, 6, 9, 3, 19, 8, 4, 22, 20, 1, 24, 11, 15, 23, 18, 16, 21, 14, 7, 25\}.$$

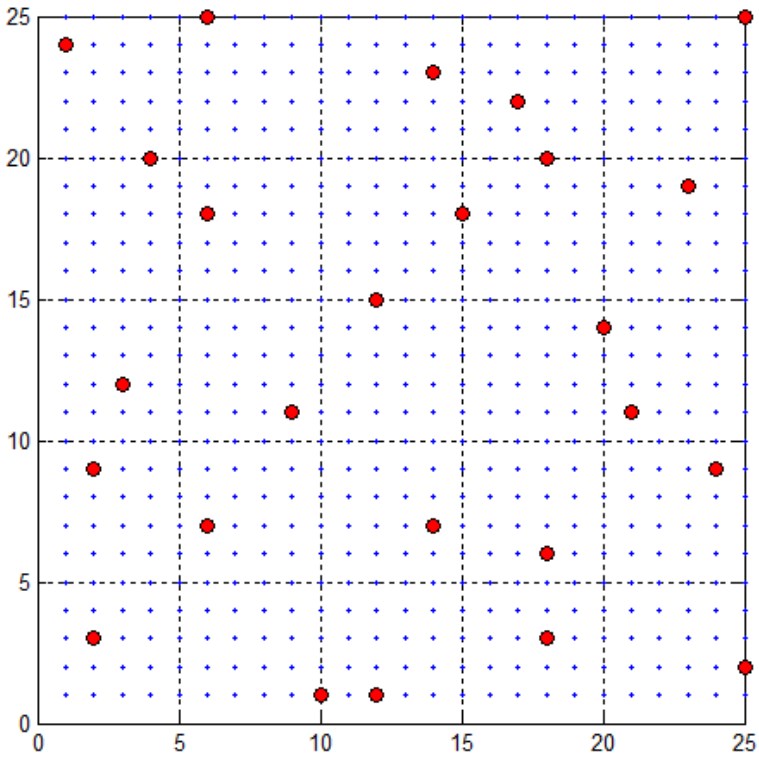


FIGURE 5.4: The final layout of irregular microphone array.

For the equidistance of sensors $d_H = 1$ cm, the histogram of the distances between the microphones is presented in Fig. 5.5 where the total number of the distinct distances is $B_k = 189$. Meanwhile, the updated cost function of the simulated annealing over the iterations is present in Fig. 5.6.

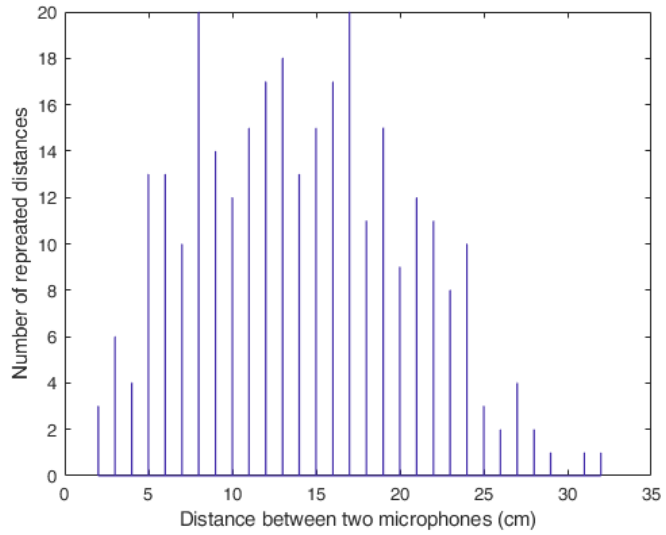


FIGURE 5.5: The histogram of distinct distances of the microphone array in Fig. 5.4.

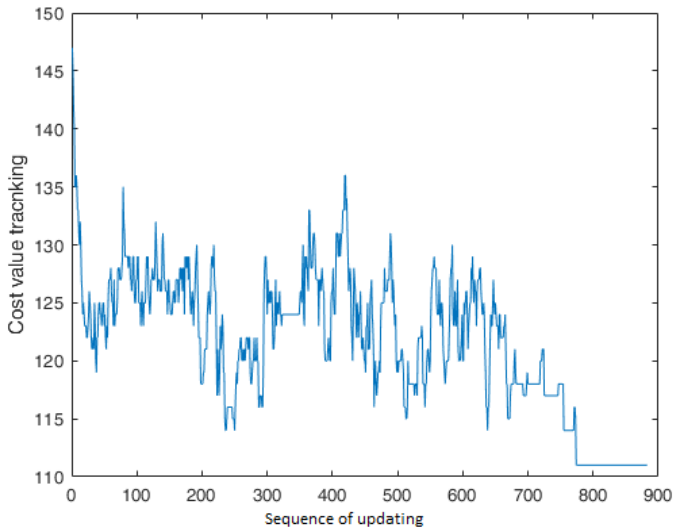


FIGURE 5.6: The updated cost values of the simulated annealing over the iterations.

Consider the irregular array archived by the simulated annealing, the spectrum of the real and imaginary part of the transformation matrix at 2.532 kHz are presented in Fig. 5.7 and Fig. 5.8, respectively. The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$.

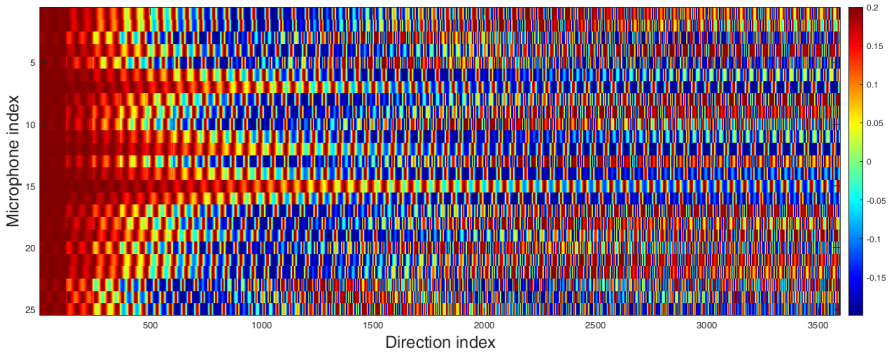


FIGURE 5.7: The spectrum of the real part of the transformation matrix at 2.532 kHz of the irregular array (25 microphones). The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$.

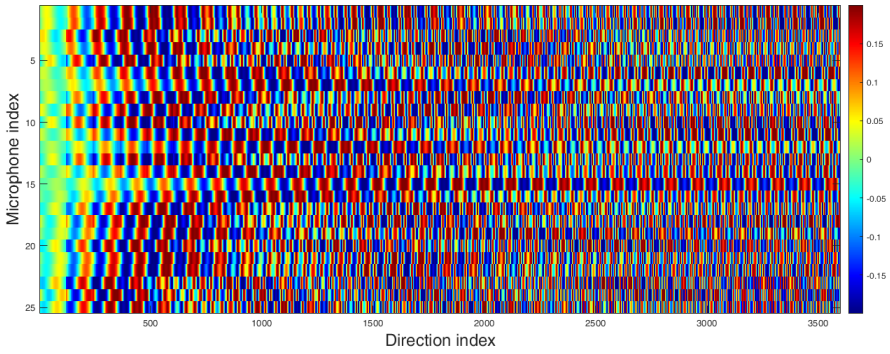


FIGURE 5.8: The imaginary part of the transformation matrix at 2.532 kHz of the irregular array (25 microphones). The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$.

For comparison, we also construct the transformation matrix for a planar uniform array with 25 microphones. The spectra of the real and imaginary parts of the uniform array's transformation matrix at 2.532 kHz are shown in Fig. 5.9 and Fig. 5.10, respectively. The figures show that the transformation matrix of the irregular array exhibited more random/low column coherence than the transformation matrix of the uniform array. More specifically, columns of $\mathbf{A}$ of irregular array and uniform array are taken randomly to form submatrices, then, with 10000 Monte Carlo trials, the maximum columns' coherence of submatrices are plotted Fig. 5.11 where the X-axis is the number of the selected columns (S-sparse).

By maximizing the degrees-of-freedom of the irregular microphone array, we increase the incoherence of the columns of $\mathbf{A}$. This results in increasing the stability of the sparse recovery algorithms, which is further elaborated on in section 5.3.2.

5.3.2 Recovery Algorithms for Sound Source Reconstruction

Before presenting the algorithm for sparse sound source reconstruction, we define some notations as follows.

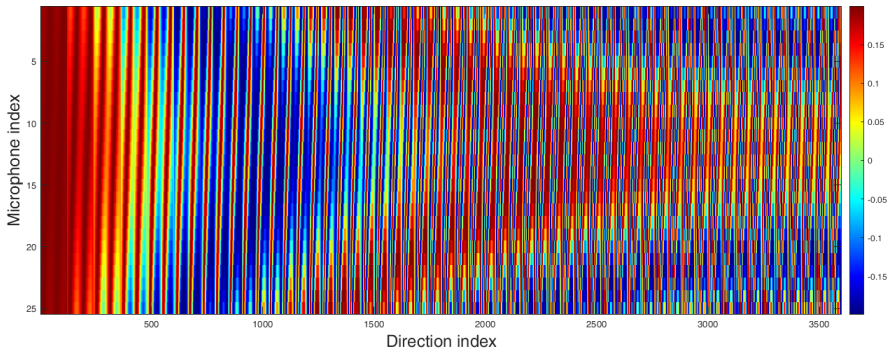


FIGURE 5.9: The spectrum of the real part of the transformation matrix at 2.532 kHz of the uniform array (25 microphones). The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$.

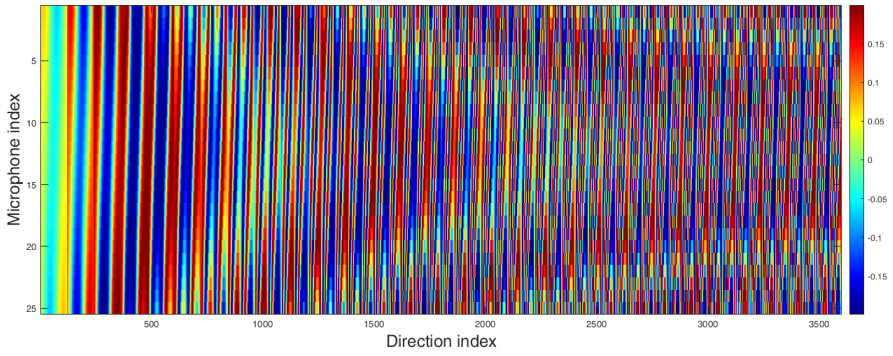


FIGURE 5.10: The spectrum of the imaginary part of the transformation matrix at 2.532 kHz of the uniform array (25 microphones). The X-axis from 1 to N is the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$, while the Y-axis from 1 to M is the microphone index where $M = 25$.

Definition 1. *A support-set* $\mathbb{T}$ *is a set of indices corresponding to the non-zero components in a vector* $\mathbf{s}$,

$$\mathbb{T} := \{i : s_i \neq 0\}. \tag{5.6}$$

Definition 2. *The complement of the support-set* $\mathbb{T}$ *is*

$$\bar{\mathbb{T}} := \{i : s_i = 0\}. \tag{5.7}$$

Definition 3. $\mathbf{s}_{\mathbb{T}}$ *is a vector formed by* $\mathbb{T}$ *elements of the vector* $\mathbf{s}$ *and* $\mathbf{A}_{\mathbb{T}}$ *is a matrix formed by* $\mathbb{T}$ *columns of the matrix* $\mathbf{A}$.

Definition 4. *The pseudo-inverse of matrix* $\mathbf{A}$ *is given by*

$$pinv(\mathbf{A}) = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H. \tag{5.8}$$

Definition 5. *A residual vector of vector* $\mathbf{x}$ *to the column space of* $\mathbf{A}_{\mathbb{T}}$ *is given by*

$$\mathbf{r} := resid(\mathbf{x}, \mathbf{A}_{\mathbb{T}}) = \mathbf{x} - \mathbf{A}_{\mathbb{T}} pinv(\mathbf{A}_{\mathbb{T}})\mathbf{x}. \tag{5.9}$$

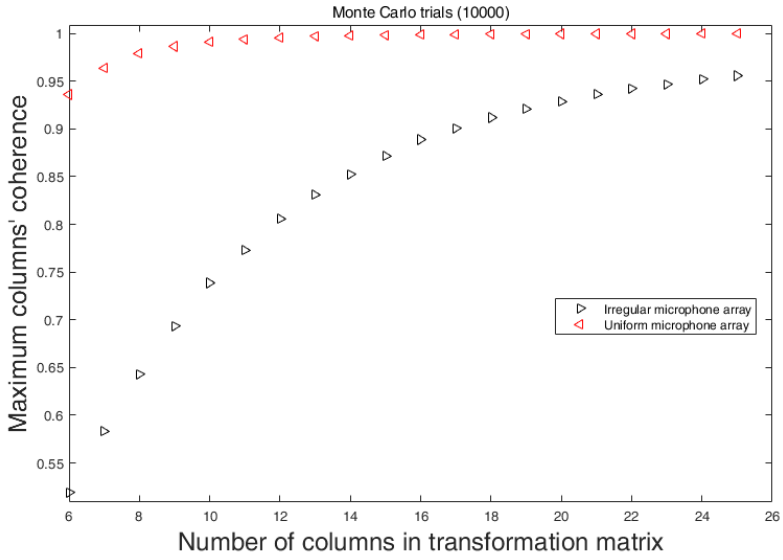


FIGURE 5.11: Analyzing the maximum columns' coherence of sub-matrices of the transformation matrix.

Definition 6. *The function max_indices*$(\mathbf{s})$ *returns indices corresponding to the largest amplitude components in the vector* $\mathbf{s}$*, if* $\mathbb{T}_{max} = max\_indices(\mathbf{s})$ *then*

$$\mathbb{T}_{max} := \{l : |s_l| \geq |s_i|,\ \forall i \in N\} \tag{5.10}$$

where N *is the size of* $\mathbf{s}$*.*

The transformation matrix $\mathbf{A}$ in (5.2) is deemed to satisfy the RIC $\delta_S(\delta_S \in (0,1))$ if δ_S is the smallest constant such that for every $\mathbf{x} \in \Sigma_S$,

$$(1-\delta_S)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1+\delta_S)\|\mathbf{x}\|_2^2. \tag{5.11}$$

Intuitively, δ_S measures "how well" the columns of $\mathbf{A}_\mathbb{S}$ for all $\mathbb{S} \subseteq \{1,\ldots,N\}$ and $|\mathbb{S}| = S$ (matrix $\mathbf{A}_\mathbb{S}$ is defined in *Definition* 3) forms approximately an orthonormal system [Mig14]. Equivalently, δ_S requires that the eigenvalue of any matrix $\mathbf{A}_\mathbb{S}^H\mathbf{A}_\mathbb{S}$ are within $1-\delta_S$ and $1+\delta_S$ so that all $\mathbf{A}_\mathbb{S}^H\mathbf{A}_\mathbb{S}$ are close to an isometric.

In general, it is difficult to check whether or not $\mathbf{A}$ satisfies the RIC δ_S. Another property of $\mathbf{A}$, which is easily verifiable and also ensures good recovery guarantees, is the coherence of the columns of $\mathbf{A}$. As can be observed in the Gram matrix of the column vectors of $\mathbf{A}$ in Fig. 5.12, several columns of $\mathbf{A}$ are highly coherent with some other columns in $\mathbf{A}$. This is especially the case with columns associated with azimuth angles that are close together, where the high coherence or columns associated with elevation angles close to zero (broad-side direction) also exhibited high coherence. Therefore, it is predicted that the basic pursuit algorithm (l_1-norm minimization) is not suitable for this system.

In CS, many algorithms could be applied to the under-determined system (5.2), including iterative hard threshold, subspace pursuit, Bayesian compressive sensing, compressive sampling matching pursuit, orthogonal matching pursuit, and basic pursuit. Here, we would select the orthogonal matching pursuit algorithm due to the following reasons. The algorithm requires less computation time than other

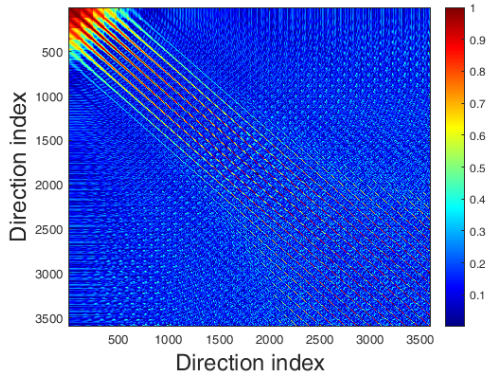


FIGURE 5.12: Spectrum of the Gram matrix ($\mathbf{A}^H\mathbf{A}$), the axes' values from 1 to N are the direction index where $N = m_1 \times m_2 = 30 \times 120 = 3600$.

methods. The orthogonal matching pursuit can reliably recover a signal with S-sparse in dimension N with M measurements with the total complexity $\mathcal{O}(SMN)$ [TG07], and it presents robustness against the transformation matrix with a high restricted isometry constants.

For comparison of the sparse recovery algorithms, we simulate three active sound sources at 2532.3 Hz impinging on the array in Fig. 5.4 with an inter-distance $d_H = 0.015\ m$: the 1^{st} active source comes from the direction $(\phi, \theta) = (80^o, 30^o)$, which has an amplitude of 1 and a phase of 60^o; the 2^{nd} active source comes from the direction $(\phi, \theta) = (230^o, 50^o)$, which has an amplitude of 0.7 and a phase of 45^o; the 3^{rd} active source comes from the direction $(\phi, \theta) = (180^o, 60^o)$, which has an amplitude of 0.5 and a phase of 35^o. The simulated sound source spectrum is presented in Fig.5.13.

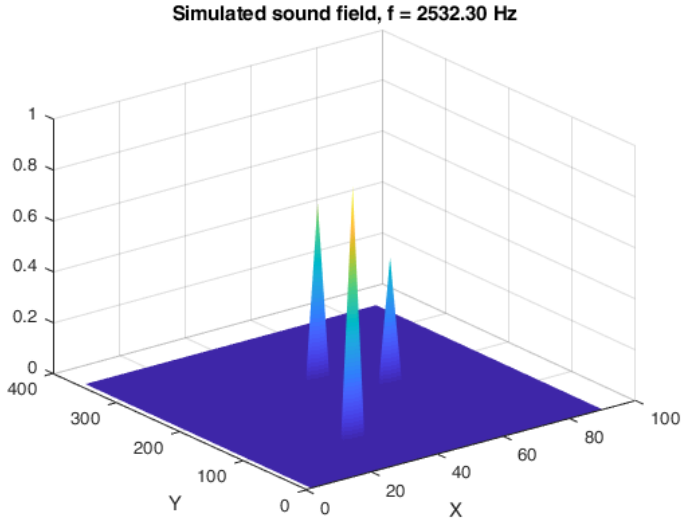


FIGURE 5.13: The simulated sound sources impinge on the array: The X-axis is the elevation angle with the range of $(0^o, 90^0)$, the Y-axis is the azimuth angle with the range of $(0^o, 360^o)$ and the Z-axis is the amplitude of the active sounds.

The signal-to-noise ratio (SNR) is 20 dB. Fig. 5.14 presents the reconstructed sound sources of iterative hard threshold (IHT), subspace pursuit (SP), Bayesian

compressive sensing (BCS), compressive sampling matching pursuit (CoSaMP), orthogonal matching pursuit (OMP) and basic pursuit (BP).

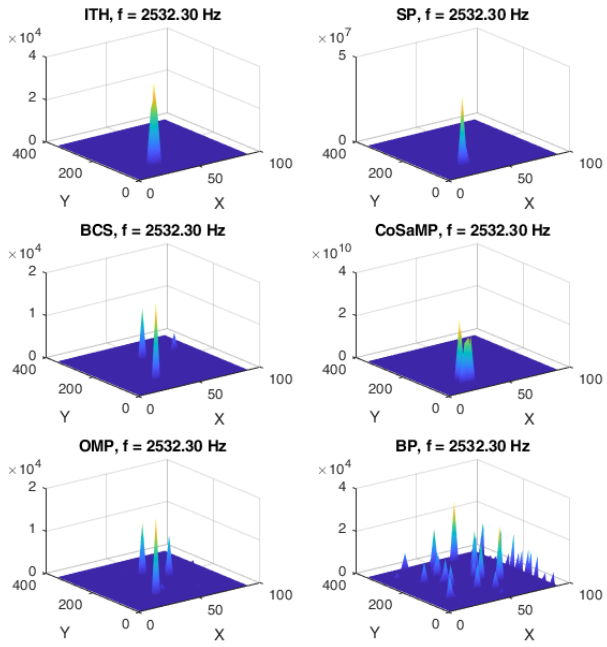


FIGURE 5.14: Comparison of sparse recovery algorithms for the sound source reconstruction: X-axis is the elevation angle with the range $(0^o, 90^0)$, Y-axis is the azimuth angle with the range $(0^o, 360^o)$ and Z-axis is the amplitude of active sounds.

To visualize the sparse data, the reconstructed sound sources in Fig. 5.14 are vectorized and depicted in Fig. 5.15. The 'black line' is the simulated sound sources, the 'cyan line' is the sound source reconstruction via the basic pursuit, the 'red line' is the sound source reconstruction via the orthogonal matching pursuit, the 'yellow line' is the sound source reconstruction via the compressive sampling matching pursuit, the 'green line' is the sound source reconstruction via the Bayesian compressive sensing, the 'blue line' is the sound source reconstruction via the subspace pursuit and the 'magenta line' is the sound source reconstruction via the iterative hard threshold.

Among all the algorithms, the orthogonal matching pursuit and Bayesian compressive sensing present the most superior performance. Furthermore, their sound source reconstructions are close to the simulated sound sources. However, the Bayesian compressive sensing results are strongly affected by the user-defined parameter, for example the variance of Gaussian noise, and the computational cost of the Bayesian compressive sensing is higher than that of the orthogonal matching pursuit. More precisely, the orthogonal matching pursuit yields a better performance than the Bayesian compressive sensing in terms of low energy active sources' estimation, as shown in Fig. 5.15: the 'red line' is almost identical with the 'black line', while the spareness of the 'green line' deviates from the sparseness of the 'black line' at the pulse with a small amplitude.

The orthogonal matching pursuit algorithm is thus selected for the sound source reconstruction. Certain properties of the orthogonal matching pursuit can be outlined to explain its superior performance. In each iteration step of the orthogonal matching pursuit, only one reliable component is added into the support-set $\mathbb{T}_k$.

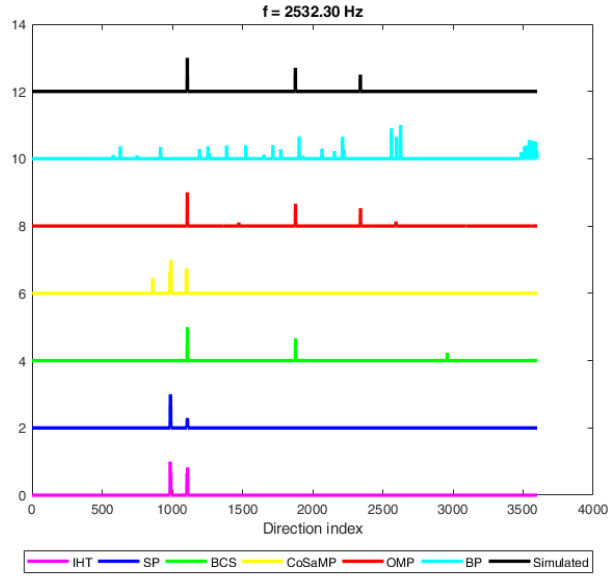


FIGURE 5.15: Comparison of sparse recovery algorithms.

This procedure indeed works well with the transformation matrix $\mathbf{A}$. Since $\mathbf{A}$ contains some columns with high coherence to others, if more than one component is newly taken, some unreliable components will be added to the support-set $\mathbb{T}_k$. Once the support-set $\mathbb{T}_k$ contains the wrong components, the residual vector $\mathbf{r}_k$ that is perpendicular to the subspace spanned by column vectors of $\mathbf{A}_{\mathbb{T}_k}$ will no longer be reliable. Then, in the next iteration, the procedure for identifying next active components will involve with an unreliable input (the residual vector), which means its outcome will no longer be reliable. However, this unexpected behavior does not occur if only one component is taken in each iteration. For example, at the first iteration, only the most reliable component is selected, then the residual vector $\mathbf{r}_1$ is orthogonal with a reliable subspace spanned by the column vectors of $\mathbf{A}_{\mathbb{T}_1}$. Therefore, in the next iteration, the next reliable component is taken successfully. Inductively, a reliable support-set is always maintained (even in the case where the transformation matrix $\mathbf{A}$ has the high column coherence).

Algorithm 2 Orthogonal matching pursuit

Input: $\mathbf{A}, \mathbf{x}, K_{max}, \epsilon$

Initialization :

1: $\mathbb{T}_0 = \emptyset$
2: $\mathbf{r}_0 = \mathbf{x}$
3: $k = 0$

Repeat

4: $k = k + 1$
5: $\mathbb{T}_{max} = \text{max\_indices}(\mathbf{A}^H \mathbf{r}_{k-1}, 1)$
6: $\mathbb{T}_k = \mathbb{T}_{k-1} \cup \mathbb{T}_{max}$
7: $\mathbf{r}_k = \text{resid}(\mathbf{x}, \mathbf{A}_{\mathbb{T}_k})$

Until $(\|\mathbf{r}_k\|_2 \geq \epsilon \|\mathbf{r}_0\|_2)$ ***and*** $(k < K_{max})$

Output: $\hat{\mathbf{s}}$ $s.t$ $\hat{\mathbf{s}}_{\mathbb{T}_k} = \text{pinv}(\mathbf{A}_{\mathbb{T}})\mathbf{x}$ and $\hat{\mathbf{s}}_{\bar{\mathbb{T}}_k} = 0$.

The procedure of orthogonal matching pursuit for the sparse sound source reconstruction is listed in Algorithm 2 where ϵ is the acceptance error value, K_{max} is the maximum number of iterations, $\mathbb{T}_k$ is the support-set of $\mathbf{s}$ at the iteration k, $\mathbf{r}_k$ is the residual vector at the iteration k and $\hat{\mathbf{s}}$ is the sparse sound source estimation. The Matlab code is provided in Appendix D.

5.3.3 Microphone Interpolation

Suppose that we have successfully estimated the sound source vector $\hat{\mathbf{s}}$ from the irregular microphone array, then it is possible to estimate the sound pressure at any point in the space. The interpolation of sound pressure at point $\mathbf{p}_x$ is the superposition of the active sounds' propagation,

$$\hat{x}(\mathbf{p}_x, f) = \sum_{k=1}^{N} \hat{s}_k(f) e^{-\frac{j2\pi f \mathbf{p}_x^T \mathbf{d}_k}{c}} \tag{5.12}$$

where $\hat{s}_k(f)$ is the k^{th} element of $\hat{\mathbf{s}}$.

In our method, we apply this interpolation for all the grid points $M \times M$ other than the real microphone positions. Finally, from M microphones, we are able to estimate the sound pressure at $M \times M$ points, which means we have an interpolated DUA.

5.3.4 Broadband Beamforming for Dense and Uniform Array

We need to determine the sensor coefficients for the interpolated DUA which satisfy the frequency-independent beam pattern. We refer to the coordinate transform method presented in Chapter 3, which comprises four steps for every frequency f:

- Step 1: define a reference beam pattern $b_{ref}(\phi, \theta)$
- Step 2: consider radius $R = \frac{fMd_H}{c}$, presenting the reference beam pattern to a gain function $b_R(\phi, \theta)$ in the spherical coordinate
- Step 3: gain function in the Cartesian coordinate $b_f(u, v)$ is achieved by transforming the surface $b_R(\phi, \theta)$ in the spherical coordinate to the Cartesian coordinate
- Step 4: apply inverse Fourier transform of $b_f(u, v)$ to achieve $\mathbf{w}(f)$

where d_H is the distance of the grids, $\mathbf{w}(f)$ is the weight values of the array at frequency f.

5.4 Numerical Simulations

In this section, we introduce a parameter for evaluating the frequency-independent characteristic. A beam pattern error indicating the average of the difference between the real BP $b_r(\rho_i, f)$ and the desired BP $b_d(\rho_i, f)$ at f is defined by [LS20]

$$\mathcal{E}(f) = \frac{\sum_{\rho_i \in \Theta} |b_r(\rho_i, f) - b_d(\rho_i, f)|}{P} \tag{5.13}$$

where Θ is the operating space of the beamforming and P is the size of Θ. Ideally, the value of $\mathcal{E}(\omega)$ is zero.

For comparison, a similar reference beam pattern of the sparse array design in [LS20] is examined:

$$b_{ref}(\phi,\theta) = \begin{cases} |\frac{\sin(\alpha\pi\theta)}{\alpha\pi\theta}|, & \theta \in (0, \frac{\pi}{2}], \alpha = \frac{4.4}{\pi}, \\ 1, & \theta = 0. \end{cases} \tag{5.14}$$

We simulate a sound source at a certain direction impinging on the irregular microphone array (the inter-distance of the grid points $d_H = 0.015\ m$), it is shown in Fig. 5.4. Here, the direction of sound sweeps over the entire space. During the sweeping process, we could solve the problem in (5.2) via the OMP to reconstruct the sound sources. An example of the solution of the equation (5.2) at 4 kHz is presented in Fig. 5.16. Next, we apply the equation (5.12) with the reconstructed sound sources to obtain the interpolated DUA. Finally, we perform the beamforming for the interpolated DUA and verify its beam patterns.

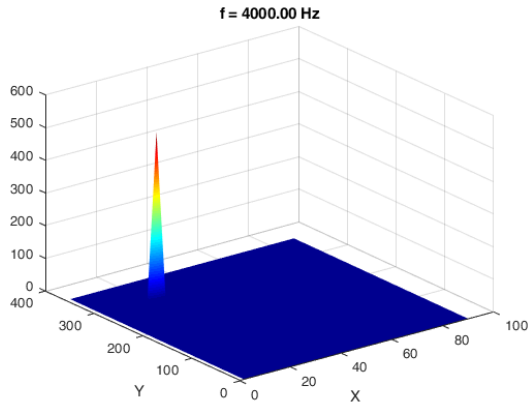


FIGURE 5.16: The reconstructed incident sound by the OMP: X-axis is the elevation angle, Y-axis is the azimuth angle, Z-axis is the strength of the incident sound.

In comparison with other sound source reconstruction methods, we also reconstruct the sound sources by applying the LSE minimization to solve the equation (5.2),

$$\hat{\mathbf{s}} = \text{pinv}(\mathbf{A})\mathbf{x}. \tag{5.15}$$

Similar to what occurred with the simulation of the OMP algorithm, an incident sound is sweeping over the whole space. The sound source spectrum of the LSE minimization method is shown in Fig. 5.17. This reconstructed sound sources is not sparse, even in a noiseless scenario.

In terms of the OMP algorithm, Fig. 5.18 and Fig. 5.19 present the simulation results of two cases, noiseless and white noise (SNR = 20 dB), respectively. In each case, the figure presents the beam patterns of the irregular microphone array (left) and the DUA (right) at different frequencies: 2 kHz, 4 kHz, 8 kHz and 10 kHz.

Following the LSE (l_2-norm) minimization, the beam patterns are obtained, as presented in Fig. 5.20 and Fig. 5.21 for the noiseless and white noise cases, respectively.

The comparisons of the beam pattern error are presented in Table 5.1. The table includes the beam pattern errors at the different frequencies (2 kHz, 4 kHz, 8 kHz and 10 kHz) for two cases, noiseless and white noise (SNR = 20 dB). Clearly, the LSE minimization fails to interpolate the DUA at the middle and high frequencies.

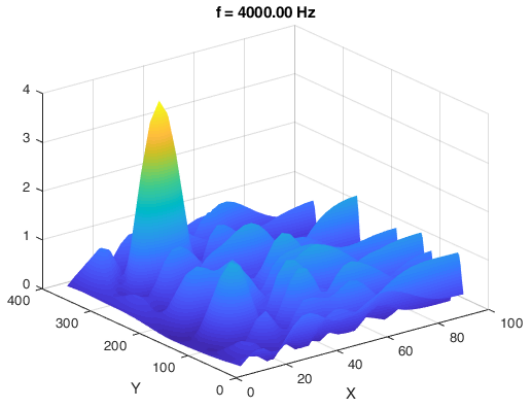


FIGURE 5.17: The reconstructed incident sounds by the l_2-norm: the X-axis is the elevation angle, the Y-axis is the azimuth angle and the Z-axis is the strength of the incident sound.

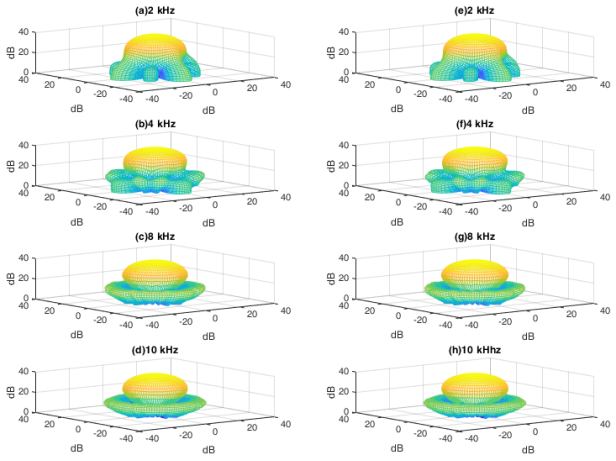


FIGURE 5.18: In the case of noiseless, using the OMP algorithm for the sound source reconstruction: Beam pattern of the irregular microphone array (left) and beam pattern of the full microphone array (right).

Even though its norm of the error in the equation (5.2) (it is minimum mean square error) is smaller than that of the OMP, the technique does not present the physical characteristics of the simulated sound sources. For the OMP algorithm, the beam patterns of the irregular microphone array can compete with the beam patterns of the DUA for all frequencies, which means that the sound source estimation obtained by the OMP algorithm is good enough to interpolate the DUA. In the case of white noise, the irregular microphone array even outperforms the DUA, which is because, if the sound source is estimated by sparse-algorithms, then estimated sound field doesn't take the white noise into account. It leads to the interpolated signal is a noise reduction signal. That is illustrated in the equations (5.1) and (5.12) where the noise term of (5.1) doesn't presented in (5.12).

In comparison with the sparse array design in [LS20], the irregular array requires fewer microphones, that is, 25 microphones compared to 81 microphones, while its frequency-independent beam pattern characteristic is better.

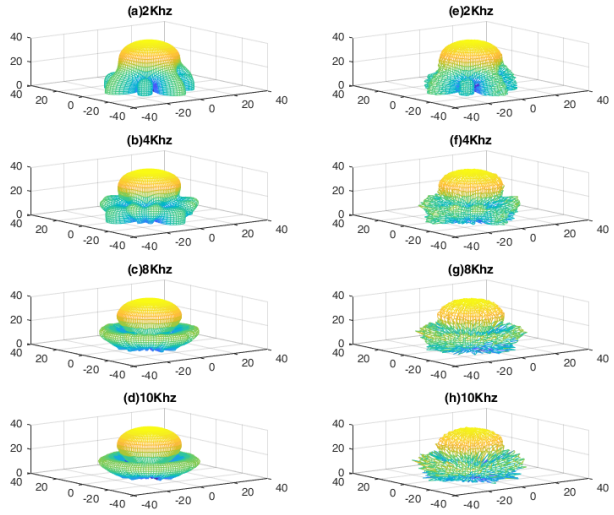


FIGURE 5.19: In the case of white noise, SNR = 20 dB, using the OMP algorithm for the sound source reconstruction: Beam pattern of the irregular microphone array (left) and beam pattern of the full microphone array (right).

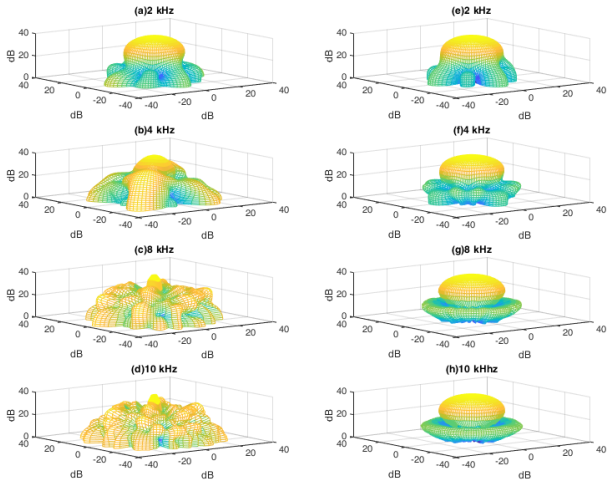


FIGURE 5.20: In the case of noiseless, using LSE minimization for the sound source reconstruction: Beam patterns of the irregular array (left) and beam patterns of the DUA (right).

5.5 Experimental Results

The proposed irregular microphone array is examined in an office room with dimensions of $3.5 \times 6 \times 3$ m^3. Here, 25 Adafruit I2S Micro-Electro-Mechanical Systems (MEMS) microphone boards are used to build the irregular microphone array in Fig. 5.4 (the inter-distance of the grid points $d_H = 0.015$ m), and one additional MEMS microphone board is used for verification purpose. The position of the verification microphone (microphone 26) is at $[-0.205, 0.08, 0]$ m in the Cartesian coordinate.

The MEMS microphone board is equipped with SPH0645LM4H-B MEMS microphone by Knowles: omnidirection, standard 24-bit I2S format, 65 dB SNR, up to 64

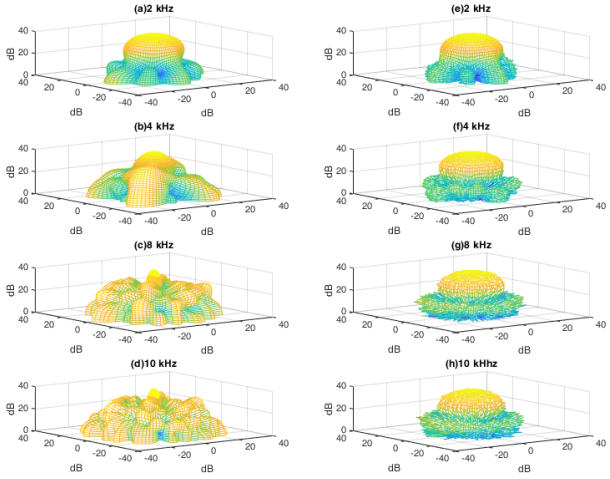


FIGURE 5.21: In the case of white noise, SNR = 20 dB, using LSE minimization for the sound source reconstruction: Beam patterns of the irregular array (left) and beam patterns of the DUA (right).

TABLE 5.1: BEAM PATTERN ERROR

	Frequency			
	2 kHz	4 kHz	8 kHz	10 kHz
OMP	0.0697	0.0388	0.0239	0.0226
DUA	0.0616	0.0306	0.0132	0.0112
LSE	0.0694	0.2008	0.2609	0.2746
OMP(noise)	0.0698	0.0389	0.0241	0.0227
DUA(noise)	0.0622	0.0382	0.0416	0.0503
LSE(noise)	0.0695	0.2008	0.2609	0.2746

kHz sampling frequency. DSP56725 chip by NXP is used to record all 26 MEMS microphones. The audio signals are processed with DC-removal and are streamed out via a TDM 30 channels 32 kHz sample frequency, 32-bit PCM (four channels are unused). A Beaglebone-Black board is used to record the TDM 30 channels from the DSP56725 by using the multichannel audio serial port in the AM3358 (Arm Cortex-A8 by Texas Instruments).

5.5.1 Experiment 1

A sound source at the direction $(\phi, \theta) = (0^o, 90^o)$ impinged on the irregular microphone array for five seconds, which contains the normal speech and a hand-clapping sound, as shown in Fig. 5.23. All the recorded signals were partitioned into overlapping frames with a frame size of 128 and an overlapping factor of 50%. A Hanning window was then applied to each frame. The windowed frame signal was added with 64 zero-padding in the head and tail before being transformed into the short-time Fourier transform (STFT) domain using a 256-point fast Fourier transform.

For comparison, the sound source reconstructions were computed offline in two cases: the OMP algorithm with $K_{max} = 6$ and the LSE minimization algorithm. Note that if $K_{max} = 25$ and $\epsilon \approx 0$, then the OMP algorithm is equivalent to the l_2-norm algorithm. Due to the limitation of the hardware (Processor: Intel(R) Core(TM) i7-6500U CPU@2.5GHz(4CPUs), Ram: 8Gb), we restricted the size of SSV to $N =$

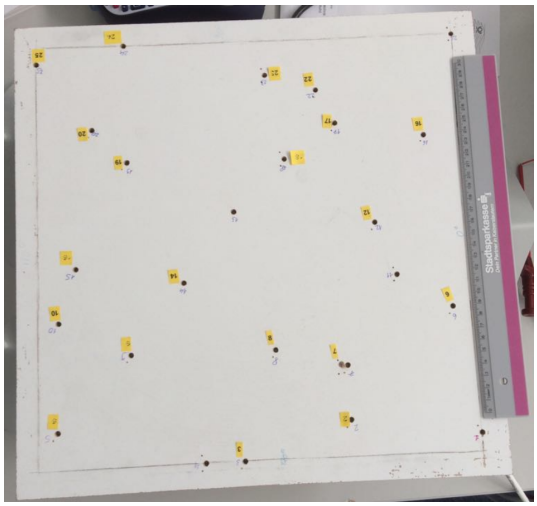


FIGURE 5.22: Experiment with the microphone array in Fig. 5.4.

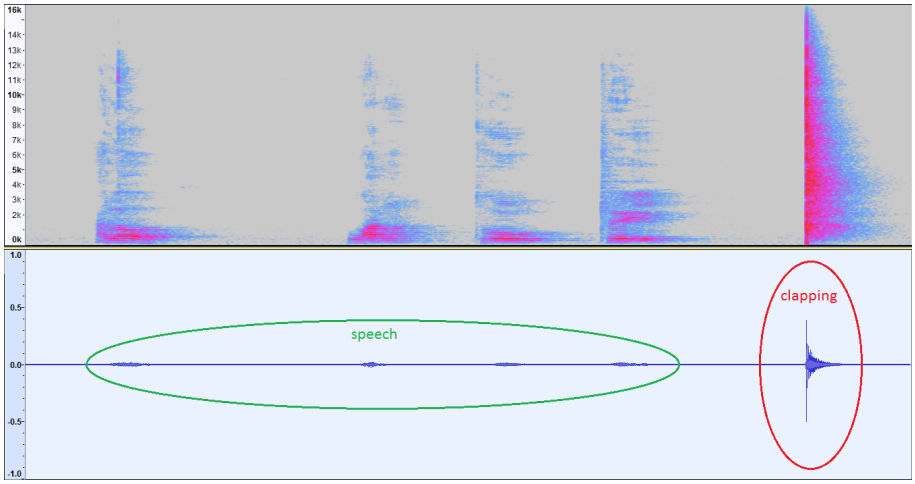


FIGURE 5.23: Spectrogram and waveform of the audio signal from microphone 5. The horizontal axis is the time, the vertical axis of the top and bottom figures are the frequency and amplitude, respectively.

$m_1 \times m_2 = 5 \times 60 = 300$. After the reconstructed sound sources were obtained, we applied formula (5.12) to interpolate the audio signal at the position of the verification microphone. The audio spectra of the real microphones and the interpolated microphones are shown in Fig. 5.24. Here, audio from the microphone 5 is on the top, with the verification microphone audio (microphone 26) below, followed by the interpolated microphone via the OMP algorithm, and, at the bottom, the interpolated microphone via LSE minimization algorithm. We applied our algorithms for the frequencies form 200 Hz to 15 kHz.

A part of the audio signals (the hand-clapping sound) is showed in Fig. 5.25. Here, audio from microphone 5 is on the top, with the verification microphone below, followed by the interpolated microphone via the OMP algorithm, and, at the bottom, the interpolated microphone via the LSE minimization algorithm.

By comparing the spectra and the waveform of the interpolated audio signals in Fig. 5.24 and Fig. 5.25, it becomes clearly that the interpolated signal achieved by the sparse sound source reconstruction (OMP) is almost identical with the verification

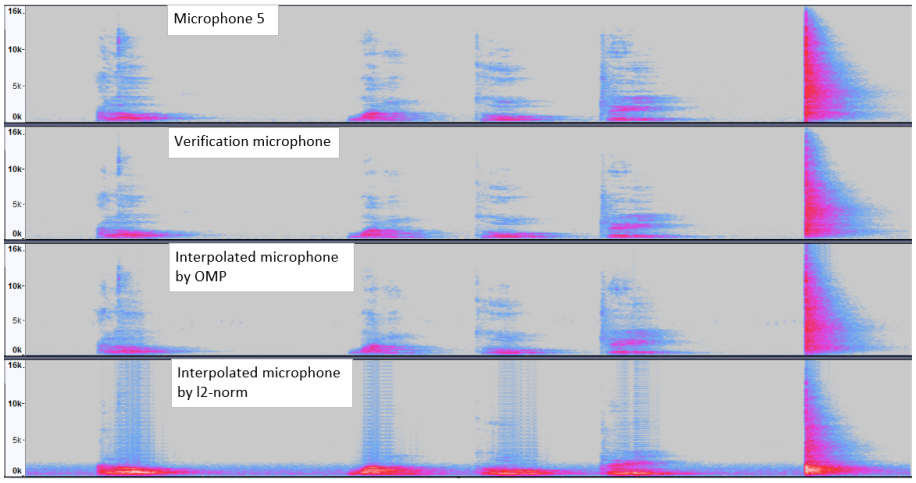


FIGURE 5.24: Spectra of the signals (from top to bottom of the figure): microphone 5, microphone 26, interpolated audio via the OMP, and interpolated audio via the LSE (l_2-norm). The horizontal axes are the time (s), the vertical axes are the frequency (kHz).

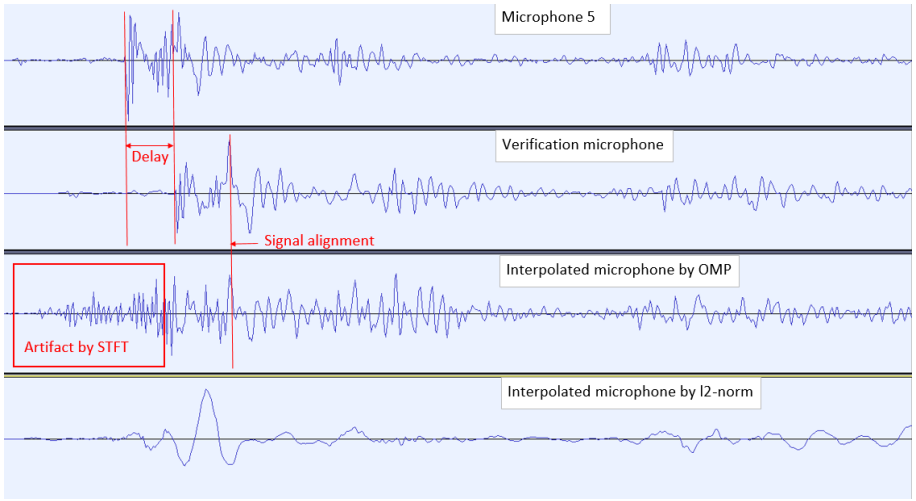


FIGURE 5.25: Waveform of the signals (from top to bottom of the figure): microphone 5, verification microphone, interpolated microphone via the OMP, and interpolated microphone via the LSE (l_2-norm). The horizontal axes are the time (s), the vertical axes are the amplitude.

signal. The delay and the amplitude of the signal are estimated successfully, which assures the interpolated signals can replace the real signals in the DUA to perform the beamforming, while the LSE minimization algorithm deteriorates the audio signal at the verification point, especially at middle and high frequencies (consistent with the results of the simulation in Section 5.4). In summary, the sound source reconstruction via the LSE minimization algorithm is unsuitable for interpolating the audio signal.

More specifically, Fig. 5.25 shows, the interpolated signal from the OMP generates the artifact at the place immediately before the clapping sound, which results from the audio signal's frame partition for carrying out the STFT. The reconstructed

sound sources were solved independently at every frame, meaning the interpolated signal of a frame cannot be certain to synchronize in phase/amplitude with the interpolated signal of the adjacent frames, especially for the pulse signal (which is asynchronous to the frame). The length of the artifact will be varied depending on the size of the frame. This is the main disadvantage of the method presented in this chapter, and the author has yet to identify the solution to this issue. Fortunately, this artifact is mitigated when applying the beamforming to the signals (shown in the bottom of Fig. 5.26), since the artifact at an interpolated microphone is a random noise, which means the beamforming could suppress this form of noise. As Fig. 5.26 shows, we carried out the interpolation for 25×25 microphones before the beamforming for the interpolated DUA was applied (the steering angle is the direction of the sound source).

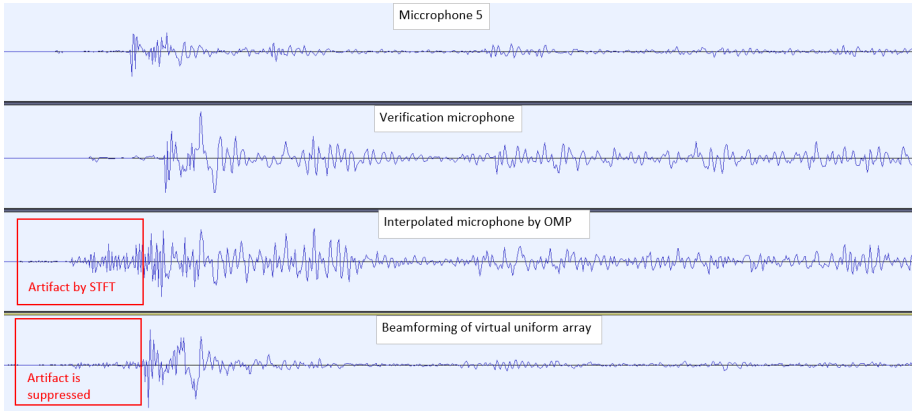


FIGURE 5.26: Waveform of the signals (from top to bottom of the figure): microphone 5, microphone 26 (verification microphone), interpolated microphone by the OMP, and beamforming. The horizontal axes are the time (s), the vertical axes are the amplitude.

In order to verify the function of beamforming, we applied the beamformings for the interpolated DUA, which steer to $(\phi, \theta) = (0^o, 90^o)$ (the direction of the source source) and $(\phi, \theta) = (180^o, 90^o)$ (the inverted direction of the source source). The results are shown in Fig 5.27 and Fig. 5.28, where the waveform and spectrum of microphone 5, the beamforming signal toward $(\phi, \theta) = (0^o, 90^o)$ and the beamforming signal toward $(\phi, \theta) = (180^o, 90^o)$ are presented.

It is clearly seen that beamforming toward the source of interest could preserve the signal of the sound source, while beamforming toward another direction will suppress the signal of the sound source, and reverberation is reduced in both cases. In this experiment, we aimed to verify the signals of the interpolated microphones and their conjunction with broadband beamforming. The properties of the beam pattern have to be verified by using an anechoic room, which was not covered here.

5.5.2 Experiment 2

Speaker 1 at the direction $(\phi, \theta) = (0^o, 0^o)$ played the sound, while a person at the direction $(\phi, \theta) = (80^o, 180^o)$ (interference sound) also spoke into the microphone array. We carried out the interpolation for the DUA with the sparse sound source reconstruction, before applying the beamforming steering in the direction of speaker

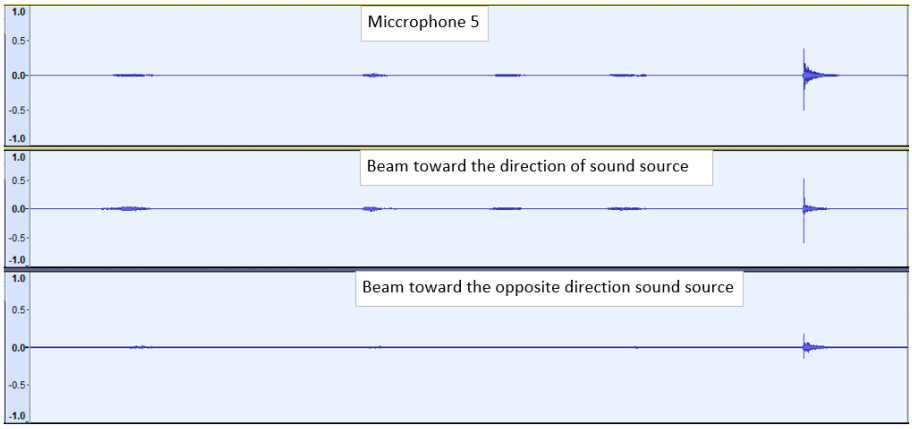


FIGURE 5.27: Waveform of the signals (from top to bottom of the figure): microphone 5, the beam signal steers to the direction of the source of interest, and the beam signal steers to the inverted direction of the source of interest. The horizontal axes are the time (s), the vertical axes are the amplitude.

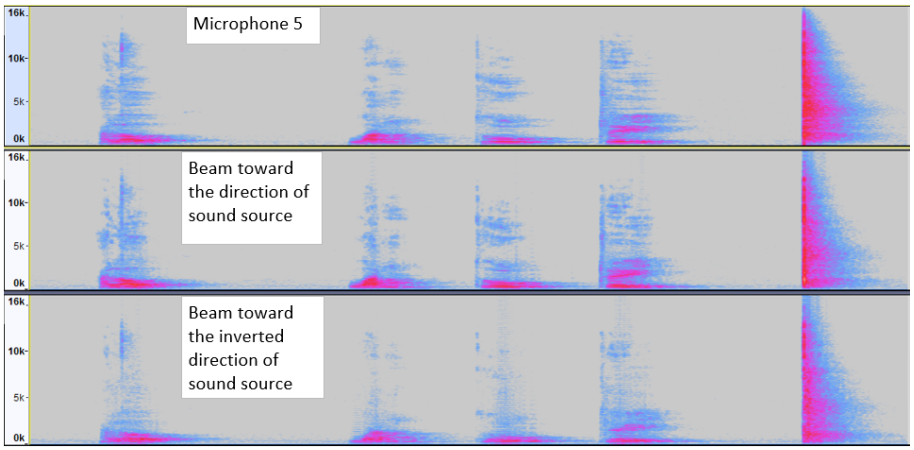


FIGURE 5.28: Spectrogram of the signals (from top to bottom of the figure): microphone 5, beam signal steers to direction of source of interest, and the beam signal steers to the inverted direction of the source of interest. The horizontal axes are the time (s), the vertical axes are the frequency (kHz).

1 $(\phi, \theta) = (0^o, 0^o)$ and comparing the beamforming signal in relation to the signal of a microphone.

As presented in Fig. 5.29, beamforming improves the quality of signal of interest while suppressing the interference sound, even in the case of overlapping speech between the source of interest and interference (zone 1, 2 in Fig. 5.29). It should be noted that testing was performed in a reverberant room.

From the results of the simulations and experiments, it was demonstrated that the proposed interpolation method successfully interpolates the DUA from the irregular microphone array. As such, the irregular microphone array could be used for broadband beamforming.

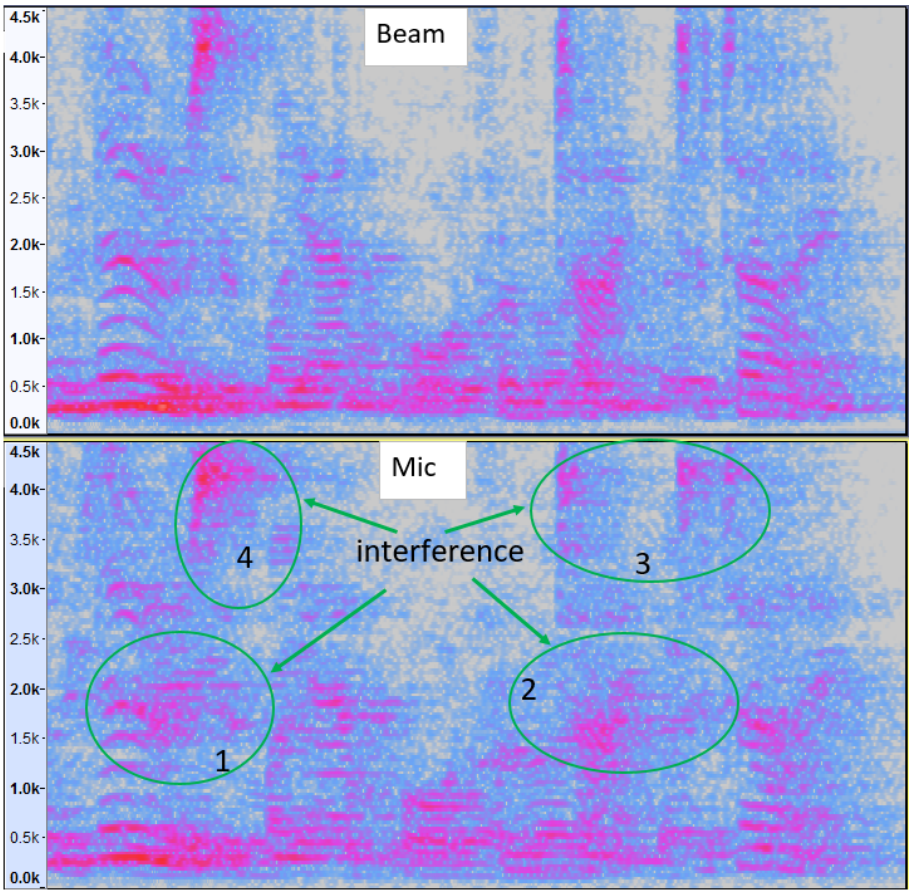


FIGURE 5.29: Spectrogram of the signals (from top to bottom of the figure): the spectrum of the beamforming steering to the speaker 1 at $(\phi, \theta) = (0^o, 0^o)$ and the spectrum of microphone 5 (there are overlapping sounds on zone 1, 2). The horizontal axes are the time (s), the vertical axes are the frequency (kHz).

5.6 Conclusions

In this chapter, a new approach for an irregular microphone array design is proposed. Based on the CS framework, we also propose a method for obtaining the interpolated microphone array with beamforming performed on the array. The method's performance is comparable to a dense and uniform array with a much larger number of microphones. In theory, in the presence of noises, the irregular microphone array could even outperform a DUA, largely because the sparse sound source reconstruction acts as the noise reduction filter for the interpolated microphones. In comparison with other sparse array designs, the number of microphones in the proposed array is reduced. For example, only 25 microphones are required in comparison to the 81 required for the sparse array in [LS20]. Furthermore, a single microphone array deployment presents a versatile design that can be used for different beam patterns since the proposed design is not based on a reference beam pattern; rather it is aimed at adopting the sparse recovery algorithms.

Generally speaking, the number of measurements M larger than four times S-sparse offers a "reasonably high" probability of successful reconstruction [CRT06; Mig14]. With the use of 25 microphones, as used in our experiments, it is feasible to reconstruct the sparse sound source with a maximum of six active sources at a

frequency. It is also feasible to measure an environment with more active sources by increasing the number of microphones in the irregular array.

Chapter 6

Discussion and Future Directions

6.1 Discussion

As mentioned over the entire content of this dissertation, beamforming is an important topic in array signal processing that enables the spatial selectivity of incoming signals to recover the signal of interest and/or suppress the unwanted components in the sensor signal. There are several important aspects of beamformers that need to be considered during design a beamformer, such as frequency-independence, robustness, steering ability, array gain, .etc. The work in this dissertation mainly focused on finding good methods for broadband beamformer designs, including deciding the number of sensors, the position of sensors and their coefficients.

Over the last decades, using uniform arrays for beamforming attracts extensive researches and is widely deployed in practical applications. The advantages of uniform arrays are the simple physical models in which researchers could apply a variety of existing mathematical theories to solve every particular problem of beamforming, such as frequency-independence, directivity factor, white noise gain and steering ability. As the demand for new features is increasing, the bandwidth of interest is also expanded. To deal with broadband signals, a uniform array needs a lot of sensors, that is sometimes impractical. Recently, based on the evolution of mathematical tools likes compressed sensing, optimization techniques, machine learning, etc., the sparse array is gaining more and more attentions, largely due to the ability to reduce the number of sensors.

In this dissertation, we considered several critical issues regarding sparse arrays with broadband beamforming. First of all, a closed-form solution of beamforming for sparse arrays is difficult to derive, since the sensor deployments are somewhat irregular and sometimes random. Different approaches with different objective functions yield different array deployments. Among various array deployments, which one is the optimal one is still not clear in mathematical sense. In Chapter 4, we propose a new approach to design the sparse arrays for data-independent beamforming that outperforms the state-of-the-art approaches, that is, with similar amount of sensors the new design could gain a better directivity factor, white noise gain as well as frequency-independence. Besides, the computational complexity of the proposed method is efficient, it spends less than one second to find the layout of a planar array, while other methods are either infeasible or time-consuming. To the best of author's knowledge, the proposed method is one of the first methods that uses a dense and uniform array as an input and the clustering algorithm as a core component. This contribution provides a new approach for sparse array design. However, we cannot claim our array is the optimal design for the sparse array, since we did not dig into mathematics to analyze this concern yet.

From a different standpoint with Chapter 4, Chapter 5 proposes an irregular microphone array in a random sense, that array has maximum the distinct distances

between sensors and it aims to be used with adaptive beamforming. This strange objective function is actually a reasonable choice for sound field reconstruction via compressed sensing framework. From an irregular microphone array, we also propose a simple method to interpolate a dense and uniform array that then is to be processed with a beamforming algorithm. The proposed method helps to reduce the number of sensors substantially, while its performance is equivalent to a dense and uniform array. Besides, the irregular microphone array is a versatile design that the desired beam pattern is arbitrary and can be changed online (at running time). However, finding such an array deployment is an NP-hard problem, in this research we use a stochastic approach to find a good array layout but we do not know how close the solution comes to optimal solutions. A further study to analyze this result is valued to evaluate the efficiency of the proposed method.

Although Chapter 3 only focuses on the uniform array design, its outcomes are also important for the proposed sparse array designs. We introduce the inverse beam pattern transform, which deduces the sensor coefficients from an arbitrary desired beam pattern. As based on the bijection of the inverse beam pattern functions, we propose the bound constraints for spatial sampling from broadband beamforming perspective. Although the uniform arrays work well with broadband beamforming, a few approaches use the information of uniform arrays as the input for sparse array design.

Last but not least, Appendix A presents the beamformer in general form where two variants of general sidelobe canceler (GSC) beamformers were proposed. One is a data-independent beamforming and another is an adaptive beamforming. The proposed methods exhibits some advantages, i.e., the computational complexity is reduced and the signal-to-noise-plus interference is increased. Besides, the method can be applied to combine multiple sensor arrays.

6.2 Future Directions

Regarding the sparse array with data-independent beamforming mentioned in Chapter 4, some parameters are empirically set in the proposed approach such as the number of the groups or the number of representative sensors for each group. These settings need to be addressed in a mathematical viewpoint in future work. Furthermore, other common classification algorithms such as the support vector machine or relevance vector machine could be applied to identify the critical sensors instead of taking the closest sensors to the group's centroid of K-means clustering.

For the method mentioned in Chapter 5 (adaptive beamforming), the extension of the irregular array to three-dimensional (3D) is a worthwhile examination. The restricted isometry property is improved substantially if we extend the array to 3D, that may increase the performances of beamforming significantly. Moreover, that 3D array could use for other applications such as sound field reconstructions, sound source localization, etc.

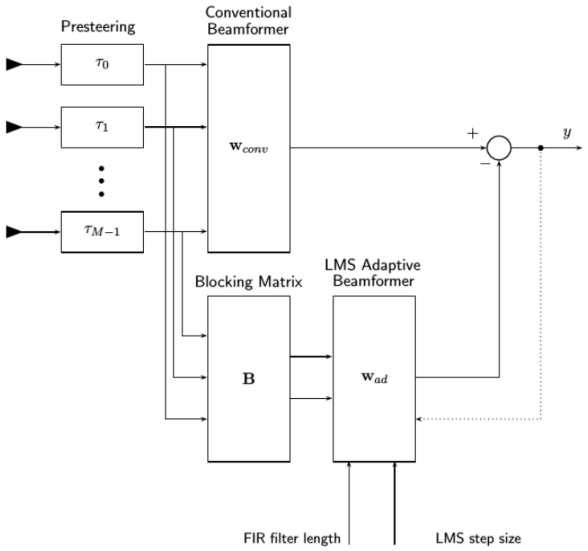
Chapter 3 introduces the transformation that transforms an arbitrary beam pattern to beamforming coefficients of an uniform array. This study only focuses on the far-field signal, a study of the transformation for near-field signal is an interesting research, since for audio applications the far-field assumption is not always assured. Furthermore, the constraint of equi-distance of sensors in X-axis and Y-axis could be removed, that is the equi-distance in X-axis could be different with that of Y-axis. In such a case, we need to build a new formula for the inverse beam pattern function based on the ellipsoid expansion.

In this work, we mainly focus on theories and simulations, other than Chapter 5 containing some experiments, more experiments to validate the results of proposed methods would consolidate the contributions of our work.

Appendix A

Alternative Approaches to Generalized Sidelobe Canceler

The generalized sidelobe canceler (GSC) decomposes the beamforming into two paths: The upper path is to preserve the desired signal, the lower path is to suppress the desired signal. More specifically, the upper path is equipped with a conventional beamformer, while the lower path is equipped with a blocking matrix and least mean squares (LMS) filters. The subtraction between the signal in the upper path and the signal in the lower path could remove the unwanted signals.



Overview of generalize sidelobe canceler.

The GSC design was proposed in the time-domain, then it is easily applied for the broadband beamforming. From beam pattern viewpoint, we propose alternative approaches to the GSC, the new algorithms mainly modify in the lower path, that is, instead of using the blocking matrix to suppress the desired signal, we design a beamformer that contains the nulls at the look direction and some other directions. We claim that the beamforming with a null at look direction performs a similar function to the blocking matrix. Moreover, the GSC subtracts two beam patterns to suppress the side-lobes, this amplifies noise when the sidelobes of the upper path beam pattern are not identical to the lower path beam pattern, especially at the null positions of the upper path beam pattern. To tackle this issue, we insert those nulls to

the lower path beam pattern to increase the similarity of the beam patterns at two paths. By doing so, we propose two variants of the GSC, one is the fixed beamforming where the adaptive filter in the lower path of the GSC is removed and the other is the adaptive beamforming where only one adaptive filter is used.

A.1 Introduction

Array signal processing (ASP) [KV96; Mai17] has been widely employed in diverse areas such as acoustics [Bra01; BCH08], radio-interferometry [RTMSJ17; Sim15], radar and sonar systems [Mai17; Hay85], wireless networks [God97; PP97; HS16] and medical imagery [LL00; Raf15]. Beamforming is an important topic in ASP [KV96], that is the process of performing spatial filters to preserve the signal from directions of interest while suppressing interfering signals and noise arriving from other directions. A fixed beamformer is a beamformer whose coefficients are independent on the measurement signals, it normally uses to suppress stationary noise. On the other hand, an adaptive beamformer is a beamformer whose coefficients depend on or adapt to measurement signals, it is capable of suppressing interferences but it suffers from computationally intense and signal distortions. In some applications, the bandwidth of signals of interest spreads over several octaves, therefore the characteristics of the beamformer should be invariant over the frequencies of interest, which is achieved via so-called broadband beamforming.

For instance, if noise fields are stationary and well-defined, then using fixed beamforming is a reasonable choice. Most fixed beamformers have closed form expression for the beamformer coefficients, it means the beamformer coefficients could be computed offline from the noise models. For example, the delay and sum beamforming (DSB) is the simplest beamforming technique where the signals of sensors are delayed to align in phase and then be summed [VT04]. Actually, the beamformer coefficents of DSB are the optimal solution for the noise suppression when only white noise is immersed in the signal. Later, superdirective beamforming (SD) [Elk00] was proposed to consider the presence of diffuse noise. However, at low frequencies, SD beamforming amplifies white noise. Therefore, the regularization of SD, considering the white noise, is commonly used in practical applications. For acoustic signals, differential microphone arrays (DMA) are used in the variety of applications. Conventional DMA is based on the spatial derivatives of the acoustic pressure field [Elk00; Elk04; BJ12]. Since the sensor spacing of the DMA is much smaller than the acoustic wavelength, the DMA is small in size which can be easily mounted into other devices. On the other hand, based on the short-time Fourier transform (STFT), spatial filtering is applied to form a differential beamformer in each subband [TE04; BCC15; ZBC14; CPC15]. The order of the differential beamformer could be designed by selecting the number of null-constraints and the type of differential beamformer could also be obtained by assigning the null positions and/or changing the optimization objective function such as maximum front-to-back ratio for a supercardioid beam pattern, maximum directivity index for a hypercardioid beam pattern, etc.

The early adaptive beamforming technique is the linearly constrained beamforming [For72], which minimizes the energy of an array output with subject to distortion-less at look direction. Over the years, the general approach for this class of beamformer has been studied as the so-called linearly constrained minimize variance (LCMV) [VT04; Dob10; SBA10] where the objective is to minimize the variance of the array output, that enables to resolve the problem from statistic viewpoint. In

LCMV, the input signals are modeled to the covariance matrix computed via multiple frames observation, then the closed form expression for the solution is derived for some simple cases. An alternative approach to linearly constrained adaptive beamforming was proposed by J. Griffiths and W. Jim in [GJ82], that is the general sidelobe canceler (GSC) beamformer. In GSC, they modified the constrained problem in [For72] to unconstrained problem by introducing a fixed beamforming at the upper path and a blocking matrix at the lower path. The lower path signals are filtered by the adaptive FIR filters before be subtracted from the upper path signal. This scheme could preserve the signal of interest (SOI) in the upper path while removing the noise in the lower path. This simple approach could be easy to deploy in the real-time implementation. Besides, the GSC is designed and processed in the time-domain, it is well-suited to broadband signals. However, the subtracting in GSC has some drawbacks, e.g., the GSC does not assure the noises/interferences in the lower path and in the upper path are almost identical, then substracting them does not suppress the noises/interferences effectively. In this chapter, we explain this issue from the beam pattern perspective. We regard the blocking matrix as the first order differential beamformings with a null at the look direction. Then, we analyze the difference between upper path's and lower path's beam patterns. In most of cases, the sidelobes of upper path beam pattern is much different with the lower path's beam pattern and the difference is varying over frequency. Then, the subtraction can not remove the sidelobes effectively. Furthermore, the blocking matrix generates multiple almost identical beam patterns at the lower path, that is somewhat redundant.

To overcome those problems, we propose modified GSCs, that is, instead of using a blocking matrix in the lower path, we design a beamformer that has a null at look direction and some nulls at the nulls' position of the upper path beamforming. Some extra nulls at the lower path make the lower path beam pattern is more similar to the sidelobes of the upper path beam pattern. Besides the shape of beam patterns, the phase of beam patterns is also important that should be aware before subtracting the signals. For instance, the phase of the upper path and lower path beam pattern are almost stable over the direction, we can use a fixed scale value in the lower path to correct the phase and amplitude of the subtracted signal. That is the case when the array layout is symmetric and the reference point is selected at the center of the array. As such, a fixed beamforming is proposed, referred to as a variant of the GSC. On the other hand, if the phase difference is varying over the direction, we use an adaptive FIR filter at the lower path. Depending on the direction of interferences at an instance time, the FIR filter can scale the amplitude and adjust the phase of the lower path beam pattern accordingly. This adpative beamforming is also considered as another variant of the GSC. We briefly summarize the contributions of our work on beamforming design as follows:

1. We present some new insight into the GSC beamformer where we claim that the response of the blocking matrix is similar with that of first order Cardiod beam patterns with the null at look direction. From a beam pattern standpoint, we draw some disadvantages of the GSC.

2. We propose a fixed beamforming design for a symmetric array layout. For this kind of array layout, we prove that the complex-valued beam pattern of superdirective beamforming contains only the real part if we select the reference point at the center of the array. The beamforming designed by this approach has a frequency-invariant mainlobe and the sidelobes are suppressed as much as possible.

3. For an arbitrary array layout, we propose an adaptive beamforming design. Its performance is not only superior to other methods, the computational complexity is also reduced. Moreover, this methodology gives the opportunity to combine multiple arrays to form a single expected beam pattern.

As the basis of our proposed approaches, we review the GSC beamforming in Section A.2. In Section A.3, we propose two alternative approaches to the GSC: A fixed beamforming with suppressed sidelobe (FBSS) and an adaptive beamformer with suppressed sidelobe (ABSS). The simulations and their results are provided in Section A.4. Finally, the conclusion is drawn in Section A.5.

A.2 Generalized Sidelobe Canceler

Consider an array with M sensors, the GSC algorithm was proposed in the time-domain [GJ82] where the signal of sensor m at the discrete time index n is defined by

$$x_m(n) = s_m(n) + n_m(n),\ m = 1, \ldots, M$$

where $s_m(n)$ and $n_m(n)$ are the signal of interest (SOI) and noise, respectively. We can present the SOI at a sensor as the wave propagation from the source to the sensor

$$s_m(n) = \mathbf{a}_m * \mathbf{s}_0$$

where $\mathbf{a}_m$ is the transfer function of the wave propagation, $\mathbf{s}_0$ is the source signal and '$*$' is the convolution operation.

Let $\mathbf{x}_A(n)$ be the time-alignment towards the look direction of $\mathbf{x}(n) = [x_1(n), \ldots, x_M(n)]^T$, normally $\mathbf{x}_A(n)$ is implemented by fraction delays of the measurement signals. Fig. A.2 plots the overview of GSC.

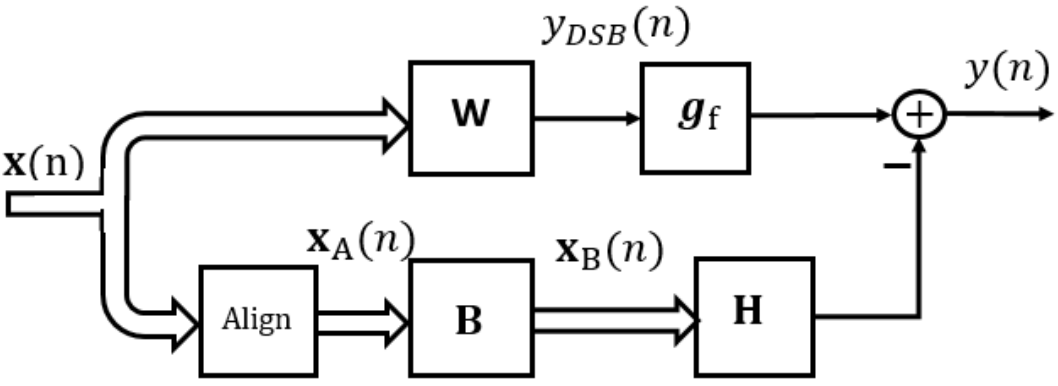


FIGURE A.2: Schematic description of the decomposition of the optimal weight vector into two orthogonal parts.

In the lower path, the SOI portion $s_m(n)$ $m = 1, \ldots, M$ is aligned and suppressed via a blocking matrix

$$\mathbf{x}_B(n) = \mathbf{B}\mathbf{x}_A(n)$$

where $\mathbf{x}_B(n)$ size of $M - 1$ is the vector output of the blocking matrix, this signal vector should exclude the SOI. To do so, the blocking matrix $\mathbf{B}$ has to fulfill the following properties [GJ82; BS01]:

- The size of the matrix is $(M-1) \times M$
- The sum of all values in one row is zero
- The matrix has to be of rank $M-1$.

An example of blocking matrix for the case $M = 4$ is

$$\mathbf{B} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}. \tag{A.1}$$

The vector $\mathbf{x}_B$ is processed with adaptive FIR filters $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_{M-1}]^T$ and then be subtracted from the upper path's signal (e.g., the output of DSB) to get the noise-reduced signal, that is

$$y(n) = \sum_{i=0}^{L-1} y_{DSB}(n-i)\mathbf{g}_f[i] - \sum_{i=1}^{M-1}\sum_{j=0}^{L-1} \{\mathbf{x}_B(n-j)\}[i]\mathbf{h}_i[j] \tag{A.2}$$

where $[.]$ indicates an element in a vector/matrix, $y_{DSB}(.)$ is the output of the DSB, $\mathbf{g}_f$ is a fixed FIR filter which ensures a specified gain and phase response for the output signal, L is the length of FIR filters.

The early paper of GSC [GJ82] used an iterative procedure to adaptive updating $\mathbf{h}_i, \forall i = 1, \ldots, (M-1)$ in the least-mean-square sense

$$\{\mathbf{h}_i[k])\}_{n+1} = \{\mathbf{h}_i[k]\}_n + \mu_0 y(n)\{\mathbf{x}_B(n-k)\}[i],\ k = 1, \ldots, L \tag{A.3}$$

where μ_0 is the normalized step size computed from a small factor β_0 by

$$\mu_0 = \frac{\beta_0}{\sum_{i=1}^{M-1}\sum_{j=0}^{L-1}(\{\mathbf{x}_B(n-j)\}[i])^2}. \tag{A.4}$$

In summary, (A.2), (A.3) and (A.4) are the procedure of the GSC algorithm. The beamforming in the upper path and the blocking matrix are selected flexible. However, how the selections affect the performance of GSC is analytically complicated.

From beam pattern standpoint, we observe that one output of the blocking matrix $\mathbf{x}_B[m],\ m = 0, \ldots, M-2$ is equivalent to the output of differential beamforming, that is delayed and subtracted the signals. In case of a uniform linear array (ULA), its output contains $M-1$ almost identical first-order Cardioid beam patterns having null at look direction.

As example for the ULA, at rotation frequency ω, the mth output element of the blocking matrix in (A.1) to the signal at an incident direction θ is given by

$$\begin{aligned} b_m(\omega,\theta) &= e^{-j\omega\frac{md_H\cos\theta}{c}}e^{-jT_m} - e^{-j\omega\frac{(m+1)d_H\cos\theta}{c}}e^{-jT_{m+1}} \\ &= e^{-j\omega\frac{md_H\cos\theta}{c}}e^{j\omega\frac{md_H}{c}} - e^{-j\omega\frac{(m+1)d_H\cos\theta}{c}}e^{j\omega\frac{(m+1)d_H}{c}} \\ &= e^{-j\omega md_H\frac{\cos\theta-1}{c}}(1 - e^{-j\omega d_H\frac{\cos\theta-1}{c}}) \end{aligned}$$

where T_m and T_{m+1} are the time delays at the sensor m and m+1, respectively, d_H is the inter-distance of sensors, c is the wave speed and j is the imaginary unit. Then,

$$|b_m(\omega,\theta)| = |1 - e^{-j\omega d_H\frac{\cos\theta-1}{c}}|$$

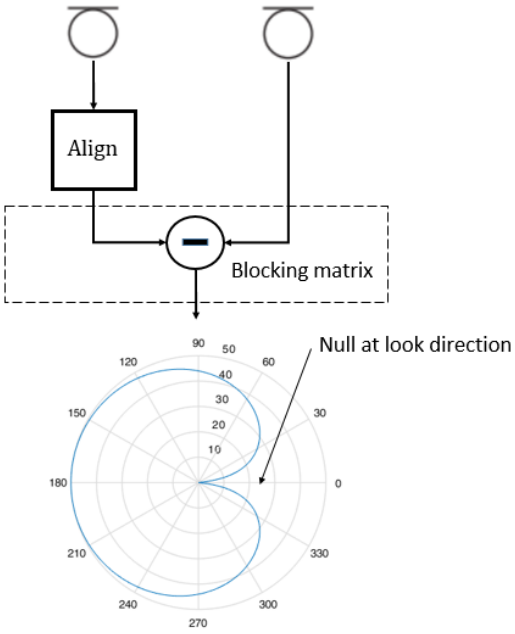


FIGURE A.3: An output of the blocking matrix.

is the first order Cardioid beam pattern with null at $\theta = 0^o$ if $d_H << \pi c/\omega$ [GB12], as illustrated in Fig. A.3.

In the upper path, if a DSB is selected, then the gain of the beam pattern obtains the maximum value at the look direction and less than that for other directions, e.g. Fig. A.4 plots the DSB beam pattern of ULA (endfire) with nine sensors (the inter-distance is 2.5 cm) at different frequencies.

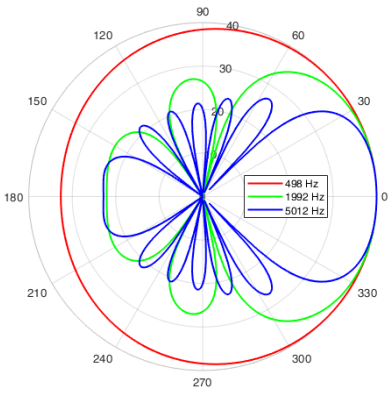


FIGURE A.4: Example for the DSB's beam patterns of the ULA at different frequencies.

It is clearly seen that the sidelobes of the upper path beam patterns are much different with the first-order Cardiod beam patterns in the lower path and the difference is varied over frequency. Therefore, the GSC needs to use adaptive filters that act as gain and phase controls for the first-order Cardiod beam patterns before subtracting them from upper path beam pattern. Also, the GSC uses $M-1$ almost

identical beam patterns in the lower path, that is somewhat redundant, because scaling and summing many identical beam patterns are equivalent to scaling a single beam pattern. That motivates us to modify the GSC in this study.

A.3 Alternative Approaches to Generalized Sidelobe Canceler Beamformer

The problem of subtracting different beam patterns in the GSC could be mitigated if we can design the lower path beamformer whose beam pattern is almost identical to the sidelobes of upper path beam pattern. That implementation is indeed feasible via the SD with multiple constraints where the nulls position can be inserted arbitrarily.

At a narrow band in STFT, let $\mathbf{x}(\omega) = [x_1(\omega), \ldots, x_M(\omega)]^T$ be a measurement vector of the array at the rotation frequency ω , then the array response with beamforming is

$$y(\omega) = \mathbf{w}(\omega)^H \mathbf{x}(\omega) \tag{A.5}$$

where $\mathbf{w}(\omega) = [w_1(\omega), \ldots, w_M(\omega)]$ is the weight vector of the beamformer. The objective of SD with multiple constraints is to minimize the noise energy with subject to distortionless at the look direction and nulls at certain directions. Assume we want to design a beam pattern having N nulls at $\theta_1, \ldots, \theta_N$ and the distortionless at θ_0, then the optimization problem is given as

$$\begin{aligned} &\underset{\mathbf{w}(\omega)}{\text{minimize}} \quad \mathbf{w}(\omega)^H (\mathbf{\Gamma}(\omega) + \mu \mathbf{I}) \mathbf{w}(\omega) \\ &\text{subject to} \\ &\quad \mathbf{w}(\omega)^H \mathbf{D}(\omega) = \mathbf{i}^T \end{aligned} \tag{A.6}$$

where $\mathbf{D}(\omega) = [\mathbf{d}(\omega, \theta_0), \mathbf{d}(\omega, \theta_1), \ldots, \mathbf{d}(\omega, \theta_N)]$ is the matrix size of $M \times (N+1)$ containing $N+1$ steering vectors, $\mathbf{\Gamma}(\omega)$, $\mathbf{I}$ are the correlation matrix of diffuse noise and white noise, respectively, μ is a number deciding the noise model and $\mathbf{i} = [1, 0, \ldots, 0]^T$ is a vector size of $N+1$.

Using the Lagrange multiplier method, the solution of SD with multiple constraints is given by [VT04]

$$\mathbf{w}(\omega) = \frac{(\mathbf{\Gamma}(\omega) + \mu \mathbf{I})^{-1} \mathbf{D}(\omega)}{\mathbf{D}(\omega)^H (\mathbf{\Gamma}(\omega) + \mu \mathbf{I})^{-1} \mathbf{D}(\omega)} \mathbf{i}. \tag{A.7}$$

For conciseness, we omit ω in the formulas in the remainder of this section whenever possible.

To manipulate the shape of beam pattern, instead of using the DSB in the upper path, we also use a SD beamfomer in the upper path where the look direction is assured by the distortionless constraint, the sidelobe's shape and beamwidth are designed by nulls' positions. For instance, the different phase response of the beam patterns at the lower and upper paths are almost equal over the spatial directions except the mainlobe region of upper path beam pattern, we could apply a fixed filter to align the phase and scale the amplitudes before subtracting them. That enables us to use fixed beamforming as an alternative approach to GSC. On the other hand, the different phase response of the beam patterns at the lower and upper paths is much varying over the spatial directions or we want to use more beamfomers at the lower path, we need to use the adaptive filter at the lower path, that is considered as a variant of GSC.

A.3.1 Fixed Beamforming with Suppressed Sidelobes

At a narrow band in STFT, we design two beamformers, one is to construct the mainlobe with expected beamwidth (upper path beam pattern) and the other is to replicate the sidelobes of the upper path beam pattern (lower path beam pattern). Then, we subtract the scale of lower path's signal from the upper path's signal to achieve a beam pattern with suprressed sidelobes. Fig. A.5 plots the block diagram of the fixed beamforming with suppressed sidelobes (FBSS).

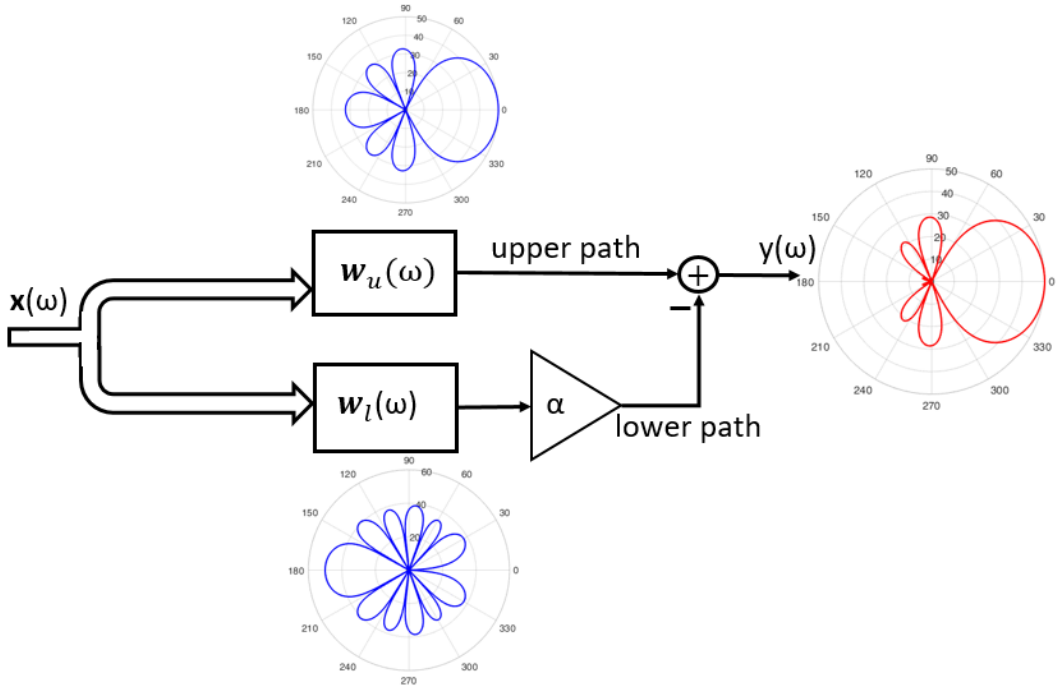


FIGURE A.5: Fixed beamforming with suppressed sidelobes.

On the upper path, the beamformer is designed by

$$\mathbf{w}_u = \frac{(\mathbf{\Gamma} + \mu\mathbf{I})^{-1}\mathbf{D}_u}{\mathbf{D}_u^H(\mathbf{\Gamma} + \mu\mathbf{I})^{-1}\mathbf{D}_u}\mathbf{i}_u \tag{A.8}$$

where $\mathbf{D}_u = [\mathbf{d}(\theta_0), \mathbf{d}(\theta_1), \ldots, \mathbf{d}(\theta_N)]$ is the matrix size of $M \times (N+1)$ containing $N+1$ steering vectors, and $\mathbf{i}_u = [1, 0, \ldots, 0]^T$ is a vector size of $N+1$.

On the lower path, the beamformer is designed by

$$\mathbf{w}_l = \frac{(\mathbf{\Gamma} + \mu\mathbf{I})^{-1}\mathbf{D}_l}{\mathbf{D}_l^H(\mathbf{\Gamma} + \mu\mathbf{I})^{-1}\mathbf{D}_l}\mathbf{i}_l \tag{A.9}$$

where $\mathbf{D}_l = [\mathbf{d}(\theta_m), \mathbf{d}(\theta_0), \mathbf{d}(\theta_1), \ldots, \mathbf{d}(\theta_N), \ldots, \mathbf{d}(\theta_{N+K})]$ is the matrix size of $M \times (N+K+2)$ containing $N+K+2$ steering vectors, θ_m is any direction that makes $\mathbf{w}_l \neq 0$ (e.g. θ_m is at the peak of sidelobes of upper path beam pattern), $\theta_{N+1}, \ldots, \theta_{N+K}$ are the extra null directions in the upper path beam pattern added by spatial aliasing effect when changing the frequency and $\mathbf{i}_l = [1, 0, 0, \ldots, 0]^T$ is a vector size of $N+K+2$.

The array response of the upper path beamforming at direction θ_m is

$$\alpha = \mathbf{w}_u^H \mathbf{d}(\theta_m) = \mathbf{i}_u^T \frac{\mathbf{D}_u^H (\mathbf{\Gamma} + \mu \mathbf{I})^{-1} \mathbf{d}(\theta_m)}{\mathbf{D}_u^H (\mathbf{\Gamma} + \mu \mathbf{I})^{-1} \mathbf{D}_u}. \tag{A.10}$$

Beamforming with suppressed sidelobes is designed by

$$\mathbf{w} = \mathbf{w}_u - \alpha \mathbf{w}_l. \tag{A.11}$$

Consider the class of arrays which are symmetric to the reference point, then the array response at the lower path and upper contain the real value only. For example, a ULA has a reference sensor at the center of array (M is an odd number), the steering vector is defined as

$$\mathbf{d}(\theta) = [e^{\frac{j\omega(M-1)d_H \cos\theta}{c}}, \ldots, e^{\frac{-j\omega(M-1)d_H \cos\theta}{c}}]^T = [\mathbf{a}^T, 1, \mathbf{a}^H]^T.$$

Similarly, we could present $\mathbf{D}_u$ as two conjunction parts

$$\mathbf{D}_u = [\mathbf{A}_u^T, \mathbf{1}, \mathbf{A}_u^H]^T$$

where $\mathbf{1}$ is the vector size of $N+1$ containing one for all element. The array response at the upper path,

$$\mathcal{B}_u(\theta) = \mathbf{w}_u^H \mathbf{d}(\theta) = \mathbf{i}_u^T \frac{\mathbf{D}_u^H (\mathbf{\Gamma} + \mu \mathbf{I})^{-1} \mathbf{d}(\theta)}{\mathbf{D}_u^H (\mathbf{\Gamma} + \mu \mathbf{I})^{-1} \mathbf{D}_u}$$

or

$$\mathcal{B}_u(\theta) = \mathbf{i}_u^T \frac{[\mathbf{A}_u^H, \mathbf{1}, \mathbf{A}_u^T](\mathbf{\Gamma} + \mu \mathbf{I})^{-1} [\mathbf{a}^T, 1, \mathbf{a}^H]^T}{[\mathbf{A}_u^H, \mathbf{1}, \mathbf{A}_u^T](\mathbf{\Gamma} + \mu \mathbf{I})^{-1} [\mathbf{A}_u^T, \mathbf{1}, \mathbf{A}_u^H]^T} \tag{A.12}$$

is a real number (see Appendix A.6: Phase response of beamforming).

We can obtain the similar result for the lower path beamforming. Then, this kind of array could be applied with the FBSS. From (A.11), it is clear that the FBSS is a beamforming that can be used to design a broadband beamforming where the mainlobe is maintained over frequency, while the sibelobes are suppressed as much as possible but may vary over frequency.

In a nutshell, we use (A.8), (A.9), (A.10) and (A.11) to design fixed beamforming for a symmetric array layout. The numerical simulation section will present some examples of this approach.

A.3.2 Adaptive Beamforming with Suppresed Sidelobe

Analogous to the GSC, in this section, the proposed method is presented in the time-domain.

In case of spatial phase response of beamforming is complicated (vary over direction), we need a robust approach that can update adaptively the phase and gain of lower path beamforming before subtracting it from the upper path beamforming, which is called an adaptive beamforming with suppressed sidelobes (ABSS). Depending on the interference directions, at a narrow band, the appropriate gain and phase shift for the lower path beamforming is updated via an adaptive FIR filter.

In Fig. A.6, beamforming of the lower path and upper path is designed similarly to the FBSS, except that the coefficients are transformed to the time-domain as a set of fixed spatial filters, i.e., $\mathbf{W}_l, \mathbf{W}_u$ are spatial filters' coefficients in time-domain for the lower path and upper path, respectively.

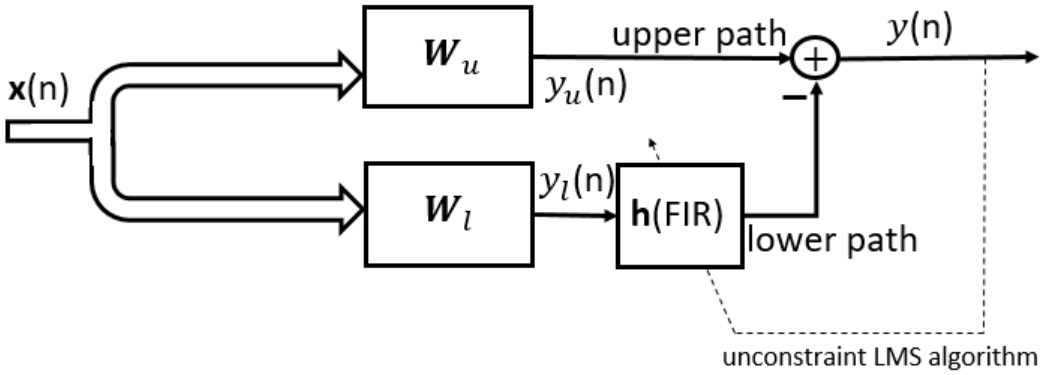


FIGURE A.6: Adaptive beamforming with suppressed sidelobes.

Beamforming at lower path is performed in the time-domain by spatial filtering

$$y_l(n) = \sum_{i=0}^{M-1} \sum_{j=0}^{L_1-1} \{\mathbf{x}(n-j)\}[i]\mathbf{W}_l[i,j] \tag{A.13}$$

where L_1 is the length of the spatial fitter of the lower path, $\mathbf{W}_l$ has a dimension of $M \times L_1$, computed from (A.9) for the frequencies of interest.

Beamforming at the upper path is performed in the time-domain by spatial filtering

$$y_u(n) = \sum_{i=0}^{M-1} \sum_{j=0}^{L_2-1} \{\mathbf{x}(n-j)\}[i]\mathbf{W}_u[i,j] \tag{A.14}$$

where L_2 is the length of the spatial fitter of the lower path (normally we set $L_1 = L_2$), $\mathbf{W}_u$ has a dimension of $M \times L_2$, computed from (A.8) for the frequencies of interest.

The final output of ABSS is given as

$$y(n) = y_u(n) - \sum_{i=0}^{L-1} y_l(n-i)\mathbf{h}[i] \tag{A.15}$$

where L is the length of the adaptive FIR filter $\mathbf{h}$.

The objective function is to minimize the energy of the output signal, that is the unconstrained optimization given by

$$E(\mathbf{h}) := \underset{\mathbf{h}}{\text{minimize}} \quad y(n)^2 = (y_u(n) - \sum_{i=0}^{L-1} y_l(n-i)\mathbf{h}[i])^2. \tag{A.16}$$

That is a quadratic function, the gradient is given explicitly by

$$\mathbf{g} = -2y(n)\mathbf{b}$$

where $\mathbf{b} = [y_l(n), \ldots, y_l(n-L+1)]^T$. The gradient descent method can be applied to update $\mathbf{h}$ iteratively,

$$\{\mathbf{h}\}_{n+1} = \{\mathbf{h}\}_n - \alpha_n\{\mathbf{g}\}_n$$

where α_n is the step size and $\{.\}_n$ is the vector/matrix at nth iteration. For very small values of α_n, the correction of $\{\mathbf{h}\}_n$ is small and the movement down the quadratic surface is slow, and as α_n is increased, the rate of descent increases. However, there

is an upper limit on how large the step size may be. For values of α_n that exceed this limit, the trajectory of $\{\mathbf{h}\}_n$ becomes unstable and unbounded [Hay09].

In Fig. A.6, we adopt the normalized least-mean-square (LMS) algorithm for the adaptive FIR filter, that is

$$\{\mathbf{h}\}_{n+1} = \{\mathbf{h}\}_n + \frac{\beta}{\mathbf{b}^T\mathbf{b}} y(n)\mathbf{b}, \tag{A.17}$$

where β is the normalized step size with $0 < \beta < 1$. With the normalization of the step size by $\mathbf{b}^T\mathbf{b}$, that avoids the gradient noise amplification when $\mathbf{b}$ is large [Hay09]. The computation in (A.17) has the complexity of $\mathcal{O}(2L)$, then the total computational complexity of the proposed method is $\mathcal{O}(ML_1 + ML_2 + 3L)$ which is less than that of the GSC, $\mathcal{O}(ML_1 + ML_2 + 3ML)$.

From the beam pattern subtracting pointview, for the instantaneous interferences, the gain and phase in the lower path beam pattern are updated accordingly to synchronize to the gain and phase of critical sidelobes of the upper path beam pattern. Updating is performed via the sense of minimizing the output signal's energy. Analogous to the GSC, this unconstrained LMS algorithm is the simplest form compared to other adaptive beamformings, e.g. Frost beamforming [For72], LCMV or minimum variance distortionless response. Compared to the GSC, the computational complexity of this method is reduced, since only one adaptive FIR filter is used instead of $M-1$ adaptive FIR filters. Moreover, the proposed method aims to avoid the drawbacks of the GSC, then its performance is superior the GSC that will be illustrated in the simulation section.

A.4 Numerical Simulation

A.4.1 Fixed Beamforming with Suppressed Sidelobe

The white noise gain (WNG) shows the ability of the array to suppress the incoherence noise, such as self-noise, array imperfection, etc. That is given by [BS01],

$$\mathcal{W}\left(\mathbf{w}(\omega)\right) = \frac{\left|\mathbf{w}^H(\omega)\,\mathbf{d}(\omega,\theta_0)\right|^2}{\mathbf{w}^H(\omega)\,\mathbf{w}(\omega)} \tag{A.18}$$

where $\mathbf{w}(\omega)$ is the weight vector of beamforming, $\mathbf{d}(\omega,\theta_0)$ is the steering vector of the look direction.

Another index used to evaluate the beam pattern is the directivity factor (DF), it measures the ability to preserve the source of interest while suppressing the signal coming from other directions [BS01],

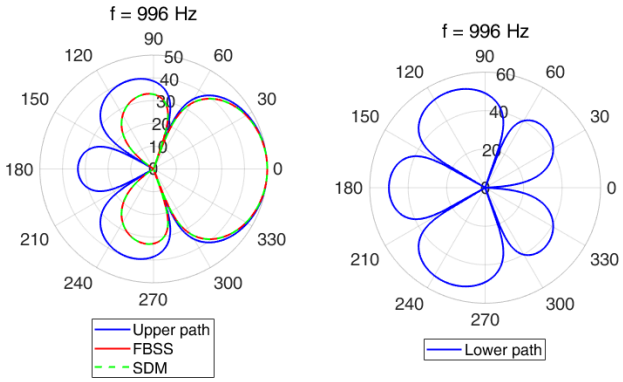
$$\mathcal{D}(\mathbf{w}(\omega)) = \frac{\left|\mathbf{w}(\omega)^H\mathbf{d}(\omega,\theta_0)\right|^2}{\mathbf{w}(\omega)^H\mathbf{\Gamma}(\omega)\mathbf{w}(\omega)^H} \tag{A.19}$$

where $\mathbf{\Gamma}(\omega)$ is the pseudo-coherence matrix of the diffuse noise field,

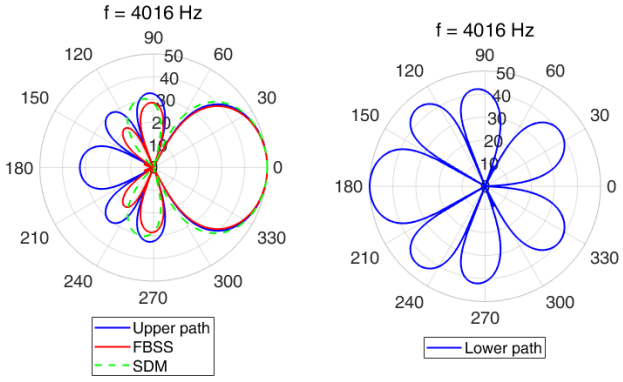
$$\mathbf{\Gamma}\left(\omega\right)[i,j] = \operatorname{sinc}\left(\frac{\omega d_{ij}}{c}\right)$$

where d_{ij} is the distance between sensor i and sensor j, c is the wave speed.

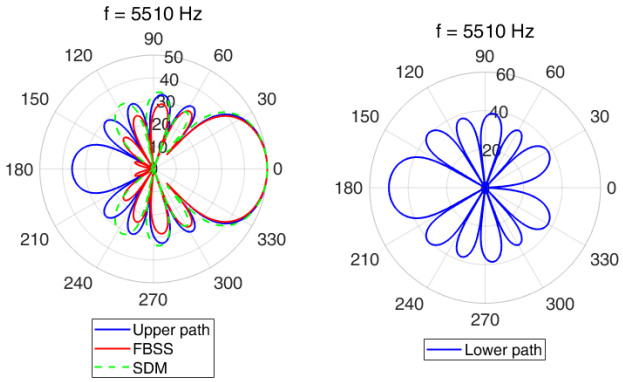
We simulated broadband beamforming for a uniform linear array of $M = 7$ microphones with the inter-distance $d_H = 0.02$ m, the reference microphone is in the



(A) 1 kHz.



(B) 4 kHz.



(C) 5.5 kHz

FIGURE A.7: Beam patterns at some frequencies: (left) upper path beam pattern, FBSS and SDM, (right) lower path beam pattern.

middle of the array. The upper path beam pattern has a look direction at 0^o and fixed nulls at 70^o, 150^o. Then, the lower path beam pattern has fixed nulls at 0^o, 70^o and 150^o, and we set $\theta_m = 180^o$. Due to spatial aliasing at high frequencies, there are some extra nulls at the upper path beam pattern, we need to insert these nulls to the lower path beam pattern as well. The noise model parameter μ in (A.7) is set 0.1 for all simulations.

In comparison, we also design superdirective beamforming with multiple constraints (SDM) with look direction at 0^o and fixed nulls at 70^o, 150^o and 180^o. In

Fig. A.7, we plotted the beam patterns at 1 kHz, 4 kHz and 5.5 kHz, it is clear that the sidelobes of the upper path beam pattern are suppressed for all frequencies, especially the sidelobe that contains $\theta_m = 180^o$ is almost completely suppressed. In comparison with the superdirective beamforming with multiple constraints, both always assure the nulls at 70^o, 150^o and 180^o, but the FBSS has smaller sidelobe regions at high frequencies, e.g., 4 kHz and 5.5 kHz in Figs A.7b and A.7c, respectively. Note that the number of nulls at Figs. A.7a, A.7b and A.7c are different, that causes the spatial aliasing effect.

Fig. A.8, Fig. A.9 and Fig. A.10 plot the broad beam pattern from 1 kHz to 6 kHz for the upper path beam pattern, lower path beam pattern and FBSS's beam pattern, respectively.

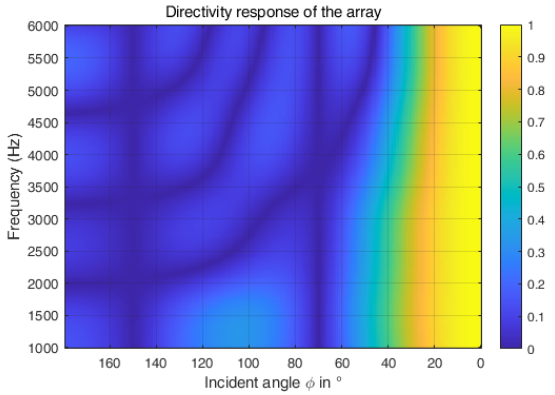


FIGURE A.8: The broadband beam pattern of the upper path.

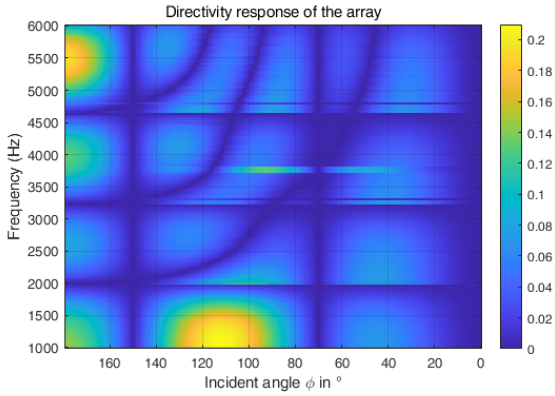


FIGURE A.9: The broadband beam pattern of the lower path.

The WNG and DF of upper path beamforming, FBSS and SDM are plotted in Fig. A.11. It is clearly seen that, the FBSS improves the DF compared to others, while the WNG is still in the acceptable range. Although the results of this simulation do not surprise us, i.e. a design with the regularization of the WNG and DF, the target of FBSS is frequency-invariant mainlobe and minimizing the sidelobes' region. This objective function is different with other approaches in the literature. Moreover, the proposed method enables to use multiple sensor arrays to form a single desired beam's shape, e.g. the upper path is an endfire array aims to build the mainlobe and the lower path is another linear array aims to reproduce the sidelobes, in such a case, combining the beamforming of two arrays can be performed via an FSBB.

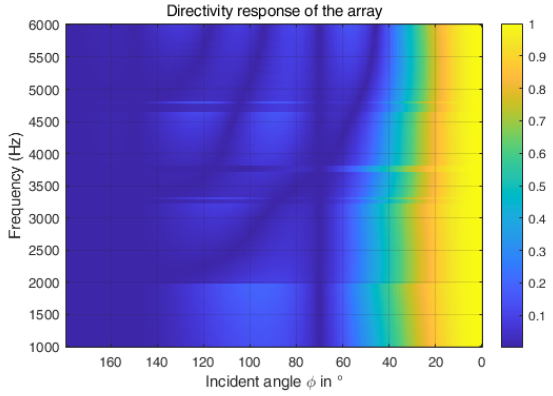
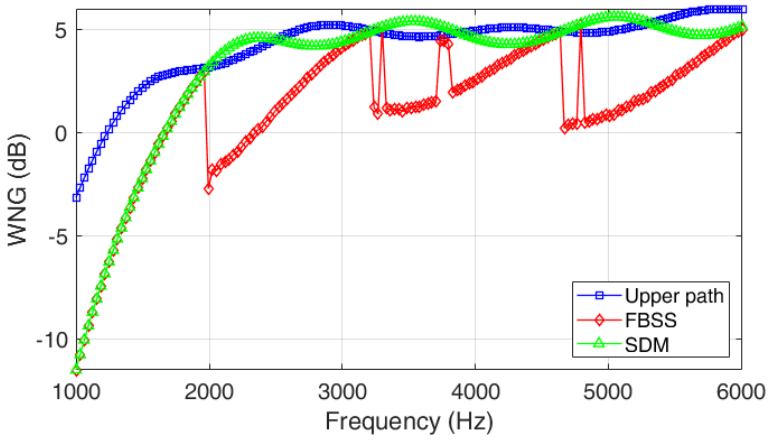
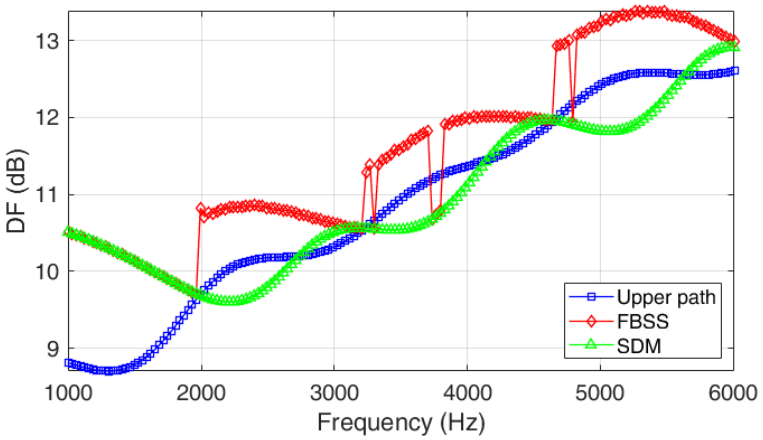


FIGURE A.10: The broadband beam pattern of the FBSS.



(A) WNG.



(B) DF.

FIGURE A.11: WNGs and DFs over frequency.

A.4.2 Adaptive Beamforming with Suppressed Sidelobe

The signal-to-noise ratio (SNR) is defined as

$$\mathrm{SNR} = 10 \log_{10} \frac{\sigma_s^2}{\sigma_n^2} \tag{A.20}$$

where σ_s^2 and σ_n^2 are the variances of SOI and additive noise, respectively.

There are two interferences impinging on the array that have the variance σ_1^2, σ_2^2, then the signal-to-interference ratio (SIR) is defined as

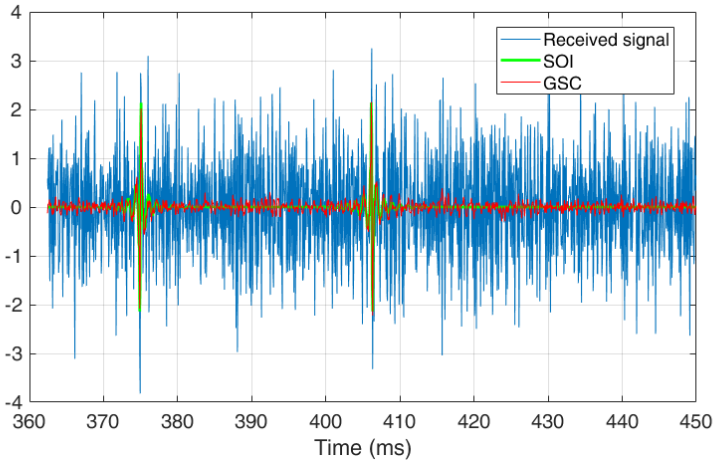
$$\text{SIR} = 10 \log_{10} \frac{\sigma_s^2}{\sigma_1^2 + \sigma_2^2}. \tag{A.21}$$

To evaluate the performance of adaptive beamforming, we measure the signal's error (SE) between the signal of interest (SOI) and beamforming output, that is

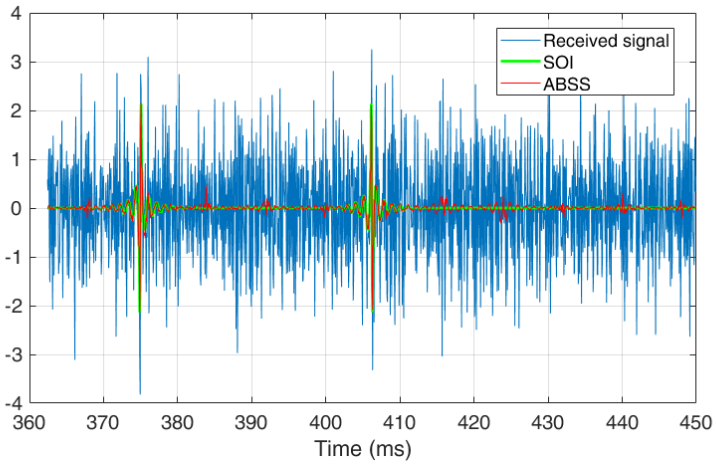
$$\text{SE} = E[(y(n) - s(n))^2] \tag{A.22}$$

where $E[.]$ is the expectation value, $y(n)$ is the beamforming signal and $s(n)$ is the SOI.

In the first simulation, we set SNR = 40 dB and $\sigma_1^2 = \sigma_2^2 = \sigma_s^2$, that is SIR = -3 dB. The SOI at 0^o contains two impulse response signals, as shown in Fig. A.13.



(A) GSC.



(B) ABSS.

FIGURE A.12: Comparison between the GSC and ABSS.

We set $L = 35$ for both ABSS and GSC. For beamformings inside ABSS, we use similar beamformings designed in Section A.4.1 for a frequency range from 1 kHz to 6 kHz and applied an inverse discrete Fourier transfrom to obtain $\mathbf{W}_l, \mathbf{W}_u$ with

$L_1 = L_2 = 128$, β is set 0.01. The waveform of beamformings, measurement signal and SOI are plotted in Fig. A.12. Both methods are able to suppress noise, while preserving the SOI in the output beamformings.

More specifically, the comparison of the ABSS and GSC from 362.5 ms to 427.5 ms is plotted in Fig. A.12 where the SEs of GSC and ABSS are 0.0064 and 0.0039, respectively.

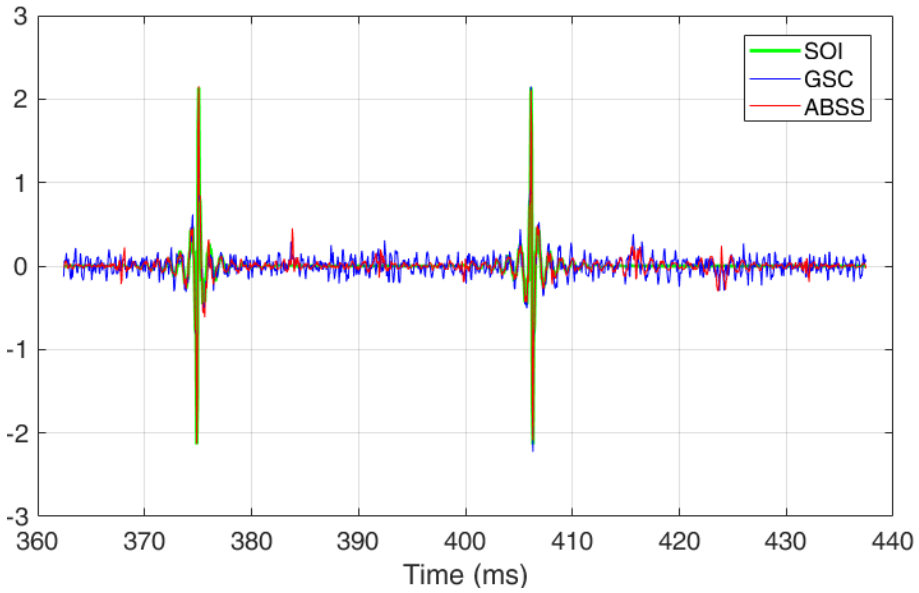


FIGURE A.13: Comparison between the GSC and ABSS from 362.5 ms to 427.5 ms.

Next, the SOI is simulated by summing a sinuous signal and two impulse responses and the SIR is changed from -30 dB to 10 dB, the SNR is set to 10 dB or 20 dB. We measure the SEs over SIR for the GSC, FBSS and ABSS with 25 Monte Carlo trials. Fig.A.14 shows that the ABSS outperforms the GSC in terms of preserving the SOI and suppressing the interferences. It is worth noting that the computational complexity of ABSS is less than that of GSC.

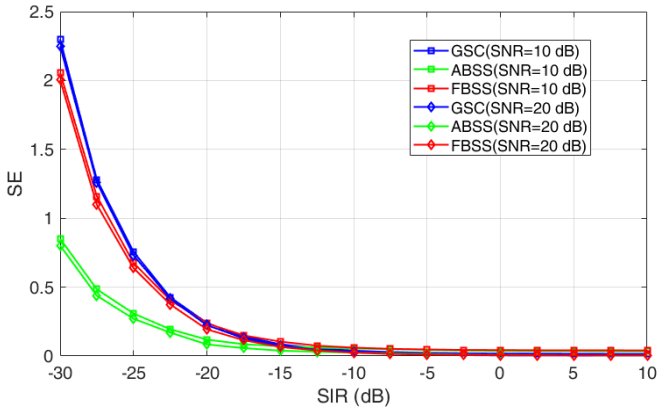


FIGURE A.14: SEs over SIR for the ABSS, FBSS and GSC, two interferences at 90^o and 150^o (a null position).

Another example, the interferences come from 90^o and 180^o (the directions are different with nulls' direction of the designed beam patterns at 70^o and 150^o), the SEs over SIR are plotted in Fig. A.15.

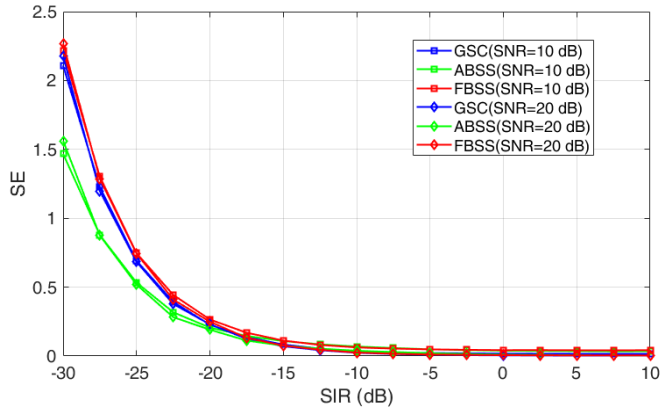


FIGURE A.15: SEs over SIR for the ABSS, FBSS and GSC, two interferences at 90^o and 180^o.

A.5 Conclusions

In this chapter, we have proposed two alternative approaches to the generalized sidelobe canceler beamformer: One is for fixed beamforming, another is for adaptive beamforming. The proposed fixed beamforming applies for a symmetric array layout and the reference point must be at the center of the array. With that constraint, the difference of phase response between the lower path beamforming and the upper path beamforming is almost constant over the direction, so we can use a fixed scale number to compensate for the difference before subtracting. For the proposed adaptive beamforming, it can apply for an arbitrary array layout. As presented in the simulation section, this proposed method outperforms the generalized sidelobe canceller in terms of interference suppression and preserving the signal of interest. Moreover, the computational complexity of this method is less than that of the generalized sidelobe canceler because the proposed method uses one adaptive FIR filter, while the GSC uses $M-1$ adaptive FIR filters. In comparison to other adaptive designs, the problem of the proposed method is the unconstrained optimization and the beamformer coefficients are updated via the normalized least-mean square algorithm, that is much simpler than the constrained optimization-based methods, such as Frost beamforming, MVDR, LCMV, etc. Besides, the proposed method could extend the lower path to multiple lower paths where each path has a different beam pattern and it may belong to different sensor arrays. That enables the ability to combine multiple arrays where one aims to build the mainlobe, others aim to suppress the sidelobes.

A.6 Appendix: Phase Response of Beamforming

Consider a complex number is defined by

$$b_m = [\mathbf{a}_m^H, 1, \mathbf{a}_m^T]\mathbf{R}[\mathbf{a}^T, 1, \mathbf{a}^H]^T$$

where $[\mathbf{a}_m^H, 1, \mathbf{a}_m^T]$ is the row m of the matrix $\mathbf{D}_u^T$ and $\mathbf{R} = (\mathbf{\Gamma} + \mu\mathbf{I})^{-1}$.

First, we prove for the case $\mu = 0$, then $\mathbf{R} = \mathbf{\Gamma}^{-1}$. Note that $\mathbf{\Gamma}(\omega)$ is the pseudo-coherence matrix of the diffuse noise field

$$\mathbf{\Gamma}\left(\omega\right)[i,j] = \operatorname{sinc}\left(\frac{\omega d_{ij}}{c}\right),$$

and the array is symmetric then $\mathbf{\Gamma}$ has a special structure, .i.e,

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{\Gamma}_1 & \gamma & \mathbf{\Gamma}_2 \\ \gamma & 1 & \gamma \\ \mathbf{\Gamma}_2 & \gamma & \mathbf{\Gamma}_1 \end{bmatrix}.$$

Moreover, $\mathbf{\Gamma}$ is a symmetric matrix, it can be decomposed as

$$\mathbf{\Gamma} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$$

where $\mathbf{V}$ is formed by eigenvectors of $\mathbf{\Gamma}$ and $\mathbf{D}$ is diagonal matrix of eigenvalues. So, we have

$$\mathbf{\Gamma}^{-1} = \mathbf{V}\mathbf{D}^{-1}\mathbf{V}^{-1}.$$

That implies $\mathbf{\Gamma}^{-1}$ has similar structure with $\mathbf{\Gamma}$, then $\mathbf{R}$ can be presented by

$$\mathbf{R} = r\begin{bmatrix} \mathbf{R}_1 & \mathbf{r} & \mathbf{R}_2 \\ \mathbf{r} & 1 & \mathbf{r} \\ \mathbf{R}_2 & \mathbf{r} & \mathbf{R}_1 \end{bmatrix}$$

where r is a real number used to scale the center's value of matrix to one, $\mathbf{r}$ is the real-valued vector and $\mathbf{R}_1, \mathbf{R}_2$ are the real-valued matrix. Thus,

$$\begin{aligned} b_m/r = {} & \mathbf{a}_m^H\mathbf{R}_1\mathbf{a} + \mathbf{a}_m^H\mathbf{r} + \mathbf{a}_m^H\mathbf{R}_2\mathbf{a}^* \\ & + \mathbf{a}^T\mathbf{r} + 1 + \mathbf{a}^H\mathbf{r} \\ & + \mathbf{a}_m^T\mathbf{R}_2\mathbf{a} + \mathbf{a}_m^T\mathbf{r} + \mathbf{a}_m^T\mathbf{R}_1\mathbf{a}^* \\ = {} & 1 + 2\mathrm{Re}\{\mathbf{a}_m^H\mathbf{R}_1\mathbf{a}\} + 2\mathrm{Re}\{\mathbf{a}_m^H\mathbf{R}_2\mathbf{a}^*\} \\ & + 2\mathrm{Re}\{\mathbf{a}_m^T\mathbf{r}\} + 2\mathrm{Re}\{\mathbf{a}^T\mathbf{r}\} \end{aligned}$$

(where $(.)^*$ is the conjugate operator, $\mathrm{Re}\{.\}$ is the real part of a complex number) is a real number. It leads to $[\mathbf{A}_u^H, \mathbf{1}, \mathbf{A}_u^T](\mathbf{\Gamma} + \mu\mathbf{I})^{-1}[\mathbf{a}^T, 1, \mathbf{a}^H]^T$ is the real-valued vector and $[\mathbf{A}_u^H, \mathbf{1}, \mathbf{A}_u^T](\mathbf{\Gamma} + \mu\mathbf{I})^{-1}[\mathbf{A}_u^T, \mathbf{1}, \mathbf{A}_u^H]^T$ is the real-valued matrix. As a result, $\mathcal{B}_u(\theta)$ in (A.12) is a real number.

For the case $\mu \neq 0$, applying Woodbury's formula [Woo50], we have

$$\begin{aligned} \mathbf{R} &= (\mu\mathbf{I} + \mathbf{V}\mathbf{D}\mathbf{V}^{-1})^{-1} \\ &= \frac{1}{\mu}\mathbf{I} - \frac{1}{\mu}\mathbf{V}(\mathbf{D}^{-1} + \frac{1}{\mu}\mathbf{V}^{-1}\mathbf{V})\frac{1}{\mu}\mathbf{V}^{-1} \\ &= \frac{1}{\mu}\mathbf{I} - \frac{1}{\mu^2}(\mathbf{V}\mathbf{D}^{-1}\mathbf{V}^{-1} + \frac{1}{\mu}\mathbf{I}) \\ &= (\frac{1}{\mu} - \frac{1}{\mu^3})\mathbf{I} - \frac{1}{\mu^2}\mathbf{\Gamma}^{-1} \end{aligned}$$

has a similar structure with $\mathbf{\Gamma}^{-1}$, then $\mathcal{B}_u(\theta)$ in (A.12) is also a real number.

Appendix B

DOA Estimation via an Annihilating Filter

An annihilating filter (AF) is a filter that can suppress the signal, it is mainly used in sampling signal reconstruction with a finite rate of innovation. Based on the properties of the AF for a uniform linear array, the direction-of-arrivals (DOAs) are deduced accurately by finding the roots of the AF. In this study, we propose a new method for DOA estimation based on the design of AF for multiple frames. The proposed method is compared to the conventional method where the AF is designed for a single frame. The proposed method can detect more DOAs and be less sensitive to noise. We also compare the proposed method to the MUSIC algorithm, the new approach can detect the DOAs with a higher resolution since its principle does not base on the search grid procedures. Moreover, the AF-based approaches use less computation than the subspace approaches in general. Also, we propose an extension for the MUSIC algorithm that takes diffuse noise into account. The simulation shows that, in the case of diffuse noise, only the extended MUSIC can estimate the DOAs properly.

B.1 Introduction

It is fact that array signal processing (ASP) [KV96; Mai17] has been widely employed in diverse areas such as acoustics [Bra01; BCH08], radio-interferometry [RTMSJ17; Sim15], radar and sonar systems [Mai17; Hay85], wireless networks [God97; PP97; HS16] and medical imagery [LL00; Raf15]. Direction-of-arrival (DOA) estimation is an important topic in ASP [KV96], that is the process of retrieving direction information of electromagnetic/sound sources by using a sensor array. Signals impinge to the array can be coherent or incoherent. The DOA for incoherent signals can be obtained by applying various subspace methods like MUSIC [Sch86] and ESPRIT [RK89]. However, the subspace-based methods are sensitive to coherent signals [LHZ12] that are challenging to separate signal and noise subspaces, leading an incorrect estimation of the spatial spectrum. To deal with coherent audio signals, various preprocessing techniques have proposed to decorrelate signals. In particular, Pillai *et al.* [PK89] suggested two different spatial smoothing techniques: forward spatial smoothing and forward backward spatial smoothing techniques for preprocessing the received signals. Recently, based on the annihilating filter's properties, Vetterli *et al.* [VMB02] proposed the finite rate of innovation concept that reconstructs the signal perfectly from uniform sampling. This reconstruction concept can be directly applied to DOA estimation where the active sources act as a

stream of Dirac (pulse). However, the method is very sensitive to noise, since the directions are deduced from the roots of an annihilating filter (AF) with complex exponential forms. Besides, in order to build a full-rank convolution matrix, the number of active sources must be less than half the number of measurements, then the number of detectable DOAs should be less than half of the number of sensors. To tackle these problems, we propose the design of AF for multiple snapshots that can detect more sources and its output is robust to noise. However, the proposed approach only works for frame-variant source signals where the strength and phase of sources vary over frame. If the sources are frame-invariant, the measurement signals at different frames are highly dependent, the proposed approach is false to estimate the DOAs because in such a case the rank of the computation matrix is less than number of active sources. In addition, the study suggests an extended MUSIC algorithm that considers white noise as well as diffuse noise that is often inevitable in some audio applications. The extended MUSIC is simple but it is useful for many practical applications, i.e. audio applications.

These remaining sections of this chapter are organized as follows. Section B.2 reviews the MUSIC algorithm from which we extend the conventional MUSIC algorithm to a general form considering diffuse noise as well as white noise. Section B.3 presents the AF design for DOA estimation which covers the approaches for a single data frame and multiple data frames. The numerical simulations and performance comparison are then presented in section section B.4. Finally, conclusions are drawn in section B.5.

B.2 MUSIC with Sound Noise Model

Consider the far-field, then the wave reaching the sensors is planar. The signal of i^{th} sensor at a rotation frequency ω is modeled as

$$p_i(\omega) = \sum_{n=1}^{N} a_i(\theta_n, \varphi_n, \omega) s_n(\omega) + n_i(\omega), \forall i = 1, \ldots, M$$

where N is the number of active sound sources, M is the number of sensors, $s_n(\omega)$ presents for the strength and phase of a source signal at direction (θ_n, φ_n), $a_i(\theta_n, \varphi_n, \omega)$ is the transfer function of the wave propagation from i^{th} sensor to a reference sensor and $n_i(\omega)$ is additive noise. For conciseness, the remaining of this chapter ω is omitted whenever possible. For the array, the output could be presented by a measurement vector

$$\begin{bmatrix} p_1 \\ p_2 \\ \cdots \\ p_M \end{bmatrix} = \begin{bmatrix} \mathbf{d}_1 & \mathbf{d}_2 & \cdots & \mathbf{d}_N \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ \cdots \\ s_N \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ \cdots \\ n_M \end{bmatrix} \tag{B.1}$$

where $\mathbf{d}_i$ is the steering vector at i^{th} direction. (B.1) could be presented in a short form

$$\mathbf{p} = \mathbf{A}\mathbf{s} + \mathbf{n}. \tag{B.2}$$

The covariance matrix of the measurement signals is defined as

$$\mathbf{R} = E[\mathbf{p}\mathbf{p}^H]$$

where $E[.]$ is the expectation operation. For noise $\mathbf{n}$, the covariance matrix is given by

$$E[\mathbf{n}\mathbf{n}^H] = \epsilon\mathbf{N} \tag{B.3}$$

where $\mathbf{N}$ is the noise correlation matrix, ϵ is a scale number representing the signal-to-noise ratio (SNR).

Suppose that noise is incoherent to the source signals, then from (B.2) and (B.3) we have

$$\mathbf{R} = \mathbf{A}\mathbf{S}\mathbf{A}^H + \epsilon\mathbf{N} \tag{B.4}$$

where $\mathbf{S}$ is the sound source covariance matrix, rank($\mathbf{S}$) = N, the diagonal elements indicate the power of the sources and the off-diagonal elements indicate the coherence between them. If the source signals are uncorrelated, $\mathbf{S}$ is a diagonal matrix. For audio applications, $\mathbf{N}$ is the combination of white noise and diffuse noise correlation matrices, then we could decompose the $\mathbf{N}$ in (B.4) to obtain a new form

$$\mathbf{R} = \mathbf{A}\mathbf{S}\mathbf{A}^H + \sigma_d^2\mathbf{\Gamma} + \sigma_w^2\mathbf{I} \tag{B.5}$$

where σ_d^2 and σ_w^2 is the noise power of diffuse noise and white noise, respectively, $\mathbf{I}$ is the identity matrix, $\mathbf{\Gamma}$ is the pseudo-coherence matrix of the diffuse noise field. $\mathbf{\Gamma}$ is given by [Elk00]

$$\mathbf{\Gamma}_{ij}(\omega) = \text{sinc}(\frac{\omega d_{ij}}{c})$$

where d_{ij} is the distance between sensor i and sensor j, c is the speed of sound. If $\sigma_d^2 + \sigma_w^2 \neq 0$, (B.5) can be rewritten as

$$\mathbf{R} = \mathbf{A}\mathbf{S}\mathbf{A}^H + (\sigma_d^2 + \sigma_w^2)\frac{\sigma_d^2\mathbf{\Gamma} + \sigma_w^2\mathbf{I}}{\sigma_d^2 + \sigma_w^2}$$

then $\epsilon = \sigma^2 = \sigma_d^2 + \sigma_w^2$ and $\mathbf{N} = \frac{\sigma_d^2\mathbf{\Gamma}+\sigma_w^2\mathbf{I}}{\sigma_d^2+\sigma_w^2} = \frac{\alpha\mathbf{\Gamma}+\mathbf{I}}{\alpha+1}$ where $\alpha = \sigma_d^2/\sigma_w^2$ is a ratio representing for the noise model.

$\mathbf{N}$ is a symmetric matrix, so it is invertible. Thus, (B.5) becomes

$$\mathbf{R}' := \mathbf{R}\mathbf{N}^{-1} = \mathbf{A}\mathbf{S}\mathbf{A}^H\mathbf{N}^{-1} + \sigma^2\mathbf{I} \tag{B.6}$$

Now we decompose $\mathbf{R}'$ to signal subspace and noise subspace by finding the eigenvalues and eigenvectors of $\mathbf{R}'$ and categorize them (based on the amplitude of eigenvalues).

$$\mathbf{R}' = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$$

where $\mathbf{V}$ is formed by the eigenvectors of $\mathbf{R}'$ and $\mathbf{D}$ is diagonal matrix of eigenvalues

$$\mathbf{D} = \begin{bmatrix} \lambda_{1s} + \sigma_1^2 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \lambda_{Ns} + \sigma_K^2 & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \sigma_{N+1}^2 & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & \cdots & \cdots & \sigma_M^2 \end{bmatrix}$$

where $\lambda_{1s}, \ldots, \lambda_{Ns}, 0, \ldots, 0$ are the eigenvalues of $\mathbf{A}\mathbf{S}\mathbf{A}^H\mathbf{N}^{-1}$ (this is rank-deficient matrix if the number of active sources is less than number of sensors). Also, $\mathbf{A}\mathbf{S}\mathbf{A}^H\mathbf{N}^{-1}$ is the semi-positive definite matrix then its eigenvalues are non-negative ($\lambda_1, \ldots, \lambda_N \geq 0$). In theory $\sigma_1^2 = \cdots = \sigma_M^2 = \sigma^2$ but in practice they are a set of smallest values. As

based on the amplitude of eigenvalues, we can separate the eigenvector of $\mathbf{R}'$ into noise subspace vectors $\mathbf{V}_N$ and signal subspace vectors $\mathbf{V}_S$

$$\mathbf{V} = [\mathbf{V}_S, \mathbf{V}_N].$$

Note that $\mathbf{V}$ is an unitary matrix, then noise subspace is orthogonal to signal subspace. We can also explain this property by a simple modification in the equations, that is, taking an column vector $\mathbf{v}_i$ in noise subspace $\mathbf{V}_N$ and multiplying it to both sides of (B.6) we obtain

$$\mathbf{R}'\mathbf{v}_i = (\mathbf{A}\mathbf{S}\mathbf{A}^H\mathbf{N}^{-1} + \sigma^2\mathbf{I})\mathbf{v}_i$$

$$\sigma^2\mathbf{v}_i = \mathbf{A}\mathbf{S}\mathbf{A}^H\mathbf{N}^{-1}\mathbf{v}_i + \sigma^2\mathbf{v}_i$$

$$\mathbf{A}\mathbf{S}\mathbf{A}^H\mathbf{N}^{-1}\mathbf{v}_i = 0$$

$$\mathbf{A}^H(\mathbf{N}^{-1}\mathbf{v}_i) = 0$$

Let us define a vector $\mathbf{v}'_i := \mathbf{N}^{-1}\mathbf{v}_i$ then

$$\mathbf{A}^H\mathbf{v}'_i = 0 \tag{B.7}$$

Or $\mathbf{d}_n^H\mathbf{v}'_i = 0$ for all $\mathbf{d}_n$ corresponding to the active sources. Since (B.7) is true for all column vector of the noise subspace $\mathbf{V}_N$, we have $\mathbf{d}_n^H\mathbf{V}'_N = 0$ where $\mathbf{V}'_N = \mathbf{N}^{-1}\mathbf{V}_N$.

Herein, we suggest the extended MUSIC that takes the diffuse noise into account, that is, the noise model ratio $\alpha = \sigma_d^2/\sigma_w^2$ is considered as an input.

- Perform eigenvalues decomposition on $\mathbf{R}' = \mathbf{R}(\frac{\alpha\mathbf{\Gamma}+\mathbf{I}}{\alpha+1})^{-1}$ to obtain the non-increasing eigenvalue $\lambda_1 \geq \cdots \geq \lambda_N > \lambda_{N+1} \geq \cdots \geq \lambda_M$.
- Based on the amplitude of eigenvalues, group the corresponding eigenvectors into two groups: group of signal subspace $\mathbf{V}_S = [\mathbf{v}_1, \ldots \mathbf{v}_N]$ and group of noise subspace $\mathbf{V}_N = [\mathbf{v}_{N+1}, \ldots \mathbf{v}_M]$.
- Modify the noise subspace to consider the diffuse noise

$$\mathbf{V}'_N = (\frac{\alpha\mathbf{\Gamma}+\mathbf{I}}{\alpha+1})^{-1}\mathbf{V}_N.$$

- Construct the power spectrum function as

$$P(\mathbf{d}_i) = \frac{1}{\mathbf{d}_i^H\mathbf{V}'^H_N\mathbf{V}'_N\mathbf{d}_i}.$$

- Search the peaks of $P(\mathbf{d}_i)$ to detect the active sources.

B.3 DOA Estimation with Annihilating Filter

Consider the uniform linear array (ULA) with the inter-distance d_H, then $\mathbf{d}_i = [1, e^{-j\omega d_H \cos\theta_i/c}, \ldots, e^{-j\omega(M-1)d_H\cos\theta_i/c}]^T$ is the steering vector towards the signal incident angle θ_i, where c is the wave speed. Then, the signal at the i^{th} sensor is

$$p_i(\omega) = \sum_{n=1}^{N} s_n(\omega)e^{-j\omega(i-1)d_H\cos\theta_n/c} + n_i(\omega), \forall i = 1, \ldots, M.$$

Thus,

$$\mathbf{p} = \sum_{n=1}^{N} \mathbf{t}_n s_n + \mathbf{n} \tag{B.8}$$

where $\mathbf{t}_n = \mathbf{d}_n = [a_n^0, a_n^1, \ldots, a_n^{M-1}]^T$, $a_n = e^{-j\omega d_H \cos\theta_n / c}$. If $\mathbf{n} \neq 0$, then $\mathbf{p}$ is the linear combination of N complex exponential vectors $\mathbf{t}_n$. Let us define a filter with the z-transform

$$A(z) = \sum_{m=0}^{N} A[m] z^{-m}$$

having N zeros at $u_n = a_n = e^{-j\omega d_H \cos\theta_n / c}, \forall n = 0, \ldots, N-1$. Then, $A(z)$ can be presented by

$$A(z) = \prod_{n=1}^{N} (1 - a_n z^{-1})$$

Note that $A[m], m = 0, \ldots, N$ is the convolution of N first-order filters with coefficients $[1, -a_n]$. It is easy to observe that the convolution $[1, -a_n] * \mathbf{t}_n^T = \mathbf{0}$. Therefore, the defined filter $A(z)$ suppresses the active directional signals in the measurement signals, that is the so-called annihilating filter [VMB02].

$$\begin{aligned}
&\big[A[0], \ldots, A[N]\big] * \mathbf{p}^T = [1, -a_1] * \cdots * [1, -a_N] \\
&* \Big(\sum_{n=1}^{N} \mathbf{t}_n^T s_n + \mathbf{n}^T\Big) \\
&= \Big(\sum_{n=1}^{N} s_n [1, -a_1] * \cdots * [1, -a_N] * \mathbf{t}_n^T\Big) \\
&\qquad + [1, -a_1] * \cdots * [1, -a_N] * \mathbf{n}^T \\
&= \mathbf{0} + [1, -a_1] * \cdots * [1, -a_N] * \mathbf{n}^T \\
&= \big[A[0], \ldots, A[N]\big] * \mathbf{n}^T.
\end{aligned}$$

By definition of $A(z)$, we know $A[0] = 1$ and $M \geq N + 1$ to complete the convolution. In case of noiseless ($\mathbf{n} = 0$), we have the definition of the AF [VMB02]

$$\big[A[0], \ldots, A[N]\big] * \mathbf{p} = 0. \tag{B.9}$$

Given the measurement signal of the array, a simple way to find the coefficients of the filter $A[m]$ is to build the equations which are deduced from (B.9)

$$\begin{bmatrix} p_1 & p_2 & \cdots & p_{N+1} \\ p_2 & p_3 & \cdots & p_{N+2} \\ \cdots & \cdots & \cdots & \cdots \\ p_N & p_{N+1} & \cdots & p_{2N} \end{bmatrix} \begin{bmatrix} A[N] \\ \cdots \\ A[1] \\ A[0] \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \cdots \\ 0 \end{bmatrix},$$

assign $A[0] = 1$, then

$$\begin{bmatrix} p_1 & p_2 & \cdots & p_N \\ p_2 & p_3 & \cdots & p_{N+1} \\ \cdots & \cdots & \cdots & \cdots \\ p_N & p_{N+1} & \cdots & p_{2N-1} \end{bmatrix} \begin{bmatrix} A[N] \\ \cdots \\ A[2] \\ A[1] \end{bmatrix} = - \begin{bmatrix} p_{N+1} \\ p_{N+2} \\ \cdots \\ p_{2N} \end{bmatrix}. \tag{B.10}$$

This system has a unique solution [VLG83], then we can find a unique set of $A[m]$. After solving (B.10), we find the roots of $A(z) = \sum_{m=0}^{N} A[m] z^{-m}$ to obtain $a_0, \ldots, a_{N-1}$,

then the direction of active sources can be achieved by

$$\theta_n = \arccos \frac{jc \log a_n}{\omega d_H}, \forall n = 0, \dots, N-1. \tag{B.11}$$

Note that, to achieve (B.10), $M \geq 2N$ needs to be satisfied that is the number of sensors is greater than or equal to two times the number of sources and the SRN needs to be very high to assure $\mathbf{n} \approx 0$. Besides, the roots of AF associated with the true DOAs stay on the unit circle, then we can utilize this property to evaluate a_n, that is

$$\text{Re}\{\log a_n\} \leq \beta \tag{B.12}$$

where Re$\{.\}$ is the real part of a complex number, β is a small value (e.g. $\beta = 0.02$). The inequality (B.12) is used to select a reliable a_n, thus we can estimate DOAs without knowing the number of DOAs in advance. In order to deal with different SNR levels, we could decrease or increase β to compromise between the accuracy and the robustness of the algorithm.

In summary, the method in [VMB02] and the constraint in (B.12) can be applied for the DOA estimation of coherence signals but the number of sources is limited and the result is sensitive to noise.

In order to detect more DOAs in noisy environments, we apply a similar idea of AF design for multiple data frames (snapshots). Suppose that the signal of active sources is time-variant, that means the amplitude and phase of the signals are varying over time, then the signals at different snapshots are almost independent. This assumption is reasonable for many applications like audio, radar, etc. Similarly to (B.10), let us build the equations for the AF from K snapshots

$$\begin{bmatrix} \mathbf{p'}_1^T \\ \mathbf{p'}_2^T \\ \dots \\ \mathbf{p'}_K^T \end{bmatrix} \begin{bmatrix} A[M-1] \\ \dots \\ A[2] \\ A[1] \end{bmatrix} = - \begin{bmatrix} p_{M,1} \\ p_{M,2} \\ \dots \\ p_{M,K} \end{bmatrix} \tag{B.13}$$

where $\mathbf{p'}_k, \forall k = 1, \dots, K$ is the measurement signal at snapshot k after removing the last value $p_{M,k}$ (value of the last sensor). Thus, we can find $A[m]$ from least-mean-square error sense, that is

$$\begin{bmatrix} A[M-1] \\ \dots \\ A[2] \\ A[1] \end{bmatrix} = -(\mathbf{P'}^H \mathbf{P'})^{-1} \mathbf{P'}^H \begin{bmatrix} p_{M,1} \\ p_{M,2} \\ \dots \\ p_{M,K} \end{bmatrix}. \tag{B.14}$$

where $\mathbf{P'} = \begin{bmatrix} \mathbf{p'}_1^T \\ \mathbf{p'}_2^T \\ \dots \\ \mathbf{p'}_K^T \end{bmatrix}$.

The solution in (B.14) is robust against noise and it is possible to detect maximum $M-1$ sources. In practice, $(\mathbf{P'}^H \mathbf{P'})^{-1}$ can be updated iteratively over the frame to reduce the complexity of the inverse operation. By applying Woodbury's formula [Woo50], we have

$$(\mathbf{P'}_{k+1}^H \mathbf{P'}_{k+1})^{-1} = (\mathbf{P'}_k^H \mathbf{P'}_k + \mathbf{p'}_{k+1} \mathbf{p'}_{k+1}^H)^{-1}$$

$$= \mathbf{B}_k^{-1} - \mathbf{B}_k^{-1}\mathbf{p'}_{k+1}(\mathbf{I} + \mathbf{p'}_{k+1}^H\mathbf{B}_k^{-1}\mathbf{p'}_{k+1})\mathbf{p'}_k^H\mathbf{B}_k^{-1}$$

where $\mathbf{B}_k = \mathbf{P'}_k^H\mathbf{P'}_k$ is the matrix of $\mathbf{P'}^H\mathbf{P'}$ at the frame k. The computation of $\mathbf{B}_{k+1}^{-1}$ has the complexity of $\mathcal{O}(M^2)$, then (B.14) has the complexity of $\mathcal{O}(KM^2)$. The total complexity of the proposed AF is $\mathcal{O}(KM^2 + M^3)$, including finding the roots of the polynomial by computing the eigenvalue decomposition [Sto+15].

From the coefficients of AF $A[m]$, applying a similar approach to the AF for a single frame, we obtain the direction of sources via (B.11).

B.4 Numerical Simulations

B.4.1 Simulations in White Noise Environment

In this section, a ULA with $M = 11$ sensors and half-wavelength inter-distance of sensors is examined. In Fig. B.1a, five incoherent sources at $\theta_0 = -24^o$, $\theta_1 = -12^o$, $\theta_2 = 0^o$, $\theta_3 = 12^o$ and $\theta_4 = 24^o$ impinge to the array, the SNR is 80 dB and there are 100 snapshots. For such a high SNR, all methods seem work well.

In Fig. B.1b, we reduce the SNR to 40 dB. The performance of AF for multiple frames (AFM) method is compatible with that of MUSIC, they estimate accurately the direction of sources, while the AF for single frame (AFS) does not detect the DOAs properly. In order to estimate the performance, we use the root-mean-squared error (RMSE) criteria defined as

$$E = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\theta_i - \bar{\theta}_i)^2},$$

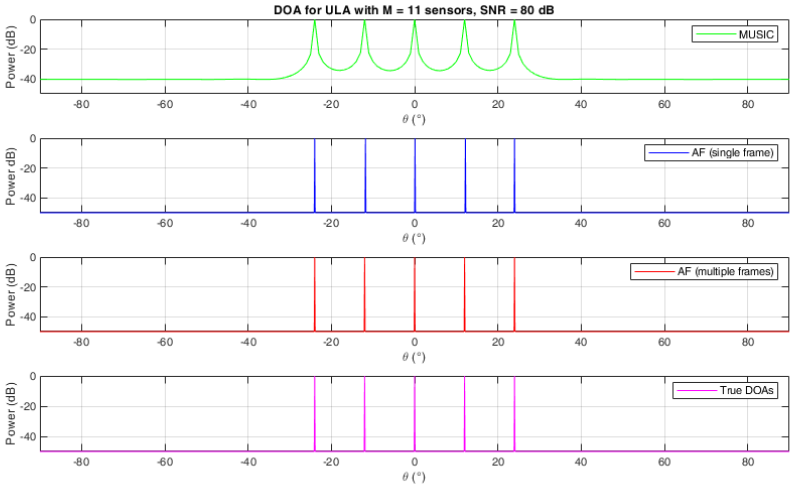
where $\bar{\theta}_i$ denotes estimated DOA of the source i and θ_i is the true DOA. Then, the RMSE of MUSIC, AFS and AFM are 0^o, 3.1^o and 0^o degree, respectively.

Fig. B.2 shows another example, where the number of active sources increases to $N = 10$. In this examination, we only compare between MUSIC and AFM approaches, since the AFS can not apply for $N > \frac{M}{2}$. At the SNR = 20 dB, the RMSEs are 0.23^o and 0.05^o for MUSIC and AFM, respectively. Fig. B.3 plots RMSE over SNR with 1000 Monte Carlo trials, it is clearly seen that the AFM outperforms the MUSIC in terms of the RMSEs.

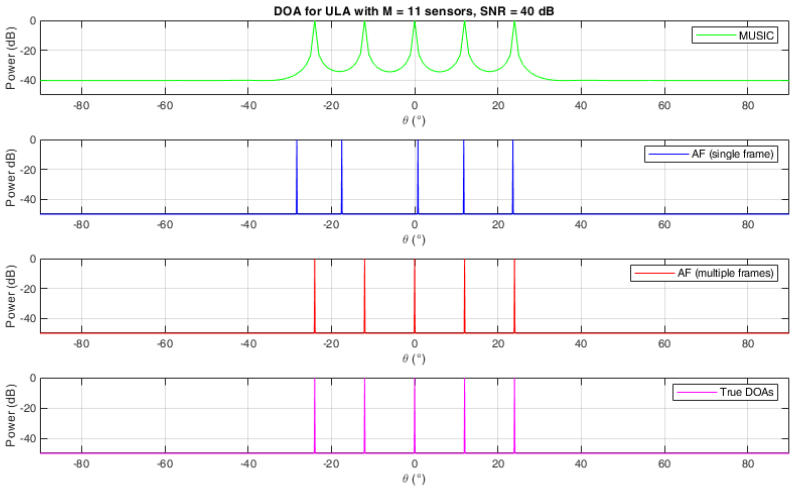
To emphasize the grid-less based property of the AF-based approach, we simulate for the off-grid case of DOAs, that is, the active sources are not placed on the search grids of MUSIC algorithm. Suppose the resolution of search grids of MUSIC is 1^o, three active sources with SNR = 20 dB at -40.5^o, 15.6^o and 20.2^o impinge to the array and the power spectrum are shown in Fig. B.4. For the MUSIC algorithms, three sources at -40^o, 16^o and 20^o are detected, while the AFM detects three sources at -40.5378^o, 15.6486^o and 20.2451^o. Then, the RMSEs of MUSIC and AFM are 0.5^o and $\approx 0^o$, respectively. Therefore, in real scenarios, the RMSE of AFM has a better performance.

B.4.2 Simulations in White Noise and Diffuse Noise Environments

For instance, the noise contains the diffuse noise with $\sigma_d^2/\sigma_w^2 = 25$, only the extended MUSIC for general noise can estimate successfully the direction of active sources, as shown in Fig. B.5. The RMSEs of MUSIC, extended MUSIC for general noise and AFM are 2.1^o, 0.0^o and 28.6^o, respectively. Note that, for this example, the inder-distance of sensors is reduced to less than half of the wavelength to achieve a



(A) SNR = 80 dB, $N = 5$.



(B) SNR = 40 dB, $N = 5$.

FIGURE B.1: Power Spectra of different methods: MUSIC, AFS, AFM and the true DOAs.

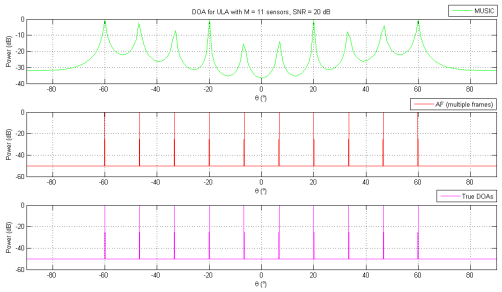


FIGURE B.2: Power Spectra of different methods (SNR = 20 dB, $N = 10$): MUSIC, AFS, AFM and the true DOAs (on-grid).

reasonable diffuse noise correlation matrix (the off-diagonal elements of $\mathbf{\Gamma}$ are not 0).

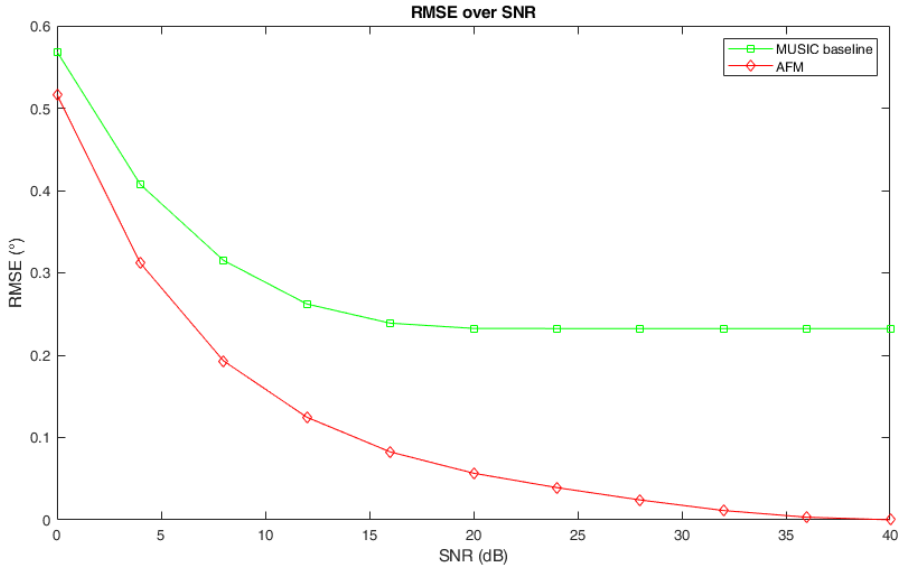


FIGURE B.3: RMSE over SNR of MUSIC and AFM, $N = 10$, 1000 Monte Carlo trials.

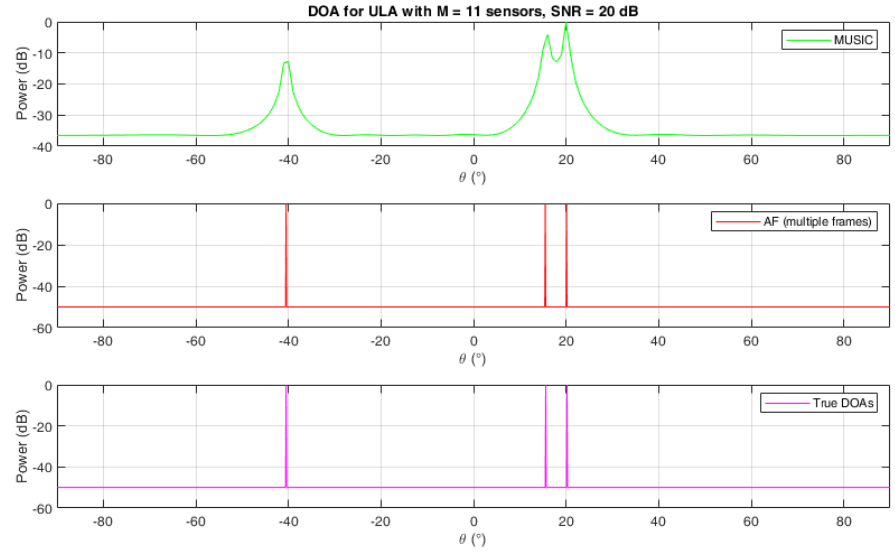


FIGURE B.4: Power Spectra of different methods (SNR = 20 dB, $N = 3$): MUSIC, annihilating filter for single frame, annihilating filter for multiple frames and the true DOAs (off-grid).

B.5 Conclusions

In this study, we have proposed an annihilating filter-based technique for DOA estimation. The proposed method processes on multiple frames under the constrain of frame-variant signals. The maximum number of detectable sources is almost twice

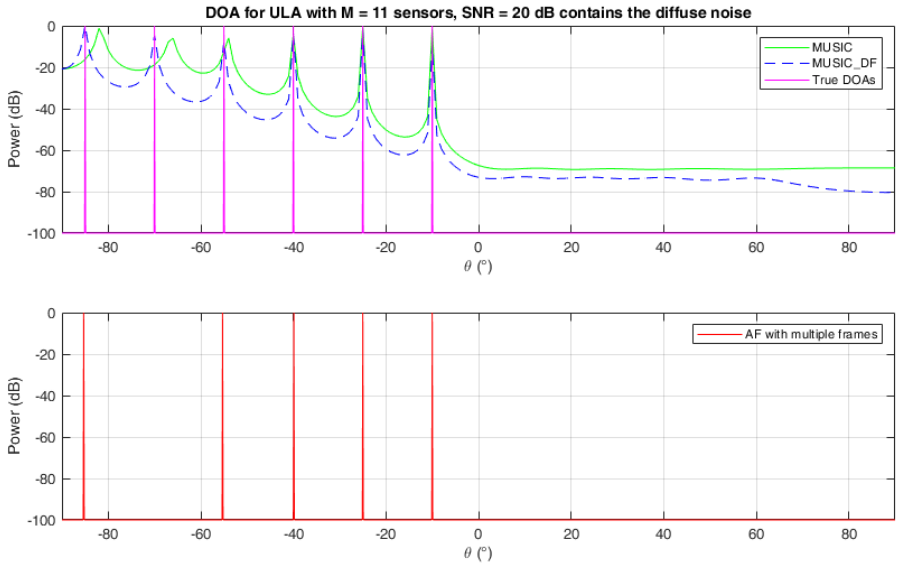


FIGURE B.5: Power Spectra of different methods (SNR = 20 dB contains the white noise and diffuse noise): MUSIC, extended MUSIC for diffuse noise, and the true DOAs are presented in the top figure, the blow is for the AFM.

times of that of annihilating filter for single frame. Besides, the proposed method is independent with the grid directions, then its RMSE's performance outperforms the MUSIC algorithm in terms of accuracy. Moreover, the computational complexity of new method is $\mathcal{O}(KM^2 + M^3)$, which is less than the computational complexity of subspace-based techniques. However, when the diffuse noise presents in the measurement signal, only the extended MUSIC, which is also newly proposed in this study, could estimate the DOA successfully.

Appendix C

Matlab Code: Simulated Annealing for Sensor Array Deployment

```
clear
clc
close all

M = 25;
Mx = floor(sqrt(M));
My = floor(sqrt(M));
Ni = 5;
Np = 100000;
T0 = 1;
e_d = 0.01;

% A random sequence
Seq = (1:M);
[dum,idx] = sort(rand(1,M));
L_r = Seq(idx(:));

C = M*(M-1)/2; % A big number
C_best = C;
Cp=[];
grid_points = zeros(Mx*My,2);
max_n = 1;
for yPos = 1: M
    for xPos = 1:M
        grid_points(xPos + M*(yPos-1), 1) = (xPos);
        grid_points(xPos + M*(yPos-1), 2) = (yPos);
    end
end

mic_pos = zeros(M,2);
np = 0;
Koordinaten=zeros(M,1);
for k = 1:Ni
   T = T0/k;                  % hyperbolic annealing law
   for l = 1:Np
       % create new array geometry by permutation
       L_p = perm_array(L_r);

       iLoop = 0;
       A=reshape(L_p,Mx,My)';
       v=1;
       for n=1:Mx
          for m=1:My
              P = A(m,n);
              nx=rem(P - 1,Mx); % from 0 to M-1
              mx=floor((P -1)/Mx); % from 0 to M-1
```

```
47                mic_pos((m - 1)*Mx + n,2) = (m-1)*Mx + mx + 1 ; % line
48                mic_pos((m - 1)*Mx + n,1) = (n-1)*Mx + nx + 1 ; % column
49                Koordinaten(v)=mic_pos((m - 1)*Mx + n,2)+1i*mic_pos((m -
    1)*Mx + n,1);
50                v=v+1;
51             end
52          end
53
54           drawnow
55           plot(grid_points(:,1), grid_points(:,2),'+','MarkerEdgeColor','
    b','MarkerFaceColor','r',...
56                'MarkerSize',2);
57           hold on
58           plot(mic_pos(:,1), mic_pos(:,2),'o','MarkerEdgeColor','k','
    MarkerFaceColor','r',...
59                'MarkerSize',6);
60           grid on
61           daspect([1 1 1]);
62           hold off
63
64           % Berechnen der Abstaende
65           for n=1:M
66              index=1+(n-1)*M;
67              D(index:index+M-1)=floor(abs(Koordinaten(n)-Koordinaten)*(1/
    e_d));
68           end
69
70           D(D==0)=[];
71           D_k = unique(D);
72           nBin = histcounts(D, D_k);
73           C1 = M*(M-1)/2 - length(nBin);
74
75           % update best solution
76           if C1 < C_best
77                C_best = C1;
78                mic_pos_best = mic_pos;
79                Koordinaten_best = Koordinaten;
80                L_best = L_p;
81           end
82            % check cost function
83           dC = C1 - C;
84           if (dC < 0) || (rand(1,1) < exp(-dC/T))
85               L_r = L_p;
86               C = C1;
87               % tracking value
88               Cp = [Cp;C];
89               np = np+1;
90               if mod(np,10) == 0
91                  disp(fprintf('%d. iteration, %d. permutation, C = %f', k
    , l, C));
92               end
93           end
94      end
95   end
96
97   figure(2);
98   plot(grid_points(:,1), grid_points(:,2),'+','MarkerEdgeColor','b','
    MarkerFaceColor','r',...
99     'MarkerSize',2);
100  hold on
101  plot(mic_pos_best(:,1), mic_pos_best(:,2),'o','MarkerEdgeColor','k','
    MarkerFaceColor','r',...
102    'MarkerSize',6);
```

```
grid on
daspect([1 1 1]);
hold off

% Berechnen der Abstaende
for n=1:M
    index=1+(n-1)*M;
    D(index:index+M-1)=floor(abs(Koordinaten_best(n)-Koordinaten_best))
    ;
end
idx = 1;
Do =zeros(M*(M-1)/2,1);
for n=1:M
    Do(idx:idx + M - n) = D((n-1)*M + n :n*M);
    idx = idx + M - n + 1;
end

Do(Do==0)=[];
Do=sort(Do);
figure(3);  histogram(Do(Do>0),2000);
xlabel('Distance between two microphones (cm)');
ylabel('Number of repreated distances');

figure(4); plot(Cp)
xlabel('Sequence of updating');
ylabel('Cost value tracking');

function y = perm_array(x)
    % randomly exchange two active sensors in an array
    y = x;
    N = length(x);
    [~,idx] = sort(rand(1,N));
    y(idx(1)) = x(idx(2));
    y(idx(2)) = x(idx(1));
end
```

Appendix D

Matlab code: Orthogonal Matching Pursuit (OMP)

```
function [Sest] = omp(A,u,tol,Kmax)
% omp: Orthogonal machting pursuit
%   Input
%       A    : measurement matrix
%       u    : measured vector
%       Kmax : sparsity of Sest
%       tol  : tolerance for approximation between successive solutions
%   Output
%       Sest : Solution found by the algorithm
%

% Initialization
Sest = zeros(size(A,2),1);
v = u;
k = 1;
numericalprecision = 1e-12;
T = [];
disIn = norm(u);
while (k <= Kmax) && (norm(v) > tol*disIn)
  y = abs(A'*v);
  [vals,z] = sort(y,'descend');
  Omega = find(y >= vals(1) & y > numericalprecision);
  T = union(Omega,T);
  b = pinv(A(:,T))*u;
  Sest = zeros(size(A,2),1);
  Sest(T) = b;
  v = u - A(:,T)*b;
  k = k+1;
end
```

Bibliography

[Bar+08] Richard Baraniuk et al. "A simple proof of the restricted isometry property for random matrices". In: *Constructive Approximation* 28.3 (2008), pp. 253–263.

[BCC15] Jacob Benesty, Jingdong Chen, and Israel Cohen. *Design of Circular Differential Microphone Arrays*. Vol. 12. Springer, 2015.

[BCH08] Jacob Benesty, Jingdong Chen, and Yiteng Huang. *Microphone array signal processing*. Vol. 1. Springer Science & Business Media, 2008.

[BD08] Thomas Blumensath and Mike E Davies. "Iterative thresholding for sparse approximations". In: *Journal of Fourier analysis and Applications* 14.5-6 (2008), pp. 629–654.

[Bis06] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[BJ01] Simmer K.U. Bitzer J. *Superdirective Microphone Arrays*. Berlin, Heidelberg: Springer, 2001.

[BJ12] Jacob Benesty and Chen Jingdong. *Study and design of differential microphone arrays*. Vol. 6. Springer Science & Business Media, 2012.

[Boc+15] Holger Boche et al. "A survey of compressed sensing". In: *Compressed sensing and its applications*. Springer, 2015, pp. 1–39.

[Bor+20] Federico Borra et al. "Efficient Implementations of First-Order Steerable Differential Microphone Arrays with Arbitrary Planar Geometry". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2020).

[Bra01] Michael Brandstein. *Microphone arrays: signal processing techniques and applications*. Springer Science & Business Media, 2001.

[BS01] Joerg Bitzer and K Uwe Simmer. "Superdirective microphone arrays". In: *Microphone arrays*. Springer, 2001, pp. 19–38.

[Buc02] Markus Buck. "Aspects of first-order differential microphone arrays in the presence of sensor imperfections". In: *European transactions on telecommunications* 13.2 (2002), pp. 115–122.

[Buc+18] Yaakov Buchris et al. "Incoherent synthesis of sparse arrays for frequency-invariant beamforming". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27.3 (2018), pp. 482–495.

[Can+08] Emmanuel J Candes et al. "The restricted isometry property and its implications for compressed sensing". In: *Comptes rendus mathematique* 346.9-10 (2008), pp. 589–592.

[Cap69] Jack Capon. "High-resolution frequency-wavenumber spectrum analysis". In: *Proceedings of the IEEE* 57.8 (1969), pp. 1408–1418.

[CDS01] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. "Atomic decomposition by basis pursuit". In: *SIAM review* 43.1 (2001), pp. 129–159.

[Cho95] Thomas Chou. "Frequency-independent beamformer with low response error". In: *1995 International Conference on Acoustics, Speech, and Signal Processing*. Vol. 5. IEEE. 1995, pp. 2995–2998.

[CMW14] Rafael E Carrillo, Jason D McEwen, and Yves Wiaux. "PURIFY: a new approach to radio-interferometric imaging". In: *Monthly Notices of the Royal Astronomical Society* 439.4 (2014), pp. 3591–3604.

[CPC15] J. Benesty C. Pan and J. Chen. "Design of robust differential microphone arrays with orthogonal polynomials". In: *The Journal of the Acoustical Society of America* 138.2 (2015), pp. 1079–1089.

[CRT06] Emmanuel J Candès, Justin Romberg, and Terence Tao. "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information". In: *IEEE Transactions on information theory* 52.2 (2006), pp. 489–509.

[CT12] Marco Crocco and Andrea Trucco. "Stochastic and analytic optimization of sparse aperiodic arrays and broadband beamformers with robust superdirective patterns". In: *IEEE Transactions on audio, speech, and language processing* 20.9 (2012), pp. 2433–2447.

[CT14] Marco Crocco and Andrea Trucco. "Design of superdirective planar arrays with sparse aperiodic layouts for processing broadband signals via 3-D beamforming". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22.4 (2014), pp. 800–815.

[CWB08] Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd. "Enhancing sparsity by reweighted l1 minimizazion". In: *Journal of Fourier analysis and applications* 14.5–6 (2008), pp. 877–905.

[DBC01] Rodney A.Kennedy Darren B.Ward and Robert C.Williamson. *Constant Directivity Beamforming*. Springer, 2001, pp. 3–17.

[DBWW95] R. A. Kennedy D. B. Ward and R. C. Williamson. "Theory and design of broadband sensor arrays with frequency invariant far-field beam patterns". In: *The Journal of the Acoustical Society of America* 97.2 (1995), pp. 1023–1034.

[DM09] Wei Dai and Olgica Milenkovic. "Subspace pursuit for compressive sensing signal reconstruction". In: *IEEE transactions on Information Theory* 55.5 (2009), pp. 2230–2249.

[Dob08] Gerhard Doblinger. "Optimized design of interpolated array and sparse array wideband beamformers". In: *2008 16th European Signal Processing Conference*. IEEE. 2008, pp. 1–5.

[Dob10] Gerhard Doblinger. "Optimization of wideband fixed beamformers with adaptive sensor calibration". In: *2010 18th European Signal Processing Conference*. IEEE. 2010, pp. 2062–2066.

[Dob68] Gordon Miller Bourne Dobson. "Forty years' research on atmospheric ozone at Oxford: a history". In: *Applied Optics* 7.3 (1968), pp. 387–405.

[Don06] David L Donoho. "Compressed sensing". In: *IEEE Transactions on information theory* 52.4 (2006), pp. 1289–1306.

[DVMP01] GD De Villiers, FBT Marchaud, and ER Pike. "Generalized Gaussian quadrature applied to an inverse problem in antenna theory". In: *Inverse Problems* 17.4 (2001), p. 1163.

[DVMP03] GD De Villiers, FBT Marchaud, and ER Pike. "Generalized Gaussian quadrature applied to an inverse problem in antenna theory: II. The two-dimensional case with circular symmetry". In: *Inverse Problems* 19.3 (2003), p. 755.

[Elk00] Gary W Elko. "Superdirectional microphone arrays". In: *Acoustic signal processing for telecommunication*. Springer, 2000, pp. 181–237.

[Elk04] Gary W Elko. "Differential microphone arrays". In: *Audio signal processing for next-generation multimedia communication systems*. Springer, 2004, pp. 11–65.

[EMK09] A. Schad E. Mabande and W. Kellermann. "Design of robust superdirective beamformer as a convex optimization problem". In: *Proc. IEEE Int.Conf. Acoust. Speech Signal Process. (ICASSP)* (2009), pp. 77–80.

[EP97] Gary W Elko and Anh-Tho Nguyen Pong. "A steerable and variable first-order differential microphone array". In: *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 12. IEEE. 1997, pp. 223–226.

[ES+18] MA El-Shorbagy et al. "A novel genetic algorithm based k-means algorithm for cluster analysis". In: *International Conference on Advanced Machine Learning Technologies and Applications*. Springer. 2018, pp. 92–101.

[EWT03] Gary W Elko, James E West, and Steve Thompson. "Differential and gradient microphone arrays". In: *The Journal of the Acoustical Society of America* 114.4 (2003), pp. 2426–2426.

[For72] OL Forst. "An algorithm for linearly constrained adaptive processing". In: *Proc. IEEE* 60.8 (1972), pp. 926–935.

[Fro72] Otis Lamont Frost. "An algorithm for linearly constrained adaptive array processing". In: *Proceedings of the IEEE* 60.8 (1972), pp. 926–935.

[GB08] Michael C Grant and Stephen P Boyd. "Graph implementations for nonsmooth convex programs". In: *Recent advances in learning and control*. Springer, 2008, pp. 95–110.

[GB12] Steven L Gay and Jacob Benesty. *Acoustic signal processing for telecommunication*. Vol. 551. Springer Science & Business Media, 2012.

[GBY09] Michael Grant, Stephen Boyd, and Yinyu Ye. *CVX: Matlab software for disciplined convex programming*. 2009.

[GHB18] J. Chen G. Huang and J. Benesty. "Insights into frequency-invariant beamforming with concentric circular microphone arrays". In: *IEEE/ACM Trans. Audio, Speech, Lang. Process.* 26.12 (2018), pp. 2305–2318.

[GJ82] Lloyd Griffiths and CW Jim. "An alternative approach to linearly constrained adaptive beamforming". In: *IEEE Transactions on antennas and propagation* 30.1 (1982), pp. 27–34.

[GM55] EN Gilbert and SP Morgan. "Optimum design of directive antenna arrays subject to random variations". In: *Bell System Technical Journal* 34.3 (1955), pp. 637–663.

[God97] Lal C Godara. "Application of antenna arrays to mobile communications. II. Beam-forming and direction-of-arrival considerations". In: *Proceedings of the IEEE* 85.8 (1997), pp. 1195–1245.

[Ham13] Richard H Hammack. *Book of proof*. Richard Hammack, 2013.

[Hay09] Monson H Hayes. *Statistical digital signal processing and modeling*. John Wiley & Sons, 2009.

[Hay85] Simon Haykin. "Array signal processing". In: *Englewood Cliffs* (1985).

[HBC17] Gongping Huang, Jacob Benesty, and Jingdong Chen. "On the design of frequency-invariant beampatterns with uniform circular microphone arrays". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 25.5 (2017), pp. 1140–1153.

[HCB18] Gongping Huang, Jingdong Chen, and Jacob Benesty. "On the design of robust steerable frequency-invariant beampatterns with concentric circular microphone arrays". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 506–510.

[HL13] Matthew B Hawes and Wei Liu. "Sparse microphone array design for wideband beamforming". In: *2013 18th International Conference on Digital Signal Processing (DSP)*. IEEE. 2013, pp. 1–5.

[HL14] Matthew B Hawes and Wei Liu. "Sparse array design for wideband beamforming with reduced complexity in tapped delay-lines". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22.8 (2014), pp. 1236–1247.

[HS16] Paul Hurley and Matthieu Simeoni. "Flexibeam: analytic spatial filtering by beamforming". In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Ieee. 2016, pp. 2877–2880.

[Hua+20] Gongping Huang et al. "A simple theory and new method of differential beamforming with uniform linear microphone arrays". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28 (2020), pp. 1079–1093.

[HW79] John A Hartigan and Manchek A Wong. "Algorithm AS 136: A k-means clustering algorithm". In: *Journal of the royal statistical society. series c (applied statistics)* 28.1 (1979), pp. 100–108.

[JG07] Dimitris G.Manolakis John G.Proakis. *Digital Signal Processing*. 4th. Inc. Division of Simon and Schuster One Lake Street Upper Saddle River, United States: Prentice–Hall, 2007.

[JXC08] Shihao Ji, Ya Xue, and Lawrence Carin. "Bayesian compressive sensing". In: *IEEE Transactions on signal processing* 56.6 (2008), pp. 2346–2356.

[KV96] Hamid Krim and Mats Viberg. "Two decades of array signal processing research: the parametric approach". In: *IEEE signal processing magazine* 13.4 (1996), pp. 67–94.

[LHZ12] Zhang-Meng Liu, Zhi-Tao Huang, and Yi-Yu Zhou. "An efficient maximum likelihood method for direction-of-arrival estimation via sparse Bayesian learning". In: *IEEE Transactions on Wireless Communications* 11.10 (2012), pp. 1–11.

[Li+13] M. Li et al. "Optimal polarised pattern synthesis of wideband arrays via convex optimisation". In: *IET Microwaves, Antennas and Propagation* 7.15 (2013), pp. 1228–1237.

[Liu+15a] Yanhui Liu et al. "Synthesis of nonuniformly spaced linear arrays with frequency-invariant patterns by the generalized matrix pencil methods". In: *IEEE Transactions on Antennas and Propagation* 63.4 (2015), pp. 1614–1625.

[Liu+15b] Yanhui Liu et al. "Synthesis of sparse arrays with frequency-invariant-focused beam patterns under accurate sidelobe control by iterative second-order cone programming". In: *IEEE transactions on antennas and propagation* 63.12 (2015), pp. 5826–5832.

[LL00] Zhi-Pei Liang and Paul C Lauterbur. *Principles of magnetic resonance imaging: a signal processing perspective*. SPIE Optical Engineering Press, 2000.

[LS20] Phan Le Son. "On the Design of Sparse Arrays With Frequency-Invariant Beam Pattern". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2020), pp. 226–238.

[LW08] Wei Liu and Stephan Weiss. "Design of frequency invariant beamformers for broadband arrays". In: *IEEE Transactions on Signal Processing* 56.2 (2008), pp. 855–860.

[LW10] Wei Liu and Stephan Weiss. *Wideband beamforming: concepts and techniques*. Vol. 17. John Wiley & Sons, 2010.

[LYF13] Zhibao Li, Ka Fai Cedric Yiu, and Zhiguo Feng. "A hybrid descent method with genetic algorithm for microphone array placement design". In: *Applied Soft Computing* 13.3 (2013), pp. 1486–1490.

[Mai17] Robert J Mailloux. *Phased array antenna handbook*. Artech house, 2017.

[McC01] Iain A McCowan. "Robust speech recognition using microphone arrays". PhD thesis. Queensland University of Technology, 2001.

[MH41] RN Marshall and WR Harry. "A new microphone providing uniform directivity over an extended frequency range". In: *The Journal of the Acoustical Society of America* 12.4 (1941), pp. 481–498.

[Mic06] Ulf Michel. "History of Acoustic Beamforming". In: *2006 Berlin Beamforming Conference*. BebeC. 2006.

[Mig14] Marco Donald Migliore. "A simple introduction to compressed sensing/sparse recovery with applications in antenna measurements". In: *IEEE Antennas and Propagation Magazine* 56.2 (2014), pp. 14–26.

[MSK09] Edwin Mabande, Adrian Schad, and Walter Kellermann. "Design of robust superdirective beamformers as a convex optimization problem". In: *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2009, pp. 77–80.

[MTR96] Vittorio Murino, Andrea Trucco, and Carlo S Regazzoni. "Synthesis of unequally spaced arrays by simulated annealing". In: *IEEE Transactions on signal processing* 44.1 (1996), pp. 119–122.

[NAR09] Mike Novey, Tülay Adali, and Anindya Roy. "A complex generalized Gaussian distribution—Characterization, generation, and estimation". In: *IEEE Transactions on Signal Processing* 58.3 (2009), pp. 1427–1433.

[Nor+14] Sven E Nordholm et al. "Broadband beamforming and optimization". In: *Academic Press Library in Signal Processing*. Vol. 3. Elsevier, 2014, pp. 553–598.

[NT09] Deanna Needell and Joel A Tropp. "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples". In: *Applied and computational harmonic analysis* 26.3 (2009), pp. 301–321.

[Ols32] Harry F Olson. "A Uni-Directional Ribbon Microphone". In: *The Journal of the Acoustical Society of America* 3.3 (1932), pp. 315–316.

[Ols46] Harry F Olson. "Gradient microphones". In: *The Journal of the Acoustical Society of America* 17.3 (1946), pp. 192–198.

[Pas15] Jordan Paschke. *Lemma 0.27: Composition of Bijections is a Bijection*. 2015.

[P.E88] Mayes P.E. *Frequency-Independent Antennas*. Boston, MA: Springer, 1988, pp. 517–637.

[Pfl17] Patrick von Pflug. "ASPECTS OF THE USE OF MEMS MICROPHONES IN PHASED ARRAY SYSTEMS". In: *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*. Vol. 255. 2. Institute of Noise Control Engineering. 2017, pp. 5093–5103.

[PK89] S Unnikrishna Pillai and Byung Ho Kwon. "Forward/backward spatial smoothing techniques for coherent signal identification". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 37.1 (1989), pp. 8–15.

[PP97] Arogyaswami J Paulraj and Constantinos B Papadias. "Space-time processing for wireless communications". In: *IEEE signal processing magazine* 14.6 (1997), pp. 49–83.

[PWV08] Gabriela Trazzi Perim, Estefhan Dazzi Wandekokem, and Flávio Miguel Varejão. "K-means initialization methods for improving clustering by simulated annealing". In: *Ibero-American Conference on Artificial Intelligence*. Springer. 2008, pp. 133–142.

[Rab75] Lawrence R Rabiner. *Theory and Application of Digital Signal Processing*. An introduction to the theory of two-dimensional signal processing. USA: Prentice-Hall, 1975. Chap. 7.

[Raf15] Boaz Rafaely. *Fundamentals of spherical array processing*. Vol. 8. Springer, 2015.

[Ras+11] Eugen Rasumow et al. "Robustness of virtual artificial head topologies with respect to microphone positioning". In: *Proceedings of Forum Acusticum*. 2011, pp. 397–402.

[RK89] Richard Roy and Thomas Kailath. "ESPRIT-estimation of signal parameters via rotational invariance techniques". In: *IEEE Transactions on acoustics, speech, and signal processing* 37.7 (1989), pp. 984–995.

[RTMSJ17] A Richard Thompson, James M Moran, and George W Swenson Jr. *Interferometry and synthesis in radio astronomy*. Springer Nature, 2017.

[SBA10] Mehrez Souden, Jacob Benesty, and Sofiène Affes. "A study of the LCMV and MVDR noise reduction filters". In: *IEEE Transactions on Signal Processing* 58.9 (2010), pp. 4925–4935.

[SC07] D. P. Scholnik and J. O. Coleman. "Optimal array-pattern synthesis for wideband digital transmit arrays". In: *IEEE J. Sel. Topics Signal Process.* 1.4 (2007), pp. 660–677.

[Sch86] Ralph Schmidt. "Multiple emitter location and signal parameter estimation". In: *IEEE transactions on antennas and propagation* 34.3 (1986), pp. 276–280.

[Sha49] C. E. Shannon. "Communication in the Presence of Noise". In: *in Proceedings of the IRE* 37.1 (1949), pp. 10–21.

[Sim15] Matthieu Martin Jean-Andre Simeoni. *Towards more accurate and efficient beamformed radio interferometry imaging*. Tech. rep. 2015.

[Son20] Phan Le Son. "Uniform Array with Broadband Beamforming for Arbitrary Beam Patterns". In: *arXiv preprint arXiv:2003.00991* (2020).

[Sto+15] Christoph Stoeckle et al. "Doa estimation performance and computational complexity of subspace-and compressed sensing-based methods". In: *WSA 2015; 19th International ITG Workshop on Smart Antennas*. VDE. 2015, pp. 1–6.

[Syd94] Carsten Sydow. "Broadband beamforming for a microphone array". In: *The Journal of the Acoustical Society of America* 96 (1994), p. 845.

[SYH07] Y. Ma S. Yan and C. Hou. "Optimal array pattern synthesis for broadband arrays". In: *The Journal of the Acoustical Society of America* 122.5 (2007), pp. 2686–2696.

[TE01] Heinz Teutsch and Gary W Elko. "First-and second-order adaptive differential microphone arrays". In: *Proc. IWAENC*. Vol. 1. Citeseer. 2001.

[TE04] H. Teutsch and G. W. Elko. "First- and second-order adaptive differential microphone arrays". In: *The Journal of the Acoustical Society of America* 115.1 (2004), pp. 57–60.

[TG07] Joel A Tropp and Anna C Gilbert. "Signal recovery from random measurements via orthogonal matching pursuit". In: *IEEE Transactions on information theory* 53.12 (2007), pp. 4655–4666.

[Tru01] A Trucco. "Synthesizing wide-band sparse arrays by simulated annealing". In: *MTS/IEEE Oceans 2001. An Ocean Odyssey. Conference Proceedings (IEEE Cat. No. 01CH37295)*. Vol. 2. IEEE. 2001, pp. 989–994.

[TT12] Ya-Ju Tsai Terence Tao Christoph Thiele. *The nonlinear Fourier transform*. Los Angeles,USA: Department of Mathematics, UCLA, 2012. Chap. 1. URL: https://www.math.uni-bonn.de/people/thiele/teaching/2012NLFT/.

[Tur66] L Richard Turner. "Inverse of the Vandermonde matrix with applications". In: (1966).

[Uri83] Robert J. Urick. *Principles of Underwater Sound*. Mc Graw-Hill Book, 1983.

[VB88] B. D. Van Veen and K. M. Buckley. "Beamforming: a versatile approach to spatial filtering". In: *IEEE ASSP Magazine* 5.2 (1988), pp. 4–24.

[Vil04] GD de Villiers. "A singular function analysis of the wideband beam pattern design problem". In: *Inverse problems* 20.5 (2004), p. 1517.

[VLG83] Charles F Van Loan and Gene H Golub. *Matrix computations*. Johns Hopkins University Press Baltimore, 1983.

[VMB02] Martin Vetterli, Pina Marziliano, and Thierry Blu. "Sampling signals with finite rate of innovation". In: *IEEE transactions on Signal Processing* 50.6 (2002), pp. 1417–1428.

[Vor13] Sergiy A Vorobyov. "Principles of minimum variance robust adaptive beamforming design". In: *Signal Processing* 93.12 (2013), pp. 3264–3277.

[VT04] Harry L Van Trees. *Optimum array processing: Part IV of detection, estimation, and modulation theory*. John Wiley & Sons, 2004.

[VVB88] Barry D Van Veen and Kevin M Buckley. "Beamforming: A versatile approach to spatial filtering". In: *IEEE assp magazine* 5.2 (1988), pp. 4–24.

[WEG87] Svante Wold, Kim Esbensen, and Paul Geladi. "Principal component analysis". In: *Chemometrics and intelligent laboratory systems* 2.1-3 (1987), pp. 37–52.

[Wia+09] Yves Wiaux et al. "Compressed sensing imaging techniques for radio interferometry". In: *Monthly Notices of the Royal Astronomical Society* 395.3 (2009), pp. 1733–1742.

[Wid+75] Bernard Widrow et al. "Adaptive noise cancelling: Principles and applications". In: *Proceedings of the IEEE* 63.12 (1975), pp. 1692–1716.

[Wil99] Earl G Williams. *Fourier acoustics: sound radiation and nearfield acoustical holography*. Academic press, 1999.

[WKW01] Darren B Ward, Rodney A Kennedy, and Robert C Williamson. "Constant directivity beamforming". In: *Microphone arrays*. Springer, 2001, pp. 3–17.

[Woo50] Max A Woodbury. "Inverting modified matrices". In: *Memorandum report* 42.106 (1950), p. 336.

[WR04] David P Wipf and Bhaskar D Rao. "Sparse Bayesian learning for basis selection". In: *IEEE Transactions on Signal processing* 52.8 (2004), pp. 2153–2164.

[WWM20] Shefeng Yanb Wenxia Wanga and Linlin Mao. "Time-domain frequency-invariant beampattern synthesis via alternating direction method of multipliers". In: *The Journal of the Acoustical Society of America* 147 (2020), p. 3372.

[Yan06] Shefeng Yan. "Optimal design of FIR beamformer with frequency invariant patterns". In: *Applied Acoustics* 67.6 (2006), pp. 511–528.

[YZL11] W. Liu Y. Zhao and R. Langley. "Application of the least squares approach to fixed beamformer design with frequency-invariant constraints". In: *IET Signal Process.* 7.3 (2011), pp. 281–291.

[ZBC14] Liheng Zhao, Jacob Benesty, and Jingdong Chen. "Design of robust differential microphone arrays". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 22.10 (2014), pp. 1455–1466.

[ZGET04] Yahong Rosa Zheng, Rafik A Goubran, and Mohamed El-Tanany. "Experimental evaluation of a nested microphone array with adaptive noise cancellers". In: *IEEE Transactions on Instrumentation and Measurement* 53.3 (2004), pp. 777–786.

[Zio20] Lawrence Ziomek. *Fundamentals of acoustic field theory and space-time signal processing*. CRC press, 2020.

Personal Information

Name	PHAN LE SON
Email	phan@eit.uni-kl.de
National	Viet Nam

Educational Background

2018-2021	PhD study at Technical University of Kaiserslautern, Germany
2012 -2014	Master study at Vietnamese German University, Viet Nam Major in Mechatronics and Sensor System
2004 -2009	Engineering study at University of Technology Ho Chi Minh City, Viet Nam Major in Mechatronics

Work Experience

Period: 2/2017 – 6/2017
Company: Technologies Transfer Center in Germany
Position: Software Engineer (Working in Germany)
Duties and Responsibilities: Beam-forming algorithm for microphone array

Period: 07/2015 – 1/2017
Company: Autonomous (a start-up company on robot design)
Position: Hardware and Embedded SW Engineer
Duties and Responsibilities:
Project:
- Microphone Array design
- IoT devices design

Period: 10/2010 – 07/2015
Company: Robert Bosch Engineer and Business Solutions VN
Position: Embedded software specialist
Project: SW function developer and system design for Vehicle Function in Engine Control Unit: Power Train System, Vehicle Devices (Brake, Clutch, Acceleration Pedal…), Electrical Supply and Driver Assistance