

# Beamforming and Array Processing

HENRY D'ASSUMPCAO AND DOUG GRAY



Em Prof Henry d'Assumpcao AO



Prof Doug Gray

# Contents

Chapter 1. INTRODUCTION	6
Chapter 2. MODELS	15
2.1. Introduction	15
2.2. Propagating fields	15
2.3. Propagation	17
2.4. Noise	20
2.5. Representation of the receiver outputs	21
2.6. Overview	23
Chapter 3. BEAM PATTERN	24
3.1. Linear array, broadside signal	24
3.2. Linear array, beamsteered	38
3.3. General array configuration	45
3.4. Beam pattern of an array in 3 dimensions	47
3.5. Overview	47
Chapter 4. SHAPING THE BEAM PATTERN	49
4.1. Introduction	49
4.2. Shading	49
4.3. Null-steering	55
4.4. Superdirectivity	59
4.5. Overview	61
Chapter 5. ARRAY GEOMETRY DESIGN	63
5.1. Introduction	63
5.2. Array symmetry	63
5.3. Beam pattern of an array of directional receivers	65
5.4. Sparse arrays	68
5.5. Redundant and non-redundant linear arrays	69
5.6. Co-array	72
5.7. Random arrays	74
5.8. Overview	74
Chapter 6. STEERING MULTIPLE BEAMS - CONVENTIONAL PROCESSING	76
6.1. Introduction	76
6.2. Multiple single-frequency signals	77
6.3. Multiple narrowband signals	79

6.4. Multiple broadband signals—time domain beamforming	80
6.5. Multiple broadband signals – frequency domain beamforming	84
6.6. Implementation of frequency domain beamforming	86
6.7. Overview	87
<b>Chapter 7. WAVEVECTORS, SPATIAL SAMPLING AND FILTERING</b>	<b>88</b>
7.1. Introduction	88
7.2. Propagating wave fields	88
7.3. Linear array—one-dimensional case	90
7.4. Wavenumber beamforming	92
7.5. Steering beams in wavevector	94
7.6. Steering independent beams	97
7.7. Varying the temporal frequency	98
7.8. Beam patterns in the frequency-wavenumber representation	100
7.9. Overview	102
<b>Chapter 8. BEAMFORMING - RANDOM PROCESSES</b>	<b>103</b>
8.1. Introduction	103
8.2. Covariance function and covariance matrix	103
8.3. Cross-spectral matrix	104
8.4. Examples of the cross-spectral matrix	105
8.5. Frequency Domain Beamforming for Random Processes	107
8.6. Examples of beamforming with the cross-spectral matrix	107
8.7. Time Delay and Sum Beamforming for Random Processes	109
8.8. Overview	109
<b>Chapter 9. ARRAY GAIN</b>	<b>111</b>
9.1. Introduction	111
9.2. Expression for array gain	112
9.3. Conventional beamformer	114
9.4. Array gain for shaded beamformer	119
9.5. Array gain with null steering	120
9.6. Array gain expressed in terms of the beam pattern	122
9.7. Overview	123
<b>Chapter 10. OPTIMAL PROCESSING</b>	<b>125</b>
10.1. Introduction	125
10.2. Maximising gain	125
10.3. Array gain vs beamsteered direction	126
10.4. Examples – Gain	129
10.5. Minimum power with linear constraint	130
10.6. Minimum mean-square error	131
10.7. Maximum likelihood estimator	132
10.8. Summary of optimal processors	134
10.9. Examples – Steered beamformer output	134
10.10. Discussion	136
10.11. Overview	138

Chapter 11. SAMPLE MATRIX BEAMFORMING	140
11.1. Introduction	140
11.2. Estimation of the cross-spectral matrix	141
11.3. Properties of the estimated cross-spectral matrix	143
11.4. Conventional beamforming using the estimated cross-spectral matrix	144
11.5. Optimal beamforming using the estimated cross-spectral matrix	145
11.6. Sample matrix inverse update	146
11.7. Overview	149
Chapter 12. GRADIENT DESCENT ALGORITHMS	151
12.1. Introduction	151
12.2. Deterministic Gradient Descent	152
12.3. Least Means Squares—Stochastic Gradient Descent Algorithm	158
12.4. Overview	162
Chapter 13. SPACE-TIME ADAPTIVE PROCESSING	163
13.1. Introduction	163
13.2. Optimal broadband processor	164
13.3. Constrained broadband gradient descent algorithm	169
13.4. Stochastic constrained broadband LMS algorithm	170
13.5. Overview	170
Chapter 14. ESTIMATION	172
14.1. Introduction	172
14.2. Classical DOA Estimation Techniques	172
14.3. Fundamental Limits to Estimation	175
14.4. Statistical Estimation Techniques	178
14.5. Crossed dipoles	185
Chapter 15. SUBSPACE METHODS	190
15.1. Introduction	190
15.2. Uncorrelated noise only	191
15.3. Single signal and uncorrelated noise	191
15.4. $L$ signals in uncorrelated noise	192
15.5. Estimating the direction of arrival	194
15.6. Pisarenko's method	195
15.7. Multiple Signal Classification (MUSIC)	196
15.8. Eigenvector method	198
15.9. Estimating the number of signals	199
15.10. ESPRIT	202
15.11. Overview	207
Appendix A. OPTIMISATION	209
A.1. Introduction	209
A.2. Extrema of functions of $K$ real variables	209
A.3. Constrained optimisation	211
A.4. Real functions of complex variables	212



A.5. Gradient descent	215
A.6. Maximum likelihood estimator	218
Appendix B. RANDOM SIGNALS AND NOISE	219
B.1. Introduction	219
B.2. Random Variable and Probability Density Functions	219
B.3. Multiple Random Variables	220
B.4. Ensemble Averaging	221
B.5. Covariance Matrix	222
B.6. Complex random vectors	223
B.7. Stationary Random Processes	224
Appendix C. MATRIX REVISION	226
C.1. Introduction	226
C.2. Definitions	227
C.3. Matrix operations	227
C.4. Special matrices	228
C.5. Quadratic forms	229
C.6. Partitioned matrices	229
C.7. Useful formulae for determinants	230
C.8. Useful formulae for matrix inverses	230
C.9. Inequalities	231
C.10. Generalised inverses	233
C.11. Kronecker product	233
C.12. Calculus	234
C.13. Eigenvalues and eigenvectors	235
Appendix D. FUNDAMENTALS OF ESTIMATION THEORY	237
D.1. Mean Value of an Estimator	238
D.2. Variance of an Estimator	238
D.3. Proof of CRLB	239
D.4. Least Squares Estimation	241
D.5. Bias of LS Estimator	242
D.6. Covariance of the LS Estimator	242
Appendix E. PROBLEMS	244
E.1. Mechanical vs Electronic steering	244
E.2. Beamwidth	244
E.3. Beampattern nulls	244
E.4. Interferences	245
E.5. Beampattern vs frequency	245
E.6. Beamwidth and array geometry	245
E.7. Beampattern of rectangular array	245
E.8. 2-receiver array	246
E.9. Grating lobes	247
E.10. Multipath arrivals	247
E.11. Towed array shape errors	248

E.12.	Null-steering 1	249
E.13.	Steering multiple nulls	249
E.14.	Symmetries	250
E.15.	Beampattern of staggered array	251
E.16.	$N$ sub-arrays of $K$ receivers	251
E.17.	Time vs frequency domain beamforming	252
E.18.	Time delays	252
E.19.	Time delays for circular array	252
E.20.	Circular array	253
E.21.	Time sampling errors	254
E.22.	Wavenumber	254
E.23.	Cross-spectral matrix	255
E.24.	Gain of 3-receiver array	255
E.25.	Array gain of rectangular array	256
E.26.	Gain in isotropic noise	256
E.27.	Null-steering 2	257
E.28.	Kantorovich inequality	259
E.29.	Spherically isotropic noise	259
E.30.	Correlated noise field	260
E.31.	Array position errors	260
E.32.	Optimal beamformer	262
Appendix.	Bibliography	264
Appendix.	List of symbols	266

## CHAPTER 1

### INTRODUCTION

A single sensor (or receiver) such as an antenna or a microphone can deduce many properties of a received signal—e.g., its *temporal* properties (waveform, spectral shape, etc.). Quite often, such sensors have an *omnidirectional* response—i.e., they are insensitive to the direction of incident signals.

Some sensors—of which a parabolic antenna is an example—have a response that varies with the direction of an incoming signal. The parabolic dishes and slotted antennas used in radar and communications are familiar examples. Such *directional* sensors can be used to discriminate in favour of signals from certain directions, to attenuate noise, or to deduce the direction of arrival of a signal. These capabilities make them extremely useful and thus they find widespread application<sup>1</sup>.

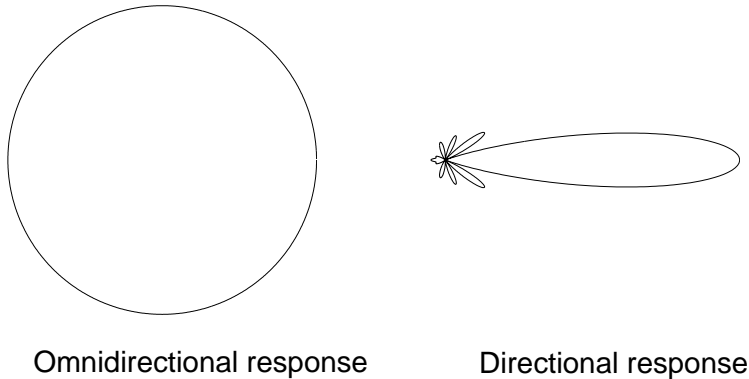


FIGURE 1.1. Illustrating directional response of a sensor

However, there are evident disadvantages in the use of directional sensors: to maximise their sensitivity to signals from different directions they must be physically rotated, and it is difficult to distinguish between more than one source simultaneously. Furthermore, their directional response is difficult to change.

To overcome these limitations, we can use multiple receivers, arranged in an *array*. An array of receivers, suitably deployed and with appropriate signal processing, provides more gain and better system performance than just a single receiver. Such

<sup>1</sup>See [24] for a detailed description of the theory and practice of such sensors.

arrays are widely used – in radar, sonar, medical diagnostics, radio and optical astronomy and seismology, for example. The configuration of receiving arrays can take many forms as illustrated in Figure 1.2. Array processing is a branch of signal processing concerned with using the outputs of spatially distributed receivers to extract information about the properties of signals and noises incident upon the array. In signal processing, we usually sample the waveform to produce a time series. With an array of receivers, we do the same in space; array processing thus operates with both temporal and spatial data and is often termed *space-time processing*.

The angular resolution of arrays depends largely on their size (loosely termed *aperture*): the bigger the aperture, the better the resolution as a rule. However, the geometry of the array needs careful design. Arrays can be densely filled (as with the grid array shown above) or *sparse* (as illustrated by the circular array above); selection of an inappropriate geometry can penalise performance. If the array is moving (for example, if it is a radar on an aircraft or satellite) it is possible to overcome the limitations of size by coherent processing of the array signals; such *synthetic aperture radars* give an effective array aperture much larger than the physical size of the array, and hence greatly increased resolution. Such techniques have been exploited in remote sensing and in radio-astronomy.

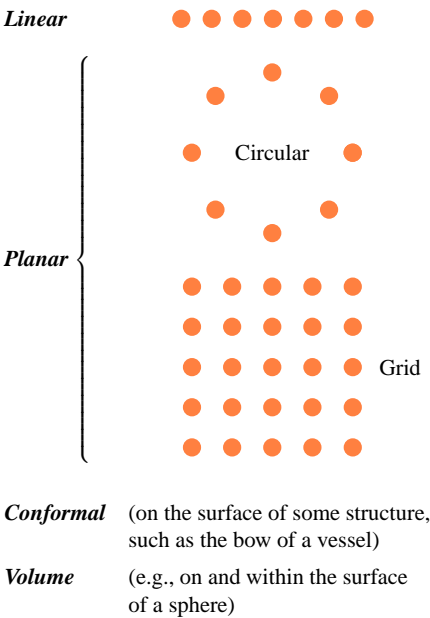


FIGURE 1.2. Some array configurations

In general we consider the outputs of receivers in some sort of field – e.g., acoustic or electromagnetic – which can be considered as the superposition of propagating waves. If the waves arise from a distant (*far-field*) source, they can usually be treated as plane waves (i.e., having a plane wavefront, as illustrated in Figure 1.3). However many of the concepts developed here can be extended to *near-field* sources for which the wavefront is curved.

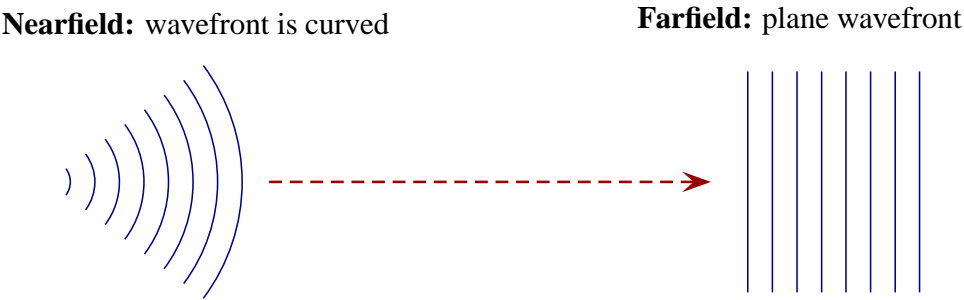


FIGURE 1.3. Illustrating near- and far-field wave propagation

Here we are not much concerned with the types of signal energy – they could be electromagnetic, acoustic, seismic, etc., – nor shall we consider in detail specific types of transducers such as antennas, hydrophones, geophones, etc. What we shall attempt is a generic formulation emphasising principles common to many types of arrays and applicable to both active and passive systems. Examples of applications from specific systems will, however, be given.

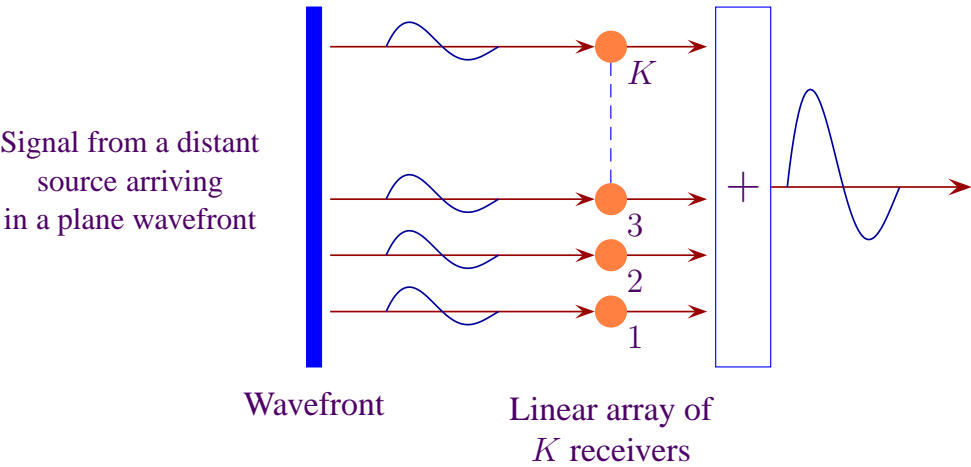


FIGURE 1.4. Conventional beamforming – array steered broadside

Essentially, the idea behind array processing is to use more than one receiver to sample the signal and noise field; by knowing more about these we can enhance the signal, suppress noise and reject interfering signals. The simplest array processor is illustrated in Figure 1.4. It shows a uniform linear array of  $K$  receivers arranged in a straight line. A signal from a far-field source at right angles to the array would arrive as a plane wavefront, striking all the receivers at the same time. If all the outputs of all the receivers are added together, those signals will be reinforced. Signals

from other directions do not reinforce and will tend to be suppressed. Similarly, noise – for example, receiver noise – is random and will not be reinforced. The effect is that signals are received preferentially from one direction, and we speak of forming a ***receive beam***.

Later we shall develop this basic idea more. We shall show that – as intuition might suggest – the more we know about the signal and the noise, the better we can enhance the signal and suppress the noise. In the example in Figure 1.4, reception is from one direction only; if we wish to reinforce a signal from another direction, we could steer the receiver beam by rotating the whole array mechanically so that it lies perpendicular to the new direction. (Some surveillance radars and communications systems scan mechanically in this way.) We shall later consider alternatives to mechanical rotation, using electronic steering to achieve a similar result. An enormous advantage of such ***electronic beamsteering*** is that it allows us to invoke the power of digital signal processors to form many beams simultaneously, thus providing instantaneous coverage over a wide angular sector.

Arrays were used at least as far back as World War I to detect submarines. With the advances in electronics after World War II, developments in signal processing, including for arrays of receivers, progressed rapidly. In recent times, the dramatic development of computers and associated digital electronics has allowed the implementation of many theoretical developments. Uses of receiver arrays include the following.

***Signal detection.*** One of the primary roles of an array of receivers is to improve the detection of signals. Often weak signals are difficult to detect and single-channel processing techniques such as filtering, spectrum analysis, etc., may not provide sufficient gain to discriminate against background noise, receiver noise or unwanted signals (***interference***) from discrete directions. By coherently processing the outputs of an array of receivers, gain is increased and thus the overall system detection threshold is lowered. This is achieved by either *suppressing noise* or by the *rejection of interfering signals*.

***Signal estimation.*** With an array of receivers distributed spatially we may, by either combining the receiver outputs or by employing multivariate statistical techniques, estimate signal properties that are usually unobtainable from a single receiver. Such properties include

- direction,
- range,
- propagation speed and
- propagation modes.

**Noise estimation.** In many cases arrays are used not to characterise particular isolated signals but to estimate the angular (or *wavenumber*) distribution of some naturally occurring or man-made noise field. For example, in sonar, vertical line arrays have been extensively used to measure the vertical directionality of noise within the ocean. In radio astronomy, large radio-telescope arrays have been used to map the distribution of radio frequency energy of far-off galaxies.

**Resolution of sources.** Whether considering either signals or noise it is often important to discriminate between different sources of energy. As mentioned earlier, arrays have the ability to differentiate energy from different directions and when this information is used in conjunction with a little physics, it often provides valuable clues for the discrimination between sources. This use of arrays – particularly with advanced high-resolution signal-processing algorithms – has been a popular area of recent research.

**Target tracking and localisation.** Once sources (*targets*) have been detected and classified, the results are commonly used as inputs to tracking algorithms. Most commonly it is the measured directional parameters, viz., azimuth and elevation, that are used. When there is more than one array, techniques such as triangulation can be used to localise targets. More recently, particularly in the sonar area, *matched field array processing*, which assumes a knowledge of the propagation conditions, has been used to develop methods which allow localisation as well as tracking to be carried out with a single array.

Applications include:

- exploration seismology (using geophones on land and hydrophones at sea)
- sonar (using hydrophones – hull-mounted, towed line arrays, sonobuoy arrays, bottom-mounted arrays)
- radar (using antennas in line or grid arrays)
- radio astronomy (using movable dishes to generate a large effective array aperture)
- ultrasound (high-frequency acoustic transducers).



FIGURE 1.5. Seismic survey vessel towing a number of hydrophone arrays

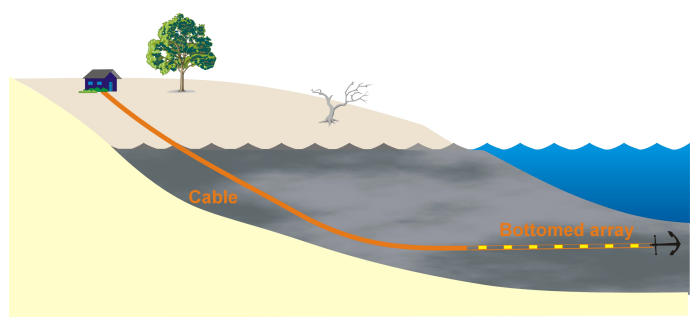


FIGURE 1.6. Arrays of hydrophones are deployed on the ocean bottom; signals are transmitted by cable to shore

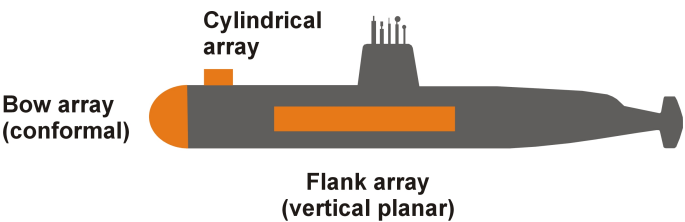


FIGURE 1.7. Submarines carry a number of arrays mounted the hull typically on the flank and conformally distributed on the bow



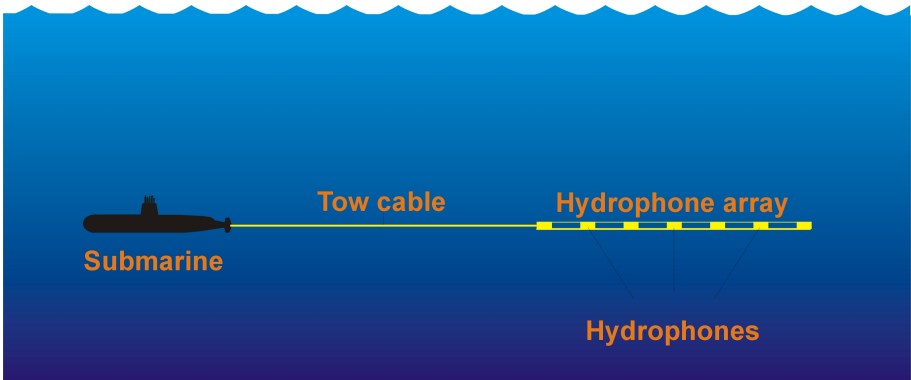


FIGURE 1.8. Very long, flexible arrays are towed behind submarines

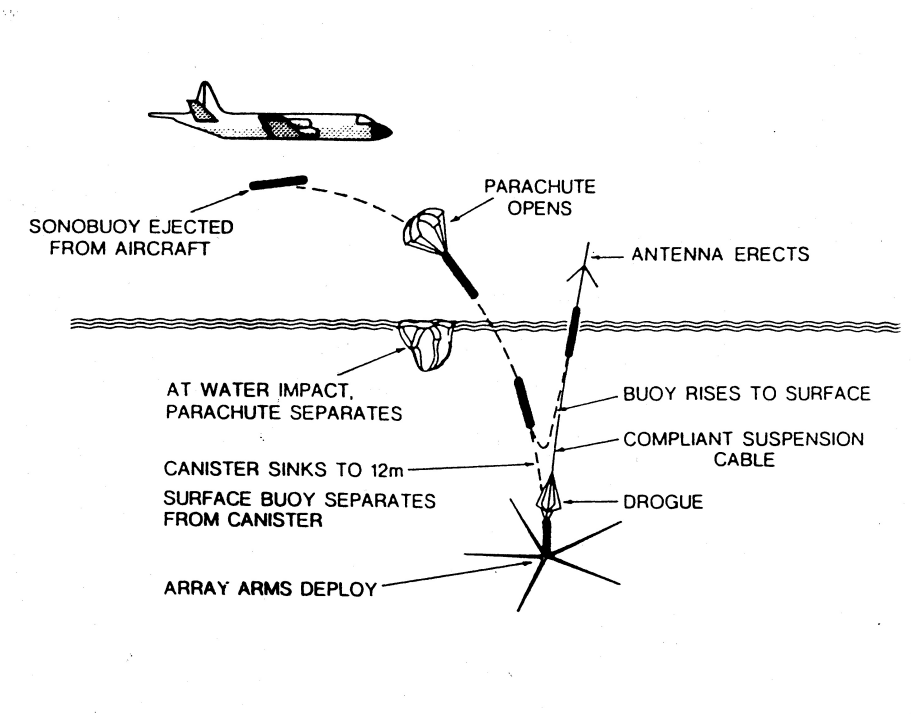


FIGURE 1.9. Passive receiving arrays of hydrophones are deployed in sonobuoys dropped from aircraft. The figure illustrates the deployment of the Australian-developed Barra sonobuoy.



FIGURE 1.10. Over-the-horizon radars operate by refracting HF radio waves from the ionosphere; they typically use antenna array kilometres in length. The figure shows one of the Australian Jindalee radar receivers.

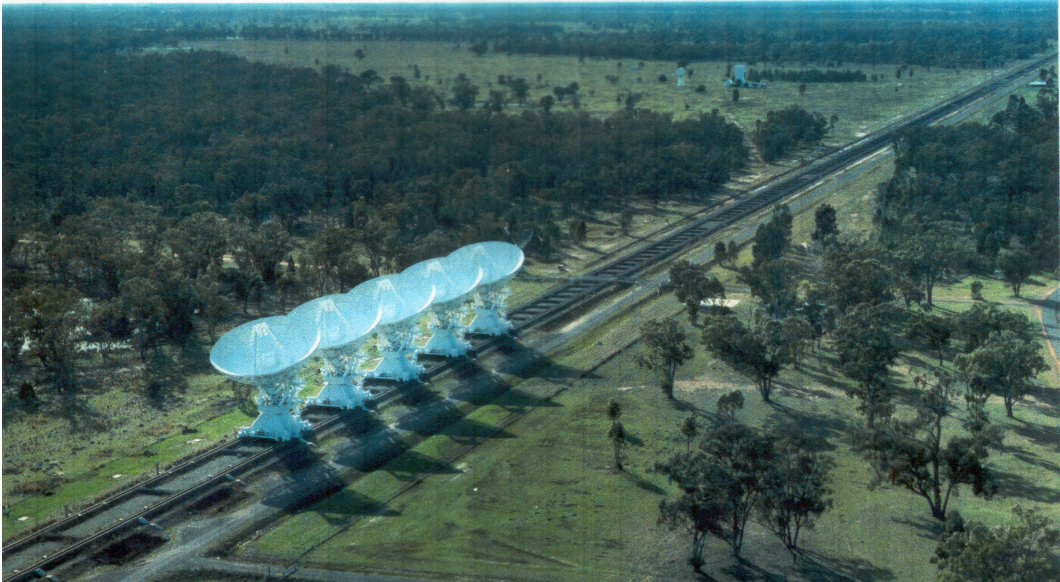


FIGURE 1.11. Radio telescopes use large paraboloidal dishes on railway tracks to give an receiving aperture of variable size. The Australia radioastronomy telescope is shown in the photograph.

We describe here some techniques for processing signals from arrays of receivers. In the first few chapters, we deal solely with *deterministic* signals. We begin in Chapter 2 with models of signal and noise. Chapters 3 and 4 give an introduction to conventional beamforming and beamsteering and some techniques for shaping the beam pattern. Then in Chapter 5 the advantages and disadvantages of various array geometries are discussed.

In Chapters 6 and 7 we introduce the concept of steered beams and wavenumber. Chapter 8 considers random processes and Chapter 9 develops expressions for *array gain*.

Various methods for optimal processing in the frequency domain are discussed in Chapter 10. In Chapter 11 we consider operations using finite samples of the data and their effect on performance.

Chapter 12 addresses *gradient descent* methods and Chapter 13 *space-time adaptive processing*. Finally, in Chapter 14, we introduce *subspace* methods.

Readers are assumed to have some familiarity with optimisation theory, probability theory and matrix operations. Some revision notes are provided in Appendix A – C.

In the text are analytical *Problems*, and hyperlinks to *Exercises* that use the Matlab-based Beamforming and Array Processing (BAP) toolbox supplied with these course notes.

▷ Problem E.1

▷ Ex 1

## CHAPTER 2

# MODELS

### 2.1. Introduction

In deriving signal processing techniques, we attempt to represent or model the real world mathematically. Not surprisingly, if our model is a close representation of the physical world, we get good results; conversely, if we select a poor model, the results can be very bad. The model we choose is thus critical to the success of our signal processing techniques when we seek to apply it. As mentioned in the preceding chapter, array processing is concerned with handling functions of both time and space; thus we need to develop a realistic model of both the temporal and the properties of our signals and of our noises.

It goes without saying that we must have a good understanding of the physical environment in which our system operates and of the design details of the system. For example, if our problem is signal processing for microwave radar, it is generally reasonable to assume that the electromagnetic waves propagate in straight lines; that assumption is not true for HF over-the-horizon radar which is affected by the earth's ionosphere, nor for long-range propagation of sound in the ocean, nor for propagation of seismic energy in the earth. In some systems, noise arises mainly from external sources; in others, the principal interference is from receiver thermal noise.

In this chapter we introduce the models used for signal and noise and the notation used. We cover:

- propagating fields
- propagation modes
- representation of receiver outputs
- signal models
- noise sources
- ambient noise.

### 2.2. Propagating fields

We consider sensors immersed in a field which is comprised of the superposition of sinusoidal waves propagating through some medium. The sensors convert the signals in the medium to electrical signals that we then process<sup>1</sup>. For the sake of

---

<sup>1</sup>In this course we assume throughout that the sensors themselves do not in any way distort the physical environment in which they are immersed. The implication is that each receiver is assumed

simplicity we shall only consider *scalar* fields (such as acoustic fields). If  $t$  is time and  $\mathbf{u}$  is the vector of coordinates of a point in space:

$$\mathbf{u} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad (2.1)$$

then the field can be denoted by  $f(\mathbf{u}, t)$ . As mentioned in Chapter 1, we shall assume throughout that the sensors are sufficiently distant from the sources of all these waves for us to be able to treat the waves as plane. Further, we assume that the medium in the vicinity of the array is *homogeneous* (i.e., its properties do not vary spatially).

Let us consider a single plane wave arriving at the array from azimuthal and elevation directions  $\theta_s$  and  $\phi_s$  respectively, as illustrated in Figure 2.1. At a single frequency  $f$ , the time at which a plane wave arrives at point  $\mathbf{u}$  is given by  $d/c$ , where

$$d = x \cos \theta_s \sin \phi_s + y \sin \theta_s \sin \phi_s + z \cos \phi_s$$

and  $c$  is the speed of propagation in the medium.

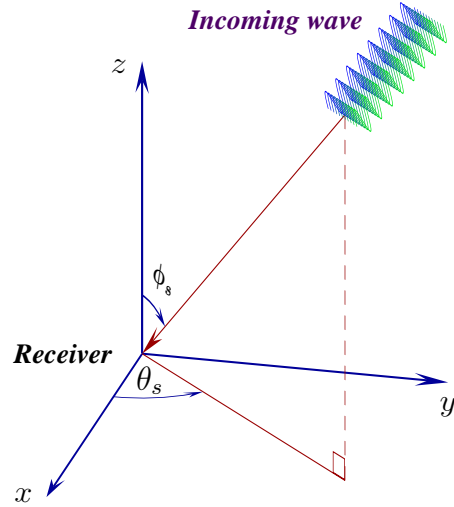


FIGURE 2.1. Coordinate system

Let us write

$$\mathbf{k}_s = \begin{bmatrix} (\mathbf{k}_s)_x \\ (\mathbf{k}_s)_y \\ (\mathbf{k}_s)_z \end{bmatrix} = \frac{2\pi}{\lambda} \begin{bmatrix} \cos \theta_s \sin \phi_s \\ \sin \theta_s \sin \phi_s \\ \cos \phi_s \end{bmatrix}, \quad (2.2)$$

where  $\lambda$  is the wavelength. Let us denote the transpose of a vector (or matrix) by  $^T$ . Then

$$\mathbf{k}_s^T \mathbf{u} = \frac{2\pi}{\lambda} [\cos \theta_s \sin \phi_s \quad \sin \theta_s \sin \phi_s \quad \cos \phi_s] \begin{bmatrix} x \\ y \\ z \end{bmatrix} = 2\pi d/\lambda, \quad (2.3)$$

so we can represent the propagating plane wave as

$$f(\mathbf{u}, t) = \alpha \exp\{i(\omega t + \mathbf{k}_s^T \mathbf{u})\}, \quad (2.4)$$

---

to be 'transparent' and that its presence does not affect the signals at arriving at any other receiver of the array.

where ( $\alpha$  is the amplitude and  $\omega = 2\pi f$  and  $\mathbf{k}_s$  is called the *signal wavevector*<sup>2</sup>. In general we shall consider a field comprised of an arbitrary number of such plane-wave signals, of different frequencies and of different directions of arrival.

### 2.3. Propagation

As discussed in Section 2.2 we shall restrict our attention to fields that can be modelled as plane waves incident upon the array. This implies that the speed of propagation  $c$  across and within the array should be constant—i.e., the medium is locally homogeneous. However, this does not mean that  $c$  need be globally constant; there are many situations in practice where  $c$  can be taken to be constant locally but varies over a large scale and, when it does, energy can propagate from source to receiver via different modes or different paths. We give some examples in the next section.

Signal propagation varies widely according to the application and affects the model. There are in practice very many, and quite different, modes of propagation and it is not possible to cover all of these. In this section we seek only to illustrate a few of the types of propagation that can occur, and point out the necessity of understanding the propagation conditions that prevail in the particular problem being addressed.

While in general these often do not affect our array processing algorithms—provided the medium is homogeneous in the vicinity of the array—it is important to understand these propagation modes when using and interpreting the results of our array processing. For example, in underwater sound propagation, we can have multipath arrivals that are correlated with one another; it is important then to include such correlation in the model.

It should be stressed however in a significant number of cases  $c$  will not be constant within the array neighbourhood. For example, in shallow water, vertical line arrays of hydrophones may extend over the whole water depth and at certain frequencies beamforming assuming plane waves may give results that are far from optimum. Furthermore, in such situations, if there is knowledge of propagation modes (and if they can be modelled well), then this can be used to advantage. This is done in *matched field processing* where the coherent processing underlying array processing is extended to include the propagation model. In this way the concept of array processing is dramatically extended to source localisation.

---

<sup>2</sup>Such a function is the solution of a wave equation of the form

$$c^2 \nabla^2 f(\mathbf{x}, t) = \frac{\partial^2}{\partial t^2} f(\mathbf{x}, t),$$

where

$$\nabla^2 \equiv \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2},$$

and  $c$  is the speed of propagation. In many applications  $c$  is constant.



Below we consider some examples of different propagation modes for different types of fields and applications.

### 2.3.1. Radio waves.

At microwave frequencies, electromagnetic waves generally propagate in straight lines through the atmosphere and space. At HF, however, electromagnetic propagation is affected by the ionosphere which is a region with ionised particles formed by the effect of solar radiation on the upper reaches of the earth's atmosphere. Its properties are measured by the *free electron density*. Because the electron density varies with altitude, an electromagnetic ray propagating through it will be bent. In Figure 2.2 we illustrate a ray that is curved towards the surface of the earth. On striking the surface, the ray will be reflected and trace a similar path through the ionosphere again.

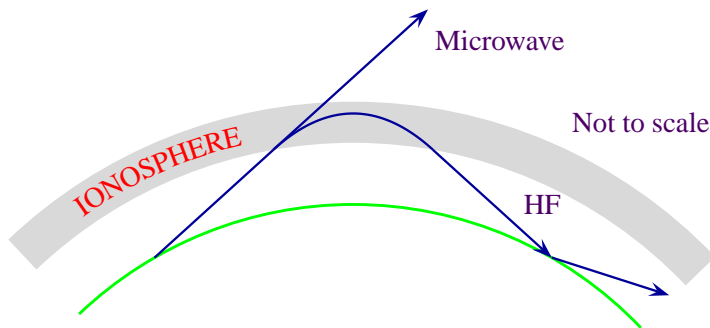


FIGURE 2.2. Propagation of electromagnetic waves

The ray paths illustrated in the figure are idealised; in practice the ionosphere is far from smooth and uniform. It may for example have two or more layers as illustrated in Figure 2.3; there might as a result be complex paths between a transmitter and a receiver. Ionospheric conditions vary with the time of day, season of year and even according to the 11-year sunspot cycle.

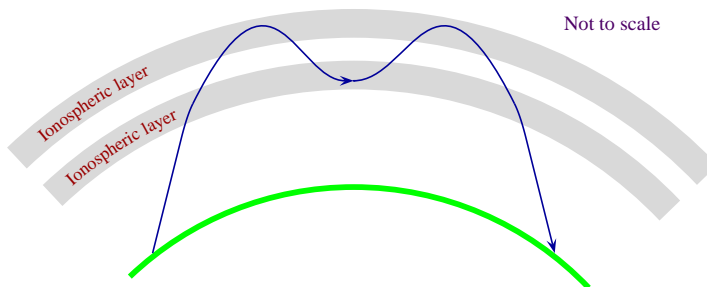


FIGURE 2.3. Complex radio propagation via the ionosphere

### 2.3.2. Acoustic waves in the sea.

There are similarly complex conditions in the ocean for acoustic waves. Propagation there is determined largely by the speed of sound in seawater which is not uniform but increases with temperature, pressure and salinity.

Near the surface of the sea, water is often well mixed, in which case the temperature is nearly constant. Below the mixed layer, temperature tends to fall rapidly in the so-called **thermocline**. At great depths the water is once again at a substantially constant temperature. These variations are illustrated in Figure 2.4.

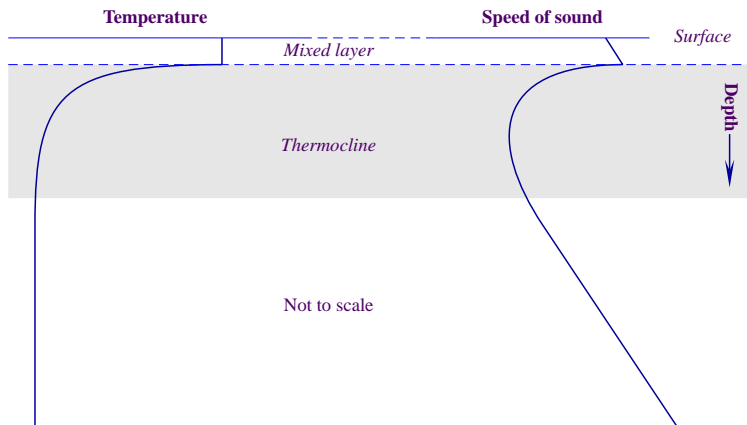


FIGURE 2.4. Temperature and speed profiles in the ocean

The speed of sound is very sensitive to temperature; in the thermocline, where the temperature falls rapidly, so does the speed of sound. On the other hand, near the surface of the sea and at great depths, the temperature is substantially constant; in those regions the effect of pressure predominates and the speed increases with depth. The result is a profile of sound speed vs depth sketched in Figure 2.5. This profile affects the propagation of sound in a complicated way. In the mixed layer near the surface, a ray at a small angle to the horizontal is bent upwards towards the surface where it is reflected. In this way sound waves are trapped in the so-called **surface duct**. Other rays at a slightly steeper angle propagate down below the mixed layer and are bent downwards and then upwards, eventually to reappear at the surface some 50 km away, as illustrated in Figure 2.5. Other rays at yet steeper angles strike the bottom.



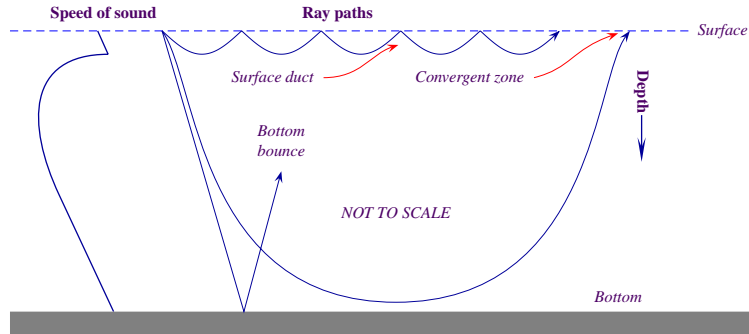


FIGURE 2.5. Temperature and speed profiles in the ocean

### 2.3.3. Seismic waves in the earth.

More complex still is the propagation of seismic waves in the earth, which is of great interest to petroleum exploration geophysicists, for example. The earth has very complex structures which affect seismic wave propagation. Not only does the speed of propagation change with rock type and with pressure (depth), but in addition there are different modes of propagation: *longitudinal waves* (or *p-waves* with vibrations in the same direction as the propagation) as well as *transverse waves* (or *s-waves* with vibrations at right angles to the direction of propagation). On encountering a boundary between two rock types, there can be mode conversion from *p*- to *s*-waves, and vice versa, as illustrated in Figure 2.6.

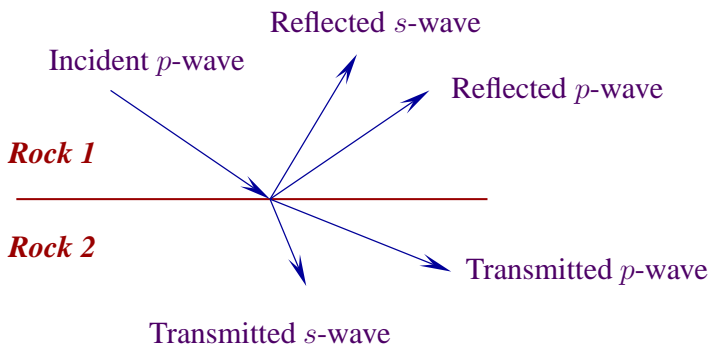


FIGURE 2.6. Partitioning of energy at a boundary between rocks

## 2.4. Noise

Noise can arise from many causes. There is *internal* noise of the system itself—for example, thermal noise in the receiver—and *external* (or *ambient*) noise from outside the system. In HF radar, external noise could, for example, result from distant thunderstorms or from other users of the HF band. In sonar, it can come

from ships, marine life, waves or rain. Seismic measurements can be contaminated by interference from mining operations.

When noise sources are uniformly distributed around the array, as illustrated in Figure 2.7, the noise is called *isotropic*.

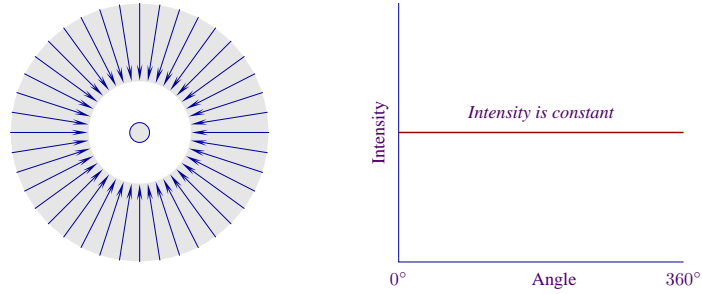


FIGURE 2.7. Isotropic noise

Figure 2.8 shows non-isotropic noise. A discrete noise source (i.e., one that comes from only one direction) is referred to as an *interference*.

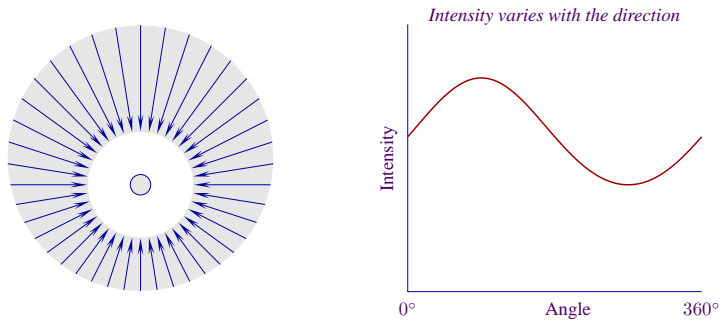


FIGURE 2.8. Non-isotropic noise

The term *interference* is used to denote some unwanted signal source arriving at the array from a discrete direction. Such interference can be deliberate (*jamming*) or accidental.

## 2.5. Representation of the receiver outputs

We shall model each receiver as a transducer which converts the energy in the incident field to electrical signals. The transducer is assumed to be free of non-linearities so that the principle of superposition applies and the receiver outputs can be represented as a linear combination of signal and noise components.

Denoting the continuous time output of the  $j^{\text{th}}$  receiver after suitable filtering and down-conversion as  $x_j(t)$  we can then write

$$x_j(t) = s_j(t) + n_j(t) \quad (2.5)$$

and for an array of  $K$  receivers we shall use the vector notation<sup>3</sup>:

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_K(t) \end{bmatrix}, \quad \mathbf{s}(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_K(t) \end{bmatrix}, \quad \mathbf{n}(t) = \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_K(t) \end{bmatrix} \quad (2.6)$$

and thus

$$\mathbf{x}(t) = \mathbf{s}(t) + \mathbf{n}(t). \quad (2.7)$$

We shall, in general, represent receiver outputs as complex quantities which we idealise to be the complex *spatial* samples of the incident field  $f(\mathbf{u}_k, t)$ , where  $\mathbf{u}_k$  is the position the  $k^{\text{th}}$  receiver. Furthermore, in most modern systems the receiver outputs are, after suitable filtering and down-conversion, sampled in time. Assuming a uniform time sampling interval  $\Delta t$  then our receiver outputs can be represented as a space-time sampling of the complex field—i.e.,  $f(\mathbf{u}_k, j\Delta t)$ . As we shall see later (in Chapter 12, this sampling in both space and time plays a fundamental and important role in array processing.

However, in many cases the time sampling issues are not of primary importance and for ease of notation we shall often use a continuous time representation of the receiver outputs. We frequently represent the receiver outputs as complex quantities. In electronic engineering terms, the real component would correspond to the output of an in-phase receiver, and the imaginary component to that of a quadrature receiver, as shown in Figure 2.9. We then have:

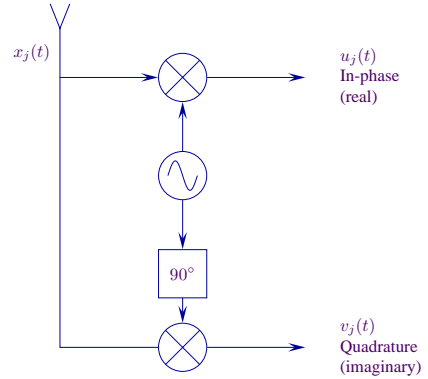


FIGURE 2.9. Complex representation

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_K(t) \end{bmatrix} = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_K(t) \end{bmatrix} + i \begin{bmatrix} v_1(t) \\ v_2(t) \\ \vdots \\ v_K(t) \end{bmatrix}. \quad (2.8)$$

Most receivers either by nature of construction or by design effect some temporal filtering of the incident field. Thus it is important to consider the frequency bandwidth of the receiver outputs as practical systems vary greatly in this aspect from

<sup>3</sup>We use lower case boldface characters for vectors and upper case boldface for matrices.

application to application. For example, the output signals from a passive sonar array of hydrophones typically can have a bandwidth from 10 Hz to 1000 Hz—a range of two decades. On the other hand, the outputs of a phased array radar may typically have a centre frequency of 10 GHz and a bandwidth of a few hundred MHz. Whilst we do not wish to describe and model specific systems in detail here, it is useful to consider a few of the generic classes that are commonly used. Bear in mind that these are not exhaustive and that for particular signals and noise we can often gain greater insight and improved array performance by considering more specific models for our receiver outputs.

## 2.6. Overview

In this chapter we have covered briefly

- the dependence of signal processing on an adequate mathematical representation of the real world,
- propagating signal and noise fields,
- how propagation varies according to the medium,
- the representation of received signals and noise as complex vectors.

In this course we assume that the signals arrive at the array as plane waves and that the medium in the vicinity of the array is homogeneous.

Noise can arise from within the receiving system ('receiver noise') or externally ('ambient noise'). Discrete noise sources (unwanted signals) are called 'interference'.

Understanding how wanted signals and unwanted noises propagate to the array often allows signal processing to be devised to discriminate between them.

The outputs of the receiving array are represented as complex vectors.

### Summary

- (1) We need a good model of our environment if our signal processing is to be effective.
- (2) Propagation of energy may not be in straight lines.
- (3) Noise can arise from the receiver system itself (*self-noise*) and from a variety of external sources (*ambient noise* and *interference*).

## CHAPTER 3

# BEAM PATTERN

### 3.1. Linear array, broadside signal

#### 3.1.1. Concept.

Consider a linear array as shown in Figure 3.1 in a *homogeneous* (i.e., uniform) medium with a single far-field signal source and assume each receiver is free of noise. Under these conditions the signal arrives at the array as a plane wavefront.

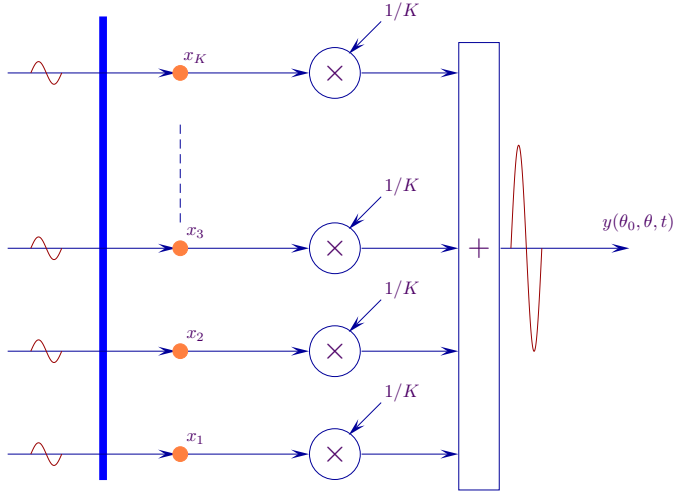


FIGURE 3.1. Conventional beamformer – steered broadside

To detect a signal arriving at right angles (**broadside**) to the array axis the outputs of all  $K$  receivers are added, as illustrated in Figure 3.1. (For mathematical convenience, we scale the amplitudes of the receiver outputs by  $1/K$ .<sup>1</sup>) If a source is in a direction normal to the array, the receivers outputs are all in phase, and when summed reinforce one another<sup>2</sup>. Signals from other directions will not be in phase and so will not be reinforced. This simple example is an elementary, but most

<sup>1</sup>The same result would of course be obtained by simply multiplying the output of the adder by  $1/K$ .

<sup>2</sup>This is an example of *coherent summation*.

useful, type of **conventional beamformer**<sup>3</sup> and the output is termed the **broadside beam**.

In all that follows it is assumed that all receivers are omnidirectional in their response and have the same sensitivity, and that the geometry of the array is known exactly. Further, although in this chapter only plane-wave signals are considered, the results obtained can be applied directly to signals of narrow bandwidth.

If  $x_j(t)$  is the output of the  $j^{\text{th}}$  receiver, the output of the array processor, i.e., the time series of the broadside beam,  $y(t)$ , is

$$y(t) = \frac{1}{K} \sum_{j=1}^K x_j(t).$$

To demonstrate how well the broadside conventional beamformer suppresses signals not from the broadside direction, the power out of the array (i.e., after summation) is plotted as the direction of the source<sup>4</sup> is varied. This function is defined to be the **polar response** or **beam pattern** of the array. By convention it is normalised to be unity in the beamsteered direction.

▷Ex 2.1–2.3

The beam pattern of an array is a function of both the horizontal angle  $\theta_s$  and the vertical angle  $\phi_s$  of the incident direction of the plane wave with respect to the array. Thus the beam pattern is a 3-dimensional surface. In most of what follows we shall consider only the beam pattern in the horizontal plane – i.e., the response of the array as a distant signal source is moved in the horizontal plane around the array.

The beam pattern of the linear array of Figure 3.1, with 15 receivers spaced half a wavelength apart, is shown on a polar plot in Figure 3.2<sup>5</sup>. Note that the beam pattern of a linear array is symmetrical about the axis of the array.<sup>6</sup>

In Figure 3.3 the same beam pattern is presented, but plotted in dB on cartesian coordinates and from  $-90^\circ$  to  $+90^\circ$ . The beam pattern is very important as it gives both a quantitative and, when plotted, a visual display of how well the beamformed output of the array discriminates between the desired signal and unwanted signals from other directions. The direction of peak response is termed the **maximum response axis (MRA)**.

The **beamwidth** (i.e., the width of the **main lobe**) is often defined as the angle between the points at which the amplitude of the beam pattern falls to half or by 3dB; the beamwidth indicates the ability to resolve signal sources whose arrival directions are closely spaced. Subsidiary peaks are termed **sidelobes**; their amplitude is a measure of susceptibility to interference by unwanted signals in directions away

<sup>3</sup>This type of beamformer is often colloquially referred to as an **add-squarer**.

<sup>4</sup>An incident plane wave.

<sup>5</sup>An expression for this will be derived shortly.

<sup>6</sup>This should be obvious from a consideration of the geometry. In fact, because of rotational symmetry, linear arrays are inherently unable to distinguish between signals arriving from a cone about the axis of the array; therefore for linear arrays only angles between  $-\pi/2$  and  $\pi/2$  will be plotted.

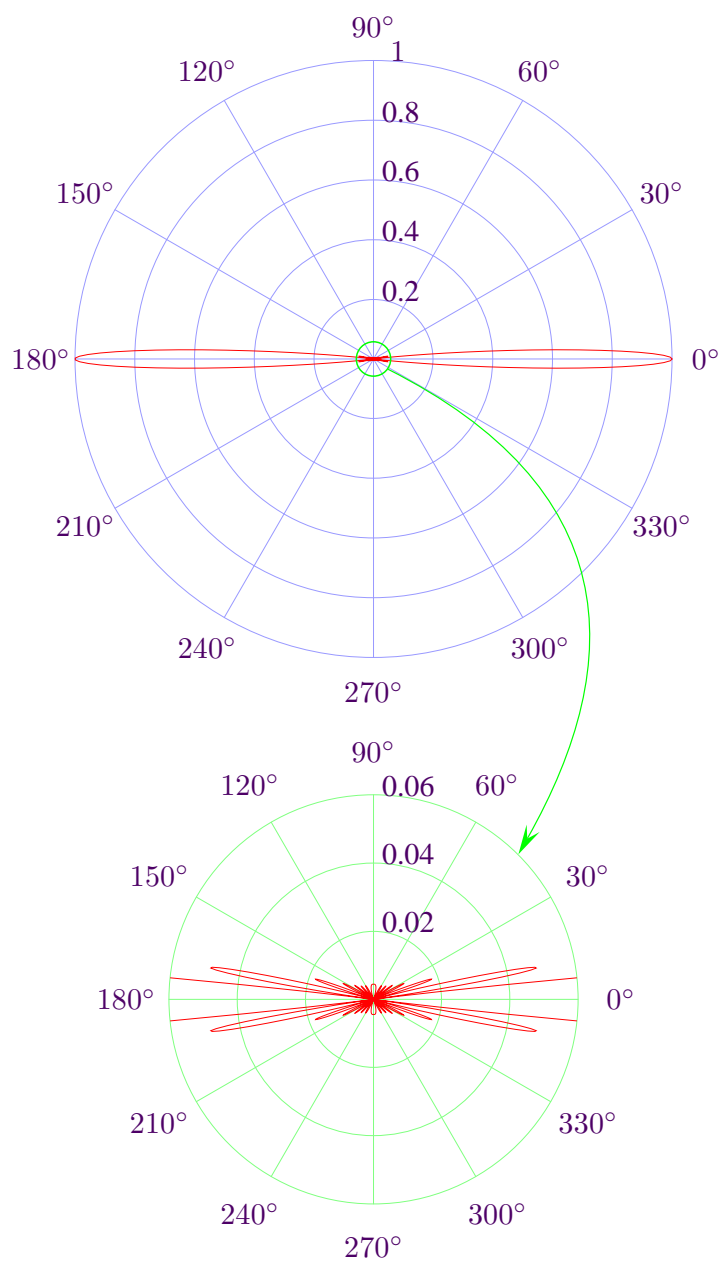


FIGURE 3.2. Polar beam pattern – array of 15 receivers uniformly spaced half a wavelength apart, steered broadside. The lower figure shows an expanded view of the sidelobes.

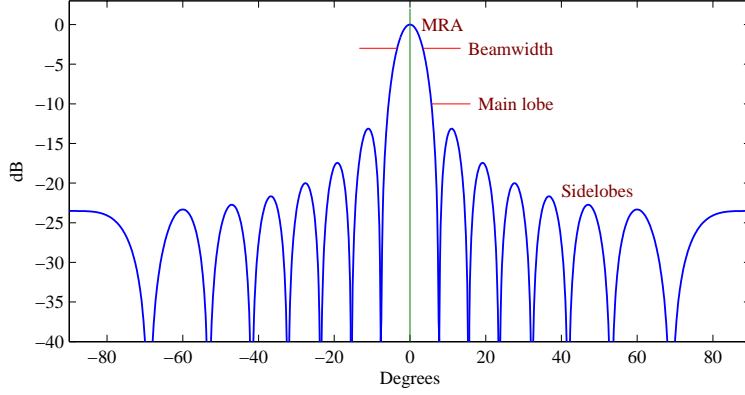


FIGURE 3.3. Cartesian beam pattern – 15 receivers uniformly spaced half a wavelength apart, steered broadside

from the main lobe. If the level of the sidelobes could be reduced to zero, interfering sources outside of the main lobe could be completely rejected. Unfortunately, that is impossible: the response can be made zero at discrete points but cannot be made zero everywhere.

### 3.1.2. Expression for the beam pattern.

An expression for the beam pattern of a narrowband linear array of equally-spaced receivers designed to receive signals from broadside is now derived. Consider a system of coordinates as illustrated in Figure 3.4, with the receivers along the  $y$ -axis. Using the notation of Section 2.2, the coordinates of the  $j^{\text{th}}$  receiver are:<sup>a</sup>

$$\mathbf{u}_j = \begin{bmatrix} 0 \\ [j-1]d \\ 0 \end{bmatrix} \quad (3.1)$$

where  $d$  is the *inter-element spacing*.

<sup>a</sup>The origin is arbitrary and is chosen for convenience.

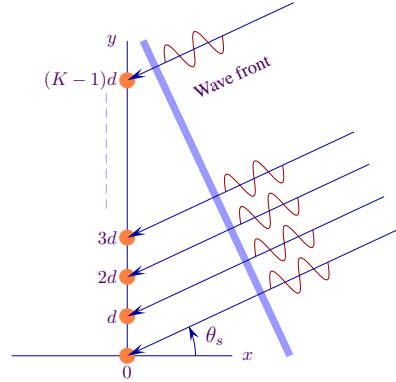


FIGURE 3.4. System of coordinates

The response of the broadside beam to a single sinusoidal signal arriving as a plane wavefront from a direction  $\theta_s$  in the horizontal plane, (ie,  $\phi_s = 90^\circ$ ) will now be considered.

From (2.4) the signal  $s_j(t, \theta_s)$  at the  $j^{\text{th}}$  receiver is given by

$$s_j(t, \theta_s) = s(t) \exp(i\mathbf{k}_s^T \mathbf{u}_j) \quad (3.2)$$



where  $s(t) = \exp\{i\omega t\}$  is the output of the first receiver located at the origin. Substituting

$$\mathbf{k}_s = \frac{2\pi}{\lambda} \begin{bmatrix} \cos \theta_s \\ \sin \theta_s \\ 0 \end{bmatrix} \quad (3.3)$$

it follows that  $s_j(t, \theta_s) = s(t) \exp\left(\frac{i2\pi(j-1)d \sin \theta_s}{\lambda}\right)$  (3.4)

(Note from Figure 3.4 that the plane wave is received at the  $j^{\text{th}}$  receiver

$$\frac{(j-1)d \sin \theta_s}{c},$$

in advance of that at the origin and the phase factor in the above expression is just the phase rotation corresponding to such a time difference.<sup>7</sup>)

The term  $\exp\left(\frac{i2\pi(j-1)d}{\lambda} \sin \theta_s\right)$

is called the *spatial phase factor*.

Writing  $z \triangleq \exp\left(\frac{i2\pi}{\lambda} d \sin \theta_s\right)$  (3.5)

and  $v_j(\theta_s) = z^{j-1}$ , (3.6)

gives  $s_j(t, \theta_s) = s(t)v_j(\theta_s)$ . (3.7)

(3.7) shows that, for narrowband signals, the receiver output can be factored into two factors: the signal waveform  $s(t)$  received at some (arbitrary) reference point (chosen for convenience to be the origin) and a spatial phase factor  $v_j(\theta_s)$  which depends only on  $\theta_s$ . This factorisation into temporal and spatial components simplifies the subsequent analysis considerably. Note also that the spatial phase factor is dependent on the ratio  $\frac{d}{\lambda}$ , a dimensionless quantity.

Using the notation introduced in Chapter 2 the vector of receiver signal outputs, i.e.,

$$\mathbf{s}(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_K(t) \end{bmatrix}, \quad (3.8)$$

is given by  $\mathbf{s}(t) = s(t)\mathbf{v}(\theta_s)$  (3.9)

---

<sup>7</sup> $c$  = speed of propagation in the medium and  $\lambda$  = wavelength of the signal.

where the vector  $\mathbf{v}(\theta_s)$ , termed the **steering vector**, is given by

$$\mathbf{v}(\theta_s) = \begin{bmatrix} 1 \\ \exp\left(\frac{i2\pi}{\lambda}d \sin \theta_s\right) \\ \exp\left(\frac{i2\pi}{\lambda}2d \sin \theta_s\right) \\ \vdots \\ \exp\left(\frac{i2\pi}{\lambda}(K-1)d \sin \theta_s\right) \end{bmatrix} \quad (3.10)$$

In this case, with no noise present and with only a single signal source, the broadside beamformer output is written as  $y(0, \theta_s, t)$ , where  $s$  denotes that this is the signal-only case, 0 denotes the broadside beam, and  $\theta_s$  is the direction of arrival of the signal. Summing the outputs of the  $K$  receivers gives

$$y(0, \theta_s, t) = \frac{1}{K} \sum_{j=1}^K s_j(t, \theta_s) = \frac{1}{K} s(t) (1 + z + z^2 + \dots + z^{K-1}). \quad (3.11)$$

This is the sum of the terms of a geometric progression and so

$$y(0, \theta_s, t) = \frac{1}{K} s(t) \left( \frac{1 - z^K}{1 - z} \right) = \frac{1}{K} s(t) z^{(K-1)/2} \left( \frac{z^{K/2} - z^{-K/2}}{z^{1/2} - z^{-1/2}} \right) \quad (3.12)$$

which, substituting for  $z$ , reduces to

$$y(0, \theta_s, t) = \frac{1}{K} s(t) \exp\left(\frac{i(K-1)\pi d}{\lambda} \sin \theta_s\right) \left( \frac{\sin\left(\frac{K\pi d}{\lambda} \sin \theta_s\right)}{\sin\left(\frac{\pi d}{\lambda} \sin \theta_s\right)} \right). \quad (3.13)$$

The instantaneous power output is

$$|y(0, \theta_s, t)|^2 = \frac{1}{K^2} |s(t)|^2 \left( \frac{\sin\left(\frac{K\pi d}{\lambda} \sin \theta_s\right)}{\sin\left(\frac{\pi d}{\lambda} \sin \theta_s\right)} \right)^2. \quad (3.14)$$

Note that  $|y(0, 0, t)|^2 = |s(t)|^2$ . The second term on the right in (3.14), which is independent of the temporal waveform of the signal, describes the response of the broadside linear array beamformer as the signal direction  $\theta_s$  is varied. This second term, when normalised to a value of unity at  $\theta_s = 0$ , is called the **beam pattern** of the array <sup>8</sup>:

$$P(0, \theta_s) = \frac{|y(0, \theta_s, t)|^2}{|y(0, 0, t)|^2} = \frac{1}{K^2} \left( \frac{\sin\left(\frac{K\pi d}{\lambda} \sin \theta_s\right)}{\sin\left(\frac{\pi d}{\lambda} \sin \theta_s\right)} \right)^2. \quad (3.15)$$

$P(0, \theta_s)$  is shown plotted in dB in Figure 3.3.

---

<sup>8</sup>Exercise: Satisfy yourself that  $\lim_{\theta_s \rightarrow 0} \frac{\sin\left(\frac{K\pi d}{\lambda} \sin \theta_s\right)}{\sin\left(\frac{\pi d}{\lambda} \sin \theta_s\right)} = K$

(3.15) can be simplified by writing

$$(\mathbf{k}_s)_y = \frac{2\pi}{\lambda} \sin \theta_s \cos(0), \quad (3.16)$$

so we have 
$$P(0, (\mathbf{k}_s)_y) = \frac{1}{K^2} \left\{ \frac{\sin((\mathbf{k}_s)_y K d / 2)}{\sin((\mathbf{k}_s)_y d / 2)} \right\}^2. \quad (3.17)$$

Note that  $(\mathbf{k}_s)_y$  is the component of the wave-vector along the axis of the array.

Finally using vector notation the beam pattern can be written as

$$P(0, \theta_s) = \frac{1}{K^2} |\mathbf{v}^H(0) \mathbf{v}(\theta_s)|^2,$$

where  $^H$  denotes the complex conjugate (Hermitian) transpose of a vector:  $\mathbf{v}^H = \mathbf{v}^{*T}$  and

$$\mathbf{v}(0) = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}.$$

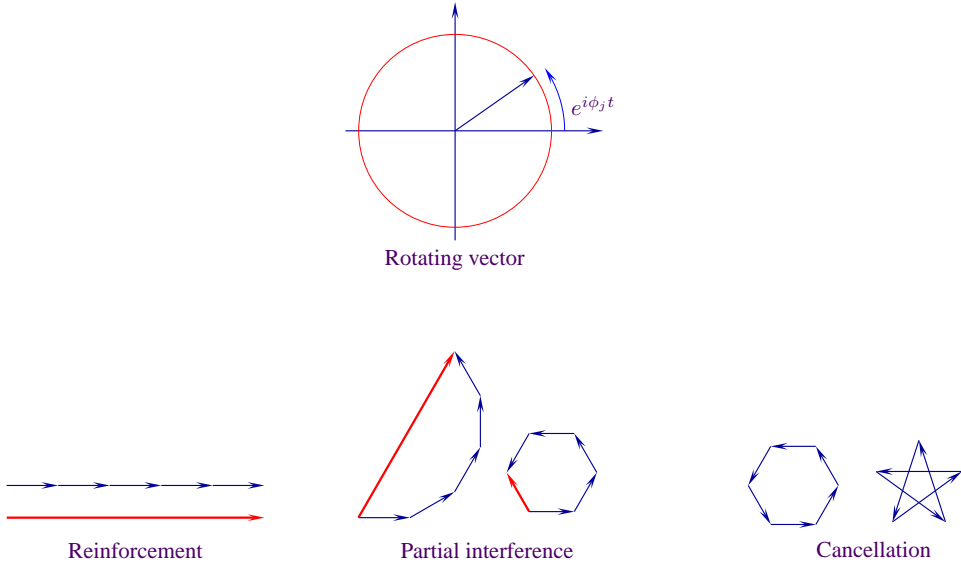


FIGURE 3.5. Phasor representation

An alternative, graphical representation of the beamformer output can be obtained by considering a phasor representation,  $\exp(i\theta_j)$ , where the phasor angle,  $\theta_j$  is given by  $\frac{2\pi(j-1)d}{\lambda} \sin \theta_s$ .

For the add-squarer illustrated in Figure 3.1, a broadside signal will arrive at all the receivers in phase and will reinforce. For other signal directions, there will be a

phase shift between the outputs of the different receivers, because of the different propagation delays, as illustrated in Figure 3.4.

For a uniformly-spaced linear array the phase shift between adjacent receivers is identical. The vectors are then rotated before addition as illustrated in Figure 3.5. It is apparent that the sum is reduced in amplitude when a phase shift is introduced, with nulls at some phase shifts (i.e., corresponding to certain directions).

### 3.1.3. Properties of the beam pattern.

#### 3.1.3.1. Inter-element spacing and array aperture.

Note that in (3.15) and (3.17) the beam pattern is dependent on  $d/\lambda$ , i.e., it is a function of the *ratio* of the inter-element spacing to the wavelength, and not of their *absolute* values.  $Kd$  in the numerator is called the **effective array aperture**; for large  $K$  it is approximately equal to the total physical length of the array  $(K - 1)d$ .

This is illustrated in the table below for a sonar and radar array. The speed of sound in water is very much less than that of radio waves, and the sizes of arrays and their operating frequencies tend to be very different for these two applications; yet when  $d/\lambda$  and  $Kd/\lambda$  are the same exactly the same beam pattern is obtained in each case.

	Sonar array	Radio array
Speed of propagation	1500 m/s	$3 \times 10^8$ m/s
Frequency	150 Hz	10 GHz
Wavelength $\lambda$	10 m	3 cm
Inter-element spacing $d$	5 m	1.5 cm
$d/\lambda$	0.5	0.5
No of elements $K$	15	15
$Kd/\lambda$	7.5	7.5
Beamwidth	$6.7^\circ$	$6.7^\circ$

3.1.3.2. *Maximum response axis and main lobe.* As planned, the maximum response axis (MRA) is in the broadside direction, and this is so for all  $K$  (i.e., whatever the number of receivers) and for all frequencies.

The **beamwidth** is defined as the angle between half-power (or 3dB) points and is a good measure of the ability of the array to resolve signal sources with similar directions of arrival. Usually the design aim is to make the beamwidth small. However, in many practical situations where the incident direction of the desired

signal is unknown and it is only possible to form a finite number of beams, some tolerance in the MRA is required – a finite beamwidth provides this.

For signals within the beamwidth, arrivals at the receivers are substantially coherent and tend to reinforce. Thus signals whose arrival directions lie within the beamwidth are unlikely to be resolved.

The 3-dB points of the main lobe of the broadside beam pattern are obtained from (3.15); writing  $\gamma = K\pi d \sin \theta_s / \lambda$ , we have

$$P(0, \theta_s) = \left( \frac{\sin \gamma}{K \sin(\gamma/K)} \right)^2 \quad (3.18)$$

$$= \frac{1}{2} \quad \text{when} \quad \frac{\sin(\gamma)}{K \sin(\gamma/K)} = \frac{1}{\sqrt{2}}. \quad (3.19)$$

For  $\gamma/K \ll 1$ , this occurs when

$$\frac{\sin \gamma}{\gamma} \simeq \pm \frac{1}{\sqrt{2}}. \quad (3.20)$$

The solution is obtained using numerical tools:  $\gamma_s \simeq \pm 1.392$ , and

$$\sin \theta_s \simeq \pm \frac{1.392\lambda}{\pi K d} \quad (3.21)$$

$$\simeq \pm 0.443 \frac{\lambda}{K d}. \quad (3.22)$$

The two 3-dB points are given by

$$\sin \theta_{3dB} = \pm \frac{0.443\lambda}{K d}, \quad (3.23)$$

so the beamwidth is given by

$$\text{Beamwidth} \simeq 2 \arcsin \left( \frac{0.443\lambda}{K d} \right). \quad (3.24)$$

For  $Kd/\lambda \gg 1$  the above expression reduces to

$$\text{Beamwidth} \sim \frac{0.88\lambda}{K d} \text{ radians, or } \frac{50\lambda}{K d} \text{ degrees.} \quad (3.25)$$

An approximation for the beamwidth that is sometimes used takes the smallest angle  $\theta_s$  at which  $P(0, \theta_s) = 0$ , viz., when

$$\theta_s = \sin^{-1} \left( \frac{\lambda}{K d} \right) = \sin^{-1} \left( \frac{c}{K f d} \right), \quad (3.26)$$

where  $c$  is the velocity of wave propagation and  $f$  is frequency.

Using this expression gives the beamwidth between the points at which the beam pattern falls to  $4/\pi^2$  or  $-3.9\text{dB}$ .

Note that beamwidth is inversely proportional to the aperture measured in wavelengths. (3.26) shows that, for a fixed array geometry, the width of the main beam is narrowed as wavelength is decreased (i.e., as frequency is increased), indicating an improvement in the ability to resolve signals at higher frequencies. These effects are illustrated in Figure 3.6, where we plot beam patterns for  $d/\lambda = 0.333, 0.5, 0.833$  and  $1.0$ .  $\lambda$  cannot be decreased without penalty: for  $\lambda \geq d$  we observe “*grating lobes*”, which are discussed in Section 3.5.<sup>9</sup>

3.1.3.3. *Nulls of the beam pattern.* When the phasors sum to zero (as illustrated in Figure 3.5), the beam pattern exhibits a null, implying exact cancellation of a signal in that direction.<sup>10</sup> So if there is an unwanted signal there, no matter how strong, it will not interfere with the desired broadside signal. Thus the number and position of such nulls is of particular interest to a system designer.

From (3.17) the response is zero when

$$(\mathbf{k}_s)_y = \frac{2n\pi}{Kd}, \quad n = \pm 1, \pm 2, \dots, \quad (3.27)$$

or 
$$\theta_n = \sin^{-1} \left( \frac{n\lambda}{Kd} \right), \quad (3.28)$$

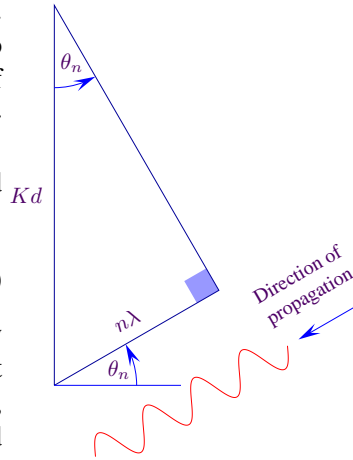
where  $\theta_n$  is the angle of the  $n^{\text{th}}$  null of the beam pattern.

The adjacent diagram provides a convenient way of remembering this formula. Whenever the total phase change due to propagation across the whole length of the array is an integral number of wavelengths, the array output is zero.

For nulls close to the main beam and  $(Kd)/\lambda \gg 1$ ,

$$\theta_n \sim \frac{n\lambda}{Kd}; \quad (3.29)$$

i.e., the first few nulls are approximately evenly spaced in angle. (This is not true in general as Figure 3.3 illustrates, but nulls *are* evenly spaced when plotted against  $(\mathbf{k}_s)_y$  or against  $\sin \theta_s$ .)



3.1.3.4. *Sidelobes.* Now consider the subsidiary peaks, termed *sidelobes* of the beam pattern. For  $K \gg 1$ , the numerator in (3.17) changes rapidly compared with the denominator and to a good approximation maxima occur when<sup>11</sup>

<sup>9</sup> $d/\lambda = 0.5$  is often referred to as the design wavelength of the array as it represents a good compromise between a narrow beamwidth and grating lobes.

<sup>10</sup>Later we shall see how this can be expressed in terms of orthogonality of vectors.

<sup>11</sup>Note that, when the beam pattern is plotted against  $(\mathbf{k}_s)_y$ , the peaks of the sidelobes lie half-way between the nulls.

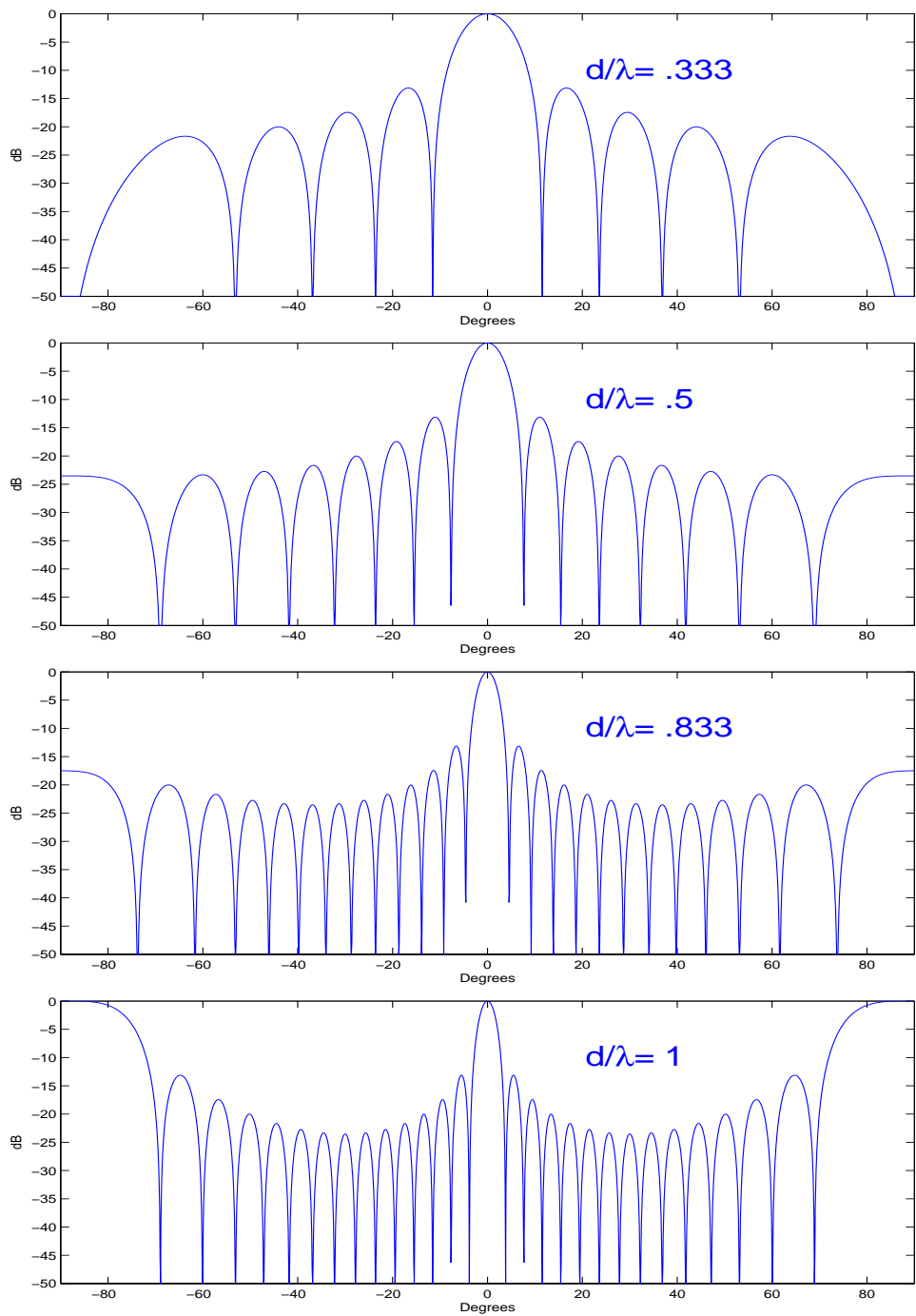


FIGURE 3.6. Beam patterns for uniform linear array of 15 receivers steered broadside, for various values of  $d/\lambda$

$$\frac{K(\mathbf{k}_s)_y d}{2} = (2n - 1)\frac{\pi}{2}, \quad n = \pm 2, \pm 3, \dots \quad (3.30)$$

or

$$(\mathbf{k}_s)_y = \frac{(2n - 1)\pi}{Kd}, \quad (3.31)$$

$$\theta_{\max, n} = \sin^{-1} \left\{ \frac{(2n - 1)\lambda}{2Kd} \right\}, \quad (3.32)$$

where  $\theta_{\max, n}$  is the angle of the  $n^{\text{th}}$  sidelobe maximum. The peaks of the sidelobes take values of:

$$\frac{1}{K^2 \left\{ \sin \left( \frac{2n-1}{2K} \pi \right) \right\}^2} \quad (3.33)$$

or approximately

$$\frac{4}{(2n - 1)^2 \pi^2}. \quad (3.34)$$

The largest sidelobe (which occurs for  $n = 2$ ) has a peak of  $4/9\pi^2$ , or  $-13.5$  dB and the sidelobe envelope falls off as approximately  $1/x^2$ . Note that these peak values are *independent of  $K$  and also of  $d/\lambda$* . However, this result ignores the effect of **grating lobes** when  $d/\lambda \geq 1$ , about which more will be said in the next section.

Ignoring grating lobes, the number of sidelobe peaks is given approximately by

$$\left[ \frac{2Kd}{\lambda} \right] - 1,$$

where  $[x]$  denotes the largest integer  $\leq x$ .

▷Problem E.2

#### 3.1.4. Grating lobes (broadside).

Consider the beam pattern of a linear array of receivers, steered at broadside, as the frequency of the incident plane-wave signal is increased (i.e., the wavelength is shortened). An example is shown in Figure 3.7 for an array of 15 receivers. At a wavelength such that  $d/\lambda = 1$ , not only is there a peak at broadside but also two others at  $\pm 90^\circ$ . At yet shorter wavelengths, many other peaks appear. Such spurious peaks are called **grating lobes**.

▷Problem E.3

▷Problem E.4

▷Problem E.5

▷Problem E.6

▷Problem E.7

▷Problem E.8

By inspection of (3.17) it is obvious that the beam pattern  $P(0, \theta_s)$  is periodic. This is a consequence of the **discrete spatial sampling** of the wave field which may, by analogy with temporal sampling of a waveform, give rise to periodic functions and hence **aliasing**. In array processing this is referred to as **spatial aliasing**.<sup>12</sup>

To understand aliasing from a physical viewpoint consider a broadside signal arriving at the array, as illustrated in Figure 3.8. In this case the spatial phase factors  $z$  are all unity  $\{1, 1, \dots, 1\}$  (i.e., the signals are **coherent**). If at the same frequency a

<sup>12</sup>By analogy with time series,  $\lambda/2$  is called the **spatial Nyquist** sampling rate.



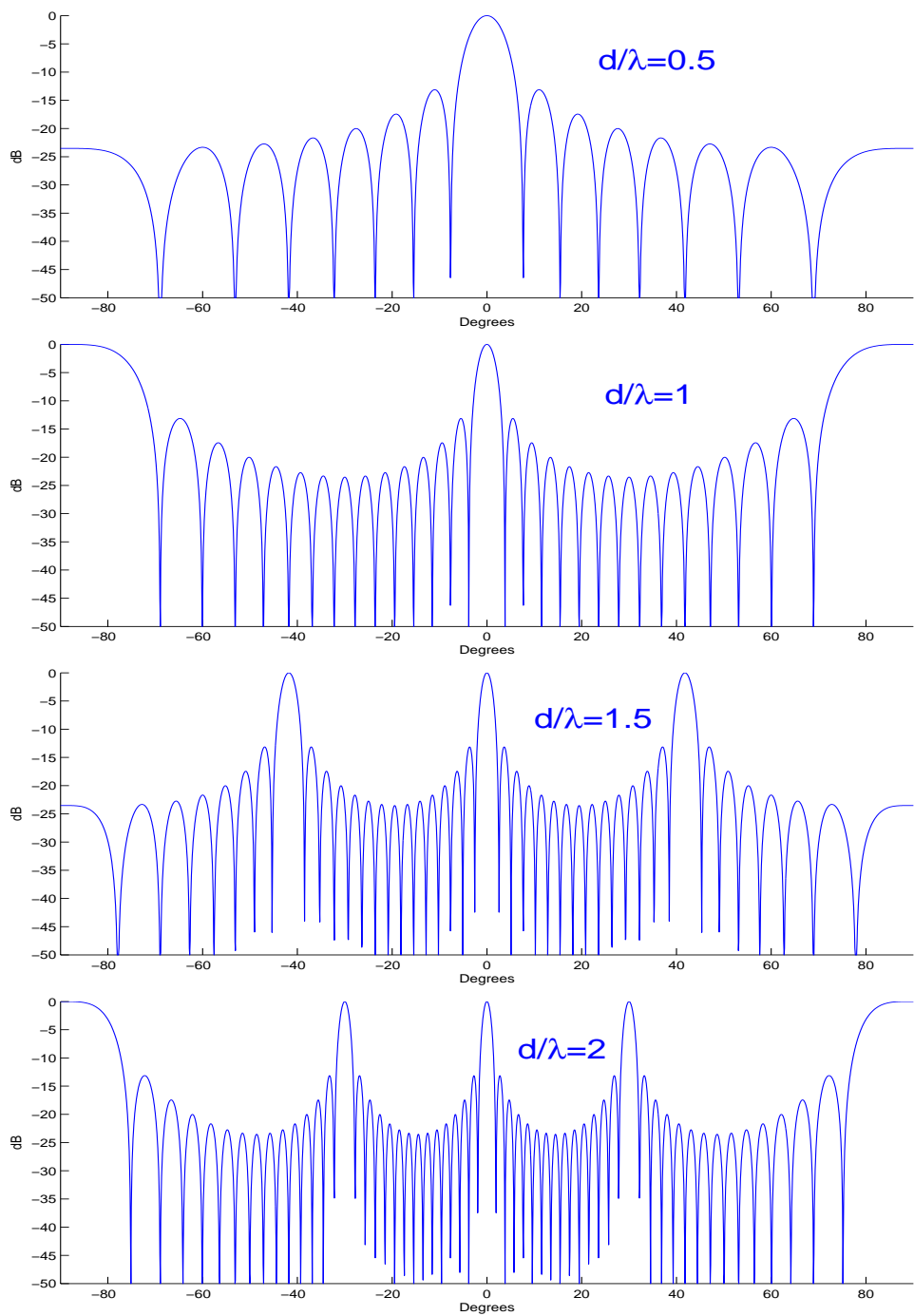


FIGURE 3.7. Illustrating grating lobes for uniform linear array of 15 receivers steered broadside.

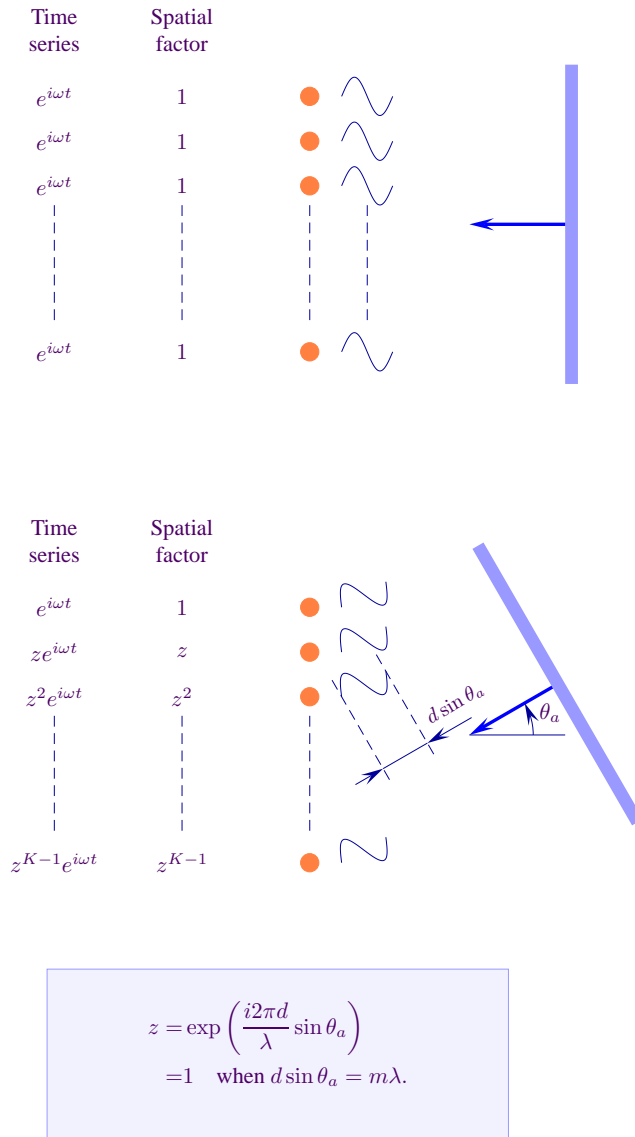


FIGURE 3.8. Illustrating spatial aliasing

signal were to arrive at the array from a direction  $\theta_a$ , the spatial phase factor would become

$$z = \exp\left(\frac{i2\pi d}{\lambda} \sin \theta_a\right).$$

When  $\sin \theta_a = \frac{m\lambda}{d}$ ,  $m = \pm 1, \pm 2, \dots$ ,  $z = 1$ , the received signals would again all be in phase and would once more be summed coherently. Thus the array would be unable to tell the difference between a signal from  $\theta_s = 0$  and a signal at  $\theta_s = \theta_a$ .

Such ambiguities manifest themselves as grating lobes and occur when the projection of the inter-element spacing  $d$  along the normal to the wavefront (i.e.,  $d \sin \theta_a$ ) is greater than  $\lambda/2$ .

Taking the broadside case and  $d/\lambda = 1$  as an example, the condition for a grating lobe is  $z = 1$ , i.e.,  $2\pi \sin \theta_a = \pm 2\pi$ , or  $\theta_a = \pm 90^\circ$ . In (3.33), the condition that  $-1 \leq \sin \theta_a \leq 1$  imposes a constraint on  $m$  that can be used to determine the number of grating lobes. For the broadside case the above constraint implies that the number of grating lobes  $= 2[d/\lambda]$ , where  $[d/\lambda]$  denotes the integer value of  $d/\lambda$ .

$d/\lambda$	No. of grating lobes	$\sin \theta_a$	$\theta_a$
1	2	$\pm 1$	$\pm 90^\circ$
2	4	$\pm 1/2, \pm 1$	$\pm 30^\circ, \pm 90^\circ$
4	8	$\pm 1/4, \pm 1/2, \pm 3/4, \pm 1$	$\pm 14.5^\circ, \pm 30^\circ, \pm 48.6^\circ, \pm 90^\circ$

Note that the grating lobes are not inter-element in angle but are inter-element in  $(\mathbf{k}_s)_y$ .

▷ Problem E.9

▷ Ex 2.4

## 3.2. Linear array, beamsteered

### 3.2.1. Concept.

Suppose it is necessary to receive a signal from some direction (say,  $\theta$ ) other than broadside, and that it is not possible to rotate the array physically. A plane wave would arrive at the receivers at different times, as shown in Figure 3.9; by delaying the output of each receiver appropriately before adding we can compensate for these differences<sup>13</sup>. The process is called **beamsteering** and  $\theta$  is the **beamsteered** direction.

For narrow-band signals, the time delays are equivalent to phase delays. Thus, rather than delay each receiver output, we could multiply it by a phase factor prior to summing. This approach to beamforming is often called **phase shift beamforming**.<sup>14</sup>

<sup>13</sup>As before, we also normalise by multiplying by  $1/K$ .

<sup>14</sup>Radar with arrays capable of being processed in this way are called phased array radars.

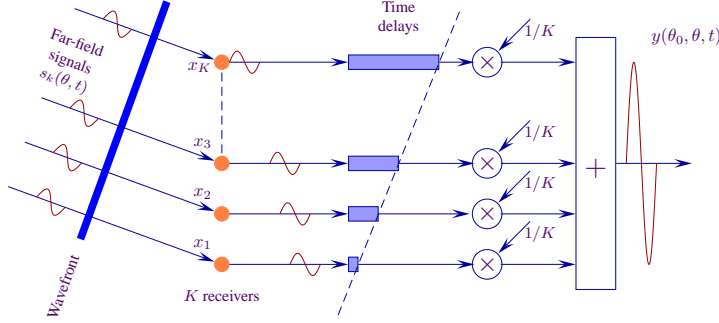


FIGURE 3.9. Beamsteering – linear array

The beamformer output,  $y(\theta, t)$ , is given by

$$y(\theta, t) = \frac{1}{K} \sum_{j=1}^K x_j(t) \exp \left( -i2\pi(j-1) \frac{d}{\lambda} \sin \theta \right) = \frac{1}{K} \mathbf{v}^H(\theta) \mathbf{x}(t) \quad (3.35)$$

▷Ex 2.5–2.6

### 3.2.2. Expression for the beam pattern.

When beamsteering in direction  $\theta$  and with a signal source incident upon the array from direction  $\theta_s$ , the output of the array is phase-shifted in one direction by the propagation delay (corresponding to  $\theta_s$ ) and in some other direction by the signal processing phase delays (corresponding to  $\theta$ ) introduced. Thus the complex beamformer output, denoted by  $y(\theta, \theta_s, t)$ , describes the output, at time  $t$ , of a beam steered in direction  $\theta$  to a plane wave incident on the array from direction  $\theta_s$ .

The vector of receiver outputs,  $\mathbf{x}(t)$  for a plane wave incident from direction  $\theta_s$  is given by

$$\mathbf{x}(t) = s(t) \mathbf{v}(\theta_s)$$

which implies that 
$$y(\theta, \theta_s, t) = \frac{1}{K} s(t) \mathbf{v}^H(\theta) \mathbf{v}(\theta_s) \quad (3.36)$$

Explicitly the complex beamformer output is

$$\begin{aligned} y(\theta, \theta_s, t) &= \frac{1}{K} s(t) \sum_{j=1}^K \exp \left( \frac{i2\pi(j-1)d}{\lambda} \sin \theta_s \right) \\ &\quad \times \exp \left( -\frac{i2\pi(j-1)d}{\lambda} \sin \theta \right) \\ &= \frac{1}{K} s(t) \sum_{j=1}^K \exp \left[ \frac{i2\pi(j-1)d}{\lambda} (\sin \theta_s - \sin \theta) \right]. \end{aligned} \quad (3.37)$$

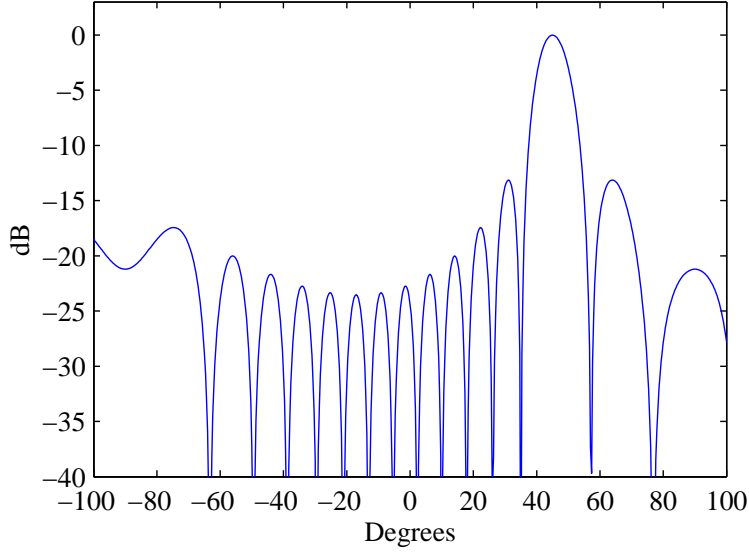


FIGURE 3.10. Beam pattern of uniform linear array of 15 receivers, beam-steered at  $45^\circ$ .  $d/\lambda = 0.5$ .

The (normalised) beam pattern is defined as before to be

$$P(\theta, \theta_s) = \frac{|y(\theta, \theta_s, t)|^2}{|y(\theta, \theta, t)|^2}. \quad (3.38)$$

which, after a manipulation similar to that in Section 3.1.2, reduces to

$$P(\theta, \theta_s) = \frac{1}{K^2} \left( \frac{\sin \left[ \frac{K\pi d}{\lambda} (\sin \theta_s - \sin \theta) \right]}{\sin \left[ \frac{\pi d}{\lambda} (\sin \theta_s - \sin \theta) \right]} \right)^2. \quad (3.39)$$

An example is shown plotted in Figure 3.10 for  $d = \lambda/2$  and  $\theta = \pi/4$ .

The right and left 3-dB points are given by:

$$\begin{aligned} \sin \theta_R &= \sin \theta + \frac{0.443\lambda}{Kd} \\ \text{and} \quad \sin \theta_L &= \sin \theta - \frac{0.443\lambda}{Kd}. \end{aligned} \quad (3.40)$$

$\theta_R$  increases with  $\theta$ ; there is no solution for  $\theta_R$  beyond

$$\sin \theta + \frac{0.443\lambda}{Kd} = 1.$$

Similarly, there is no solution for  $\theta_L$  beyond <sup>15</sup>

$$\sin \theta - \frac{0.443\lambda}{Kd} = -1.$$

<sup>15</sup>These are known as the "scan limits"[40, p.56], [11, p.125].

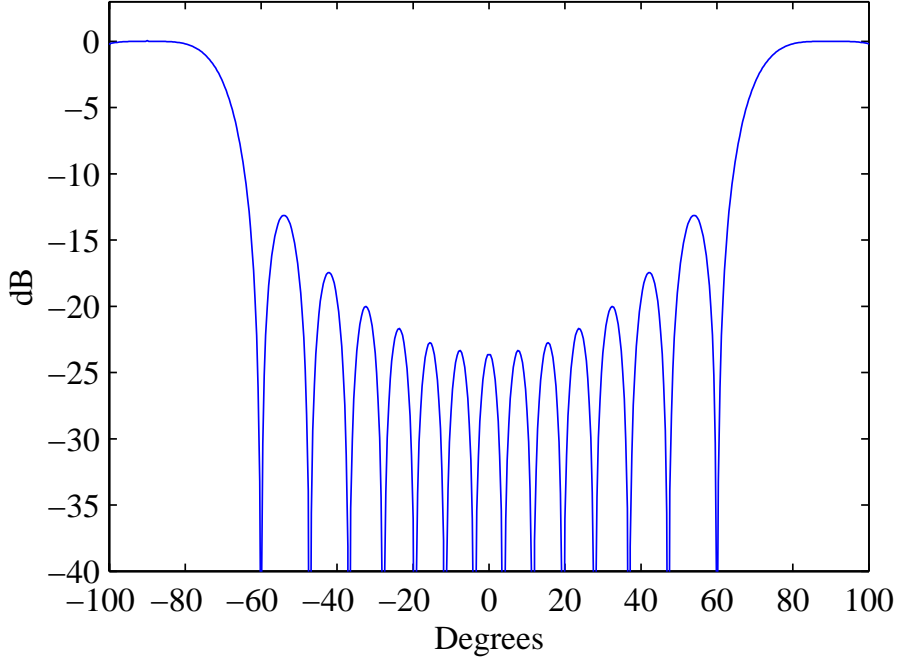


FIGURE 3.11. Illustrating grating lobes. Beam pattern of uniform linear array of 15 receivers, steered at endfire.  $d/\lambda = 0.5$

Within these bounds, the half-power bandwidth is given by

$$\begin{aligned} \text{Bandwidth} &= \theta_R - \theta_L \\ &\simeq \arcsin\left(\sin\theta + \frac{0.443\lambda}{Kd}\right) - \arcsin\left(\sin\theta - \frac{0.443\lambda}{Kd}\right). \end{aligned} \quad (3.41)$$

When  $\theta = \pm\pi/2$  we have the so-called **endfire** case. The beam pattern for  $d/\lambda = 0.5$  is shown plotted in Figure 3.11. Note that it has two main lobes – a main beam and a grating lobe – and is symmetrical about  $\theta_s = 0$ .

The 3-dB point is given by

$$\theta_R \simeq \arcsin\left(1 - 0.443\frac{\lambda}{Kd}\right) \quad (3.42)$$

$$\text{Beamwidth} \simeq \pi - 2 \arcsin\left(1 - 0.443\frac{\lambda}{Kd}\right). \quad (3.43)$$

The bandwidth between 3-dB points is plotted in Figure 3.2.2 against  $Kd/\lambda$  for several values of  $\theta$ .

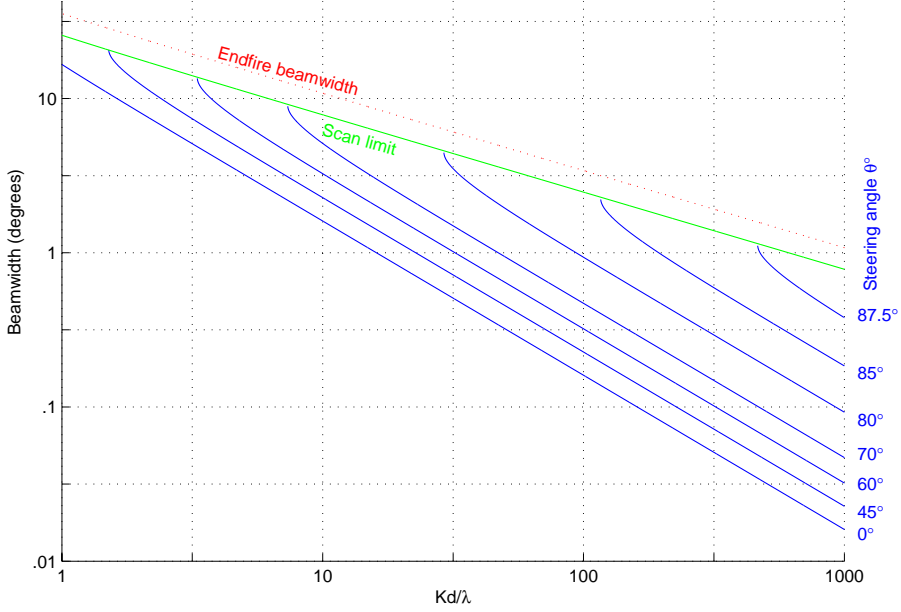
FIGURE 3.12. Beamwidth vs  $Kd/\lambda$  for various steering angles  $\theta$ 

Figure 3.13 shows another endfire beam, with  $d = \lambda/4$ . This beam pattern has a single main lobe that is broader than that of the array steered at broadside (shown in Figure 3.11).

Note that, as the beam is steered away from broadside, the beamwidth increases. This can be explained physically as due to a reduction in the effective size of the aperture.

### 3.2.3. Grating lobes (steered beams).

The general condition for grating lobes for a signal arriving from direction  $\theta_a$ , is as before

$$z = \exp\left(\frac{i2\pi d}{\lambda}(\sin \theta_a - \sin \theta)\right) = 1 \quad (3.44)$$

or<sup>16</sup>

$$\frac{2\pi d}{\lambda} \sin \theta_a = \frac{2\pi d}{\lambda} \sin \theta - 2m\pi, \quad m \neq 0. \quad (3.45)$$

As in the broadside case,  $m$  is constrained by the requirement that

$$\begin{aligned} -1 &\leq \sin \theta_a \leq 1 \\ \Rightarrow -\frac{d}{\lambda}(1 + \sin \theta) &\leq m \leq \frac{d}{\lambda}(1 - \sin \theta). \end{aligned} \quad (3.46)$$

<sup>16</sup>When  $m = 0$ ,  $\theta_a = \theta$  giving the main beam, not a grating lobe.

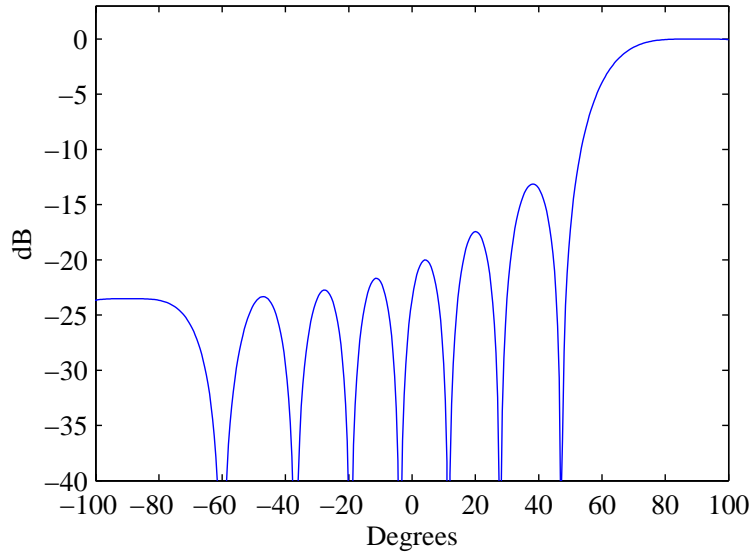


FIGURE 3.13. Beam pattern of uniform linear array of 15 receivers steered at endfire.  $d/\lambda = 0.25$

Now consider the case in which  $\theta = \pi/2$ , i.e, the endfire case. (3.45) becomes

$$\frac{d}{\lambda} \sin \theta_a = \frac{d}{\lambda} + m, \quad m \neq 0 \quad (3.47)$$

or 
$$\sin \theta_a = 1 + \frac{m\lambda}{d} \quad (3.48)$$

$$\Rightarrow -\frac{2d}{\lambda} \leq m < 0. \quad (3.49)$$

$d/\lambda$	No. of grating lobes	$\sin \theta_a$	$\theta_a$
1/2	1	-1	$-90^\circ$
1	2	0, -1	$0^\circ, -90^\circ$
3/2	3	$\pm 1/3, -1$	$\pm 19.5^\circ, -90^\circ$
2	4	$-1, \pm 1/2, 0$	$-90^\circ, \pm 30^\circ, 0^\circ$

The grating lobes for the endfire case are shown plotted in Figure 3.14.



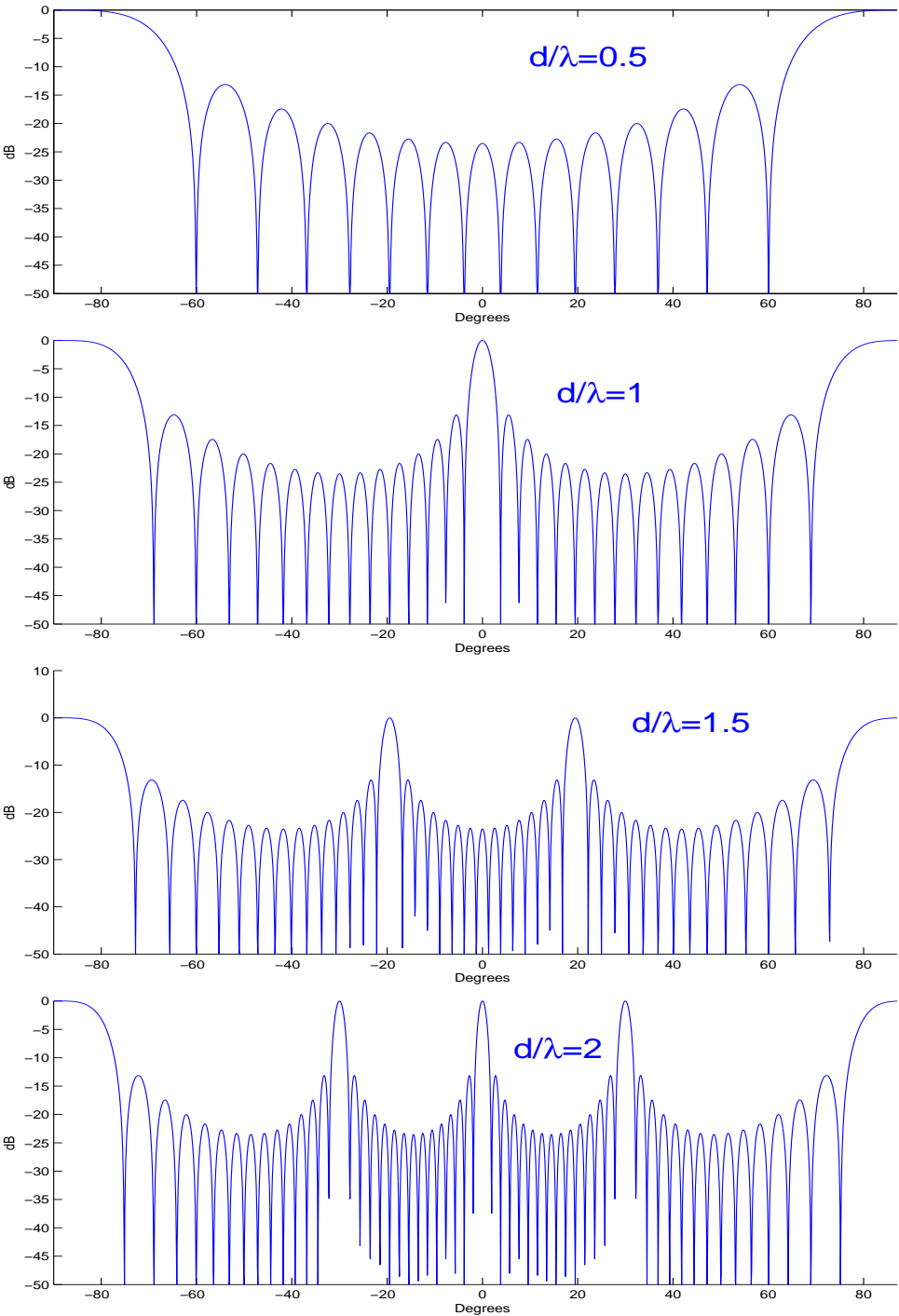


FIGURE 3.14. Illustrating grating lobes for uniform linear array of 15 receivers steered at endfire.

### 3.3. General array configuration

#### 3.3.1. Expression for the beam pattern.

By introducing the appropriate time delays to compensate for the different times of arrival, the concept of beamsteering can be extended to an array of arbitrary shape in two or three dimensions as illustrated in Figure 3.15. Again for the single-frequency plane wave considered in this Chapter the time delays can be implemented as phase shifts.

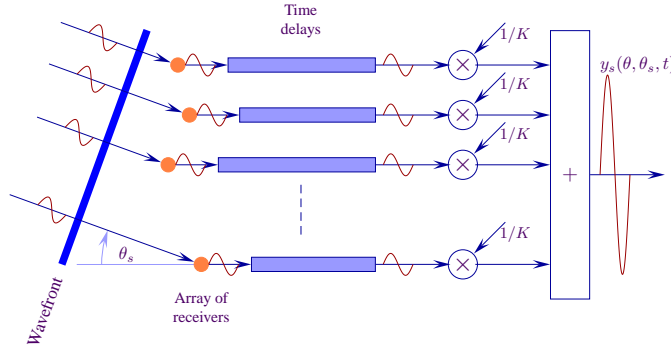


FIGURE 3.15. Beamsteering – general array geometry

To determine the values of the delays, draw an imaginary plane normal to the beamsteered direction. The actual position of the plane is arbitrary. The propagation distance to the  $j^{\text{th}}$  receiver,  $d_j$ , is given by the distance from the receiver to that plane.<sup>17</sup>

For a beamsteered direction  $(\theta, \phi)$  in azimuth and elevation, the phase delays are given by  $\mathbf{k}^T \mathbf{u}_j$ , where

$$\mathbf{k} = \frac{2\pi}{\lambda} \begin{bmatrix} \cos \theta \sin \phi \\ \sin \theta \sin \phi \\ \cos \phi \end{bmatrix}, \quad (3.50)$$

$$\mathbf{u}_j = \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix}, \quad (3.51)$$

and  $(x_j, y_j, z_j)$  are the coordinates of the  $j^{\text{th}}$  receiver.<sup>18</sup>

<sup>17</sup>Note that *any* plane normal to the beamsteered direction can be used. If the delays are implemented using some form of storage medium (e.g., a buffer storage or some delay line) it is economical to use the plane which minimises the storage requirements – i.e., the plane through the receiver furthest from the signal source.

<sup>18</sup>The reader should not confuse the  $x$ -coordinate,  $x_j$ , with the output of the receiver,  $x_j(t)$ .

The output beam,  $y(\mathbf{k}, t)$ , is then given by

$$y(\mathbf{k}, t) = \frac{1}{K} \sum_{j=1}^K x_j(t) \exp(-i\mathbf{k}^T \mathbf{u}_j) = \frac{1}{K} \mathbf{v}^H(\mathbf{k}) \mathbf{x}(t) \quad (3.52)$$

and the steering vector by

$$\mathbf{v}(\mathbf{k}) = \begin{bmatrix} \exp(i\mathbf{k}^T \mathbf{u}_1) \\ \exp(i\mathbf{k}^T \mathbf{u}_2) \\ \vdots \\ \exp(i\mathbf{k}^T \mathbf{u}_K) \end{bmatrix}. \quad (3.53)$$

An expression for the beam pattern can readily be derived by considering the output of the beamformer when the input signal is a plane wave from a direction specified by  $\mathbf{k}_s$ . In this case the vector of receiver outputs is given by

$$\mathbf{x}(t) = s(t) \mathbf{v}(\mathbf{k}_s).$$

Substituting this in (3.52), the *complex beamformer output* is then given by

$$\begin{aligned} y(\mathbf{k}, \mathbf{k}_s, t) &= \frac{1}{K} s(t) \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_s) \\ &= \frac{1}{K} s(t) \sum_{j=1}^K \exp(-i\mathbf{k}^T \mathbf{u}_j) \exp(i\mathbf{k}_s^T \mathbf{u}_j) \\ &= \frac{1}{K} s(t) \sum_{j=1}^K \exp(i(\mathbf{k}_s - \mathbf{k})^T \mathbf{u}_j) \end{aligned} \quad (3.54)$$

and hence the beam pattern is

$$\begin{aligned} P(\mathbf{k}, \mathbf{k}_s) &= \frac{|y(\mathbf{k}, \mathbf{k}_s, t)|^2}{|y(\mathbf{k}, \mathbf{k}, t)|^2} \\ &= \frac{|\mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_s)|^2}{K^2} \\ &= \frac{\left| \sum_{j=1}^K \exp\left\{i(\mathbf{k}_s - \mathbf{k})^T \mathbf{u}_j\right\} \right|^2}{K^2} \end{aligned} \quad (3.55)$$

### 3.4. Beam pattern of an array in 3 dimensions

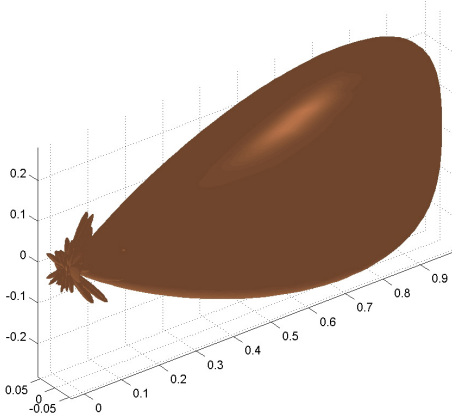


FIGURE 3.16. Beam pattern in 3 dimensions

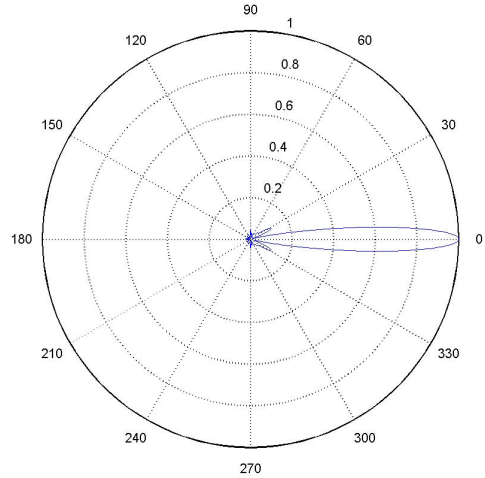


FIGURE 3.17. Horizontal beam pattern (2 dimensions)

It was mentioned earlier that the beam pattern, being a function of the horizontal ( $\theta_s$ ) and vertical ( $\phi_s$ ) angles of arrival of the far-field signal, is a 3-dimensional surface.

An example of this is shown in Figure 3.16. The horizontal beam patterns we have been considering thus far are horizontal sections through the origin (see Figure 3.17).

▷Ex 2.10–2.12

▷Problem E.10

▷Problem E.11

### 3.5. Overview

In this chapter the following concepts have been introduced

- conventional beamforming,
- two- and three-dimensional beam patterns,
- beamwidth,
- sidelobes,
- beamsteering for arrays of any configuration,
- grating lobes,

with illumination by example. The reader should now be able to calculate beam patterns and have some understanding of the dependence of the beam pattern on the operating frequency and beamsteered direction.

### Summary

- (1) Conventional beamformers delay or phase-shift receiver signals to compensate for propagation delays before summing them.
- (2) Linear arrays are inherently unable to distinguish between signals arriving from a cone centred on the array axis.
- (3) For a uniform linear array, for large  $K$ ,

$$\text{Beamwidth} \sim \frac{50}{\text{aperture in wavelengths}} \text{ degrees}$$

$$\text{Nulls occur at } \theta_n = \sin^{-1} \left( \frac{n}{\text{aperture in wavelengths}} \right)$$

$$\text{Peaks of sidelobes take values of } \frac{4}{(2n-1)^2 \pi^2}$$

which are independent of the number of receivers and the inter-element spacing.

- (4) With suitable processing, beams may be steered in any direction, for arrays of any geometry.
- (5) When the inter-element spacing is larger than the wavelength, spatial aliasing occurs which generates grating lobes.
- (6) For a uniform linear array steered broadside, the number of grating lobes  $= 2[d/\lambda]$ .
- (7) Grating lobes are not equally spaced in angle.

## CHAPTER 4

# SHAPING THE BEAM PATTERN

### 4.1. Introduction

The beam patterns in Chapter 3 exhibit substantial sidelobes. Any unwanted signal in the direction of one of these sidelobes will interfere with the wanted signal. We are thus motivated to seek means of reducing the level of sidelobes. We would also like to have some control over the width of the main lobe of the beam pattern.

Many techniques have been devised to modify the beam pattern. In the following sections we address *shading* and *null-steering* and briefly mention *superdirectivity*.

Recall from (3.52) that, for phase-shift beamforming,

$$y(\mathbf{k}, t) = \frac{1}{K} \mathbf{v}^H(\mathbf{k}) \mathbf{x}(t)$$

and for a far-field signal this reduces to

$$y(\mathbf{k}, \mathbf{k}_s, t) = \frac{1}{K} s(t) \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_s).$$

We see that the conventional beamformer multiplies the receiver outputs  $s(t)v_j(\mathbf{k}_s)$  by a factor  $1/K v_j^*(\mathbf{k}) = 1/K \exp(-i\mathbf{k}^T \mathbf{u}_j)$  before summing them.

In what follows, instead of using the conventional beamsteering vector  $\mathbf{v}(\mathbf{k})/K$ , we use a generalised weighting vector  $\mathbf{w}$

$$\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_K \end{bmatrix} \quad (4.1)$$

in order that the beam pattern might have certain desirable properties.

### 4.2. Shading

#### 4.2.1. Introduction.

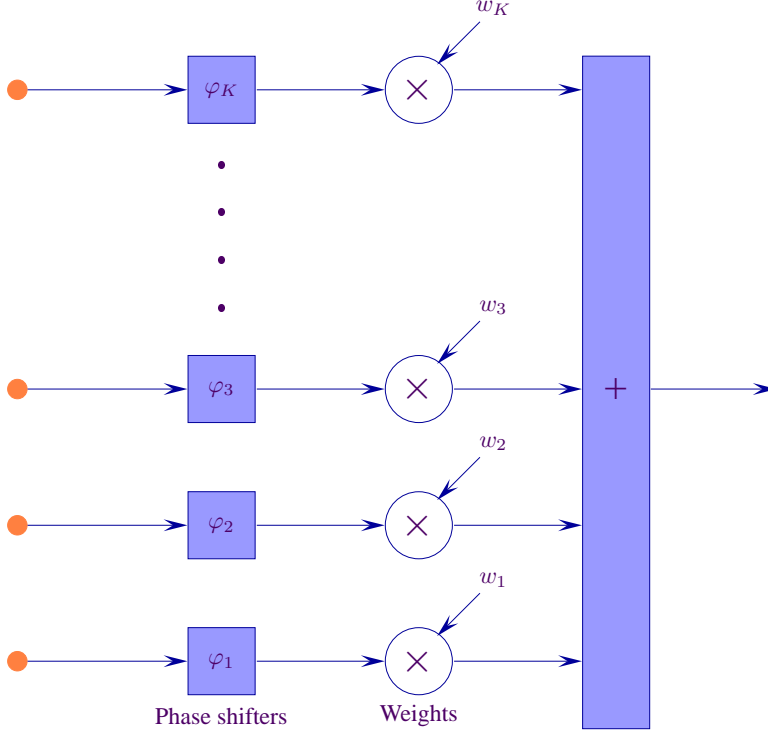


FIGURE 4.1. Illustrating shading for a uniform linear array

One simple way of improving the beam pattern is to phase-shift the receiver outputs by an amount  $\varphi_j$ , where  $\varphi = -i\mathbf{k}^T \mathbf{u}_j$  as discussed in Chapter 3, and then multiply by some real factor  $\alpha_j$ ,  $j = 1, \dots, K$  before summation. This is illustrated in Figure 4.1 for a uniform linear array. Typically the weighting factors for the receivers at the ends of the array,  $\alpha_1$  and  $\alpha_K$  would be the least, and weights would be larger for receivers nearer the centre. The technique is called **shading**<sup>1</sup>. It can be used for arrays of any configuration but the theory is most developed for linear arrays of equally spaced receivers.

Instead of beamforming using the beamsteering vector for the conventional beamformer,  $\mathbf{v}(\mathbf{k})$ , we use a general **weighting vector**  $\mathbf{w} = [w_j]$ , where

$$w_j = \alpha_j v_j(\mathbf{k}).$$

#### 4.2.2. Linear arrays.

Several popular shading schemes, with corresponding beam patterns, are shown in Table 1 and in Figures 4.2 and 4.3. In each case, the magnitudes of the weights are shown plotted alongside the beam pattern.<sup>2</sup>

<sup>1</sup>A similar technique is employed to improve the estimation of spectra from time series.

<sup>2</sup>Many of the popular shading schemes used in spectral analysis – Hann, Bartlett and Blackman, for example – reduce the weights at the extremes of the set of time samples to zero; the observation

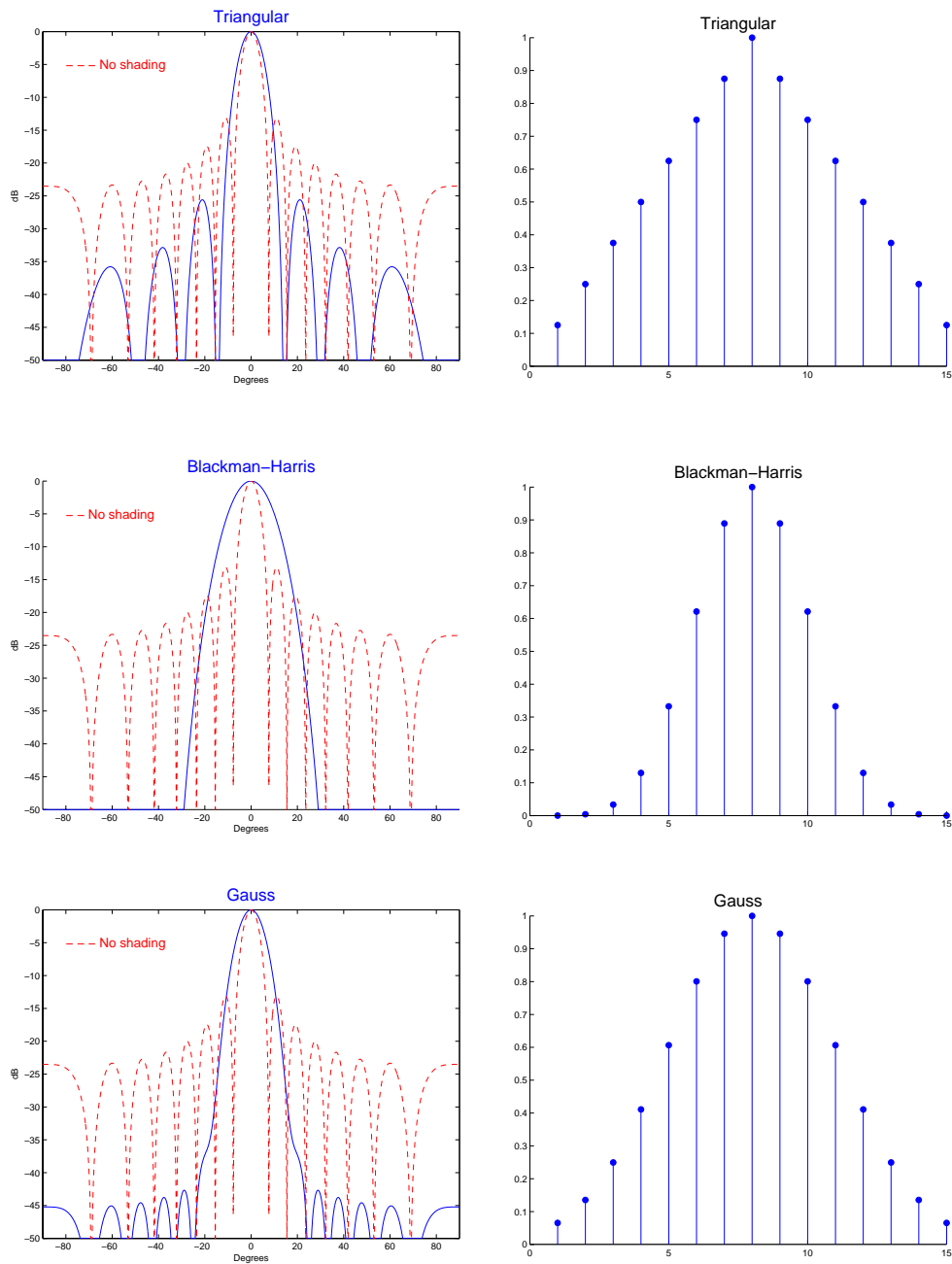


FIGURE 4.2. Beam patterns and weights with Triangular, Blackman-Harris and Gauss shading

time is effectively reduced because the first and last samples are discarded. With time sampling this is usually not of much consequence but with spatial sampling the loss of two receivers is significant. Here we modify those shading schemes by (notionally) increasing the number of receivers by two before applying the shading formula.



TABLE 1. Shading function  $w(\varrho)$ 

Type	Function
Boxcar	$w(\varrho) = 1$
Blackman-Harris	$w(\varrho) = 0.35875 + 0.48829 \cos(2\pi\varrho/D) + 0.14128 \cos(4\pi\varrho/D) + 0.01168 \cos(6\pi\varrho/D).$
Gauss	$w(\varrho) = \exp \left\{ - \left( \alpha' \varrho (K - 1) / (2KD) \right)^2 \right\}$
Hamming	$w(\varrho) = .54 + .46 \cos(\varrho)$
Hanning	$w(\varrho) = .5 \left( 1 + \cos(\varrho(K - 1) / (K + 1)) \right)$
Kaiser	$w(\varrho) = I_0 \left( \beta' \sqrt{1 - (2\varrho/D)^2} \right) / I_0(\beta')$
Triangular	$w(\varrho) = 1 - \left\{ (2\varrho(K - 1)) / (D(K + 1)) \right\}$
$\varrho$ = distance from the centre of the array $D = 2 \max(\varrho)$ $\alpha'$ and $\beta'$ adjust the trade-off between beamwidth and sidelobe levels	

In each of these beam patterns, the sidelobes have been reduced (compared to the unshaded conventional beamformer) but at the cost of increasing the width of the main lobe. By varying the shading, we can trade between sidelobe level and main lobe width.

This notion motivated the development of a processor to give a beam pattern in which each sidelobe has the same peak value. This ***Chebyshev***<sup>3</sup> processor is illustrated in Figure 4.4 for three different values of the peak sidelobe level; in each case, the beam pattern of the conventional beamformer (i.e., with no shading<sup>4</sup>) is also displayed. The trade-off between beamwidth and sidelobe level is evident. Computation of the shading coefficients is readily implemented using standard signal processing software packages.

<sup>3</sup>Also called ***Dolph-Chebyshev*** shading.

<sup>4</sup>When there is no shading—all the weights  $w(j) = 1$ —the processor is also known as ***boxcar*** shading.

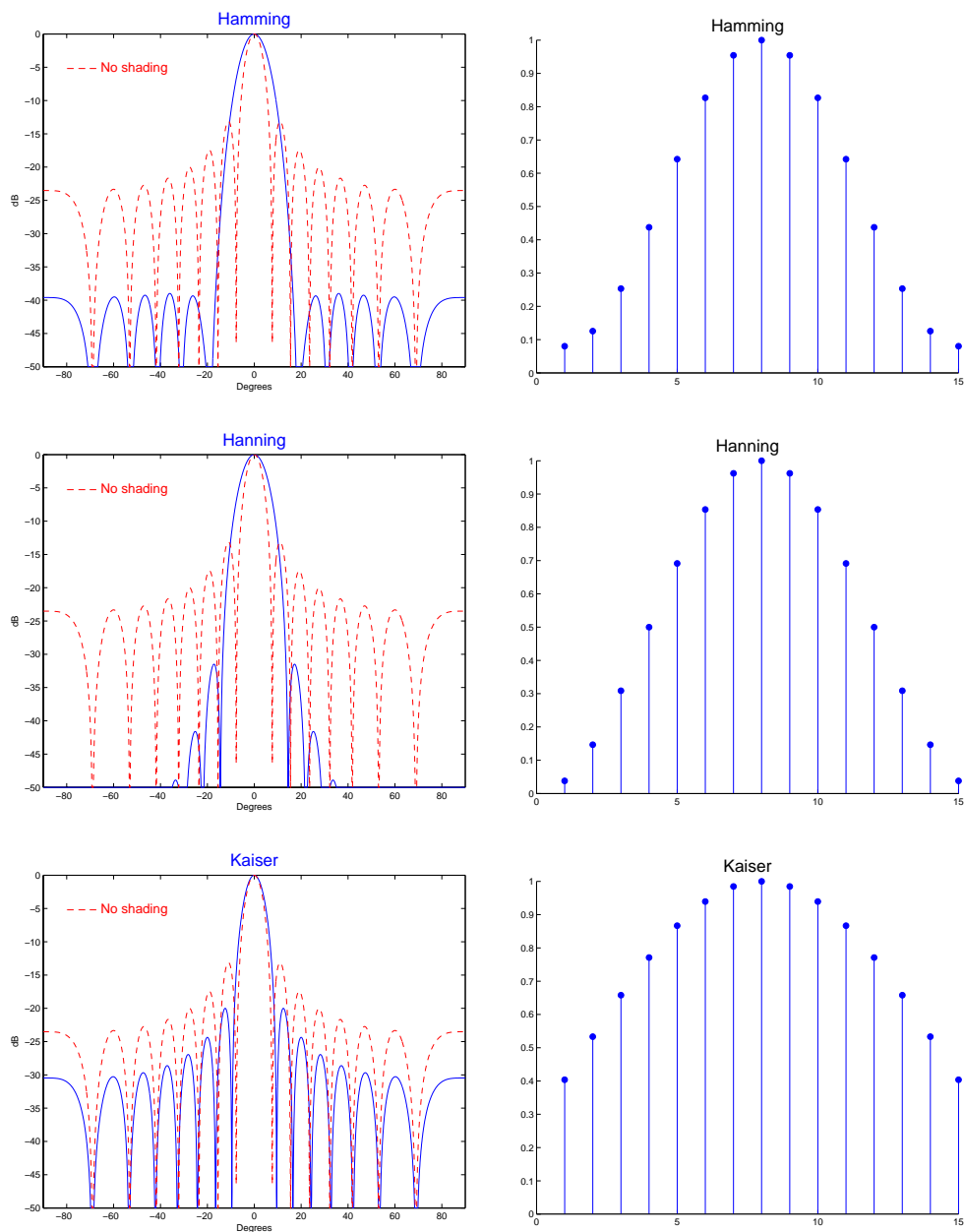


FIGURE 4.3. Beam patterns and weights with Hamming, Hanning and Kaiser shading

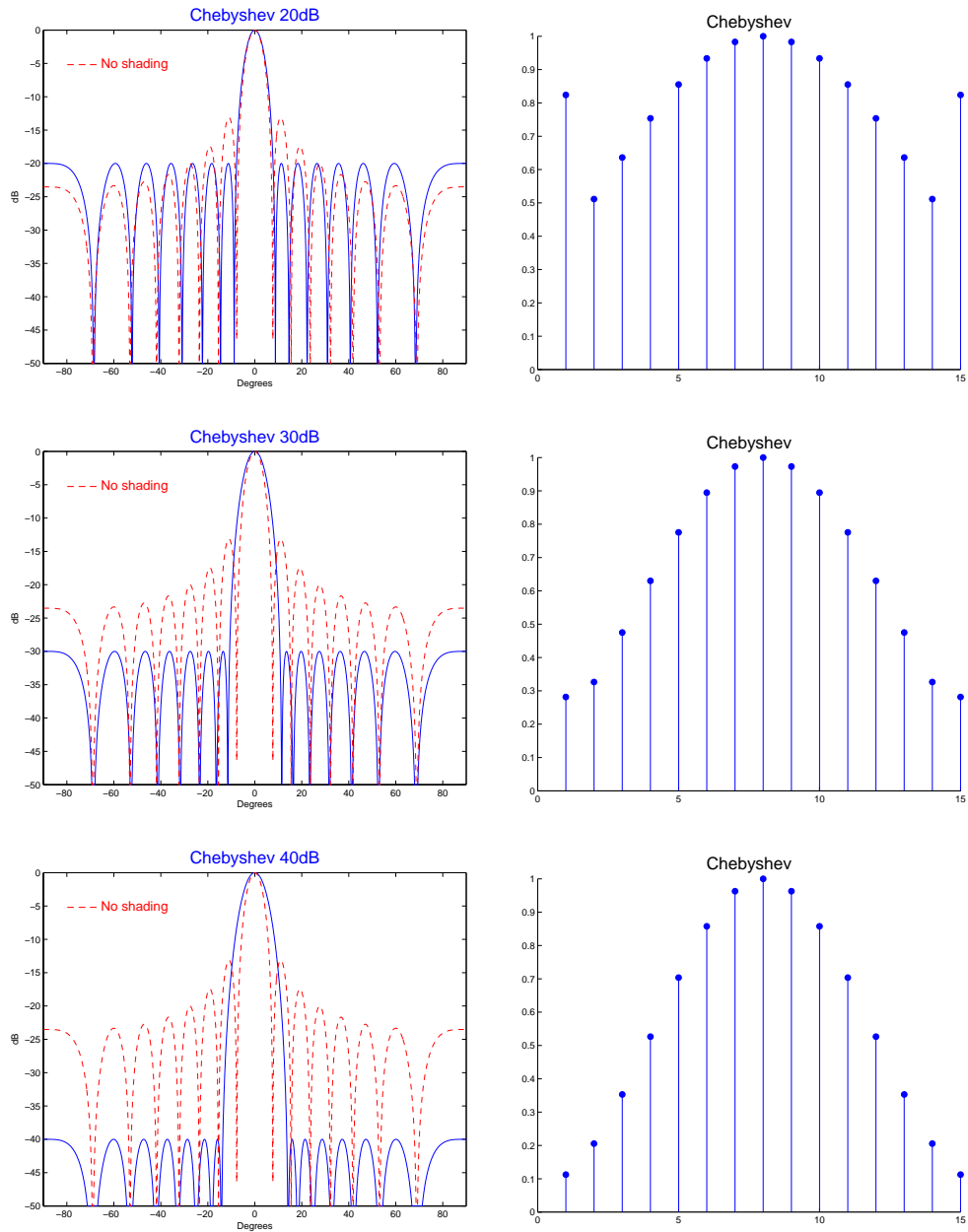


FIGURE 4.4. Dolph-Chebyshev shading and weights with side-lobes at  $-20$ ,  $-30$  and  $-40$  dB

4.2.3. Planar arrays.

For planar arrays, shading is related to the radial distance of the receiver from the centre of the array. Four examples – using Gauss, Kaiser, Hamming and Hanning shading – are given in Figure 4.5

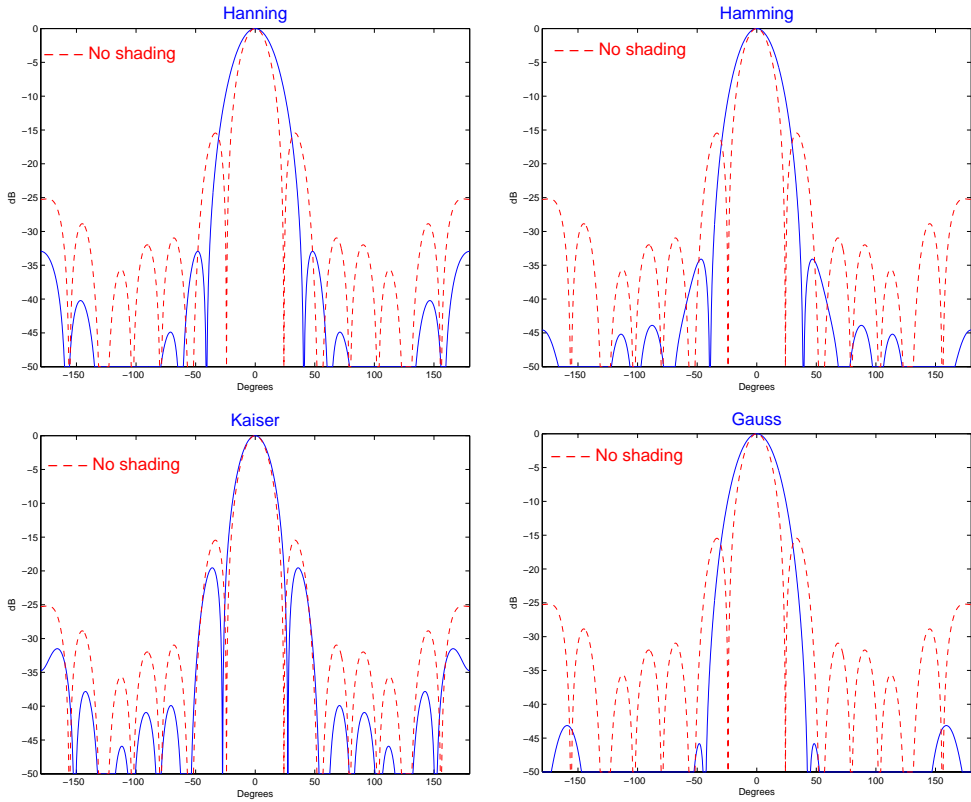


FIGURE 4.5.  $7 \times 7$  regular rectangular array with x-spacing = y-spacing =  $0.35\lambda$ . Hanning, Hamming, Kaiser and Gauss shading

▷Ex 3.1–3.2

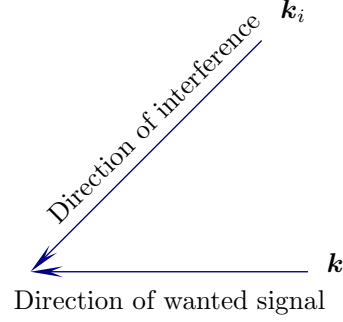
### 4.3. Null-steering

Shading techniques described previously reduce sidelobes in general and are effective where there are many interfering sources distributed around the array. However, very often one is faced with interference, whether accidental or deliberate, from *discrete* far-field sources. It is then desirable to generate a beam pattern with nulls in the directions of the interfering sources<sup>5</sup>.

<sup>5</sup> The concept can readily be extended to sources that are close by—i.e., when the interference arrives in a wavefront that is not plane.

### 4.3.1. Single null.

We begin by considering a beamformer which generates a single null and **later** see how we can generate many nulls. Let  $\mathbf{w}$  be the vector of weights for the beamformer (which we wish to derive),  $\mathbf{v}(\mathbf{k})$  the steering vector corresponding to the direction  $\mathbf{k}$  of the desired signal and  $\mathbf{v}(\mathbf{k}_i)$  that corresponding to the interference direction  $\mathbf{k}_i$ . As usual, we want the summed output of the beamformer to be unity in the direction of the wanted signal and zero in the direction ( $\mathbf{k}_i$ ) of the interference.



In other words, we need simultaneously to satisfy the following two equations:

$$\mathbf{w}^H \mathbf{v}(\mathbf{k}) = 1 \quad (4.2)$$

$$\text{and} \quad \mathbf{w}^H \mathbf{v}(\mathbf{k}_i) = 0. \quad (4.3)$$

We can combine (4.2) and (4.3) into a single equation:

$$\begin{bmatrix} \mathbf{v}^H(\mathbf{k}) \mathbf{w} \\ \mathbf{v}^H(\mathbf{k}_i) \mathbf{w} \end{bmatrix} = \begin{bmatrix} \mathbf{v}^H(\mathbf{k}) \\ \mathbf{v}^H(\mathbf{k}_i) \end{bmatrix} \mathbf{w} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (4.4)$$

Let us define the  $(2 \times K)$  matrix

$$\mathbf{A} = \begin{bmatrix} \mathbf{v}^H(\mathbf{k}) \\ \mathbf{v}^H(\mathbf{k}_i) \end{bmatrix} = [\mathbf{v}(\mathbf{k}) \mathbf{v}(\mathbf{k}_i)]^H. \quad (4.5)$$

Substituting into (4.4) we have

$$\mathbf{A} \mathbf{w} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (4.6)$$

This represents two equations with  $K$  unknowns. For  $K > 2$  there is no unique solution.<sup>6</sup> Let us consider one very specific solution, where the weight vector is a combination of the signal and interference vectors:

$$\mathbf{w} = p_1 \mathbf{v}(\mathbf{k}) + p_2 \mathbf{v}(\mathbf{k}_i) = \mathbf{A}^H \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}. \quad (4.7)$$

Pre-multiplying by  $\mathbf{A}$  and applying the constraint (4.6), we have

---

<sup>6</sup>Except for the special case in which  $\mathbf{A}$  is of rank 2—i.e., it has only two non-zero eigenvalues.

$$\mathbf{A} \mathbf{w} = \mathbf{A} \mathbf{A}^H \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (4.8)$$

Note that

$$\begin{aligned} \mathbf{A} \mathbf{A}^H &= \begin{bmatrix} \mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}) & \mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_i) \\ \mathbf{v}^H(\mathbf{k}_i)\mathbf{v}(\mathbf{k}) & \mathbf{v}^H(\mathbf{k}_i)\mathbf{v}(\mathbf{k}_i) \end{bmatrix} \\ &= \begin{bmatrix} K & \mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_i) \\ \mathbf{v}^H(\mathbf{k}_i)\mathbf{v}(\mathbf{k}) & K \end{bmatrix} \end{aligned} \quad (4.9)$$

and hence, if the inverse exists,

$$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = (\mathbf{A} \mathbf{A}^H)^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (4.10)$$

Substituting (4.10) in (4.7) we obtain<sup>7</sup>

$$\mathbf{w} = \mathbf{A}^H \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \mathbf{A}^H (\mathbf{A} \mathbf{A}^H)^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \triangleq \mathbf{A}^\dagger \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (4.11)$$

It is obvious from direct substitution that  $\mathbf{w}$  satisfies (4.4) and (4.6). This particular solution of (4.6) is called the **minimum-norm solution**.

We can gain a little insight by calculating  $(\mathbf{A} \mathbf{A}^H)^{-1}$ . After a little manipulation of (4.10) we obtain

$$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = (\mathbf{A} \mathbf{A}^H)^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{K}{K^2 - |\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_i)|^2} \begin{bmatrix} 1 \\ -\left\{ \frac{\mathbf{v}^H(\mathbf{k}_i)\mathbf{v}(\mathbf{k})}{K} \right\} \end{bmatrix}. \quad (4.12)$$

---

<sup>7</sup>There are infinitely many solutions to (4.4). The general solution may be written as

$$\mathbf{w} = \mathbf{A}^- \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

where  $\mathbf{A}^-$  denotes a so-called **generalised inverse** or **g-inverse** of the matrix  $\mathbf{A}$ . Any g-inverse will satisfy (4.4).  $\mathbf{A}^\dagger$  is called the **Moore-Penrose generalised inverse**, or **pseudo-inverse** of  $\mathbf{A}$  and is the one which minimises the **Euclidean norm** ( $\mathbf{w}^H \mathbf{w} = \sum_{j=1}^K |w_j|^2$ ). In one sense, it gives the null-steering processor that is least sensitive to noise and errors in the beamforming. A general solution for  $\mathbf{w}$  is  $\mathbf{w} = \mathbf{A}^\dagger \begin{bmatrix} 1 \\ 0 \end{bmatrix} + (\mathbf{I} - \mathbf{A}^\dagger \mathbf{A}) \mathbf{z}$ , where  $\mathbf{z}$  is arbitrary. When  $\mathbf{z} = \mathbf{0}$  we have the minimum-norm solution.

Note that the second entry,  $p_2$ , is proportional to the conventional beamformer output in the direction  $\mathbf{k}_i$ . In effect we are subtracting the signal entering via a sidelobe, and then renormalising the beamformer response.

The beam pattern of an array with a single steered null is illustrated in Figure 4.6.

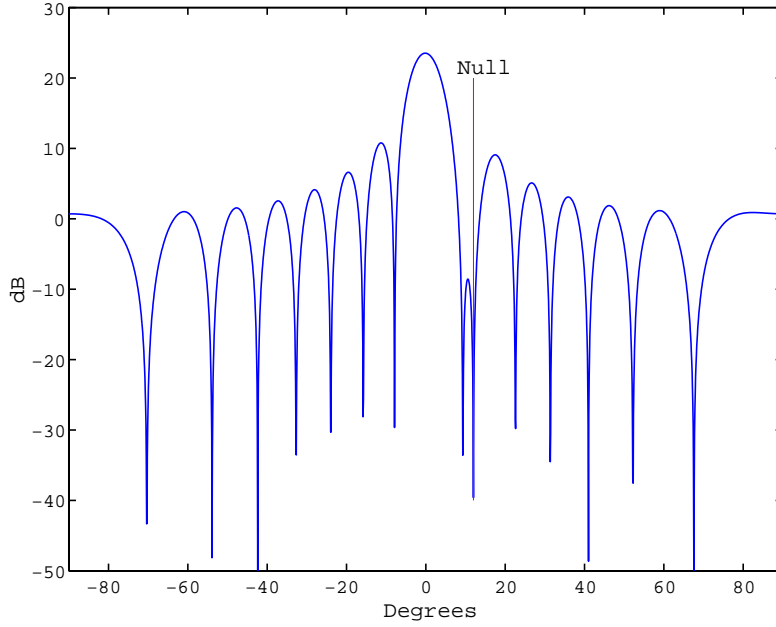


FIGURE 4.6. Beam pattern of array with single steered null

▷ Problem E.12

#### 4.3.2. $L < K$ nulls.

It is not difficult to extend these results to generate a beam pattern with  $L < K$  nulls, by defining the  $((L + 1) \times K)$  matrix

$$\mathbf{A} = \begin{bmatrix} \mathbf{v}^H(\mathbf{k}) \\ \mathbf{v}^H(\mathbf{k}_1) \\ \vdots \\ \mathbf{v}^H(\mathbf{k}_L) \end{bmatrix} \quad (4.13)$$

and solving

$$\mathbf{A}\mathbf{w} = \boldsymbol{\delta}_1^{(L+1)}, \quad (4.14)$$

where  $\boldsymbol{\delta}_1^{(L+1)}$  is the  $((L + 1) \times 1)$  vector:

$$\boldsymbol{\delta}_1^{(L+1)} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (4.15)$$

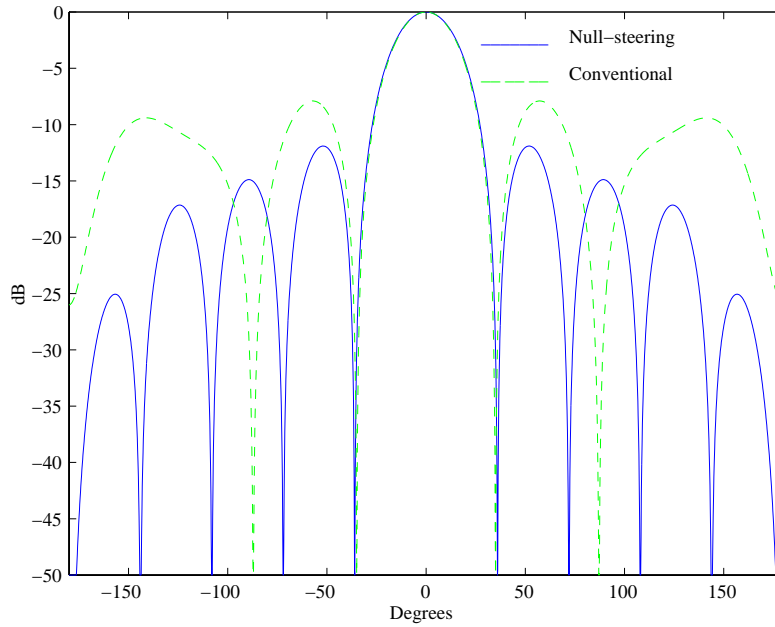


FIGURE 4.7. Beam pattern of circular array with nine steered nulls

When  $(\mathbf{A} \mathbf{A}^H)$  is non-singular, the minimum-norm solution is

$$\mathbf{w} = \mathbf{A}^H (\mathbf{A} \mathbf{A}^H)^{-1} \boldsymbol{\delta}_1^{(L+1)}. \quad (4.16)$$

The limit is reached when  $L = K - 1$ ; if  $\mathbf{A}$  is non-singular, there is then a unique solution:

$$\mathbf{w} = \mathbf{A}^{-1} \boldsymbol{\delta}_1^{(K)}. \quad (4.17)$$

Figure 4.7 shows the beam pattern of a 10-element circular array of receivers spaced  $0.4\lambda$  apart, with both conventional beamforming and with 9 steered nulls. It is apparent that with null-steering the beam pattern is significantly improved over that of the conventional beamformer.

However, this technique must be applied judiciously. Figure 4.8 shows the beam pattern of an array with two nulls placed too close to the beamsteered direction. The width of the main lobe is made smaller compared to that of the conventional beamformer, but at the cost of an increase in the levels of all of the sidelobes.

▷ Problem E.13

▷ Ex 3.3–3.4

#### 4.4. Superdirectivity

With conventional beamforming, the width of the beam between half-power points is given approximately by

$$\text{beamwidth} \cong 0.88 \lambda / \text{aperture radians}, \quad (4.18)$$



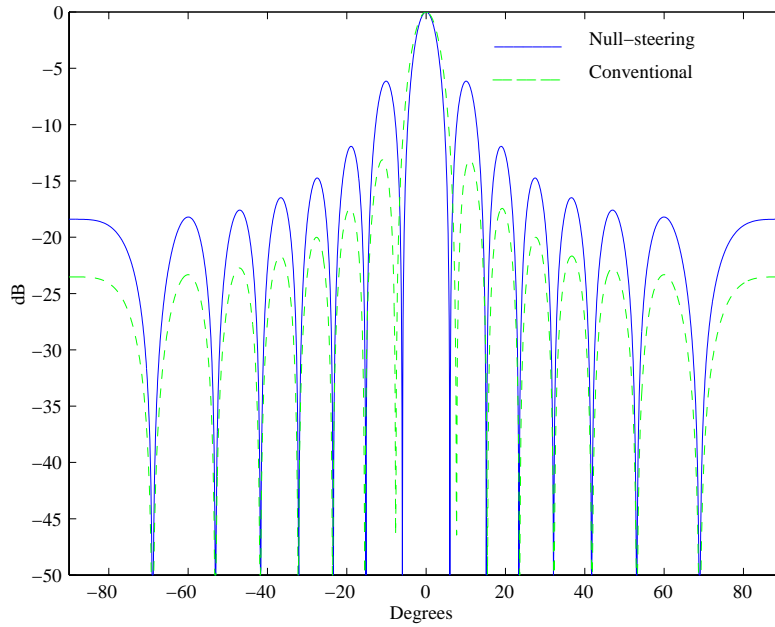


FIGURE 4.8. Beam pattern of linear array with two steered nulls

where *aperture* is the width of the aperture of the array and  $\lambda$  is the wavelength. At low frequencies,  $\lambda$  is large and so the beamwidth is as well.

In previous sections we have been concerned with suppressing sidelobes. It is also possible to process receiver outputs in order to make the beamwidth narrower than that from conventional beamforming.

Beamformers that give narrower beams than does the conventional processor are called *superdirective*.

Figure 4.9 shows the beam pattern of a so-called *minimum bias beamformer*[9] for a linear array of 15 elements, steered broadside and with elements spaced  $\lambda/4$  apart. Also shown for comparison is the beam pattern for a conventional beamformer. Note that the beamwidth of the minimum bias beamformer is narrower than that of the conventional. Not unexpectedly, there is a price to pay for any of these superdirective processors: the processing is more complex and the magnitude of the weights tends to be large. In the particular case shown in Figure 4.9, the ratio of the absolute value of the maximum to minimum weight is  $\cong 260$ . The result is that the processor is more sensitive than the conventional beamformer to noise and any imperfections in the processing or in the assumed model.

In some cases, narrowing the main lobe generates higher sidelobes. As a general rule, the higher the performance demanded, the more sensitive is the processor.

There will be further examples of superdirectivity in later chapters.

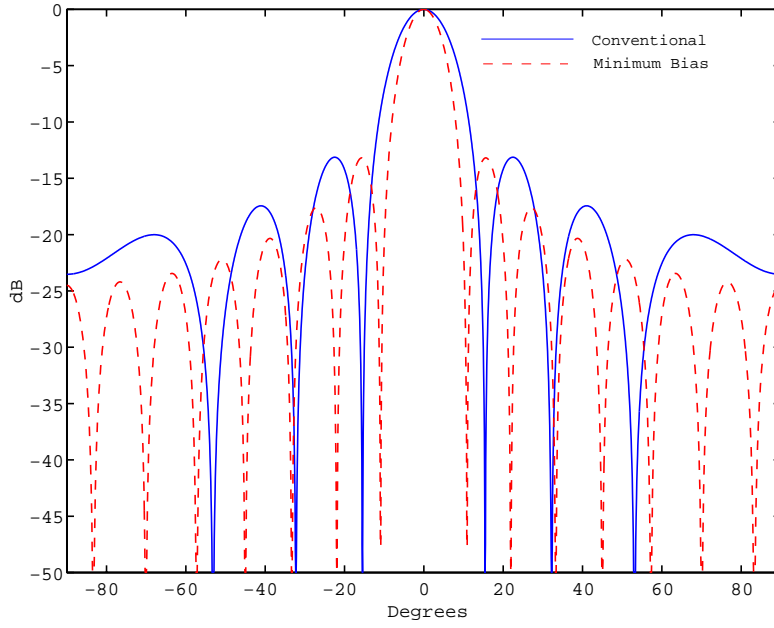


FIGURE 4.9. Illustrating superdirectivity

### 4.5. Overview

In this chapter we have introduced several techniques for using weights to shape the beam pattern : shading, null-steering and superdirectivity. Shading reduces the level of sidelobes, null-steering places up to  $K - 1$  nulls in the beam pattern, and superdirectivity reduces the width of the main lobe.

### Summary

- (1) *Shading* is a technique which multiplies the delayed receiver outputs by real weights in order to suppress sidelobes. In general, the magnitude of weights is larger for receivers near the centre of the array than for those at the periphery.
- (2) There are many shading schemes available and expressions have been given for some popular ones.
- (3) *Chebyshev* shading gives a beampattern with all sidelobe peaks of the same amplitude.
- (4) With shading, the penalty for low sidelobes is an increase in the beamwidth.
- (5) Interfering signals from discrete directions can be cancelled completely (in theory at least) by *null-steering*. For an array of  $K$  receivers up to  $K - 1$  nulls can be steered.
- (6) At low frequencies, where the beampattern for conventional beamformers is broad, it is possible to get narrower beamwidths by *superdirective* beamforming.

## CHAPTER 5

# ARRAY GEOMETRY DESIGN

### 5.1. Introduction

The geometry of the array plays a dominant part in the performance of the system. For example, the beam pattern is determined entirely by the geometry and the wavelength of the signal. We have seen how a linear array is unable inherently to distinguish between signals arriving from anywhere within a cone centred on the axis of the array.

In this Chapter we explore some of the effects arising from the geometry of the array – some desirable and others not so.

It must be stressed that the design of the geometry is inevitably a compromise between many factors – such as physical size, mechanical construction, cost, computing load, beamwidth and sidelobe levels. These frequently override signal processing criteria and force the adoption of a less than optimal array configuration.

### 5.2. Array symmetry

Here the effect of symmetry on the beam pattern of an array is considered qualitatively. It has already been noted that a linear array – a highly symmetrical structure – has a beam pattern with two main lobes symmetrical about the array axis, as illustrated in Figure 3.2.

If an array has a high degree of symmetry – for example, if it is triangular as illustrated in Figure 5.1 – then it will tend to have higher sidelobes in directions corresponding to the ambiguous lobes of each of the sides. This is illustrated in Figure 5.2 which shows the beam patterns of three sub-arrays (i.e., of each of the three sides of the array of Figure 5.1), and of the whole array of 30 receivers.

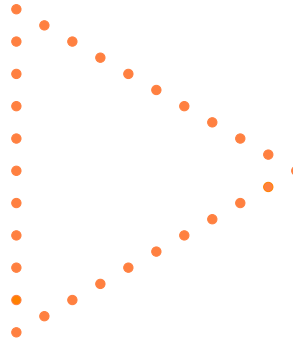


FIGURE 5.1. Example of a highly symmetrical array

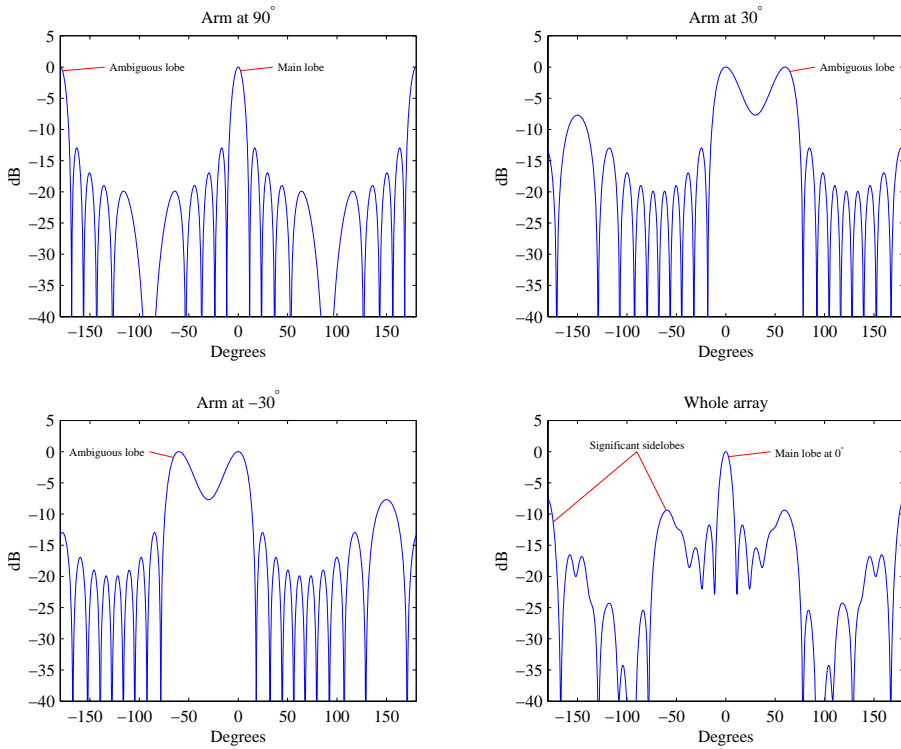


FIGURE 5.2. Illustrating how the ambiguous lobes of sub-arrays affect the beam pattern

The spacing between adjacent receivers is  $\lambda/2$  and the beamsteered direction is  $0^\circ$ . The sub-arrays have spurious lobes at  $180^\circ$  and  $\pm 60^\circ$  which result in significant sidelobes in the beam pattern of the whole array. Qualitatively the height of these sidelobes would be of the order of  $\left(\frac{1}{3}\right)^2$  or approximately -10dB.

It must be stressed that the beam pattern for the whole array is *not* the sum of the beam patterns of the individual sub-arrays. Each sub-array can be considered separately but to obtain the output of the whole array the sub-array outputs with *appropriate phase delays* should be added. Nevertheless, if the beam pattern of a sub-array has a large unwanted sidelobe, it indicates a predisposition to there being a significant sidelobe in the beam pattern of the whole array. As a general rule, it is desirable to avoid symmetry in array geometry (an extreme example is to spread the receivers randomly), but practical considerations frequently override this desire.

▷ Problem E.14

▷ Ex 4.1–4.4

### 5.3. Beam pattern of an array of directional receivers

Thus far it has been assumed that each receiver is omnidirectional – i.e., that it is by itself insensitive to the direction of arrival of any signal. In practice this is often not so: for example, a vertical radio dipole has a response that varies with angle in the vertical plane.

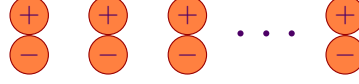


FIGURE 5.3. Array of horizontal dipoles

Consider the beam pattern of an array of *identical* directional receivers each oriented in the same direction, as illustrated in the sketch. Let  $A_s^{\text{rec}}(\mathbf{k}_s)$  be the complex response of each receiver to a signal of wavenumber  $\mathbf{k}_s$ . The vector of receiver outputs when a plane wave is incident upon the array from direction  $\mathbf{k}_s$  is then given by

$$\mathbf{x}(t) = s(t)A_s^{\text{rec}}(\mathbf{k}_s)\mathbf{v}(\mathbf{k}_s) \quad (5.1)$$

Then the output of a beamsteered in direction  $\mathbf{k}$  is then given by

$$y(t) = \frac{1}{K}s(t)A_s^{\text{rec}}(\mathbf{k}_s)\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_s) \quad (5.2)$$

The (normalised) beam pattern of the directional array is therefore given by

$$P^{\text{dir}}(\mathbf{k}, \mathbf{k}_s) = \frac{|\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_s)|^2}{K^2} \frac{|A_s^{\text{rec}}(\mathbf{k}_s)|^2}{|A_s^{\text{rec}}(\mathbf{k})|^2}. \quad (5.3)$$

and so

$$P^{\text{dir}}(\mathbf{k}, \mathbf{k}_s) = P_s^{\text{rec}}(\mathbf{k}, \mathbf{k}_s)P_s^{\text{omni}}(\mathbf{k}, \mathbf{k}_s). \quad (5.4)$$

This is the very useful **principle of beam pattern multiplication**.

*The beam pattern of an array of identical directional receivers is the product of the beam pattern of an array of omnidirectional receivers, with the same geometry, and the beam pattern of the receivers themselves.*

There are many instances where this principle is used. In radio astronomy, arrays of large identical dishes are used, as sketched in Figure 5.4; each dish has the same beam pattern and is pointed in the same direction. The spacing of the dishes is deliberately made large so as to obtain very high resolution. If omnidirectional receivers were used there would be grating lobes. However, the beam pattern of the dishes serves to suppress the grating lobes. This effect is illustrated in Figure 5.5 for a linear array of 15 inter-element dipoles steered at broadside, with each dipole oriented so that its maximum response is aligned with the beamsteered direction (but note that in radio astronomy the beamwidths would be much narrower than shown.)

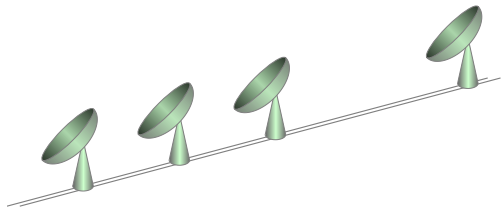


FIGURE 5.4. Array of dishes

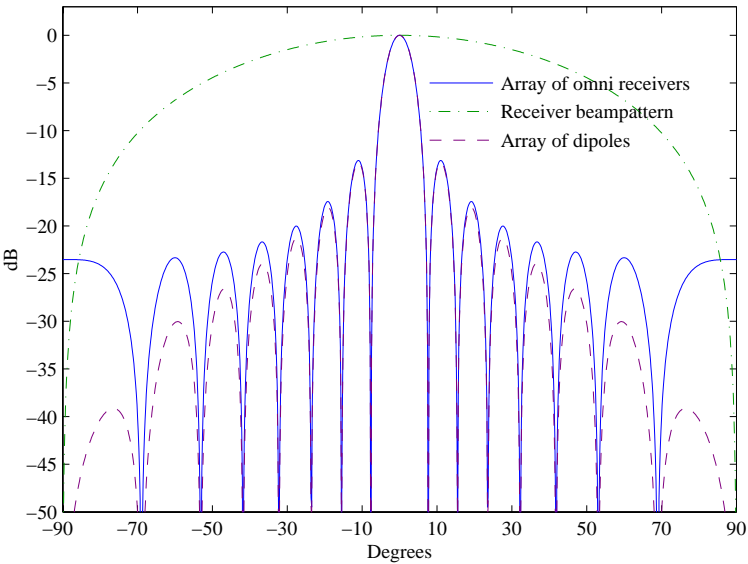


FIGURE 5.5. Beam pattern of linear array of dipoles

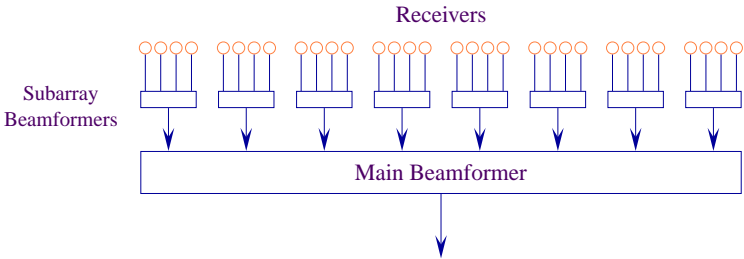


FIGURE 5.6. Partitioning into sub-arrays

In HF radar, very large linear arrays are used. To reduce the cost of beamforming, it is possible to partition the array into small identical *sub-arrays* and to form beams

for each of them. Each sub-array may then be thought of as a separate directional receiver.

A similar approach is also used in sonar which utilises towed arrays of hydrophones.



FIGURE 5.7. Towed array of groups of hydrophones

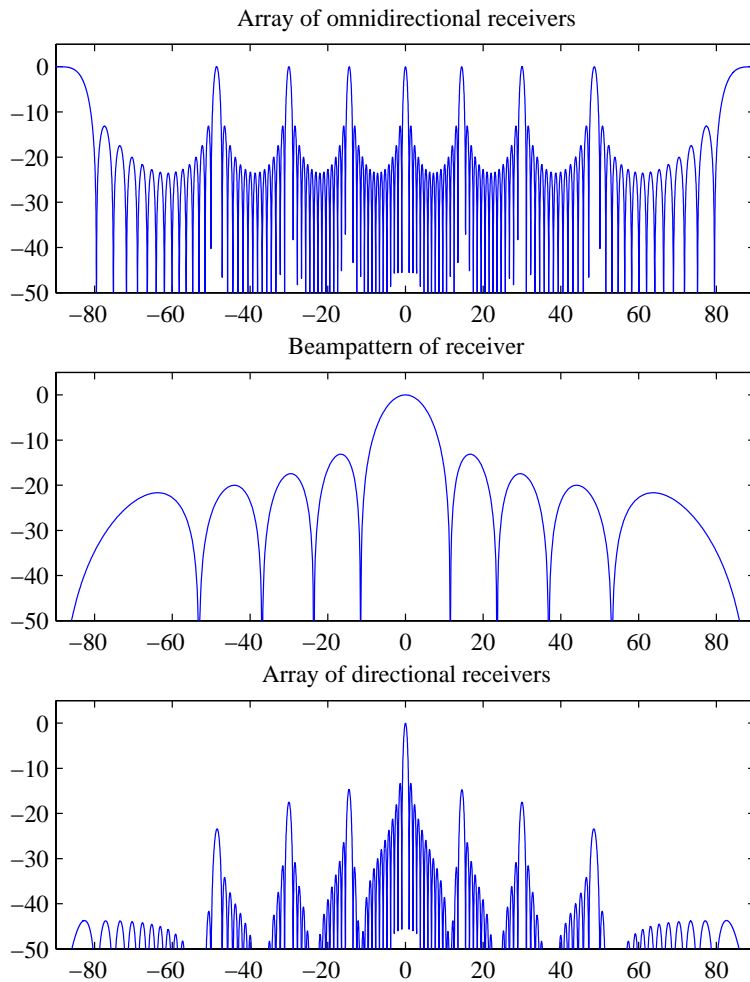


FIGURE 5.8. Suppressing grating lobes using receiver directivity

In this case, the hydrophones are often arranged in groups and their outputs added to reduce the acoustic flow noise. Each group has a broadside beam pattern (albeit usually rather broad). Beamforming is then done using each group as if it were a single (directional) receiver.

▷ Problem E.15

▷ Problem E.16

▷ Ex 4\_5–4\_6



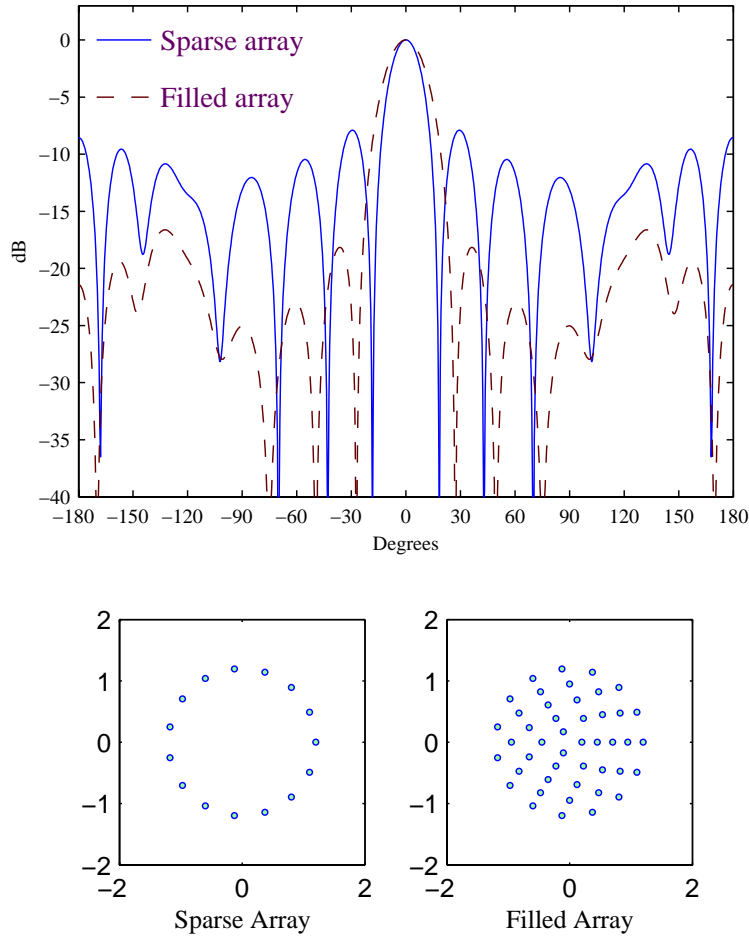


FIGURE 5.9. Beam pattern of circular arrays – sparse and filled

### 5.4. Sparse arrays

Thus far we have been considering arrays with receivers that are distributed to fill evenly the space between the extremes of the array. Such *filled arrays* give a beam pattern with relatively narrow main lobe and low sidelobes.

The width of the main lobe is determined primarily by the array *aperture* of the array (its spatial extent)<sup>1</sup>. However, the computational cost of beamforming varies approximately as the cube of the number of receivers. Thus there is naturally interest in the search for arrays with the same aperture but with fewer receivers. Such *sparse arrays* have been extensively studied.

<sup>1</sup>Let the shortest spacing between receivers be  $d$ , the distance between receivers at the extremes of the array be  $D$ , and the wavelength  $\lambda$ ; then the aperture – in units of wavelength – is normally taken to be  $(D + d)/\lambda$ .

Figure 5.9 shows the horizontal beam pattern  $P(0, \theta)$  of a plane array with 15 receivers arranged uniformly on a circle of radius  $1.2\lambda$  and with adjacent elements spaced close to  $\lambda/2$  apart. The beamsteered direction  $\theta$  is on a radial line in the  $x$ -direction.

Note that the level of sidelobes is much higher than for the linear array. This is because the array is relatively sparse (i.e., the number of receivers within the area or volume spanned by the array is relatively small); there are no receivers within the circle.

Also in Figure 5.9 is the beam pattern of the same array filled with additional elements arranged in concentric circles, with radii of reduced in steps of  $0.25\lambda$ . It is evident that the beam pattern is much improved, but at the cost of a three-fold increase in the number of elements. The beam pattern of such circular arrays is substantially independent of the beamsteered direction.

▷Ex 4\_7–4\_8

### 5.5. Redundant and non-redundant linear arrays

An array takes spatial samples of the incoming wave field. Consider the regular linear array of  $K$  receivers separated by a distance  $d$ , and a signal arriving as a plane wavefront. This samples the wave field at regular intervals: any two receivers separated by the same distance will receive the incoming signal with the same phase difference<sup>a</sup>. Thus there is one sample at a spacing of  $(K - 1)d$ , two at a spacing of  $(K - 2)d$ , ...,  $(K - 1)$  samples at a spacing of  $d$  and  $K$  samples at a spacing of zero. In a homogeneous noise field (and in the absence of receiver noise) only a single sample would suffice. A filled array, then, might take many redundant spatial samples.



<sup>a</sup>A parameter of great importance is the correlation between the outputs of the receivers. The average output of the sum-and-square beamformer, for example, can be expressed as the weighted sum of these correlation functions. However, a proper discussion of this has to await the introduction of random variables in Chapter 7.

FIGURE  
5.10. Filled  
linear array

The question could be asked whether a linear array can be designed such that there are no redundant samples. The answer is that, for  $K \leq 4$  there is, as illustrated in Figure 5.11, where the successive spacings are 1, 3 and 2. With this array, every possible spacing, from  $d$  to  $6d$ , is exhibited once and only once by the array. This is a so-called *perfect array*<sup>a</sup>.

<sup>a</sup>The term 'perfect' is loose: it only means that all spacings – except the zero spacing – occur only once; otherwise, these arrays have no particularly attractive properties.

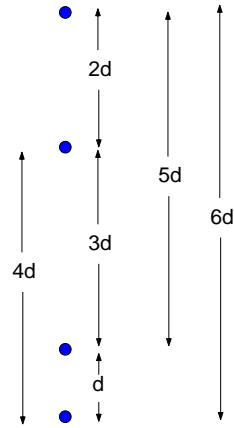


FIGURE 5.11. Perfect sparse linear array

Unfortunately for  $K > 4$  it is only possible to find only

- **non-redundant** arrays, where most but not all spacings appear only once, and the number of gaps is minimised, or
- **redundant** arrays, where every spacing appears at least once and in some cases more than once, and the number of redundant spacings is minimised.

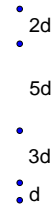


FIGURE 5.12. Non-redundant sparse linear array

Figure 5.12 shows a non-redundant array of 5 receivers, with every spacing from  $d$  to  $11d$  (except  $6d$ ) appearing once and only once.

Redundant			Non-redundant		
K	Aperture	Spacings	K	Aperture	Spacings
3	4	•1•2•	3	4	•1•2•
4	7	•1•3•2•	4	7	•1•3•2•
5	10	•1•3•3•2•	5	12	•1•3•5•2•
6	14	•1•5•3•2•2•	6	18	•1•3•6•2•5•
7	18	•1•3•6•2•3•2•	7	26	•1•3•6•8•5•2•
8	24	•1•3•6•6•2•3•2•	8	35	•1•3•5•6•7•10•2•
9	30	•1•3•6•6•6•2•3•2•	9	45	•1•4•7•13•2•8•6•3•
10	37	•1•2•3•7•7•7•4•4•1•	10	56	•1•5•4•13•3•8•7•12•2•

Spacings for redundant and non-redundant arrays

Figure 5.13 shows a redundant array of 5 receivers, with every spacing from  $d$  to  $9d$  appearing at least once ( $3d$  appears twice).

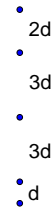


FIGURE 5.13. Redundant sparse linear array

The beam patterns of a 10-element linear non-redundant array and a 55-element linear filled array are shown in Figure 5.14. Since the apertures of the two arrays are the same, so are the widths of their main lobes. However, the properties of their sidelobes are markedly different: those for the non-redundant array (like other sparse arrays) tend to take an average value of  $10 \log_{10} K$ .

Similar results comparing redundant and filled arrays are shown in Figure 5.14

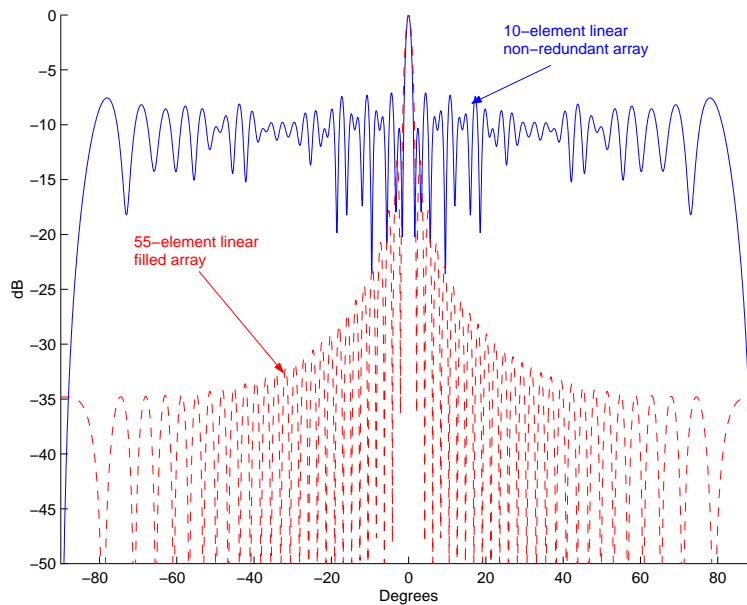


FIGURE 5.14. Beam pattern of non-redundant 10-element linear array compared with a filled linear array with the same aperture (55 receivers)

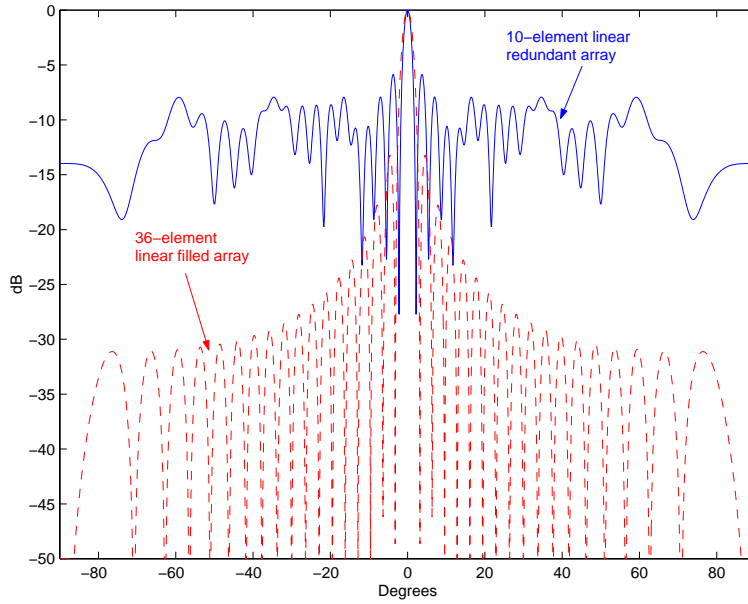


FIGURE 5.15. Beam pattern of redundant 10-element linear array compared with a filled linear array with the same aperture (36 receivers)

## 5.6. Co-array

A convenient way of visualising the efficiency of the spatial sampling by an array is to plot the histogram of array separations. This so-called *co-array* is shown in Figure 5.16 for a 10-element filled linear array.

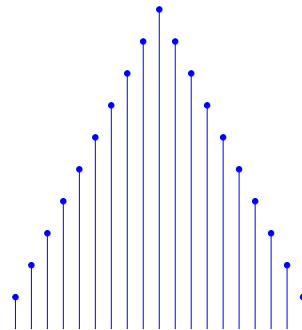


FIGURE 5.16. Co-array for 10-element filled linear array

The co-array for a redundant 5-element array is shown in Figure 5.17. For  $K$  receivers there are of course always  $K$  samples at the origin (i.e., a spacing of 0), but apart from sampling twice for a spacing of 3, this array samples all other spacing only once.

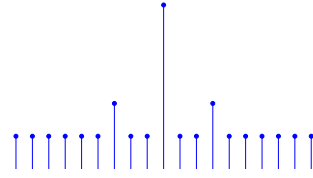


FIGURE 5.17. Co-array for 5-element redundant array

For planar arrays the co-array is a two-dimensional plot. It is easier to design planar arrays without redundancy than linear arrays. A special case is the so-called **Haubrich array** shown in Figure 5.18 with its co-array in Figure 5.19. The point at the origin, with a multiplicity of 6, has been as a small point in red for convenience. The even spread of the other points, and the absence of any multiple samples, shows that this geometry provides efficient sampling.

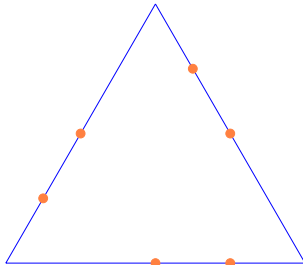


FIGURE 5.18. Haubrich array

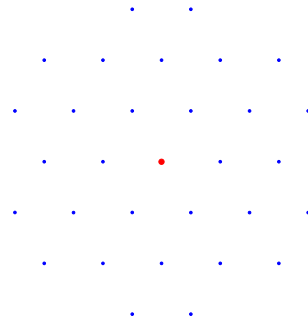


FIGURE 5.19. Co-array for Haubrich array

The co-array for a circular array with eight receivers is shown in Figure 5.20: each point where the spatial wave field is sampled appears as a dot, with the diameter of the dot indicating the multiplicity (i.e., how many times that point in space has been sampled).

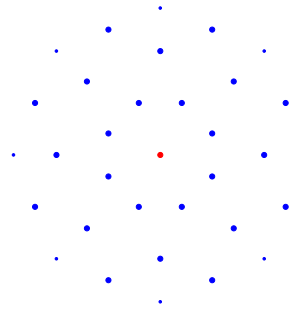
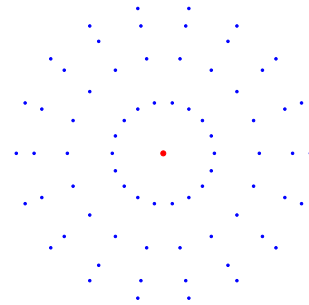


FIGURE 5.20. Co-array for 8-element circular array

Figure 5.21 shows the result for a nine-element circular array, which is quite different from the preceding one for eight elements. Here the sampling is quite efficient, with no points sampled more than once (except, of course, the origin).



▷Ex 4\_13–4\_14

FIGURE 5.21. Co-array for 9-element circular array

## 5.7. Random arrays

There has been experimentation with *random arrays* – in which the positions of the receivers are placed randomly but with some constraints. For example, one could require that all the receivers have to lie within a circle of some specified diameter, and that no pair of receivers are closer than some minimum spacing.

Like all sparse arrays, random arrays tend to have the same beamwidth as a filled array occupying the same overall length, area or volume, but exhibit sidelobes higher than those of the filled array.

▷Ex 4\_15–4\_16

## 5.8. Overview

In this chapter the following concepts have been introduced

- effect of array symmetry,
- principle of beam pattern multiplication

- filled and sparse arrays
- redundant and non-redundant linear arrays
- co-arrays
- random arrays

The reader should have some understanding of the desirable and undesirable aspects of array geometry.

### Summary

- (1) Symmetry in an array can produce high sidelobes.
- (2) The beampattern of an array of identical directional receivers is the product of the beampattern of the array of omnidirectional receivers with the same geometry, and the beampattern of an individual receiver.
- (3) Sparse arrays (including random arrays) tend to have narrower beamwidths and higher sidelobes than filled arrays of the same physical size.
- (4) With more than 4 receivers, it is not possible to design a ‘perfect’ linear array (i.e., with every possible spacing occurring once and only once).
- (5) The co-array allows the efficiency of spatial sampling to be visualised.



## CHAPTER 6

# STEERING MULTIPLE BEAMS - CONVENTIONAL PROCESSING

### 6.1. Introduction

In this chapter processing the array receiver outputs to form multiple beams steered in different directions is considered. In a field consisting of a superposition of a number of plane-wave signals the formation of multiple beams is obviously essential. Whilst it is possible to steer the beam by mechanically rotating the array this approach is not always practical or desirable and the modern preferred approach is to beam-steer – *electronically* or *numerically* – multiple beams simultaneously as illustrated in Figure 6.1.

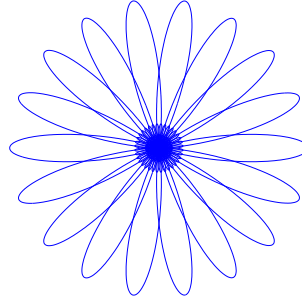


FIGURE 6.1. Overlapping beams from 42-receiver filled circular array

Consider the situation where one or more strong signal sources are incident upon the array from unknown directions. Forming many beams simultaneously and plotting the power output of the beamformer against the steered direction of these beams will result in peaks in the directions of the signal sources, thus enabling the direction of arrival of each signal to be estimated. Such a function is usually termed the ***Steered Beamformer Power Output***<sup>1</sup>

As discussed, above beams may be steered sequentially or simultaneously. Scanning some field of view sequentially is easier computationally but leads to a loss in detection performance and the delay in revisiting a particular direction is a disadvantage.

---

<sup>1</sup>The reader should be careful to distinguish between the *beam pattern* and the *beamformer output*. The beam pattern is the output of the beam for a *fixed* steering direction as a single source is moved around the array. It is independent of the environment in which the array is placed. The beamformer output is the output signal as the beam is steered in different directions. It varies according to the signals and noise incident on the array.

There are evident practical advantages in the ability to survey all directions simultaneously. The penalty is, of course, a manifold increase in the computation required. The cost is driven primarily by:

- the number of receivers
- the bandwidth of the signal
- the number of beams
- the aperture of the array (which is related to the number of receivers but also to the geometry of the array).

Very roughly, the computational cost tends to vary as the cube of the number of receivers.

In practice the following cases often occur:

- the plane waves all have the same frequency, in which case multiple phase shifted beams using multiple steering vectors are formed;
- the plane waves are modulated by narrowband signals and have carriers of almost identical frequencies. Beamforming in this case can be achieved similarly to above or by using frequency domain beamforming; or
- the plane waves have many different frequencies and are best considered as signals with a broad power spectrum - often termed *broadband sources*. Beamforming is achieved in either the frequency domain or the time domain

In this chapter techniques for forming multiple beams in each of the above cases is considered. As shall be shown, the beam pattern of the array still has a strong influence on determining the number of beams required and the system performance in the presence of multiple signals.

▷ Problem E.17

## 6.2. Multiple single-frequency signals

In this case there are a number of signal sources, each of identically the same frequency so that all the signals arriving at the array are *coherent*. The input to each receiver output is then a linear superposition of plane waves of a constant frequency,  $f_c$  but with different phases due to their different directions of arrival.

A practical example of when this occurs is when there are two or more propagation paths from the same signal—for example, when there is one direct path from the source to the array, and a second path reflected from some large fixed object (such as a building), as illustrated in Figure 6.2. The two arrivals from the same signal can be fully coherent, but in general will not be in phase.

The output beam of a phase shift beamformer is, from Chapter 3, given by

$$y(\theta, \phi, t) = \frac{1}{K} \mathbf{v}^H(\theta, \phi) \mathbf{x}(t). \quad (6.1)$$

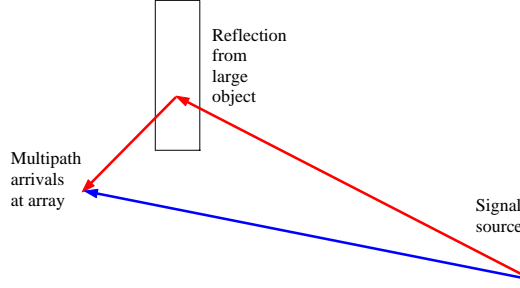


FIGURE 6.2. Illustrating multiple arrivals

To form multiple beams  $\theta, \phi$  are varied over a set of angles; in practice this will usually be discrete but in theory can be considered to be continuous. The range of angles will be determined by the operational use of the array but in general for surveillance arrays coverage of the full angular region  $[0 < \theta \leq 2\pi, 0 < \phi \leq \pi]$  is required. In the following wavevectors, rather than angles, will be used and the above (6.1) will be written as

$$y(\mathbf{k}, t) = \frac{1}{K} \mathbf{v}^H(\mathbf{k}) \mathbf{x}(t). \quad (6.2)$$

To complete the description of the beamformed signal,  $y(\mathbf{k}, t)$ , a representation of the receiver outputs in terms of the incident signals is required. As an introduction, consider the situation where there are two plane waves of the same single frequency  $f_c$  incident upon the array from angles corresponding to wavevectors  $\mathbf{k}_1$  and  $\mathbf{k}_2$ . The output of the  $j^{\text{th}}$  receiver is given by

$$\begin{aligned} x_j(t) &= s_1 \exp(i2\pi f_c t) \exp(i2\pi \mathbf{k}_1^T \mathbf{u}_j) \\ &\quad + s_2 \exp(i2\pi f_c t) \exp(i\varphi) \exp(i2\pi \mathbf{k}_2^T \mathbf{u}_j) \end{aligned}$$

or in vector notation

$$\mathbf{x}(t) = \{s_1 \mathbf{v}(\mathbf{k}_1) + s_2 \exp(i\varphi) \mathbf{v}(\mathbf{k}_2)\} \exp(i2\pi f_c t) \quad (6.3)$$

Note the scalars  $s_1$  and  $s_2$  are the magnitudes of each signal incident upon the array and  $\varphi$  is their relative phase. In the example illustrated in Figure 6.2  $\varphi$  is determined by the difference in path lengths of the direct and reflected signals. The beamformer output, as a function of the steering angle,  $\mathbf{k}$  is given by

$$\begin{aligned} y(\mathbf{k}, t) &= \frac{1}{K} \mathbf{v}^H(\mathbf{k}) \mathbf{x}(t) \\ &= \frac{1}{K} \{s_1 \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_1) + s_2 \exp(i\varphi) \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_2)\} \exp(i2\pi f_c t) \end{aligned} \quad (6.4)$$

The *steered beamformer output power*,  $p_{\text{conv}}(\mathbf{k})$ , defined<sup>2</sup> by

$$p_{\text{conv}}(\mathbf{k}) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T |y(\mathbf{k}, t)|^2 dt$$

<sup>2</sup>The subscript <sub>conv</sub> is used to denote conventional beamforming

can be shown to reduce to

$$p_{\text{conv}}(\mathbf{k}) = \frac{1}{K^2} (s_1^2 |\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_1)|^2 + s_2^2 |\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_2)|^2 + 2\Re(s_1 s_2^* \exp(i\varphi) \mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_1)\mathbf{v}^H(\mathbf{k}_2)\mathbf{v}(\mathbf{k})))$$

The first two terms in (6.4) are the beam patterns when the beam is steered in directions with wavevectors  $\mathbf{k}_1$  and  $\mathbf{k}_2$ , multiplied by the signal powers  $s_1^2$  and  $s_2^2$  respectively. Note, however, that the beamformer power output is in general *not* simply the sum of the weighted beam patterns because of the presence of the last term in the above expression representing the interference effects between the two coherent arrivals.

When the arrival directions are well separated relative to a beamwidth and the two signals are easily resolvable and the output beam power is, to a good approximation, the sum of the two beam patterns since the magnitude of

$$\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_1)\mathbf{v}^H(\mathbf{k}_2)\mathbf{v}(\mathbf{k})$$

is small. If the main lobes of the two beam patterns overlap, then the effect on  $p_{\text{conv}}(\mathbf{k})$  could be either reinforcement or cancellation, depending on the relative phase  $\varphi$ . This effect is particularly important for low grazing angle multipath signals propagating over the earth's surface.

If one signal is weak, it can be obscured by the sidelobes of the stronger signal. If the angular separation is decreased to be comparable with the beamwidth the signals will no longer be resolvable.

This discussion illustrates the need to control both the array beamwidth and sidelobe levels as discussed in Chapter 4.

### 6.3. Multiple narrowband signals

In this section narrowband signals that are not coherent with each other are considered. The vector of receiver outputs is now given by

$$\mathbf{x}(t) = \{s_1(t) \exp(i2\pi f_{c1}t) \mathbf{v}(\mathbf{k}_1) + s_2(t) \exp(i2\pi f_{c2}t) \mathbf{v}(\mathbf{k}_2)\} \quad (6.5)$$

where the bandwidth of  $s_i(t)$  is small compared its carrier frequency,  $f_{ci}$  and it is assumed that the carrier frequencies  $f_{c1}$  and  $f_{c2}$  are close. In this case the output power of the beamformer can be shown to be given by

$$p_{\text{conv}}(\mathbf{k}) = \frac{1}{K^2} (\sigma_1^2 |\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_1)|^2 + \sigma_2^2 |\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_2)|^2) \quad (6.6)$$

where

$$\sigma_i^2 = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T |s_i(t)|^2 dt$$

provided

$$\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T s_1(t) s_2^*(t) \exp(i2\pi(f_{c1} - f_{c2})t) dt = 0$$

In order for the above to hold either the carrier frequencies need to be slightly different or the signal waveforms  $s_i(t)$  need to be uncorrelated narrowband random processes.<sup>3</sup>

In this situation, the beamformer output power is simply the sum of the outputs from the output powers of the individual signals.

In the case of just two uncorrelated signal arrivals, (6.6) becomes

$$p_{\text{conv}}(\mathbf{k}) = \sigma_1^2 P(\mathbf{k}, \mathbf{k}_1) + \sigma_2^2 P(\mathbf{k}, \mathbf{k}_2). \quad (6.7)$$

The beamformer output power is thus the sum of the two beam patterns, scaled by the signal powers and the number of receivers squared. The  $K^2$  term is the gain in signal power resulting from the use of beamforming. This result can be generalised to any number of independent signal arrivals.

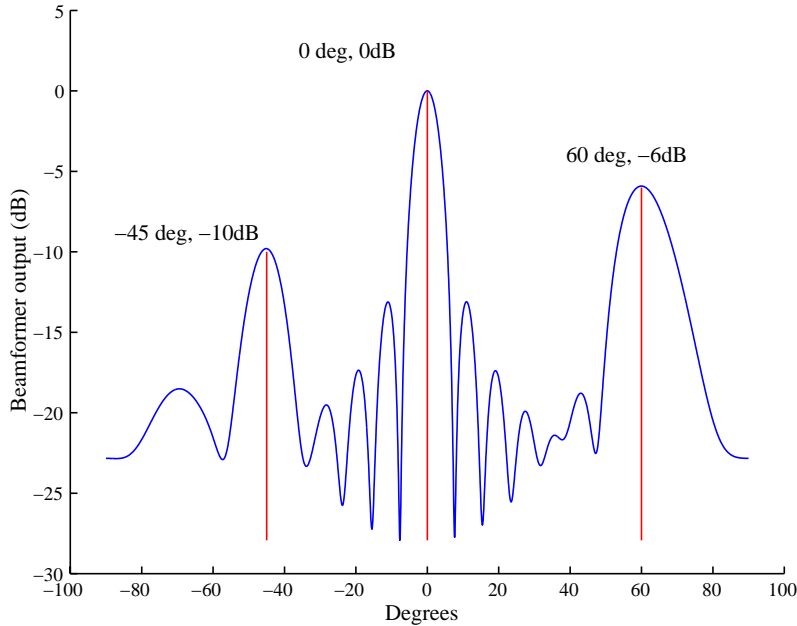


FIGURE 6.3. Steered beamformer output for 3 incident sources

Figure 6.3 illustrates the beamformer power output of a linear array of 15 receivers, with adjacent receivers spaced half a wavelength apart. There are 3 independent arrivals, at  $-45^\circ$ ,  $0^\circ$  and  $60^\circ$ , and of strength -10dB, 0dB and -6dB respectively.

#### 6.4. Multiple broadband signals—time domain beamforming

Consider the case where the signals incident upon the array have a power spectrum consisting of both continuous broadband energy and spectral line components.

<sup>3</sup>A more formal explanation of this principle is given in Chapter 7, which deals with beamforming signals whose waveforms are random processes.

Such situations will always occur when the elements of the array are wide-band receivers. Typically these receivers occur in

- acoustics and sonar
- specialised communications and radar applications.

The wide bandwidth of the receiver outputs prevents the use a single phase factor to represent the propagation delay so alternative approaches to beamforming must be used. In these cases the more direct approach of physically delaying the receiver outputs to compensate for the propagation delays across the array aperture is adopted. The delays can be chosen such that, at each receiver, the output waveforms due to a signal from the direction of interest are identical and so can be added to reinforce the signal of interest. This is the key idea of **time delay and sum beamforming**; it is a direct and intuitive form of coherent addition and is considered in this section.

#### 6.4.1. Time Delays.

To illustrate, consider a circular array of 8 receivers equally spaced around a circle of radius  $r$  and located in the  $x$ - $y$  plane as illustrated in Figure 6.4: For this array each receiver's position,  $\mathbf{u}_j$ , is most conveniently written in terms of polar coordinates as

$$\mathbf{u}_j = \begin{bmatrix} r \cos \theta_j \\ r \sin \theta_j \\ 0 \end{bmatrix}, \quad (6.8)$$

where for such circular arrays

$$\theta_j = 2(j-1)\pi/K.$$

FIGURE 6.4. Circular array geometry

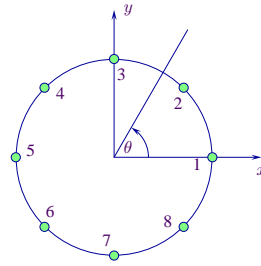
For a plane wave incident upon the array from directions  $\theta$  and  $\phi = 90^\circ$ , the time delay for the  $j^{\text{th}}$  receiver relative to the origin is given by

$$\tau'_j(\theta) = \frac{r}{c} \cos(\theta_j - \theta). \quad (6.9)$$

Since negative time delays cannot be implemented in practise we add a bulk delay  $T_0$ , such that

$$\tau_j(\theta) = T_0 + \tau'_j(\theta) \quad (6.10)$$

is always positive.  $T_0$  defines the time delay reference plane; a beam steered in direction  $\theta$  is formed by delaying the receiver outputs by  $\tau_j(\theta)$  and then adding these delayed receiver outputs as illustrated in Figure 6.5.



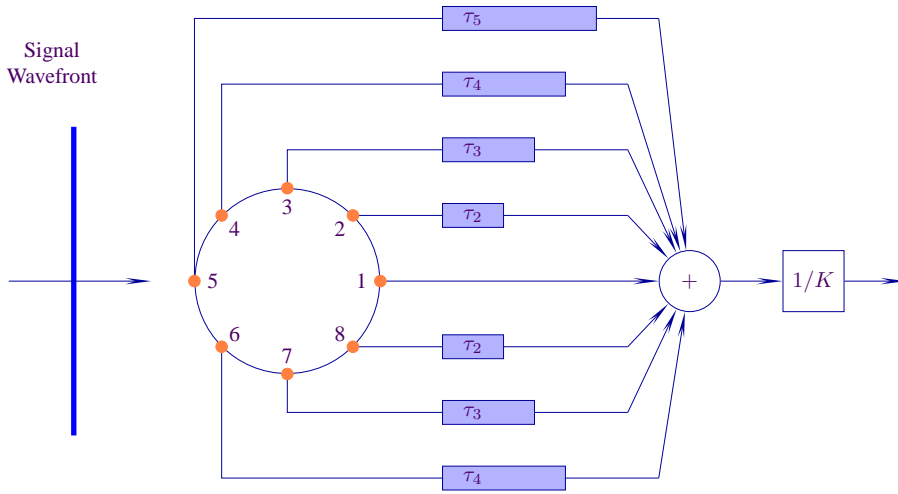


FIGURE 6.5. Time Delay and Sum for a Circular Array

This beamformer is called the **time delay and sum beamformer**. If the beam output is squared it is often called the “**add-squarer**”<sup>4</sup>.

The time series output of such a beamformer can be written as:

$$y(\theta, t) = \frac{1}{K} \sum_{j=1}^K x(t - \tau_j(\theta)) \quad (6.11)$$

or in general

$$y(\theta, \phi, t) = \frac{1}{K} \sum_{j=1}^K x(t - \tau_j(\theta, \phi)), \quad (6.12)$$

where

$$\tau_j(\theta, \phi) = \frac{x_j \cos \theta \sin \phi + y_j \sin \theta \sin \phi + z_j \cos \phi}{c} \quad (6.13)$$

and in the above the arbitrary time  $T_0$  has been ignored.

Note the following regarding the time delay and sum beamformer

- (a) In general it is desirable to simultaneously form not just one, but a number of beams (e.g. for sonar arrays with a large number of elements this number can be in the hundreds or thousands). Digital technology allows these beams to be formed simultaneously.
- (b) Apart from a bulk time delay, the signal waveform characteristics in the look direction are preserved—this property is particularly important if the array is being used to beamform transient signals. Note because of the

<sup>4</sup>In this example we have chosen to perform the scaling after the summation.

frequency dependence of the beam pattern this is not true for interferences “leaking” into the steered direction through sidelobes—their waveforms are distorted.

▷ Problem E.18

▷ Problem E.19

▷ Problem E.20

### 6.4.2. Time Delay Errors - Digital Beamforming.

In practice the digital receiver outputs are sampled in time and thus the time delay  $\tau_j(\theta, \phi)$  used the time delay and sum beamformer can only be accurately approximated to within half a sampling interval. Since the actual propagation delays are continuous this will lead to errors. The errors are termed *time quantisation errors* and their effect on the beam pattern is of the form:

- (a) General increase in sidelobe structure with a few isolated bumps, or
- (b) Maintenance of general sidelobe behaviour with isolated high peaks.

Case (a) corresponds to the situation in which the time quantisation errors are periodic across the array aperture, whilst (b) corresponds to the situation where the time quantisation errors are not.

In general beam-pattern degradations result in performance degradations. In most practical situations, the degradation to array gain in uncorrelated or isotropic noise fields is minimal as usually the main-lobe response is not significantly degraded. However high sidelobes in the beam pattern, case (a) above, can significantly degrade array performance if there are strong interferences in the directions of these sidelobes. Similarly an overall increase in sidelobe level, case(b) above, can lead to poorer performance in the presence of distributed clutter.

#### Example.

If  $\Delta T$  is the sampling interval consider the case, for a uniform linear array, where the steering direction,  $\theta$ , satisfies  $(d/c) \sin \theta = T/2$  which implies that the relative propagation delay between receivers separated by a distance  $2d$  is equal to the sampling interval. Since, without resorting to interpolation schemes, it is not possible to achieve a relative delay between adjacent receivers of  $T/2$ , time quantisation errors will result. This is illustrated below and it can be seen that the time quantisation errors in this case will be periodic.



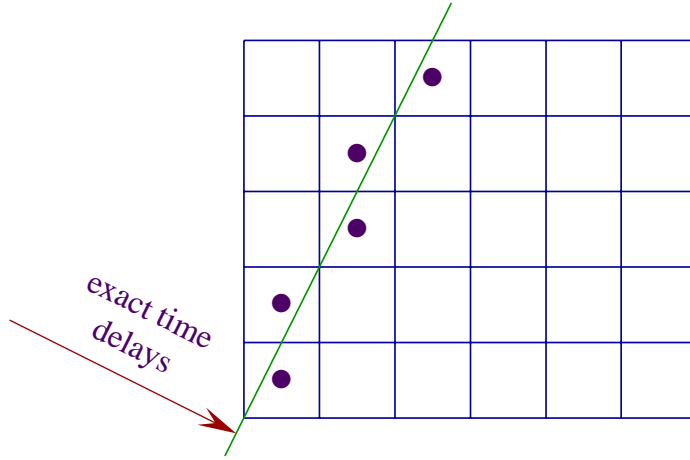


FIGURE 6.6. Delay Errors

Note that odd receivers have the correct relative time delays and similarly for the even receivers, so the only error is that between the subsets of odd and even receivers. In such a case the principle of beam-pattern multiplication can be used to analyse the errors.

Time quantisation errors can be minimised by

- (a) Increasing the sampling rate—as a rule of thumb most time delay and sum beamformers use a sampling frequency about eight times the highest frequency in the signal.
- (b) Use interpolation—this can be efficiently implemented for beamformers using both polyphase filters and post beamformer interpolation. Even simple linear interpolation between samples can significantly reduce the beam-pattern degradation.

▷ Problem E.21

### 6.5. Multiple broadband signals – frequency domain beamforming

**Frequency domain beamforming** is an approach where the Fourier transform of each receiver output is formed and at each frequency the receiver outputs are beamformed by phase multiplication. This technique is often used when we need to spectrum analyse the beam outputs to detect weak narrowband signals buried in noise and if the number of beams to be formed is greater than the number of receivers it can be more efficient to Fourier transform the receiver outputs rather than the beamformer outputs.

Let  $\tilde{y}(\theta, \phi, f)$  denote the continuous Fourier transform of a beam steered in direction  $\{\theta, \phi\}$ , i.e.,

$$\tilde{y}(\theta, \phi, f) = \int_{-\infty}^{\infty} y(\theta, \phi, t) e^{-2\pi i f t} dt. \quad (6.14)$$

Substituting for  $y(\theta, \phi, t)$  from (6.12) results in

$$\tilde{y}(\theta, \phi, f) = \frac{1}{K} \int_{-\infty}^{\infty} \sum_{j=1}^K x_j \{t - \tau_j(\theta, \phi)\} e^{-2\pi i f t} dt. \quad (6.15)$$

Interchanging the order of integration and summation gives

$$\tilde{y}(\theta, \phi, f) = \frac{1}{K} \sum_{j=1}^K \int_{-\infty}^{\infty} x_j \{t - \tau_j(\theta, \phi)\} e^{-2\pi i f t} dt. \quad (6.16)$$

However a translation in the time domain corresponds to multiplication by a phase term in the frequency domain, i.e., if

$$\tilde{x}_j(f) = \int_{-\infty}^{\infty} x_j(t) e^{-2\pi i f t} dt \quad (6.17)$$

then

$$\int_{-\infty}^{\infty} x_j \{t - \tau_j(\theta, \phi)\} e^{-2\pi i f t} dt = \tilde{x}_j(f) e^{-2\pi i f \tau_j(\theta, \phi)}. \quad (6.18)$$

(6.15) becomes

$$\tilde{y}(\theta, \phi, f) = \frac{1}{K} \sum_{j=1}^K \tilde{x}_j(f) e^{-2\pi i f \tau_j(\theta, \phi)}. \quad (6.19)$$

Thus the Fourier transform of  $y(\theta, \phi, t)$  can be expressed as a phase-weighted summation of the Fourier transformed receiver outputs.

Recall that the **steering vector**,  $\mathbf{v}(\theta, \phi, f)$ , is given by

$$\mathbf{v}(\theta, \phi, f) = \begin{bmatrix} \exp(2\pi i f \tau_1(\theta, \phi)) \\ \exp(2\pi i f \tau_2(\theta, \phi)) \\ \vdots \\ \exp(2\pi i f \tau_K(\theta, \phi)) \end{bmatrix}. \quad (6.20)$$

$\tilde{\mathbf{x}}(f)$ , the vector of Fourier transformed receiver outputs, is given by

$$\tilde{\mathbf{x}}(f) = \begin{bmatrix} \tilde{x}_1(f) \\ \tilde{x}_2(f) \\ \vdots \\ \tilde{x}_K(f) \end{bmatrix} \quad (6.21)$$

When forming beams it is convenient to normalise the steering vector by multiplying by  $1/K$  so that the resultant beam pattern takes a value of unity in the beam-steered direction. With this notation the beam output,  $\tilde{y}(\theta, \phi, f)$  can be written as the Hermitian product

$$\tilde{y}(\theta, \phi, f) = \frac{1}{K} \mathbf{v}^H(\theta, \phi, f) \tilde{\mathbf{x}}(f). \quad (6.22)$$

In practice the continuous Fourier transform is approximated by taking the discrete Fourier transform of the receiver outputs. If only a few beams are to be formed then this technique is inefficient as the computational cost in evaluating the Fourier

transform for each receiver may be prohibitive. However, if the system requirements are to form more beams than there are receivers and then to determine the spectral content of each beam by Fourier transforming then this approach may be preferred.

For deterministic signals in the frequency domain the output power is given by  $|\tilde{y}(\theta, \phi, f)|^2$ ; substituting from (6.22) we have

$$|\tilde{y}(\theta, \phi, f)|^2 = \frac{1}{K^2} \mathbf{v}^H(\theta, \phi, f) \tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}^H(f) \mathbf{v}(\theta, \phi, f). \quad (6.23)$$

The above is an Hermitian form of the matrix  $\tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}^H(f)$ . Similar expressions naturally arise when beamforming signals represented as random processes - see Chapter 7.

▷Ex 5\_1

### 6.6. Implementation of frequency domain beamforming

Implementing frequency domain beamforming in practice requires taking the Fourier transform of the receiver outputs. For sampled data this is commonly implemented using the fast Fourier transform (FFT) algorithm. The finite length and frequency resolution of the FFT algorithm can result in errors. In most practical implementations these errors are negligible and we do not consider them here.

A block diagram for implementing frequency domain beamforming is shown in Figure 6.7.

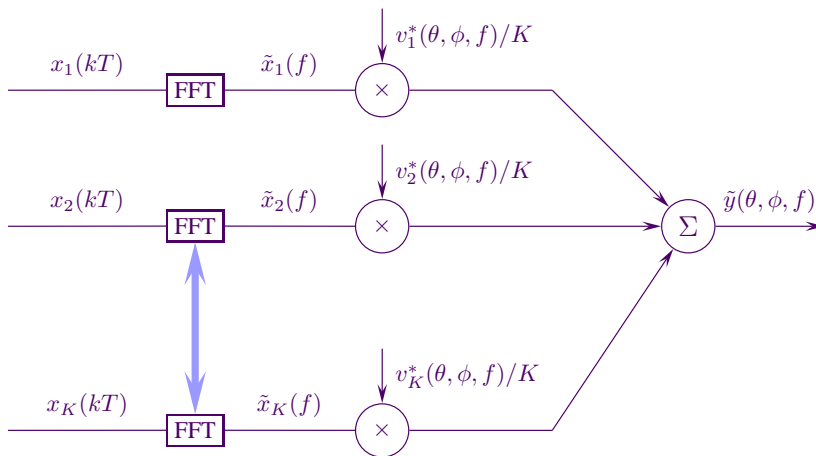


FIGURE 6.7. Block diagram for frequency domain beamforming

## 6.7. Overview

In this Chapter we have introduced the concept of steering multiple beams simultaneously.

Three cases have been considered:

- Multiple coherent single-frequency signals and
- Multiple non-coherent narrowband signals
- Broadband signals

### Summary

- (1) If the arrival directions of coherent single-frequency signals are well separated relative to the beamwidth, the output beam power is approximately equal to the sum of the beam patterns.
- (2) When the main lobes of the beam patterns of two or more coherent single-frequency signals overlap, the output beam power depends critically on relative phase(s) between them.
- (3) In the case of non-coherent signals, the output power is simply the sum of the individual beam patterns, multiplied by the powers of the arrivals.
- (4) Processing of single-frequency and narrowband signals can be implemented using phase-shift beamforming.
- (5) Broadband signals can be processed in the time domain or, using fourier transforms, in the frequency domain.
- (6) In time-domain beamforming, consideration needs to be given to time quantisation errors.

## CHAPTER 7

# WAVEVECTORS, SPATIAL SAMPLING AND FILTERING

### 7.1. Introduction

In Chapter 2 the concept of *wavevector* was introduced and was shown to be related to angular coordinates  $(\theta, \phi)$  and wavelength  $\lambda$  by the following relationship:

$$\mathbf{k} = \begin{bmatrix} k_x \\ k_y \\ k_z \end{bmatrix} = \frac{2\pi}{\lambda} \begin{bmatrix} \cos \theta \sin \phi \\ \sin \theta \sin \phi \\ \cos \phi \end{bmatrix} \quad (7.1)$$

The wavevector is a very elegant representation and for that reason this chapter is devoted to its usefulness in beamforming. A uniform linear array will be used to show the analogy between wavevectors and one-dimensional filters. The results can be extended to more general arrays.

Three main reasons favouring the wavevector approach are briefly considered below.

- The general solution to the wave equation when the propagation speed is constant across the array aperture is a superposition of plane waves. These plane waves are most conveniently and compactly described in terms of their wavevectors.
- The wavevector  $\mathbf{k}$  and frequency  $\omega$  are Fourier transforms of the spatial and temporal variables  $\mathbf{u}$  and  $t$  respectively. This suggests implementation structures for beamforming that utilise *multidimensional Fourier transforms*; these are considered later in this chapter.
- Wavevectors allow the properties of an array to be succinctly quantified and give insight into their properties.

### 7.2. Propagating wave fields

Recall that in Chapter 2 the coordinate vector  $\mathbf{u}$  was defined as:

$$\mathbf{u} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (7.2)$$

and the wave field was denoted by  $f(\mathbf{u}, t)$ . Assuming the medium to be locally homogeneous in the vicinity of the array, the wave field resulting from a *single*

far-field source, as illustrated in Figure 1.3, can be written as:

$$f(\mathbf{u}, t) = \alpha(\mathbf{k}_s, \omega) \exp\{i(\omega t + \mathbf{k}_s^T \mathbf{u})\}, \quad (7.3)$$

where  $\alpha(\mathbf{k}_s, \omega)$  is the complex amplitude at the origin and  $|\alpha(\mathbf{k}_s, \omega)|^2$  is the power of the plane wave.

It is often possible to model the wave field as a linear superposition of signals from many far-field sources with different frequency components as illustrated and discussed in Section 2.4. The resultant wave field is calculated by integrating over all wavevector components and frequencies and can be represented as

$$f(\mathbf{u}, t) = \frac{1}{(2\pi)^4} \int d\omega \iiint \alpha((\mathbf{k}_s)_x, (\mathbf{k}_s)_y, (\mathbf{k}_s)_z, \omega) \quad (7.4)$$

$$\times \exp\{i(\omega t + x(\mathbf{k}_s)_x + y(\mathbf{k}_s)_y + z(\mathbf{k}_s)_z)\} d(\mathbf{k}_s)_x d(\mathbf{k}_s)_y d(\mathbf{k}_s)_z$$

$$= \frac{1}{(2\pi)^4} \int \alpha(\mathbf{k}_s, \omega) \exp\{i(\omega t + \mathbf{k}_s^T \mathbf{u})\} d\omega d\mathbf{k}_s. \quad (7.5)$$

(7.5) shows that the *space-time* field,  $f(\mathbf{u}, t)$ , is related to the complex amplitude,  $\alpha(\mathbf{k}_s, \omega)$ , by a four-dimensional Fourier transform, where the three-dimensional wavevector  $\mathbf{k}_s$  and frequency  $\omega$  are the natural Fourier coordinates corresponding to the position vector  $\mathbf{u}$  and time  $t$ , respectively<sup>1</sup>.

---

<sup>1</sup>For plane waves propagating at speed  $c$  the values of  $(\mathbf{k}_s)_x$ ,  $(\mathbf{k}_s)_y$ ,  $(\mathbf{k}_s)_z$  and  $\omega$  in (7.5) would be constrained to lie on the multidimensional cone defined by:

$$(\mathbf{k}_s)_x^2 + (\mathbf{k}_s)_y^2 + (\mathbf{k}_s)_z^2 = \omega^2 / c^2.$$

However, we shall not impose this constraint: in (7.5) the variables of integration will range over  $\pm\infty$ . This may seem in conflict with the idea of propagating plane waves and not to be of practical significance. However there are important physical situations where the range needs to be extended for a complete physical description of all the signals of interest. For example, in the design of towed sonar arrays, signals can propagate along the mechanical structure of the array at a speed other than  $c$  (the speed of sound in sea-water); in this situation the complete wavevector domain is required to fully understand the effect of array self noise.

### 7.3. Linear array—one-dimensional case

Consider the uniform linear array of Figure 7.1 steered in a direction  $\theta$ . One can assume, without loss of generality, that the array is aligned along the  $y$ -axis and the  $\phi$  steering direction is chosen to be  $\pi/2$ . Let  $\tilde{x}_j(f)$  be the Fourier transformed output of the  $j^{\text{th}}$  receiver at frequency  $f$  and located at position  $\mathbf{u}_j$ . Then, from (3.52) the output of a frequency-domain beam steered in direction  $\theta$  is

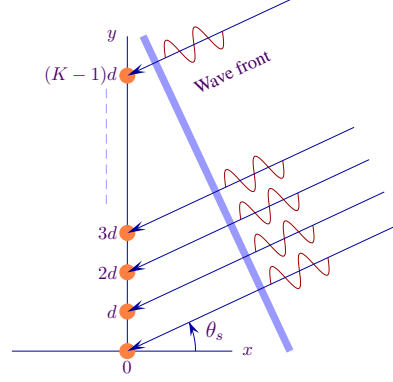


FIGURE 7.1. System of coordinates

$$\tilde{y}(\theta, f) \equiv \tilde{y}(\theta, \pi/2, f) = \frac{1}{K} \left\{ \sum_{j=1}^K \tilde{x}_j(f) \exp(-i \mathbf{k}^T \mathbf{u}_j) \right\}. \quad (7.6)$$

Since

$$\mathbf{u}_j = \begin{bmatrix} 0 \\ (j-1)d \\ 0 \end{bmatrix},$$

(7.6) can be written as

$$\tilde{y}(\theta, f) = \frac{1}{K} \left\{ \tilde{x}_1(f) + \tilde{x}_2(f)z^{-1} + \tilde{x}_3(f)z^{-2} + \cdots + \tilde{x}_K(f)z^{-(K-1)} \right\}, \quad (7.7)$$

where

$$z = \exp\left(\frac{2\pi i d \sin \theta}{\lambda}\right) = \exp(ik_y d). \quad (7.8)$$

Dispensing with the angular variable,  $\theta$ , and working with the wavevector component,  $k_y$  gives

$$\tilde{y}(k_y, f) = \frac{1}{K} \left\{ \sum_{j=1}^K \tilde{x}_j(f) \exp(-i(j-1)k_y d) \right\}. \quad (7.9)$$

This expression looks remarkably like a Fourier series where, instead of a time sampling of  $T$  a spatial sampling interval of  $d$  is used, and instead of  $\omega$ , as the transformed variable  $k_y$  is used, i.e.,

$$\begin{array}{ccc} T & \Leftrightarrow & \omega \\ \updownarrow & & \updownarrow \\ d & \Leftrightarrow & k_y \end{array}$$

In (7.9),  $k_y = \frac{2\pi}{\lambda} \sin \theta$  is the  $y$ -component of the wavevector; and since

$$\frac{ck_y}{2\pi} = \frac{c}{\lambda} \sin \theta = f \sin \theta, \quad (7.10)$$

it follows that  $k_y$  may be interpreted as a *spatial frequency*<sup>2</sup>. To understand this from a physical point of view, consider Figure 7.2 which depicts a sinusoidal signal arriving at a vertical linear array located along the  $y$ -axis. The horizontal axis denotes any line in the  $x - z$  plane. The wavelength of the incoming signal can be resolved into two components of length  $\lambda' = \lambda / \sin \theta_s$  along and  $\lambda / \cos \theta_s$  normal to the array, respectively.

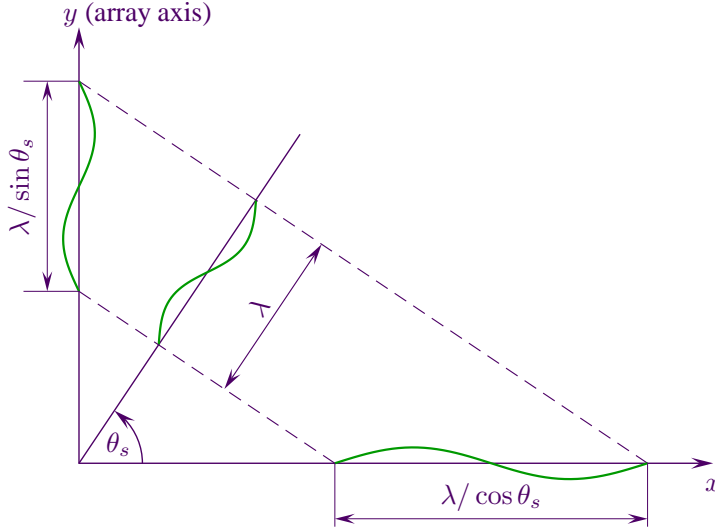


FIGURE 7.2. Projection of wavelength along a linear array

The components of the plane wave,  $\exp\{i(\mathbf{k}_s)_x x\}$  and  $\exp\{i(\mathbf{k}_s)_z z\}$ , are unity by virtue of the array geometry. However, the component along the  $y$ -axis, termed the *spatial factor* and given by  $\exp\{i(\mathbf{k}_s)_y (j-1)d\}$ , is non-zero and can, as illustrated in Figure 7.2, be interpreted as a plane wave of wavelength  $\lambda / \sin \theta_s$  propagating along the array from end-fire. As the incidence angle of the plane wave, viz.,  $\theta_s$  is changed so is the wavelength of the  $y$  component. Some examples of the spatial factor for an array such that  $d/\lambda = 1/2$  are given below.

a. *End-fire signal* (i.e.,  $\theta_s = \pi/2 \Rightarrow \lambda' = \lambda$ )

<sup>2</sup>In seismic applications,  $\frac{\sin \theta}{\lambda}$  is called *slowness*.



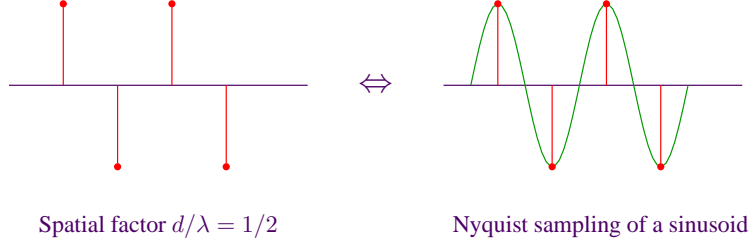


FIGURE 7.3. Analogy between spatial and temporal sampling: end-fire

The analogy between time series analysis is apparent: in the end-fire case and at half-wavelength spacing, an end-fire signal is analogous to sampling a sine wave whose frequency is the Nyquist rate.

*b. Broadside signal (i.e.,  $\theta_s = 0 \Rightarrow \lambda' = \infty$ )*



FIGURE 7.4. Analogy between spatial and temporal sampling: broadside

Similarly, the broadside example is analogous to sampling a d.c. signal in the time domain. Furthermore, these examples illustrate that the spatial separation of receivers is analogous to the time interval between samples for time series analysis.

### 7.4. Wavenumber beamforming

The idea of forming beams in a direction  $\theta$  by phase-shifting the receiver outputs so that, for a signal from that direction, they are all brought into phase was introduced in Chapter 6; this can be generalised to the concept of forming a beam for a signal with a particular wavevector component  $k_y$ .<sup>3</sup>

The output of such a beam can then be written as:

$$\tilde{y}(k_y, f) = \frac{1}{K} \mathbf{v}^H(k_y) \tilde{\mathbf{x}}(f), \quad (7.11)$$

<sup>3</sup>It should be noted that not all values of  $|k_y|$  will correspond to actual angles of incidence; for  $|k_y| > 2\pi/\lambda$  there is no such real angle. Nevertheless, as will be discussed later, this generalised concept is still very useful.

where, for this linear array with receivers located along the  $y$ -axis, the **steering vector as a function of the wavevector component**  $k_y$ ,  $\mathbf{v}(k_y)$ , is given as:

$$\mathbf{v}(k_y) = \begin{bmatrix} 1 \\ \exp(ik_y d) \\ \exp(ik_y 2d) \\ \vdots \\ \exp[ik_y (K-1)d] \end{bmatrix} \quad (7.12)$$

and  $\tilde{\mathbf{x}}(f)$ , is the vector of Fourier transformed receiver outputs.

Since the spatial sampling is uniform and (7.11) is a discrete Fourier transform, the output  $\tilde{y}(k_y, f)$  is periodic in  $k_y$ . It can readily verified by inspection of the phase factors that  $\mathbf{v}(k'_y, f) = \mathbf{v}(k_y, f)$  if  $k'_y = k_y + m2\pi/d$ . Thus the period of  $\mathbf{v}(k_y, f)$ , and hence  $\tilde{y}(k_y, f)$ , is  $2\pi/d$ , i.e.,

$$\tilde{y}(k_y, f) = \tilde{y}(k_y + m2\pi/d, f)$$

where  $m$  is any integer.

To illustrate this the output power,  $|\tilde{y}(k_y, f)|^2$  of a uniform linear array of 16 receivers as a function of  $k_y$  is plotted below. This example assumes a signal incident upon the array from  $30^\circ$  of broadside and a frequency such that  $d/\lambda = 1/2$ . The periodicity of  $2\pi/d$  is obvious.

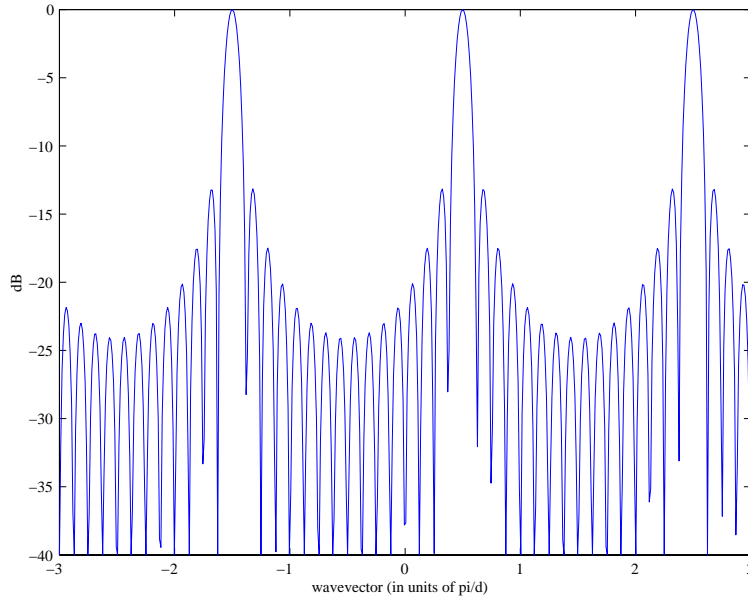


FIGURE 7.5. Periodicity of the wavevector beamformer

Since the peak has been normalised to unity, the above is equal to the beam pattern  $P(k_y, (\mathbf{k}_s)_y)$  of a linear array steered in the wavevector direction  $k_y = 0.5$  and plotted as a function of the wavevector component  $(\mathbf{k}_s)_y$ . From (3.55) it can be

shown that, for a uniform linear array of  $K$  receivers, the beam pattern is given by

$$P(k_y, (\mathbf{k}_s)_y) = \frac{1}{K^2} \left( \frac{\sin(Kd((\mathbf{k}_s)_y - k_y)/2)}{\sin(d((\mathbf{k}_s)_y - k_y)/2)} \right)^2 \quad (7.13)$$

which can be written, in steering vector notation, as

$$P(k_y, (\mathbf{k}_s)_y) = \frac{1}{K^2} |\mathbf{v}^H(k_y) \mathbf{v}((\mathbf{k}_s)_y)|^2. \quad (7.14)$$

Note in particular that:

- (i) since  $\tilde{x}_j(f)$  is complex, the spatial Fourier transform,  $\tilde{y}(k_y, f)$  and hence the beam pattern, is not symmetric about the origin;
- (ii) because of the periodic nature of the beam pattern only the region

$$-\pi/d < k_y \leq \pi/d.$$

is considered;

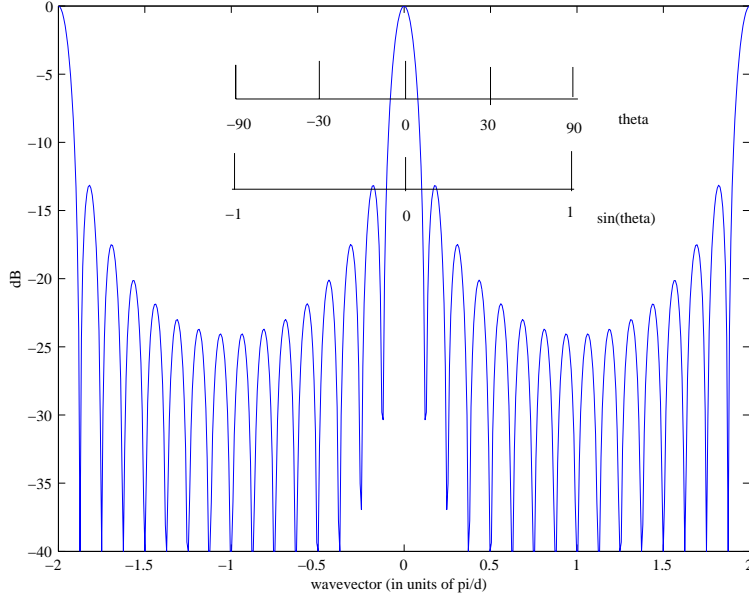
- (iii) the beam pattern is analogous to the response function of a bandpass filter, and in this sense beamforming is analogous to spatial filtering;
- (iv) observe however that periodicity in  $\theta$  that must always be present is not the same as the periodicity in  $k_y$ .

### 7.5. Steering beams in wavevector

First consider the beam pattern for a beam steered in the broadside direction, i.e.,  $k_y = 0$  at  $d/\lambda = 1/2$ . In this case

$$(\mathbf{k}_s)_y = 2\pi \sin \theta_s / \lambda = \pi \sin \theta_s / d$$

and the beam pattern will have its maxima at values of  $(\mathbf{k}_s)_y$  equal to integer multiples of  $\pi/d$ . This is plotted below both as a function of  $(\mathbf{k}_s)_y$  and the corresponding values of  $\sin \theta_s$  and  $\theta_s$ .

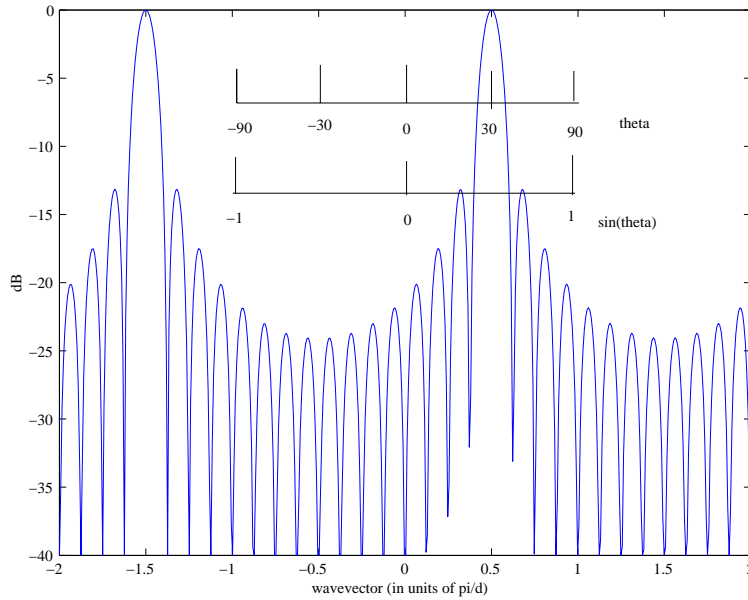
FIGURE 7.6. Steering at broadside— $0^\circ$ 

Note that a linear scale in  $(\mathbf{k}_s)_y$  implies a nonlinear scale in  $\theta_s$  and that the range of values of  $(\mathbf{k}_s)_y$  from  $-\pi/d$  to  $\pi/d$  is equivalent to  $\theta_s$  ranging from  $-90^\circ$  to  $90^\circ$ .

Now consider the case of steering a beam in direction  $\theta$ . As discussed in Section 5.3, this requires that the output from each receiver is multiplied by a phase shift, i.e., the  $j^{\text{th}}$  receiver is multiplied by  $\exp(-i(j-1)dk_y)$ , where  $k_y = 2\pi/\lambda \sin \theta$ . From Fourier transform theory a phase multiplication in one domain results in a shift in the transform domain. Since wavevector beamforming is a spatial transform, steering in the spatial domain by phase multiplication results in a translation in the  $k_y$ , wavevector, domain. (This is analogous to modulating a signal onto a carrier in the time series case.) This results in the following change of axis

$$k'_{ys} = (\mathbf{k}_s)_y - k_y,$$

where  $k_y$  is the steering direction of the beam. Illustrated below is the beam pattern response, as a function of  $(\mathbf{k}_s)_y$  for the steering angle of  $30^\circ$ . In this case  $k_y = \pi/d \sin(30^\circ)$  or, in the normalised units of  $\pi/d$ ,  $1/2$ . The maximum of the beam pattern occurs when  $(\mathbf{k}_s)_y$  equals this value since the beam pattern is just a translated version of that shown in Figure 7.6. Again the corresponding values of  $\theta_s$  and  $\sin \theta_s$  and indicated in the axis below.

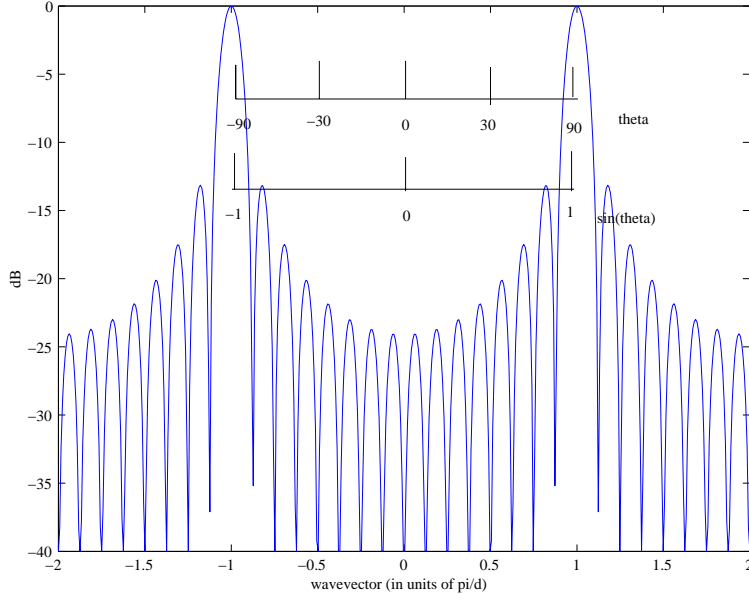
FIGURE 7.7. Steering at  $30^\circ$ 

Note that in wavevector coordinates the beamwidth is independent of steering direction, however in angular coordinates this is not the case; because of the nonlinear relation between  $\theta$  and  $\sin \theta$  the beamwidth, in angular coordinates, increases as we steer towards end-fire.

As a further example consider the wavevector beam pattern for a beam steered at  $90^\circ$ . In this case the axis is translated by an amount  $\sin(90^\circ)\pi/d$ , i.e., one unit of  $\pi/d$ .

Note that translation holds in the wave-vector domain but not in the angular domain. Thus in contrast to the situation in Chapter 3 in which the beam patterns looked markedly different as the steering direction changed it can now be seen that this difference is simply due to the mapping of  $k_y \rightarrow \theta$ . As the examples illustrate, this simple observation has the following important ramifications.

- The envelope and shape of the sidelobe pattern is independent of the steering direction. For example, it is obvious that the height of highest sidelobe ( $-13.5$  dB relative to the peak) is independent of the steering direction.
- The broadening of the main beam when steering towards end-fire is solely due to the  $\sin \theta$  deformation.
- The number of nulls is independent of the steering direction and are equispaced in  $(\mathbf{k}_s)_y$ , i.e.  $\sin \theta_s$ .

FIGURE 7.8. Steering at  $90^\circ$ 

### 7.6. Steering independent beams

In Chapter 5 the concept of forming multiple beams was introduced and some practical considerations effecting the number of beams formed was discussed. This issue will now be considered from a theoretical point of view.

First consider the case of two beams steered in direction  $\theta_1$  and  $\theta_2$ . If the difference  $|\theta_1 - \theta_2|$  is small relative to their beamwidths then it would be expected that the beam outputs would be similar, whilst if the difference is large then it would be expected that their outputs would be more or less independent. Thus the first problem to consider is how to determine, at half wavelength, the steering directions and number of independent beams needed to cover the angular region from  $-\pi/2$  to  $\pi/2$ . As will be seen, use of the above translational properties in wavevector allows an easy determination of the number and directions of these beams.

Two beams are defined to be independent if their steering vectors are linearly independent which is equivalent to requiring that their steering vectors be orthogonal. Denoting the values of  $k_y$  corresponding to  $\theta_1$  and  $\theta_2$  as  $k_1$  and  $k_2$  then the orthogonality condition can be expressed as

$$\mathbf{v}^H(k_1)\mathbf{v}(k_2) = 0. \quad (7.15)$$

Reflecting back to Chapter 3 this is equivalent to requiring that the steering direction,  $k_1$  of the first beam be chosen to be at the same position as a null of the beam pattern of the second beam. Since the beam pattern has  $K - 1$  nulls equally spaced in  $k_y$  on the region  $-\pi/2$  to  $\pi/2$  the nulls are separated from the main lobe by

multiples of  $\Delta k_y = 2\pi/Kd$ . Thus two beams are orthogonal if  $k_1 = k_2 + m\Delta k_y$  for  $m$  any integer not equal a multiple of  $K$ .

Furthermore, as there are  $K - 1$  nulls in the beam pattern it follows that in addition to the broadside beam there are  $K - 1$  other independent beams at values of  $k_y$  given by

$$k_y^{(j)} = 2\pi j/Kd = 2\pi/\lambda \sin \theta_j \quad \text{for } j = -K/2, \dots, K/2 \text{ and } j \neq 0.$$

Alternatively, in angular coordinates the steering directions of these independent beams is given by  $\sin \theta_j = j\lambda/Kd$ . Thus taking a spatial Fourier transform is equivalent to forming the complete set of independent beams.

The progressive phase factor for steering in direction  $\theta_j$ ,  $z(\theta_j)$ , is given by

$$z(\theta_j) = \exp(2\pi i d/\lambda \sin \theta_j)$$

which, on substituting for  $\sin \theta_j$  above, reduces to

$$z(\theta_j) = \exp(2\pi i j/K)$$

which is the discrete Fourier transform<sup>4</sup>. Thus taking a spatial Fourier transform is equivalent to forming the complete set of independent beams.

If only a finite number of beams are formed and a desired signal is incident upon the array from a direction slightly different to that of the nearest beam then this signal will be suppressed. For equi-spaced beams this loss is greatest when the signal's  $k_y$  direction lies midway between any two beams. In the case where all beams are independent, this loss is  $\sim 3.9$  dB and in many practical applications is too great. In such cases extra beams more closely spaced in  $k_y$  are steered. For example, steering  $2K$  beams equally spaced in  $k_y$ , reduces this loss to around 1.1 dB. In this case the progressive phase factor can be written as

$$z(\theta_j) = \exp(\pi i j/K) \quad \text{for } j = 0, 1, 2, \dots, 2K - 1.$$

In practice this can be efficiently implemented by zero padding the vector of receiver outputs and taking a discrete Fourier transform of size  $2K$ .

## 7.7. Varying the temporal frequency

The situation considered above was one in which the wavelength of the incident signal was twice the receiver spacing. Now consider varying the frequency of the signal, thus changing the ratio  $d/\lambda$ . First consider how the ranges of  $k_y$ , i.e.,  $[-\pi/d, \pi/d]$  and  $\sin \theta$  are related. In order to satisfy the necessary constraint that

$$-1 \leq \sin \theta \leq 1$$

or equivalently that

$$-2\pi/\lambda \leq (2\pi/\lambda) \sin \theta \leq 2\pi/\lambda$$

---

<sup>4</sup>Choosing the number of receivers,  $K$ , to be a power of 2, allows the efficiency of the FFT to be used to effect the beamforming.

it is required that

$$-\pi/d(2d/\lambda) \leq k_y \leq \pi/d(2d/\lambda).$$

Defining “physical angles” to be those that correspond to  $-1 \leq \sin \theta \leq 1$  it follows that the corresponding range of  $k_y$  is determined by the factor  $2d/\lambda$ . For example at a frequency such that  $\lambda = 4d$  all angles lie within the range  $[-\pi/2d, \pi/2d]$  whilst conversely for  $\lambda = d$  the range of  $k_y$  is  $[-2\pi/d, 2\pi/d]$ . Thus at frequencies below the half-wavelength of the array the range of  $k_y$  corresponding to all possible physical angles is compressed whilst at frequencies above the half-wavelength of the array the range is expanded. This is best illustrated graphically below where the range of  $k_y$  versus the frequency of the incident signal is illustrated.

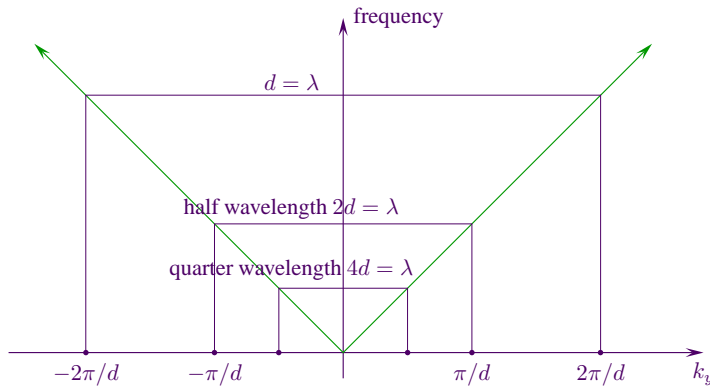


FIGURE 7.9. Frequency-wavenumber representation

With some looseness of terminology this has been traditionally termed the frequency-wavenumber representation. Since  $k_y = 2\pi f \sin \theta / c$  it follows that straight lines radiating from the origin correspond to particular values of  $\theta$ . Directions of particular significance are the end-fire lines corresponding to  $\sin \theta = \pm 1$  and the broadside line corresponding to  $\sin \theta = 0$ . The end-fire lines  $k_y = \pm 2\pi f / c$  split the frequency-wavenumber representation into two regions..

- (i) Within the interior connected region all values of  $k_y$  correspond to values of  $\theta$  lying between  $\pm\pi/2$ . Depending on terminology this region is often termed the “physical” or “real” region. In acoustics and sonar it is sometimes called the “acoustic region”. Lines of constant bearing have slope  $(2\pi f \sin \theta / c)^{-1}$ . In particular the broadside signal has an infinite slope as indicated above.
- (ii) The two regions such the  $|k_y| > 2\pi f / c$  would not be expected to contain energy propagating across the array as a plane wave with speed  $c$ . However often energy propagating through the superstructure of an array at a different speed  $c'$  is picked up by the array receivers and appears in these



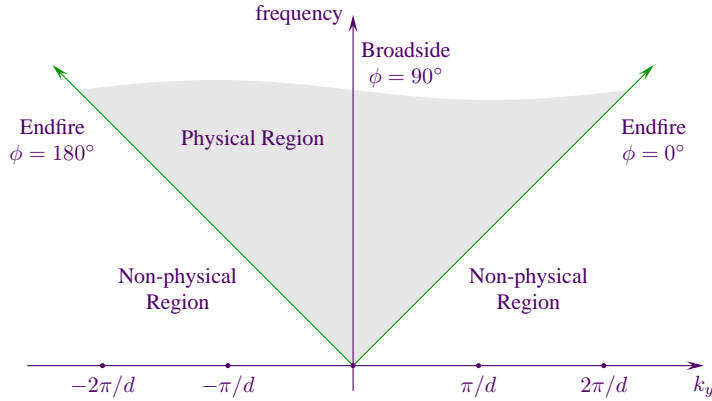


FIGURE 7.10. Regions of the frequency-wavenumber space

regions. This often occurs for towed sonar arrays and this additional energy appears as unwanted array self noise. Steering beams in this region determines how much of this unwanted noise is present.

### 7.8. Beam patterns in the frequency-wavenumber representation

The expression for the beam pattern, i.e.,

$$P_s(k_y, (\mathbf{k}_s)_y) = \frac{1}{K^2} \left( \frac{\sin(Kd((\mathbf{k}_s)_y - k_y)/2)}{\sin(d((\mathbf{k}_s)_y - k_y)/2)} \right)^2$$

shows that, as a function of  $(\mathbf{k}_s)_y$  and  $f$ , the beam pattern is independent of  $f$ . In the example below the broadside beam patterns for a 15 element array at  $d/\lambda$  of 1,  $1/2$  and  $1/4$  are illustrated. They are all identical but are simply shifted in frequency. The number of sidelobes within the ‘physical region’ varies with frequency. Also illustrated, for  $d/\lambda = 1$  are the grating lobes which as can be seen are simple due to periodicity of the Fourier transform induced by the uniform spatial sampling.

In general the beam pattern at a particular steering angle,  $\theta$ , and frequency,  $f_0$  can be simply obtained by the following translations. First slide the broadside beam pattern at  $d/\lambda = 1/2$  vertically to the desired frequency. Next, calculate the value of  $k_y$  by  $2\pi/cf_0 \sin \theta$  and slide the beam pattern horizontally to that value of  $k_y$ . In all translations the periodic nature of  $P(k_y, (\mathbf{k}_s)_y)$  must be taken into account. In summary all that is required is the basic shape of the broadside beam pattern at  $d/\lambda = 1/2$ . An example of this is shown below where the beam pattern for the following three cases: (a) broadside beam at  $d/\lambda = 1$ , (b)  $-30$  degree beam at  $d/\lambda = 1/2$  and (c) end-fire beam at  $d/\lambda = 1/4$  are plotted.

The concepts introduced here can readily be generalised to arbitrary arrays where a four-dimensional transform to the  $k_x, k_y, k_z, f$  domain can be effected.

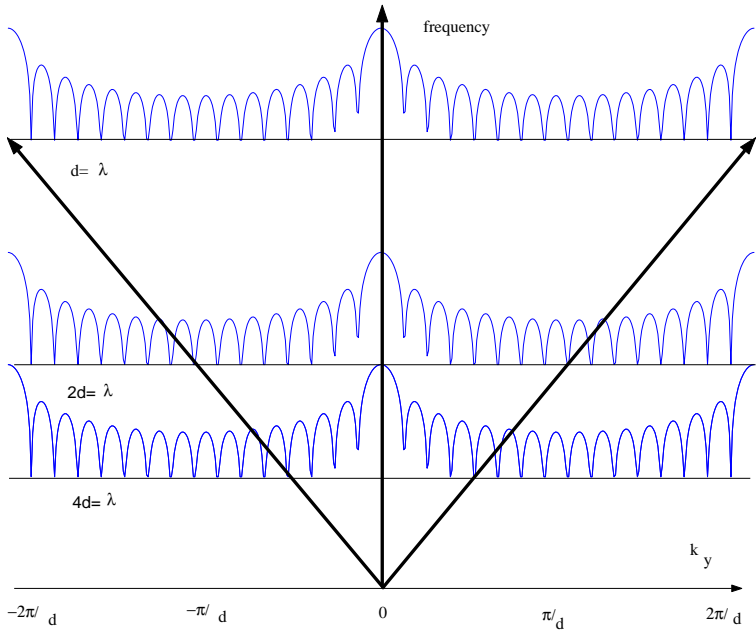


FIGURE 7.11. Frequency invariance of the wavevector beam pattern

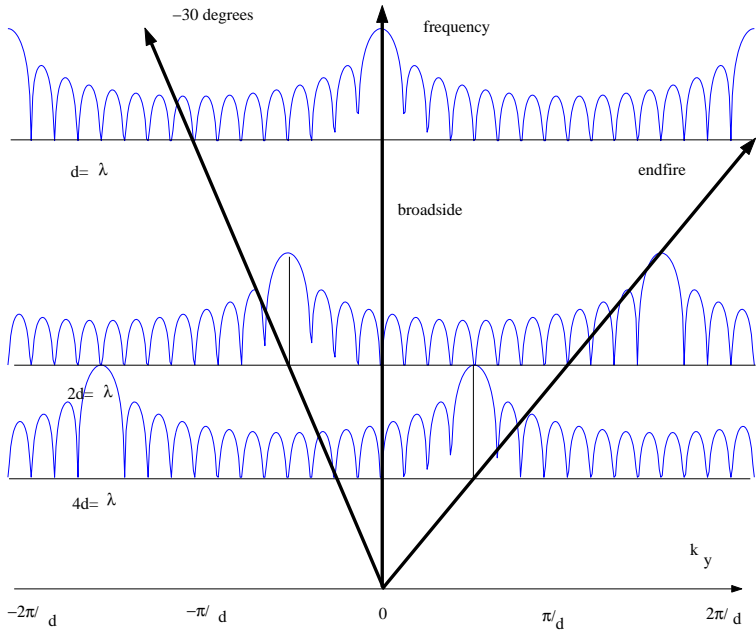


FIGURE 7.12. Frequencies and angles

## 7.9. Overview

This Chapter has emphasised the importance of the wavevector:

- in representing plane waves that traverse the array aperture,
- its relationship to the spatial variable  $\mathbf{u}$ ,
- wavevector beamforming,
- steering beams in wavevector, and
- beam patterns in frequency-wavenumber.

### Summary

- (1) When the space-time wave field  $f(\mathbf{u}, t)$  comprises the linear superposition of plane waves, it is the four-dimensional Fourier transform of the value at the origin  $\alpha(\mathbf{k}, \omega)$ .
- (2) For a linear array, the wavevector  $\mathbf{k}$  is the Fourier transform of the spatial variable  $\mathbf{u}$ .
- (3) In wavevector coordinates, the beamwidth of a linear array is independent of the steering direction.
- (4) For a linear array, the two-dimensional plot of steered beamformer output as a function of frequency and wavenumber has a “physical” or “real” region  $|k_y| \leq 2\pi f/c$ . The other regions where  $|k_y| > 2\pi f/c$  would not correspond to energy propagating with speed  $c$ , but is often provides useful indication of other, extraneous signals propagating with different velocities.

## CHAPTER 8

# BEAMFORMING - RANDOM PROCESSES

### 8.1. Introduction

So far beamforming discussions have been restricted to *deterministic* far field sources. However in practice an array is operated in the presence of both internal and external noise sources. For example, in sonar, random noise generated by the wind on the surface of the sea is a common source of external noise, whilst in radar and communications systems it is often internal receiver thermal noise that limits system performance. Furthermore the desired signals themselves may be noisy. For example, a spread spectrum communications or GPS signal can, from an array processing point of view, conveniently be represented as a random process.

In this Chapter the receiver outputs are represented as random processes and expressions for the mean output power of time and frequency domain beamformers are derived.<sup>1</sup>

### 8.2. Covariance function and covariance matrix

Consider the case where the outputs  $x_i(t)$  and  $x_j(t)$  of a pair of receivers are real random processes. In this case the **cross-covariance** function is defined as

$$\mathbf{R}_{X_i, X_j}(t, t) = E\{x_i(t), x_j(t)\}, \quad (8.1)$$

where  $E\{\cdot\}$  is formally defined by

$$E\{x_i(t_1), x_j(t_2)\} = \iint u_1 u_2 p(x_i(t_1) = u_1, x_j(t_2) = u_2) du_1 du_2 \quad (8.2)$$

and

$$p(x_i(t_1) = u_1, x_j(t_2) = u_2)$$

is the joint p.d.f of  $x(t_1)$  and  $x(t_2)$ .

Generalising to the case of a random vector the **covariance matrix**<sup>2</sup> of the receiver outputs of an array may be written as

---

<sup>1</sup>The reader is invited to revise some basic theory of random processes presented in Appendix B.

<sup>2</sup>Strictly this is a cross-covariance matrix but by convention its common name, covariance matrix, is used.

$$\begin{aligned} \mathbf{R}_x(t_1, t_2) &= E\{\mathbf{x}(t_1)\mathbf{x}^T(t_2)\} \\ &= \begin{bmatrix} E\{x_1(t_1), x_1(t_2)\} & E\{x_1(t_1), x_2(t_2)\} & \cdots & E\{x_1(t_1), x_K(t_2)\} \\ E\{x_2(t_1), x_1(t_2)\} & E\{x_2(t_1), x_2(t_2)\} & \cdots & E\{x_2(t_1), x_K(t_2)\} \\ \vdots & \vdots & \ddots & \vdots \\ E\{x_K(t_1), x_1(t_2)\} & E\{x_K(t_1), x_2(t_2)\} & \cdots & E\{x_K(t_1), x_K(t_2)\} \end{bmatrix} \end{aligned} \quad (8.3)$$

In many cases the output of each receiver is fed into a quadrature receiver prior to beamforming and, as discussed, in Chapter 2 the in-phase and quadrature components out of such a receiver are represented as a complex time series. In this case the covariance matrix of receiver outputs is defined as

$$\mathbf{R}_x(t_1, t_2) = E\{\mathbf{x}(t_1)\mathbf{x}^H(t_2)\}$$

where  $^H$  denotes the Hermitian transpose.

If the random processes are jointly stationary then the cross-covariance function is a function of time differences, i.e.,

$$\mathbf{R}_x(t_1, t_2) = \mathbf{R}_x(t_1 - t_2). \quad (8.4)$$

An important case is when the receiver outputs are correlated Gaussian random processes. In the complex Gaussian case the joint p.d.f. of the receiver outputs,  $p_{\mathbf{x}}(\mathbf{x}(t))$  is given by

$$p_{\mathbf{x}}(\mathbf{x}(t)) = \frac{1}{\pi^K |\mathbf{R}_x|} \exp(-\mathbf{x}^H(t) \mathbf{R}_x^{-1}(t_1, t_1) \mathbf{x}(t))$$

### 8.3. Cross-spectral matrix

In Chapter 6 frequency domain beamforming was considered using the Fourier transform of the output of the  $j^{\text{th}}$  receiver. Denoting this as  $\tilde{x}_j(f)$ , it is defined as

$$\tilde{x}_j(f) = \int_{-\infty}^{\infty} x_j(t) e^{-2\pi i f t} dt. \quad (8.5)$$

Since  $x_j(t)$  is a random process it follows that  $\tilde{x}_j(f)$  is a random function. Let us write

$$(\mathbf{R}_x)_{ij}(f) = E\{\tilde{x}_i(f)\tilde{x}_j^*(f)\}. \quad (8.6)$$

and define the **cross-spectral matrix**  $\mathbf{R}_x(f) \triangleq [(\mathbf{R}_x)_{ij}(f)]$ .

Letting  $\tilde{\mathbf{x}}(f)$  be the vector of the Fourier transformed receiver outputs:

$$\tilde{\mathbf{x}}(f) = \begin{bmatrix} \tilde{x}_1(f) \\ \tilde{x}_2(f) \\ \vdots \\ \tilde{x}_K(f) \end{bmatrix}, \quad (8.7)$$

it follows that

$$\mathbf{R}_x(f) = E\{\tilde{\mathbf{x}}(f)\tilde{\mathbf{x}}^H(f)\}. \quad (8.8)$$

### 8.4. Examples of the cross-spectral matrix

The cross-spectral matrices of signal and noise are defined as

$$\mathbf{R}_s(f) = E \{ \tilde{\mathbf{s}}(f) \tilde{\mathbf{s}}^H(f) \} \quad (8.9)$$

and

$$\mathbf{R}_n(f) = E \{ \tilde{\mathbf{n}}(f) \tilde{\mathbf{n}}^H(f) \}. \quad (8.10)$$

respectively.

If the signal and noise are uncorrelated, then

$$\begin{aligned} \mathbf{R}_x(f) &= E \{ \tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}^H(f) \} \\ &= E \{ (\tilde{\mathbf{s}}(f) + \tilde{\mathbf{n}}(f))(\tilde{\mathbf{s}}^H(f) + \tilde{\mathbf{n}}^H(f)) \} \\ &= E \{ \tilde{\mathbf{s}}(f) \tilde{\mathbf{s}}^H(f) \} + E \{ \tilde{\mathbf{n}}(f) \tilde{\mathbf{s}}^H(f) \} \\ &\quad + E \{ \tilde{\mathbf{s}}(f) \tilde{\mathbf{n}}^H(f) \} + E \{ \tilde{\mathbf{n}}(f) \tilde{\mathbf{n}}^H(f) \} \\ &= E \{ \tilde{\mathbf{s}}(f) \tilde{\mathbf{s}}^H(f) \} + 0 + 0 + E \{ \tilde{\mathbf{n}}(f) \tilde{\mathbf{n}}^H(f) \} \\ &= \mathbf{R}_s(f) + \mathbf{R}_n(f). \end{aligned} \quad (8.11)$$

When the signal comprises a superposition of  $N$  independent random processes, its cross-spectral matrix is the sum of the cross-spectral matrices of the  $N$  random process.

Recall that the vector of receiver outputs at frequency  $f$  for a signal incident upon the array with wave-vector  $\mathbf{k}_s$  can be written as

$$\mathbf{s}(\mathbf{k}_s, f) = \tilde{s}(f) \begin{bmatrix} \exp(i\mathbf{k}_s^T \mathbf{u}_1) \\ \exp(i\mathbf{k}_s^T \mathbf{u}_2) \\ \vdots \\ \exp(i\mathbf{k}_s^T \mathbf{u}_K) \end{bmatrix} = \tilde{s}(f) \mathbf{v}(\mathbf{k}_s), \quad (8.12)$$

where  $\tilde{s}(f)$  is the Fourier component of the incident signal at frequency  $f$  and  $\mathbf{u}_j$  is the position of the  $j$ -th receiver. In this case the cross-spectral matrix is given by

$$\begin{aligned} \mathbf{R}_s(f) &= E \{ \tilde{\mathbf{s}}(f) \tilde{\mathbf{s}}^H(f) \} \\ &= E \{ \tilde{s}(f) \mathbf{v}(\mathbf{k}_s) \tilde{s}^*(f) \mathbf{v}^H(\mathbf{k}_s) \} \\ &= E \{ \tilde{s}(f) \tilde{s}^*(f) \} \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s) \\ &= \sigma_s^2(f) \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s), \end{aligned} \quad (8.13)$$

where as  $\sigma_s^2(f)$  is the power of the incident signal at frequency  $f$ . In the presence of noise, which is uncorrelated with the signal, the cross-spectral matrix of the receiver output at frequency  $f$  is given by

$$\mathbf{R}_x(f) = \sigma_s^2(f) \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s) + \mathbf{R}_n(f). \quad (8.14)$$

The above expression can readily be extended to two random signals in noise where the signals are uncorrelated with each other:

$$\mathbf{R}_x(f) = \sigma_1^2(f) \mathbf{v}(\mathbf{k}_1) \mathbf{v}^H(\mathbf{k}_1) + \sigma_2^2(f) \mathbf{v}(\mathbf{k}_2) \mathbf{v}^H(\mathbf{k}_2) + \mathbf{R}_n(f). \quad (8.15)$$

For  $L$  uncorrelated signals the receiver output vector is

$$\mathbf{x}(f) = \sum_{\ell=1}^L s(\mathbf{k}_\ell) \mathbf{v}(\mathbf{k}_\ell) + \mathbf{n}(f) \quad (8.16)$$

Defining the  $(K \times L)$  matrix  $\mathbf{V}$  and the  $L$ -vector  $\mathbf{s}(f)$ :

$$\mathbf{V}(f) = \begin{bmatrix} \mathbf{v}(\mathbf{k}_1) & \mathbf{v}(\mathbf{k}_2) & \cdots & \mathbf{v}(\mathbf{k}_L) \end{bmatrix} \quad (8.17)$$

$$\text{and } \mathbf{s}(f) = [s(\mathbf{k}_1) \ s(\mathbf{k}_2) \ \cdots \ s(\mathbf{k}_L)]^T, \quad (8.18)$$

$$\text{we have } \mathbf{x}(f) = \mathbf{V}(f) \mathbf{s}(f) + \mathbf{n}(f). \quad (8.19)$$

$\mathbf{V}(f)$  is called the *array manifold matrix*. If the  $L$  signals are all independent of one another, the cross-spectral matrix becomes

$$\begin{aligned} \mathbf{R}_x(f) &= E\{\mathbf{x}(f) \mathbf{x}(f)^H\} \\ &= \mathbf{R}_s(f) + \mathbf{R}_n(f) \\ &= \mathbf{V}(f) \mathbf{S}(f) \mathbf{V}(f)^H + \mathbf{R}_n(f), \end{aligned} \quad (8.20)$$

$$\text{where } \mathbf{S}(f) = \text{diag}[\sigma^2(\mathbf{k}_1) \ \sigma^2(\mathbf{k}_2) \ \cdots \ \sigma^2(\mathbf{k}_L)]. \quad (8.21)$$

In the limit as the number of signals  $L$  approaches infinity we replace the above summation by an integral and write

$$\mathbf{R}_s(f) = \int \sigma^2(\mathbf{k}) \mathbf{v}(\mathbf{k}) \mathbf{v}^H(\mathbf{k}) d\mathbf{k}. \quad (8.22)$$

The above expression is often used as a model for distributed noise—as the following two examples illustrate.

(a) *Spherically isotropic noise*

Spherically isotropic noise is modelled by independent sources of equal strength uniformly distributed on the surface of a sphere of infinite radius. The field at the receiver then consists of an integral of plane waves whose power is uniform in angle and hence  $\mathbf{k}$  and the cross-spectral matrix of isotropic noise can be written as

$$\mathbf{R}_{\text{iso}}(f) = \int \sigma^2(\mathbf{k}) \mathbf{v}(\mathbf{k}) \mathbf{v}^H(\mathbf{k}) d\mathbf{k} \quad (8.23)$$

and it can be shown that  $\mathbf{R}_{\text{iso}}(f)$  reduces to

$$(\mathbf{R}_{\text{iso}})_{ij}(f) = \sigma_n^2(f) \frac{\sin[2\pi|\mathbf{u}_i - \mathbf{u}_j|/\lambda]}{2\pi|\mathbf{u}_i - \mathbf{u}_j|/\lambda}, \quad (8.24)$$

where  $\sigma_n^2(f)$  is the isotropic noise power at a receiver.

(b) *2-D Isotropic noise*

This case is similar to above except we assume the independent sources are located on a infinite plane. Note that this model is often used in sonar to model acoustic noise generated by the combined action of wind and waves on the surface of the ocean.

Without proof the cross-spectral matrix is stated to be

$$(\mathbf{R}_{2\text{d-iso}})_{ij}(f) = \sigma_n^2(f) J_0(2\pi|\mathbf{u}_i - \mathbf{u}_j|/\lambda), \quad (8.25)$$

where  $J_0$  is the zero-th order spherical Bessel function.

### 8.5. Frequency Domain Beamforming for Random Processes

In Chapter 5 it was shown that the Fourier transform  $\tilde{y}(\mathbf{k}, f)$  of the output of a conventional beamformer is given by

$$\tilde{y}(\mathbf{k}, f) = \frac{1}{K} \mathbf{v}^H(\mathbf{k}) \tilde{\mathbf{x}}(f) \quad (8.26)$$

Since  $x_j(t)$  is a random process it follows that  $\tilde{y}(\mathbf{k}, f)$  is a random function. The mean output power,  $p_{\text{conv}}(\mathbf{k})$ , of the beam steered in direction  $(\theta, \phi)$  is then given by

$$\begin{aligned} p_{\text{conv}}(\mathbf{k}) &= E \{ |\tilde{y}(\mathbf{k}, f)|^2 \} \\ &= E \left\{ \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}^H(f) \mathbf{v}(\mathbf{k}) \right\} \\ &= \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) E \{ \tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}^H(f) \} \mathbf{v}(\mathbf{k}) \\ &= \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{v}(\mathbf{k}) \end{aligned} \quad (8.27)$$

which expresses the mean output power in terms of the cross-spectral matrix.

### 8.6. Examples of beamforming with the cross-spectral matrix

#### 8.6.1. Signal and noise uncorrelated.

If the receiver outputs due to signal and noise are uncorrelated then

$$\mathbf{R}_x(f) = \mathbf{R}_s(f) + \mathbf{R}_n(f) \quad (8.28)$$

and thus it follows that

$$p_{\text{conv}}(\mathbf{k}) = \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{v}(\mathbf{k}) + \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{v}(\mathbf{k}) \quad (8.29)$$

This can be generalised to the case where the field at the array comprises the superposition of  $N$  independent random processes incident upon the array; the output beamformer power is the sum of the output beamformer powers of the  $N$  random processes.



### 8.6.2. Noises uncorrelated with one another.

When the noises at the receivers are uncorrelated with one another and have equal variance, the cross-spectral matrix is

$$\mathbf{R}_n = [E\{n_i(f)n_j^*(f)\}] = [\sigma_n^2 \delta_{jk}] \quad (8.30)$$

$$= \sigma_n^2 \mathbf{I}. \quad (8.31)$$

In this case the output power  $p_{\text{conv}}(\mathbf{k})$  of a beam steered in direction  $\mathbf{k}$  is given by

$$\begin{aligned} p_{\text{conv}}(\mathbf{k}) &= \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{v}(\mathbf{k}) \\ &= \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \sigma_n^2 \mathbf{I} \mathbf{v}(\mathbf{k}) \\ &= \frac{1}{K} \sigma_n^2 \end{aligned} \quad (8.32)$$

which is independent of the steering direction, i.e., the output beam power is independent of direction.

### 8.6.3. Single arrival.

For a single signal incident upon the array from direction  $\mathbf{k}_s$  with no noise the cross-spectral matrix of the receiver outputs can, from (8.13) be written as

$$\mathbf{R}_s(f) = \sigma_s^2(f) \mathbf{v}(\mathbf{k}) \mathbf{v}^H(\mathbf{k}) \quad (8.33)$$

and consequently the output power of a beam steered in direction  $\mathbf{k}$  given by

$$\frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{v}(\mathbf{k})$$

becomes

$$\frac{\sigma_s^2(f)}{K^2} \mathbf{v}^H(\mathbf{k}_s) \mathbf{v}(\mathbf{k}) \mathbf{v}^H(\mathbf{k}_s) \mathbf{v}(\mathbf{k})$$

which, apart from the power term,  $\sigma_s^2(f)$ , is the beam pattern of the array. Note that the maximum of this function occurs at  $\mathbf{k} = \mathbf{k}_s$ .

### 8.6.4. $L$ arrivals.

The above example can be generalised to the case in which  $L$  plane waves are incident upon the array from directions  $\mathbf{k}_1, \mathbf{k}_2, \dots, \mathbf{k}_L$ .

In this case the output of the conventional beamformer steered in direction  $\mathbf{k}$  can be shown to be given by

$$\sum_{\ell=1}^L \frac{\sigma_\ell^2(f)}{K^2} \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_\ell) \mathbf{v}^H(\mathbf{k}_\ell) \mathbf{v}(\mathbf{k}), \quad (8.34)$$

which can be recognised as the weighted sum of  $L$  beam patterns, with each beam pattern weighted by the power of the plane wave.

### 8.7. Time Delay and Sum Beamforming for Random Processes

Recall from Chapter 5 that the time series output of such a beamformer can be written as:

$$y(t, \theta, \phi) = \frac{1}{K} \sum x_j(t - \tau_j(\theta, \phi))$$

where

$$\tau_j(\theta, \phi) = \frac{1}{c} \{x_j \cos \theta \sin \phi + y_j \sin \theta \sin \phi + z_j \cos \phi\}$$

Since the receiver outputs are random processes it follows that  $y(t, \theta, \phi)$  is also a random process. Because of the random nature we are generally mainly interested in its mean output power which is given by

$$\begin{aligned} E \{ |y(t, \theta, \phi)|^2 \} &= E \left\{ \frac{1}{K^2} \left| \sum_{k=1}^K x_k(t - \tau_k(\theta, \phi)) \right|^2 \right\} \\ &= E \left\{ \frac{1}{K^2} \sum_{k=1}^K \sum_{l=1}^K x_k(t - \tau_k(\theta, \phi)) x_l(t - \tau_l(\theta, \phi)) \right\} \\ &= \frac{1}{K^2} \sum_{k=1}^K \sum_{l=1}^K E \{ x_k(t - \tau_k(\theta, \phi)) x_l(t - \tau_l(\theta, \phi)) \} \\ &= \frac{1}{K^2} \sum_{k=1}^K \sum_{l=1}^K \{ \mathbf{R}(\tau_k(\theta, \phi) - \tau_l(\theta, \phi)) \}_{kl} \end{aligned} \quad (8.35)$$

provided the random processes are stationary. This approach is rarely used in practice and is only given for completeness.

### 8.8. Overview

This Chapter the receiver outputs are represented as random processes and expressions have been derived for the mean power of beamformers. Central to the derivations are the cross-covariance function and cross-spectral matrix.

## Summary

### Definitions:

- (1) Cross-covariance function

$$\mathbf{R}_{X_i, X_j}(t, t) = E\{x_i(t), x_j(t)\}.$$

- (2) Cross-covariance matrix

$$\mathbf{R}_x(t_1, t_2) = E\{\mathbf{x}(t_1)\mathbf{x}^H(t_2)\}$$

- (3) Cross-spectral matrix

$$\begin{aligned}\mathbf{R}_x(f) &= [(\mathbf{R}_x)_{ij}(f)], \\ (\mathbf{R}_x)_{ij}(f) &= E\{\tilde{x}_i(f)\tilde{x}_j^*(f)\}.\end{aligned}$$

### Relationships

- (1) For stationary random processes,

$$\mathbf{R}_x(t_1, t_2) = \mathbf{R}_x(t_1 - t_2).$$

- (2) For complex  $\mathbf{x}(t)$ , the joint p.d.f. is given by

$$p_{\mathbf{x}}(\mathbf{x}(t)) = \frac{1}{\pi^K |\mathbf{R}_x|} \exp(-\mathbf{x}^H(t) \mathbf{R}_x^{-1} \mathbf{x}(t)).$$

- (3) If signal and noise are uncorrelated,

$$\mathbf{R}_x(f) = \mathbf{R}_s(f) + \mathbf{R}_n(f).$$

- (4) When the signal comprises a superposition of  $L$  independent plane-wave arrivals of power  $\sigma_1^2, \sigma_2^2, \dots, \sigma_L^2$ , the cross-spectral matrix of receiver outputs can be expressed as

$$\mathbf{R}_x = \mathbf{V} \mathbf{S} \mathbf{V}^H + \mathbf{R}_n,$$

where  $\mathbf{S} = \text{diag}[\sigma_s^2(\mathbf{k}_1) \ \sigma_s^2(\mathbf{k}_2) \ \dots \ \sigma_s^2(\mathbf{k}_L)]$

and  $\mathbf{V} = [\mathbf{v}(\mathbf{k}_1) \ \mathbf{v}(\mathbf{k}_2) \ \dots \ \mathbf{v}(\mathbf{k}_L)]$ .

- (5) For spherically isotropic noise,

$$(\mathbf{R}_{\text{iso}})_{ij} = \sigma_n^2 \frac{\sin[2\pi|\mathbf{u}_i - \mathbf{u}_j|/\lambda]}{2\pi|\mathbf{u}_i - \mathbf{u}_j|/\lambda}.$$

- (6) For 2-D (cylindrically) isotropic noise,

$$(\mathbf{R}_{\text{2d-iso}})_{ij} = \sigma_n^2 J_0(2\pi|\mathbf{u}_i - \mathbf{u}_j|/\lambda).$$

## CHAPTER 9

# ARRAY GAIN

### 9.1. Introduction

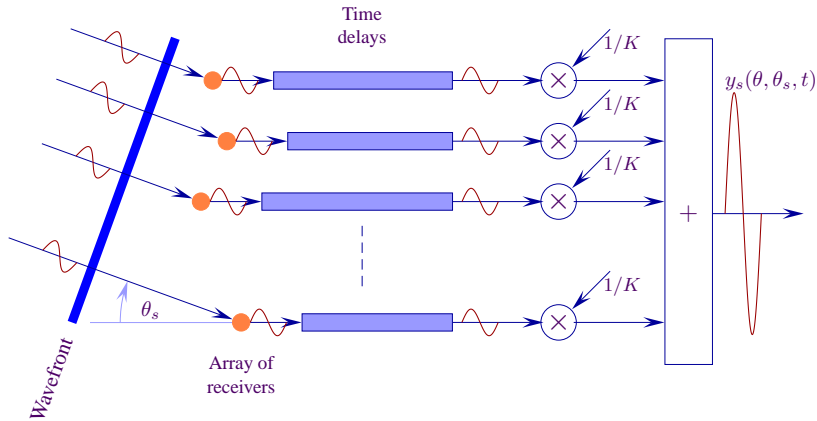


FIGURE 9.1. Beamsteering—general array

Recall that the conventional beamformer introduces delays (or, for narrowband signals, phase shifts) so as to bring all the wanted signals into phase.

This beamformer not only discriminates against unwanted signals (i.e., *interference*) but also suppresses noise. Consider the case (typical of internal noise) in which the noises from all receivers have the same noise power (variance) and are statistically independent of one another. The wanted signals are brought into phase and summed, as illustrated in Figure 9.2, so the amplitude is increased  $K$  times (and hence the power is increased  $K^2$  times).

However, the noises, being independent of one another, have random phases so the resultant noise *power* is increased only  $K$  times. The resulting signal-to-noise ratio is thus improved by a factor of  $K$  (or  $10 \log_{10} K$  dB). *This result applies to both narrowband and broadband beamformers.*

## 9.2. Expression for array gain

In what follows, we operate only in the frequency domain and assume that all signals and noises are zero-mean random processes.

Consider the signal-to-noise ratio at the output of an array of arbitrary geometry; beamforming is done by multiplying by complex scalars  $w_k^*(\mathbf{k})$ ,  $k = 1, \dots, K$ . Signals are assumed to arrive as plane waves. The output of the beamformer is

$$\sum_{k=1}^K w_k^*(\mathbf{k}) \tilde{x}_k(f) = \mathbf{w}^H(\mathbf{k}) \tilde{\mathbf{x}}(f) \quad (9.1)$$

and the mean power output of the array is  $E\{|\mathbf{w}^H(\mathbf{k}) \tilde{\mathbf{x}}(f)|^2\}$ ,

where  $\tilde{\mathbf{x}}(f) = \begin{bmatrix} \tilde{x}_1(f) \\ \tilde{x}_2(f) \\ \vdots \\ \tilde{x}_K(f) \end{bmatrix}$ ,

$\tilde{x}_k(f)$  is the output of the  $k^{\text{th}}$  receiver at frequency  $f$  and, as before,  $\mathbf{k}$  is the wavevector corresponding to the beamsteered direction. If the receiver output comprises signal from a direction with wavevector  $\mathbf{k}_s$  plus additive noise, we can write:

$$\tilde{\mathbf{x}}(f) = \tilde{\mathbf{s}}(\mathbf{k}_s, f) + \tilde{\mathbf{n}}(f), \quad (9.2)$$

where  $\tilde{\mathbf{s}}(\mathbf{k}_s, f)$  and  $\tilde{\mathbf{n}}(f)$  are the signal and noise vectors, both random complex  $K$ -dimensional vectors. For a plane wavefront arriving at the array, the signal

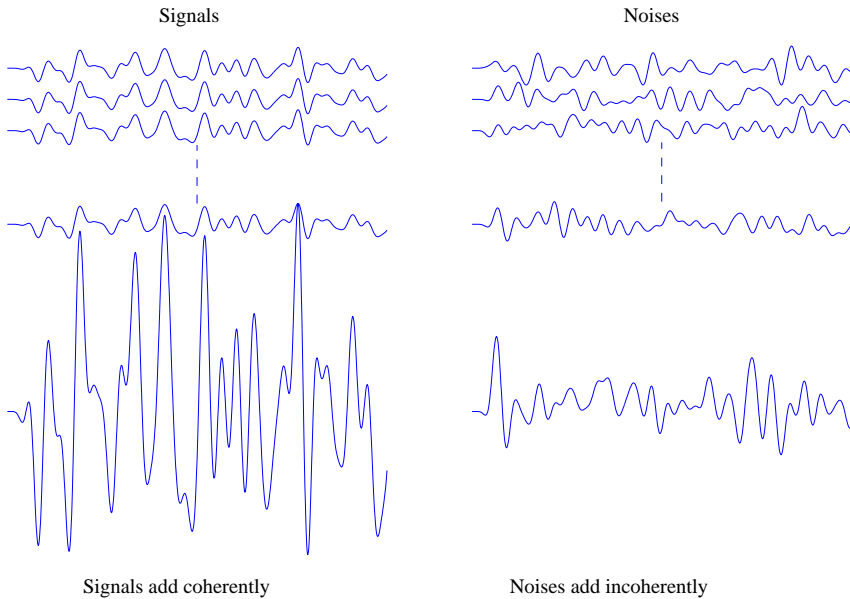


FIGURE 9.2. Illustrating coherent and incoherent summation

output from the array is  $\tilde{s}(\mathbf{k}_s, f) = \tilde{s}(f) \mathbf{v}(\mathbf{k}_s)$  (see (8.12)). Hence

$$\tilde{\mathbf{x}}(f) = \tilde{s}(f) \mathbf{v}(\mathbf{k}_s) + \tilde{\mathbf{n}}(f). \quad (9.3)$$

The signal and noise variances at the receivers are defined to be<sup>1</sup>

$$\sigma_s^2(f) = E\{|\tilde{s}(f)|^2\}, \quad \sigma_n^2(f) = E\{|\tilde{n}_k(f)|^2\}, \quad k = 1, \dots, K. \quad (9.4)$$

**Signal-to-noise ratio** is defined in the usual way:

$$SNR \triangleq \frac{\text{output power when only signal is present}}{\text{output power due to noise alone}}. \quad (9.5)$$

Hence the signal-to-noise ratio at a single receiver is

$$SNR_{\text{receiver}} = \sigma_s^2(f) / \sigma_n^2(f). \quad (9.6)$$

In the absence of noise and when the signal arrives from direction  $\mathbf{k}_s$  and the beam is steered in direction  $\mathbf{k}$ , the array output power is

$$\begin{aligned} p_s &= E \left\{ |\mathbf{w}^H(\mathbf{k}) \tilde{s}(\mathbf{k}_s, f)|^2 \right\} \\ &= E \left\{ \mathbf{w}^H(\mathbf{k}) \tilde{s}(\mathbf{k}_s, f) \tilde{s}^H(\mathbf{k}_s, f) \mathbf{w}(\mathbf{k}) \right\} \\ &= \mathbf{w}^H(\mathbf{k}) E \left\{ \tilde{s}(\mathbf{k}_s, f) \tilde{s}^H(\mathbf{k}_s, f) \right\} \mathbf{w}(\mathbf{k}) \\ &= \mathbf{w}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{w}(\mathbf{k}). \end{aligned} \quad (9.7)$$

When no signal is present, the output power from the array is

$$\begin{aligned} p_n &= E \left\{ |\mathbf{w}^H(\mathbf{k}) \tilde{\mathbf{n}}(f)|^2 \right\} \\ &= E \left\{ \mathbf{w}^H(\mathbf{k}) \tilde{\mathbf{n}}(f) \tilde{\mathbf{n}}^H(f) \mathbf{w}(\mathbf{k}) \right\} \\ &= \mathbf{w}^H(\mathbf{k}) E \left\{ \tilde{\mathbf{n}}(f) \tilde{\mathbf{n}}^H(f) \right\} \mathbf{w}(\mathbf{k}) \\ &= \mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k}). \end{aligned} \quad (9.8)$$

$\mathbf{R}_s(f)$  and  $\mathbf{R}_n(f)$  are the signal and noise cross-spectral matrices which were defined in (??) and (??). Note that the diagonal elements of  $\mathbf{R}_n(f)$  are

$$[\mathbf{R}_n(f)]_{jj} = E \left\{ \tilde{n}_j(f) \tilde{n}_j^*(f) \right\} = \sigma_n^2(f) \quad \forall j \quad (9.9)$$

and hence

$$\sigma_n^2(f) = \frac{1}{K} \text{Tr}\{\mathbf{R}_n(f)\}, \quad (9.10)$$

where  $\text{Tr}(\cdot)$  is the trace of a matrix which is defined to be

$$\text{Tr}(\mathbf{A}) \triangleq \sum_j a_{jj}. \quad (9.11)$$

Similarly

$$\sigma_s^2(f) = \frac{1}{K} \text{Tr}\{\mathbf{R}_s(f)\}. \quad (9.12)$$

---

<sup>1</sup>The receivers are assumed to be identical, so the receiver noise powers are equal.

The signal-to-noise ratio of the beamformer output using  $\mathbf{w}(\mathbf{k})$  is

$$SNR_{\text{beam}} = \frac{p_s}{p_n} = \frac{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{w}(\mathbf{k})}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})}. \quad (9.13)$$

The **array gain** is defined to be the ratio of the  $SNR$  at the output of the beamformer to that at a receiver:

$$G \triangleq \frac{SNR_{\text{beam}}}{SNR_{\text{receiver}}}. \quad (9.14)$$

Substituting (9.6), (9.10), (9.12) and (9.13) in (9.14) we obtain the following expression for the array gain:

$$\begin{aligned} G &= \left( \frac{\sigma_n^2(f)}{\sigma_s^2(f)} \right) \left( \frac{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{w}(\mathbf{k})}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})} \right) \\ &= \left( \frac{\text{Tr}\{\mathbf{R}_n(f)\}}{\text{Tr}\{\mathbf{R}_s(f)\}} \right) \left( \frac{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{w}(\mathbf{k})}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})} \right). \end{aligned} \quad (9.15)$$

Note that (9.15) applies to any beamformer. For the conventional unshaded beamformer, the weights are equal to the phase factors ( $\mathbf{w}(\mathbf{k}) = \mathbf{v}(\mathbf{k})$ ). For shaded beamformers they take the form  $\{w_j(\mathbf{k})\} = \{\alpha_j v_j(\mathbf{k})\}$  but in general we can use any weights to achieve a desired result<sup>2</sup>.

For a plane wave signal with a wavevector  $\mathbf{k}_s$  (see (8.13))

$$\mathbf{R}_s(f) = \sigma_s^2(f) \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s),$$

and hence, substituting in (9.15), the array gain for a plane-wave signal is

$$\begin{aligned} G &= \left( \frac{\sigma_n^2(f)}{\sigma_s^2(f)} \right) \left( \frac{\sigma_s^2(f) \mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s) \mathbf{w}(\mathbf{k})}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})} \right) \\ &= \frac{\text{Tr}(\mathbf{R}_n(f))}{K} \left( \frac{|\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_s)|^2}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})} \right). \end{aligned} \quad (9.16)$$

Note that the expression for gain does not involve the signal variance  $\sigma_s^2(f)$ , but only the signal direction  $\mathbf{k}_s$ .

We are particularly interested in the array gain when the beam is steered directly at the wanted signal (i.e.,  $\mathbf{k} = \mathbf{k}_s$ ). In what follows we shall assume this to be so unless specifically stated to the contrary.

### 9.3. Conventional beamformer

We now derive expressions for the array gain for the unshaded (“*boxcar*”) conventional beamformer in a few special cases. For this beamformer, when the beam is

<sup>2</sup>In Chapter 9 we shall pursue this thought further.

steered in the signal direction,  $\mathbf{w}(\mathbf{k}_s) = \frac{1}{K} \mathbf{v}(\mathbf{k}_s) = \frac{1}{K} \mathbf{v}(\mathbf{k})$  so (9.16) becomes

$$G_{\text{conv}} = K \frac{\text{Tr}\{\mathbf{R}_n(f)\}}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{v}(\mathbf{k})}. \quad (9.17)$$

### 9.3.1. Independent noises.

Here we consider the case in which the noises at the receivers are all independent of one another and have identical power (variance). Then

$$\mathbf{R}_n(f) = \sigma_n^2(f) \mathbf{I}, \quad (9.18)$$

and

$$\text{Tr}\{\mathbf{R}_n(f)\} = K \sigma_n^2(f). \quad (9.19)$$

Substituting (9.18) and (9.19) in (9.17), the array gain is seen to be

$$G_{\text{conv}} = K \quad (9.20)$$

as expected.

The power from a conventional beamformer steered in the signal direction in the absence of signal and in the absence of noise are

$$\begin{aligned} p_n &= \mathbf{v}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{v}(\mathbf{k}) = K \sigma_n^2(f), \\ p_s &= \mathbf{v}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{v}(\mathbf{k}) = K^2 \sigma_s^2(f). \end{aligned} \quad (9.21)$$

### 9.3.2. Single interfering signal.

Next we take the case in which there is a single interfering source of power  $\sigma_i^2(f)$  from direction with wavevector  $\mathbf{k}_i$ , and no noise in the system. Then the cross-spectral matrix of the noise,  $\mathbf{R}_n(f)$ , takes the same form as the signal in (9.2) in the previous example, but with  $\sigma_i^2(f)$  and  $\mathbf{k}_i$  instead of  $\sigma_s^2(f)$  and  $\mathbf{k}_s$ :

$$\mathbf{R}_n(f) = \sigma_i^2(f) \mathbf{v}(\mathbf{k}_i) \mathbf{v}^H(\mathbf{k}_i). \quad (9.22)$$

The signal-to-noise ratio at the receivers is

$$SNR_{\text{receiver}} = \sigma_s^2(f) / \sigma_i^2(f) \quad (9.23)$$

and at the output of the beamformer is

$$\begin{aligned} SNR_{\text{beam}} &= \frac{K^2 \sigma_s^2(f)}{\sigma_i^2(f) \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_i) \mathbf{v}^H(\mathbf{k}_i) \mathbf{v}(\mathbf{k})} \\ &= \left( \frac{\sigma_s^2(f)}{\sigma_i^2(f)} \right) \frac{1}{|\mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_i)|^2 / K^2}. \end{aligned} \quad (9.24)$$

The array gain is then

$$G_{\text{conv}} = \frac{1}{|\mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_i)|^2 / K^2}. \quad (9.25)$$



Note that the expression in the denominator,  $|\mathbf{v}^H(\mathbf{k})\mathbf{v}(\mathbf{k}_i)|^2/K^2 = P(\mathbf{k}, \mathbf{k}_i)$ , the value of the beam pattern of the array in direction  $\mathbf{k}_i$  when beamsteered in the direction  $\mathbf{k}$ . Hence in this case the array gain is

$$G_{\text{conv}} = \frac{1}{P(\mathbf{k}, \mathbf{k}_i)}. \quad (9.26)$$

If the interfering source is located in a null of the beam pattern,  $P(\mathbf{k}, \mathbf{k}_i) = 0$  and the gain becomes indefinitely large (in theory at least). This is to be expected because the beamforming in this idealised case, and for this particular frequency, has completely eliminated the interference.<sup>3</sup> When the beam is steered in a direction coincident with the interference,  $P(\mathbf{k}_i, \mathbf{k}_i) = 1$  and the gain is unity (i.e., there is no improvement in using an array over a single receiver).

### 9.3.3. Single interfering source and independent noises.

Next we consider the case in which there is a single interfering source of power  $\sigma_i^2(f)$  and wavevector  $\mathbf{k}_i$  and independent noise of power  $\sigma_n^2(f)$  which is the same for all receivers. Then

$$\mathbf{R}_n = \sigma_n^2(f)\mathbf{I} + \sigma_i^2(f)\mathbf{v}(\mathbf{k}_i)\mathbf{v}^H(\mathbf{k}_i), \quad (9.27)$$

$$SNR_{\text{receiver}} = \frac{\sigma_s^2(f)}{\sigma_n^2(f) + \sigma_i^2(f)}, \quad (9.28)$$

$$SNR_{\text{beam}} = \frac{K^2\sigma_s^2(f)}{K\sigma_n^2(f) + \sigma_i^2(f)|\mathbf{v}^H(\mathbf{k}_i)\mathbf{v}(\mathbf{k})|^2} \quad (9.29)$$

so

$$G_{\text{conv}} = \frac{K(1 + \beta)}{1 + \beta KP(\mathbf{k}, \mathbf{k}_i)}, \quad (9.30)$$

where  $\beta \triangleq \sigma_i^2(f)/\sigma_n^2(f)$  is the interference-to-receiver noise ratio (**INR**).

If  $\beta \ll 1$ ,  $G_{\text{conv}} \simeq K$ , as would be expected – this is the result when there is only independent noise present.

If  $\beta \gg 1$ ,  $G_{\text{conv}}$  varies from 1 (when the beam is steered directly at the interference) to  $K\beta$  (when the beam is steered in a direction such that the interference is at one of the nulls of the beam pattern. Array gain is plotted in Figure 9.3 as a function of signal arrival direction for a linear array with 15 elements spaced half a wavelength apart; there is a single 0 dB interference at  $45^\circ$  and 0 dB of independent self-noise.

As would be expected, the gain is high when the signal and interference directions are well separated.

---

<sup>3</sup>In practice there is always some internal noise present, as considered in the next section, in which case the gain is large but remains finite.

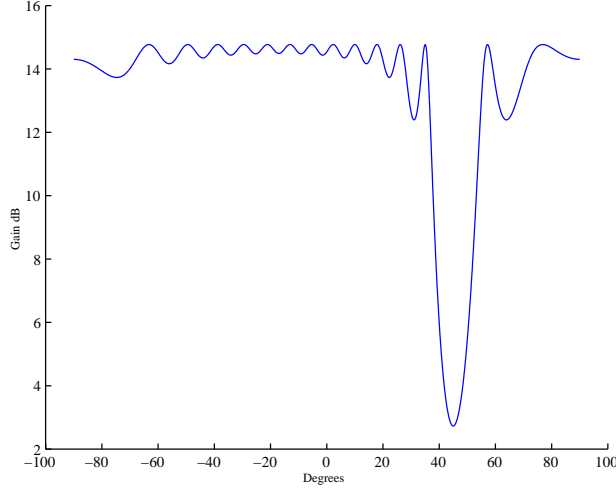


FIGURE 9.3. Array gain vs beamsteered direction. 15 receivers, single interference at  $(\theta_i = 45^\circ)$ ,  $(\phi_i = 90^\circ)$ ,  $INR = 0\text{dB}$

▷ Problem E.24

▷ Problem E.25

▷ Ex 6.1–6.4

#### 9.3.4. Spherically isotropic noise, uniform linear array steered at broadside.

Consider next the case of a uniform linear array, unshaded and steered at broadside, in a spherically isotropic noise field.<sup>4</sup> In this case the steering vector is

$$\mathbf{v}(\mathbf{k}) = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad (9.31)$$

and it can be shown that the elements of the cross-spectral matrix take the form<sup>5</sup>

$$[\mathbf{R}_n]_{jk} = \frac{\sigma_n^2(f) \sin(2\pi d(j-k)/\lambda)}{2\pi d(j-k)/\lambda}. \quad (9.32)$$

This is a special matrix called a symmetric **Toeplitz** matrix with the form:

$$\mathbf{R}_n = \sigma_n^2(f) \begin{bmatrix} 1 & r_2 & r_3 & \cdots & \cdots & r_K \\ r_2 & \mathbf{I} & r_2 & \cdots & \cdots & r_{K-1} \\ r_3 & \ddots & \ddots & \ddots & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \cdots & \ddots & \ddots & \ddots & r_2 \\ r_k & r_{K-1} & \cdots & r_3 & r_2 & \mathbf{I} \end{bmatrix}, \quad (9.33)$$

where

<sup>4</sup>In the sonar community, array gain in the case of spherically isotropic noise, no interfering sources and no internal noise, is also called the **directivity index**.

<sup>5</sup>See Chapter 8.

$$r_m = \frac{\sin(2\pi d(m-1)/\lambda)}{2\pi d(m-1)/\lambda}. \quad (9.34)$$

The array output power in the absence of signal is

$$\begin{aligned} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_n \mathbf{v}(\mathbf{k}) &= \mathbf{1}^T \mathbf{R}_n \mathbf{1} \\ &= [1 \quad \cdots \quad 1] \begin{bmatrix} \mathbf{R}_n \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \\ &= \sum_{k=1}^K \sum_{j=1}^K [\mathbf{R}_n]_{jk} \\ &= \sigma_n^2(f) \left\{ K + 2 \sum_{k=1}^{K-1} (K-k) r_k \right\} \\ &= \sigma_n^2(f) \left( K + 2 \sum_{k=1}^{K-1} (K-k) \frac{\sin(2\pi k d/\lambda)}{2\pi k d/\lambda} \right). \end{aligned} \quad (9.35)$$

Substituting (9.35) in (9.17) we have an expression for the array gain:

$$G_{\text{conv}} = \frac{K}{1 + \frac{2}{K} \sum_{k=1}^{K-1} (K-k) \frac{\sin(2\pi k d/\lambda)}{2\pi k d/\lambda}} \quad (9.36)$$

Let us see how array gain varies with  $d/\lambda$ . At  $d/\lambda = 0$ ,

$$\mathbf{R}_n = \sigma_n^2(f) \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix} = \sigma_n^2(f) \mathbf{1} \mathbf{1}^T, \quad (9.37)$$

and

$$\sum_{j,k} [\mathbf{R}_n]_{jk} = \sigma_n^2(f) K^2, \quad (9.38)$$

so substituting in (9.38) and (9.19) the array gain for  $d/\lambda = 0$  is  $G_{\text{conv}} = 1$ . As would be expected, there is no gain compared to a single receiver because we have in effect coalesced our array into a single point.

When  $d$  is a multiple of  $\lambda/2$ ,

$$\mathbf{R}_n = \sigma_n^2(f) \mathbf{I}, \quad (9.39)$$

and

$$G_{\text{conv}} = K. \quad (9.40)$$

Note that this is the same value for gain as we had obtained for independent noises (see (9.20)).<sup>6</sup> Array gain is shown plotted against  $d/\lambda$  in Figure 9.4 for an array of 15 receivers, for both the broadside and end-fire cases (the latter case is considered in the next section).

As frequency is increased, the array gain oscillates about the value  $K$ . Note that the gain can be increased significantly above  $K$  by judicious choice of array spacing.

<sup>6</sup> The noises in this particular case are uncorrelated but not necessarily independent. In the special case of Gaussian noise (i.e., with a normal distribution), *uncorrelated* implies *independent*.

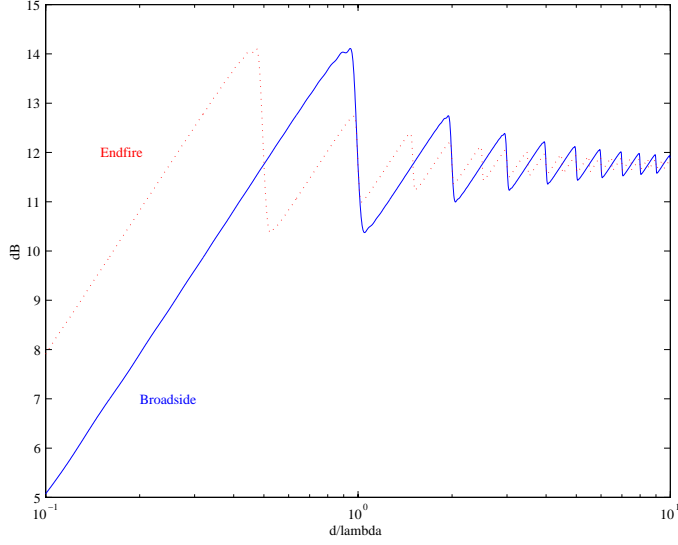


FIGURE 9.4. Array gain for broadside and end-fire arrays, plotted against  $d/\lambda$

### 9.3.5. Spherically isotropic noise, uniform linear array steered at end-fire.

This case is identical with the previous one (steered broadside), except that here, for the end-fire case,

$$\mathbf{v}(\mathbf{k}) = \begin{bmatrix} 1 \\ \exp(i2\pi d/\lambda) \\ \exp(i4\pi d/\lambda) \\ \vdots \\ \exp(i2(K-1)\pi d/\lambda) \end{bmatrix}. \quad (9.41)$$

When  $d = n\lambda/4$ , where  $n$  is an integer, it can be shown that the array gain is again  $K^7$ .

Array gain for the end-fire case is also shown plotted in Figure 9.4.

▷ Problem E.26

## 9.4. Array gain for shaded beamformer

Here we consider the case in which the noises are independent – i.e.,  $\mathbf{R}_n = \sigma_n^2(f)\mathbf{I}$ , and the beam is steered in the signal direction ( $\mathbf{k} = \mathbf{k}_s$ ). Substituting in (9.16), we have:

$$G_{\text{shaded}} = \frac{|\mathbf{w}^H(\mathbf{k})\mathbf{v}(\mathbf{k})|^2}{\mathbf{w}^H(\mathbf{k})\mathbf{w}(\mathbf{k})} \quad (9.42)$$

<sup>7</sup>The proof is left as an exercise for the reader.

TABLE 1. Gain for large  $K$ 

<i>Type of shading</i>	$\frac{G_{\text{shaded}}}{G_{\text{unshaded}}}$	$10 \log \left( \frac{G_{\text{shaded}}}{G_{\text{unshaded}}} \right)$ dB
Boxcar (unshaded)	1	0
Bartlett	0.747	-1.3
Blackman	0.577	-2.4
Hamming	0.732	-1.4
Hanning	0.669	-1.7
Kaiser (-30 dB sidelobes)	0.947	-0.2
Triangular	0.750	-1.2

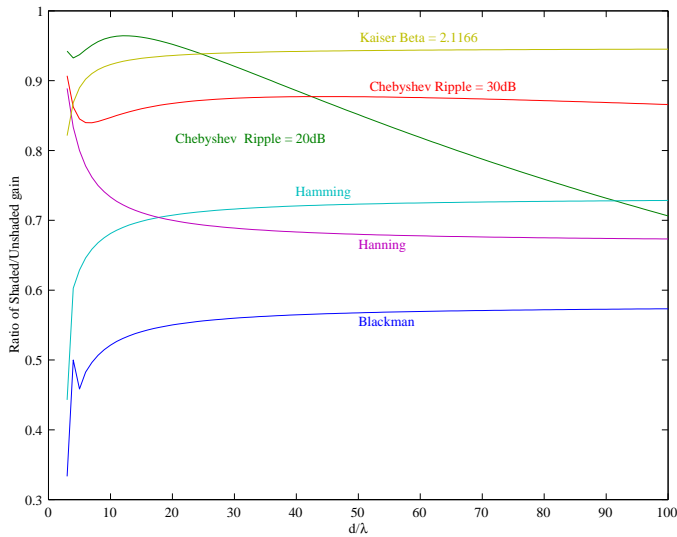


FIGURE 9.5. Array  $\text{gain}/K$  for uniformly spaced linear array with 15 receivers,  $d/\lambda = 0.5$ , plotted against  $K$ , independent noises.

The gain with shading is less than that for the unshaded conventional beamformer. Figure 9.5 shows a plot of the ratio of the gains with and without shading when the noise is uncorrelated. The ratio varies with the number of receivers  $K$ , and—for other than the Chebyshev case—tends to approach a constant value for large  $K$ . Table 9.4 displays the ratio of the gains with and without shading, for a uniform linear array of 300 receivers spaced half a wavelength apart.

### 9.5. Array gain with null steering

Here we again consider the case in which the noise is independent from receiver to receiver. Recall from (9.42) that

$$G = \frac{|\mathbf{w}^H(\mathbf{k})\mathbf{v}(\mathbf{k})|^2}{\mathbf{w}^H(\mathbf{k})\mathbf{w}(\mathbf{k})}$$

and that, for a beamformer that steers  $L$  nulls,

$$\mathbf{w}(\mathbf{k}) = \mathbf{A}^H (\mathbf{A} \mathbf{A}^H)^{-1} \boldsymbol{\delta}_1^{(L+1)},$$

and<sup>8</sup>

$$\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}) = 1$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{v}^H(\mathbf{k}) \\ \mathbf{v}^H(\mathbf{k}_1) \\ \vdots \\ \mathbf{v}^H(\mathbf{k}_L) \end{bmatrix},$$

$$\mathbf{A} \mathbf{A}^H = \begin{bmatrix} \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}) & \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_1) & \cdots & \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_L) \\ \mathbf{v}^H(\mathbf{k}_1) \mathbf{v}(\mathbf{k}) & \mathbf{v}^H(\mathbf{k}_1) \mathbf{v}(\mathbf{k}_1) & & \\ \vdots & & \ddots & \\ \mathbf{v}^H(\mathbf{k}_L) \mathbf{v}(\mathbf{k}) & \cdots & & \mathbf{v}^H(\mathbf{k}_L) \mathbf{v}(\mathbf{k}_L) \end{bmatrix},$$

and

$$\boldsymbol{\delta}_1^{(L+1)} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{[(L+1) \times 1]}$$

Then

$$\begin{aligned} \mathbf{w}(\mathbf{k})^H \mathbf{R}_n \mathbf{w}(\mathbf{k}) &= \sigma_n^2 \boldsymbol{\delta}_1^{(L+1)T} (\mathbf{A} \mathbf{A}^H)^{-1} \mathbf{A} \mathbf{A}^H (\mathbf{A} \mathbf{A}^H)^{-1} \boldsymbol{\delta}_1^{(L+1)} \\ &= \sigma_n^2 \boldsymbol{\delta}_1^{(L+1)T} (\mathbf{A} \mathbf{A}^H)^{-1} \boldsymbol{\delta}_1^{(L+1)} \\ &= \sigma_n^2 [(\mathbf{A} \mathbf{A}^H)^{-1}]_{11} \quad \text{and} \end{aligned} \tag{9.43}$$

$$G = \frac{1}{[(\mathbf{A} \mathbf{A}^H)^{-1}]_{11}}. \tag{9.44}$$

We now make use of the following inequality[33, p.74]:

*If  $\mathbf{A}_{[K \times K]}$  is positive definite and  $\mathbf{A}^{-1} = [a^{ij}]$ , then  $a^{ii} \geq 1/a_{ii}$ , with equality if and only if  $a_{ij} = 0$  for  $j = 1, \dots, i-1, i+1, \dots, K$ .*

Noting that  $\mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}) = K$ , we have

$$G \leq K,$$

with equality if and only if

$$\mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_1) = \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_2) = \dots = \mathbf{v}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_L) = 0$$

---

<sup>8</sup>This was the constraint imposed when setting up the null-steering beamformer.

– in other words, if and only if all the steered nulls coincide with the naturally-occurring nulls of the conventional beamformer. (Of course, in this case, the null-steering beamformer accomplishes nothing at all, as it only steers nulls where they would have occurred anyway.)

We conclude that

*the gain of a null-steered beamformer in the presence of uncorrelated noise is always less than that of the conventional beamformer, unless all steered nulls coincide with the nulls of the conventional beamformer.*

As a rule of thumb, the greater the suppression of the sidelobes or of the main lobe, the lower the gain of the null-steering beamformer.

### 9.6. Array gain expressed in terms of the beam pattern

Let us consider again the case in which the noise field can be modelled as arising from a large number  $N$  of independent far-field sources. Recall the earlier result from Chapter 7 that the noise cross-spectral matrix is

$$\mathbf{R}_n = \sum_{m=1}^N \sigma_m^2(\mathbf{k}_m) \mathbf{v}(\mathbf{k}_m) \mathbf{v}^H(\mathbf{k}_m), \quad (9.45)$$

where  $\sigma_m^2(\mathbf{k}_m)$  and  $\mathbf{k}_m$  are the power and wavevector of the  $m^{\text{th}}$  far-field source, respectively.

From (9.15) and (9.45), the array gain for a general weighting vector  $\mathbf{w}(\mathbf{k})$  is

$$\begin{aligned} G &= \left( \frac{\text{Tr}(\mathbf{R}_n(f))}{\text{Tr}(\mathbf{R}_s(f))} \right) \left( \frac{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{w}(\mathbf{k})}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})} \right) \\ &= \left( \frac{K \sum_m \sigma_m^2}{K \sigma_s^2(f)} \right) \left( \frac{\sigma_s^2(f) \mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s) \mathbf{w}(\mathbf{k})}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})} \right) \\ &= \frac{(\sum_m \sigma_m^2) |\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_s)|^2}{\sum_m \sigma_m^2 |\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_m)|^2}. \end{aligned} \quad (9.46)$$

Recalling that the beam pattern for a beam steered in direction  $\mathbf{k}$  is

$$P(\mathbf{k}, \mathbf{k}_s) = \frac{|\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}_s)|^2}{K^2} \quad (9.47)$$

and that by convention the weights  $\mathbf{w}(\mathbf{k})$  are scaled so that  $P(\mathbf{k}_s, \mathbf{k}_s) = 1$ , then the expression for array gain can be written in terms of the beam pattern:

$$G = \frac{\sum_m \sigma_m^2}{\sum_m \sigma_m^2 P(\mathbf{k}, \mathbf{k}_m)}. \quad (9.48)$$

If the noise far-field sources can be considered to be continuously distributed, then (see Chapter 8, (8.22), (8.23)), (9.45) becomes an integral:

$$\mathbf{R}_n = \int \sigma(\mathbf{k}) \mathbf{v}(\mathbf{k}) \mathbf{v}^H(\mathbf{k}) d\mathbf{k}. \quad (9.49)$$

If instead of the wavevector  $\mathbf{k}_m$  we use angles  $(\theta, \phi)$ , then the array gain is

$$G = \frac{\int_{4\pi} N(\theta, \phi) d\Omega}{\int_{4\pi} N(\theta, \phi) P((\theta, \phi), (\theta_s, \phi_s)) d\Omega} \quad (9.50)$$

where  $N(\theta, \phi)$  is the power distribution of the far-field noise sources and

$$P((\theta, \phi), (\theta_s, \phi_s))$$

is the array beam pattern, expressed here as a function of angles  $(\theta, \phi)$  and signal arrival angles  $(\theta_s, \phi_s)$ ; integrals are taken over solid angle  $\Omega$ .

## 9.7. Overview

In this chapter we have derived an expression for the gain of an array and illustrated with a number of simple examples. We have seen the tradeoff between array gain and sidelobe level when we use shading or null-steering. We have also shown how array gain can be expressed in terms of the beam pattern.



### Summary

(1) Array gain  $G$  is defined as the ratio of signal-to-noise ratio at the output of the array, to that at one of the receivers.

(2) The general expression for array gain is

$$G = \left( \frac{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_s(f) \mathbf{w}(\mathbf{k})}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})} \right) \left( \frac{\text{Tr}\{\mathbf{R}_n(f)\}}{\text{Tr}\{\mathbf{R}_s(f)\}} \right).$$

(3) If only independent noise is present then

(a) for the conventional beamformer,  $G = K$ ,

(b) with shading,  $G < K$ ,

(c) with null-steering,  $G < K$  unless all the nulls are coincident with those of the conventional beamformer.

(4) For a linear array of equi-spaced receivers immersed in spherically isotropic noise, the cross-spectral matrix is Toeplitz.

(5) If the noise field is due solely to far-field independent sources, then the array gain can be expressed in terms of the beampattern  $P(\mathbf{k}, \mathbf{k}_m)$ :

$$G = \frac{\sum_m \sigma_m^2}{\sum_m \sigma_m^2 P(\mathbf{k}, \mathbf{k}_m)}.$$

## CHAPTER 10

# OPTIMAL PROCESSING

### 10.1. Introduction

The conventional beamformer and the shading techniques we have covered in Chapters 3 and 4 use weights of the form

$$w_j(\mathbf{k}) = \alpha_j v_j(\mathbf{k}), \quad j = 1, \dots, K \quad (10.1)$$

where

$$v_j(\mathbf{k}) = \exp(i\mathbf{k}^T \mathbf{u}_j), \quad (10.2)$$

$\mathbf{u}_j$  is the vector of coordinates of the  $j^{\text{th}}$  receiver and the  $\{\alpha_j\}$  are real-valued weights designed to control sidelobe levels.

Then we considered more general weights  $w_j(\mathbf{k})$  that were designed to steer nulls in the beam pattern.

A natural extension is to select weights so as to optimise the signal processing in some way. Various “optimal” processors have been proposed, with different criteria for optimality. Here we consider

- maximising the array gain
- minimum power with linear constraint
- minimum mean-square error criterion
- maximum likelihood criterion

In each case we shall work only in the frequency domain.

Readers are invited to revise some basic tools for multivariate optimisation, given in [Appendix A](#).

### 10.2. Maximising gain

#### 10.2.1. Optimal weights.

Since array gain is a useful measure of the performance of the system, we are led naturally to try to maximise it. Recall from (9.16) that the array gain for a beam steered in the direction,  $\mathbf{k}$ , of a plane wave incident upon the array *from the same direction* is given by

$$G = \left( \frac{\text{Tr}(\mathbf{R}_n(f))}{K} \right) \left( \frac{|\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k})|^2}{\mathbf{w}^H(\mathbf{k}) \mathbf{R}_n(f) \mathbf{w}(\mathbf{k})} \right). \quad (10.3)$$

We wish to maximise  $G$  with respect to  $\mathbf{w}(\mathbf{k})$ . In Appendix A it is shown that

$$G = \frac{|\mathbf{w}^H(\mathbf{k})\mathbf{v}|^2}{(\mathbf{w}^H(\mathbf{k})\mathbf{R}\mathbf{w}(\mathbf{k}))}$$

is a maximum when

$$\mathbf{R}\mathbf{w}(\mathbf{k}) = \left( \frac{\mathbf{w}^H(\mathbf{k})\mathbf{v}}{\mathbf{w}^H(\mathbf{k})\mathbf{R}\mathbf{w}(\mathbf{k})} \right) \mathbf{v}, \quad (10.4)$$

so, if the inverse exists,

$$\mathbf{w}_{\max \text{ gain}}(\mathbf{k}) = \alpha \mathbf{R}_n^{-1}(f)\mathbf{v}(\mathbf{k}), \quad (10.5)$$

where  $\alpha$  is some scaling factor. Since  $|\alpha|^2$  appears in both the numerator and denominator of the expression for  $G$ , its actual value is *arbitrary*. Substituting (10.5) into (10.3) gives an expression for the maximum gain:

$$G_{\max \text{ gain}} = \frac{\text{Tr}(\mathbf{R}_n(f))}{K} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k}) \quad (10.6)$$

There are some aspects of this solution that are unsatisfactory.

- (1) The expression  $\mathbf{w}_{\max \text{ gain}}(\mathbf{k}) = \alpha \mathbf{R}_n^{-1}(f)\mathbf{v}(\mathbf{k})$  does not inform us about what value of  $\alpha$  we might use in practice.
- (2) Usually we cannot estimate the cross-spectral matrix of *noise*,  $\mathbf{R}_n(f)$ , but only that of *signal plus noise*,  $\mathbf{R}_x(f) = \mathbf{R}_s(f) + \mathbf{R}_n(f)$ .

We shall derive in Section 10.3, using a different criterion, an optimal processor that overcomes these difficulties.

### 10.3. Array gain vs beamsteered direction

Figure 10.1 shows array gain plotted against direction of arrival for a 15-element array with adjacent receivers equally spaced half a wavelength apart, with optimal and conventional beamforming; there are present two interferences at  $\pm 45^\circ$ , of amplitudes 0 and  $-6$  dB and uncorrelated noise of amplitude 0 dB. Note that the gain drops in the vicinity of the interferences but is otherwise high. Note also that the optimal and conventional gains are approximately equal in the directions of the interferences. It will be shown shortly that the gain of the optimal beamformer is always greater than or equal to that of the conventional beamformer.

In Figure 10.2 is shown plotted array gain for the maximum gain beamformer vs beamsteered direction, with several levels of uncorrelated receiver noise and a 0 dB interfering signal at an angle of  $45^\circ$ . Note that, the lower the noise level, the higher the gain in all directions except very close to the interference.

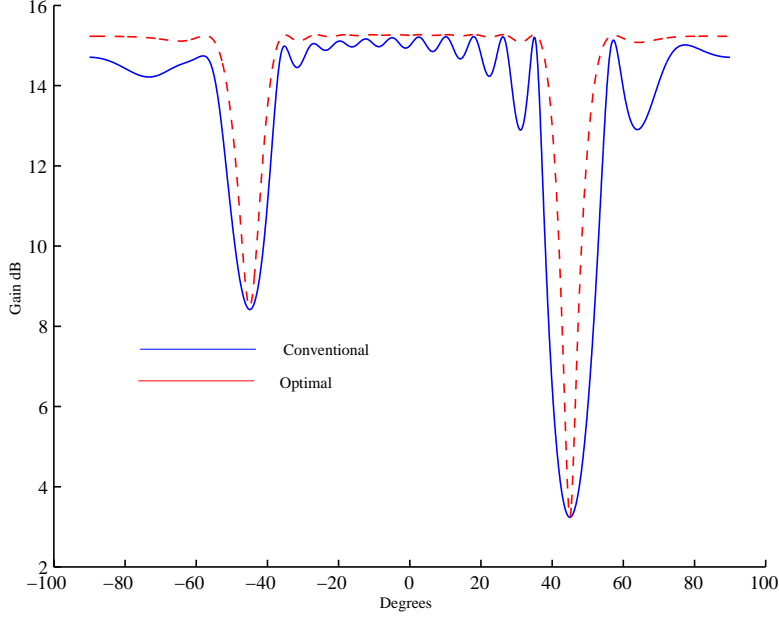


FIGURE 10.1. Array gain plotted against beamsteered direction for optimal and conventional beamforming. There are two interferences present, at  $\theta_i = \pm 45^\circ$  and  $INR = 0\text{dB}$  and  $INR = -6\text{dBdB}$ .

### 10.3.1. Comparison of Optimal and Conventional Processors.

It is useful to compare the gain from this optimal processor with that from the conventional beamformer. For the unshaded conventional beamformer,  $w(\mathbf{k}) = \mathbf{v}(\mathbf{k})/K$ , so from (10.3) and (10.6),

$$\frac{G_{\text{max gain}}}{G_{\text{conv}}} = \frac{\{\mathbf{v}^H(\mathbf{k})\mathbf{R}_n(f)\mathbf{v}(\mathbf{k})\}\{\mathbf{v}^H(\mathbf{k})\mathbf{R}_n^{-1}\mathbf{v}(\mathbf{k})\}}{K^2}.$$

Here we make use of the *Kantorovich inequality*[3, p. 70] which states that, for a positive definite Hermitian symmetric  $[K \times K]$  matrix  $\mathbf{R}$  with eigenvalues

$$\{\lambda_{\text{max}} \equiv \lambda_1 \geq \dots \lambda_k \geq \lambda_K \equiv \lambda_{\text{min}}\},$$

$$(\mathbf{x}^H \mathbf{x})^2 \leq (\mathbf{x}^H \mathbf{R} \mathbf{x})(\mathbf{x}^H \mathbf{R}^{-1} \mathbf{x}) \leq \frac{(\mathbf{x}^H \mathbf{x})^2}{4} \left\{ \left( \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} \right)^{1/2} + \left( \frac{\lambda_{\text{min}}}{\lambda_{\text{max}}} \right)^{1/2} \right\}^2. \quad (10.7)$$

Hence

$$1 \leq \frac{G_{\text{max gain}}}{G_{\text{conv}}} \leq \frac{1}{4} \left\{ \left( \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} \right)^{1/2} + \left( \frac{\lambda_{\text{min}}}{\lambda_{\text{max}}} \right)^{1/2} \right\}^2. \quad (10.8)$$

The above inequality reveals some interesting features of the optimal processor compared to the conventional.

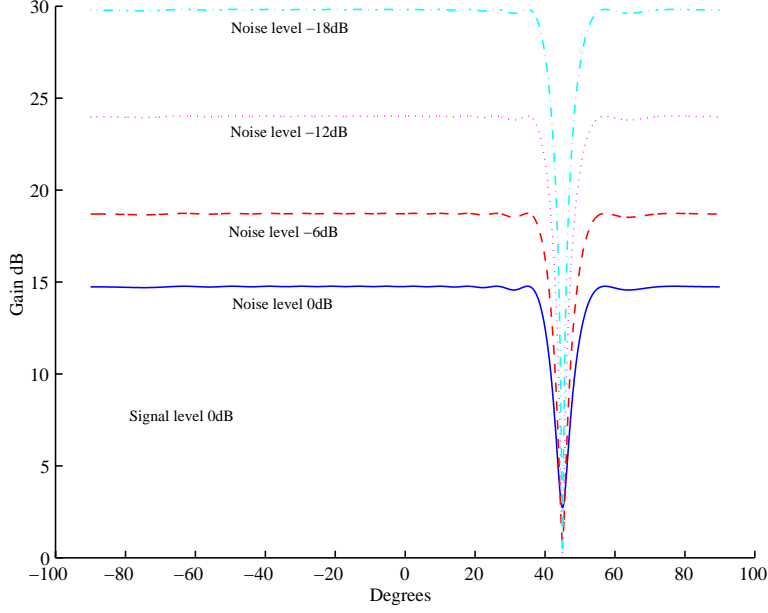


FIGURE 10.2. Array gain for maximum gain beamformer plotted against beamsteered direction for several values of noise. Single interference at  $\theta_i = 45^\circ$  with  $INR = 0, -6, -12$  and  $-18$ dB.

- (1) The gain of the optimal processor is always equal to or greater than that of the conventional.
- (2) When  $\mathbf{R}_n(f) = \sigma_n^2 \mathbf{I}$ ,  $\lambda_{\max} = \lambda_{\min}$  and  $G_{\max \text{ gain}} = G_{\text{conv}} = K$  – there is no improvement, as is expected.
- (3) Because the upper limit involves the ratio of  $\lambda_{\max}/\lambda_{\min}$ , we have the following important result.

*The optimal processor can only give a large improvement over the conventional if  $\lambda_{\max}/\lambda_{\min} \gg 1$  – i.e., if  $\mathbf{R}_n(f)$  is ill-conditioned.*

- (4) In practice there are constraints on the values that the signal vector  $\mathbf{v}(\mathbf{k})$  can take – because they must correspond to actual arrival directions – but it is illuminating to explore what would happen if there were no such constraints.

Let  $\{\mathbf{p}_1, \dots, \mathbf{p}_K\}$  be the eigenvectors of  $\mathbf{R}_n(f)$  corresponding to the eigenvalues  $\{\lambda_{\max} \equiv \lambda_1 \geq \dots \lambda_k \geq \lambda_K \equiv \lambda_{\min}\}$ .

- (a) If  $\mathbf{v}(\mathbf{k})$  could take a value equal to any eigenvector of  $\mathbf{R}_n(f)$ , then the lower bound would be reached, and again  $G_{\max \text{ gain}} = G_{\text{conv}}$ . Situations when this can occur are given in the examples to follow.
- (b) If  $\mathbf{v}(\mathbf{k})$  could take a value proportional to  $(\mathbf{p}_1 + \mathbf{p}_K)$ , then the upper bound would be attained, and  $G_{\max \text{ gain}}/G_{\text{conv}} = \frac{\lambda_1}{\lambda_K} + \frac{\lambda_K}{\lambda_1} + 2$ .

▷ Problem E.28

▷ Ex 7.1–7.4

## 10.4. Examples – Gain

### 10.4.1. Uncorrelated noise.

Consider the case in which the noises at the receivers are all uncorrelated with one another. Then

$$\mathbf{R}_n(f) = \sigma_n^2 \mathbf{I} \quad (10.9)$$

which, substituted in (10.7), gives:

$$G_{\max \text{ gain}} = K, \quad (10.10)$$

the same result as for the conventional unshaded beamformer. In this case, then, the optimal beamformer gives no improvement over the conventional beamformer, whatever the beamsteered direction.

### 10.4.2. Uncorrelated noises, single interference.

Here we include a single interference with wavevector  $\mathbf{k}_i$ , so

$$\mathbf{R}_n(f) = \sigma_n^2 \mathbf{I} + \sigma_i^2 \mathbf{v}(\mathbf{k}_i) \mathbf{v}^H(\mathbf{k}_i). \quad (10.11)$$

To calculate  $\mathbf{R}_n^{-1}$  we use the well-known Woodbury matrix identity<sup>1</sup>

$$(\mathbf{A} + c\mathbf{u}\mathbf{v}^H)^{-1} = \mathbf{A}^{-1} - \frac{c\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^H\mathbf{A}^{-1}}{1 + c\mathbf{v}^H\mathbf{A}^{-1}\mathbf{u}}. \quad (10.12)$$

Writing the *INR*  $\beta = \sigma_i^2/\sigma_n^2$ , and applying (10.12) to (10.11) gives

$$\mathbf{R}_n^{-1}(f) = \frac{1}{\sigma_n^2} \left( \mathbf{I} - \frac{\beta \mathbf{v}(\mathbf{k}_i) \mathbf{v}^H(\mathbf{k}_i)}{1 + K\beta} \right) \quad (10.13)$$

$$\text{and } G_{\max \text{ gain}} = \frac{K(1 + \beta)}{1 + K\beta} (1 + K\beta - K\beta P(\mathbf{k}, \mathbf{k}_i)), \quad (10.14)$$

where as before  $P(\mathbf{k}, \mathbf{k}_i)$  is the beam pattern of the array.

▷ Problem E.29

▷ Problem E.30

<sup>1</sup>Exercise: verify by direct multiplication.

### 10.4.3. Errors in beamsteered direction.

Thus far in this Chapter, we have been assuming (see (10.3)) that the beamsteered direction,  $\mathbf{k}$ , is exactly equal to the direction of the desired signal  $\mathbf{k}_s$  but in practice this may not be so.

Applying (9.16) we find that, when we maintain the distinction between  $\mathbf{k}$  and  $\mathbf{k}_s$ , the gain is:

$$G = \frac{\text{Tr}(\mathbf{R}_n(f))}{K} \left( \frac{|\mathbf{v}^H(\mathbf{k}) \mathbf{R}^{-1} \mathbf{v}(\mathbf{k}_s)|^2}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})} \right). \quad (10.15)$$

Figure 10.2 shows a plot of gain versus beamsteered direction for several SNRs. The fact that the peak response of the optimal beamformer is so sharp (particularly for strong signal-to-noise ratios) means that if the beamsteered direction differs from the direction of the wanted signal – even slightly – the processor will tend to consider the signal as an interference and reject it.

## 10.5. Minimum power with linear constraint

Here we consider a processor whose output is of the form

$$y(\mathbf{k}, f) = \mathbf{w}^H(\mathbf{k}) \tilde{\mathbf{x}}(f) \quad (10.16)$$

Since noise contributes to the power out of the array processor, it would be desirable to minimise the array output power:

$$\begin{aligned} E\{|y(\mathbf{k})|^2\} &= E\{\mathbf{w}^H(\mathbf{k}) \tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}(f)^H \mathbf{w}(\mathbf{k})\} \\ &= \mathbf{w}^H(\mathbf{k}) E\{\tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}(f)^H\} \mathbf{w}(\mathbf{k}) \\ &= \mathbf{w}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{w}(\mathbf{k}). \end{aligned} \quad (10.17)$$

We want to scale  $\mathbf{w}(\mathbf{k})$  so that the beam pattern is unity in the beamsteered direction, i.e.,

$$\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}) = 1 = \mathbf{v}^H(\mathbf{k}) \mathbf{w}(\mathbf{k}). \quad (10.18)$$

The problem here is therefore stated as:

$$\begin{aligned} &\text{Minimise } \mathbf{w}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{w}(\mathbf{k}) \\ &\quad \text{subject to } \mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}) = 1. \end{aligned}$$

This is an example of *optimisation subject to a linear constraint*, which is discussed in Appendix A. Because we are minimising  $\mathbf{w}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{w}(\mathbf{k})$  (the power out of the beamformer) whilst maintaining the MRA at unity, this is commonly called the **Minimum Variance Distortionless Response (MVDR)** processor.

Using (A.39) the solution is

$$\mathbf{w}_{\text{MVDR}}(\mathbf{k}) = \frac{\mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})}. \quad (10.19)$$

This expression for the weights under this criterion looks rather like (10.6) for the maximum array gain criterion but with important differences: (10.19) involves  $\mathbf{R}_x(f)$  instead of  $\mathbf{R}_n(f)$ , and the scaling factor  $\alpha$  is given explicitly.

Let us explore this similarity a little further. Recall that for a plane wave arrival

$$\mathbf{R}_x(f) = \mathbf{R}_s(f) + \mathbf{R}_n(f) = \sigma_s^2 \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s) + \mathbf{R}_n(f). \quad (10.20)$$

To calculate  $\mathbf{R}_x^{-1}(f)$  we use (10.12) to obtain, after a little manipulation,

$$\mathbf{w}_{\text{MVDR}}(\mathbf{k}) = \frac{\mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})} = \frac{\mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})}. \quad (10.21)$$

This is an important result. It shows that  $\mathbf{w}_{\text{max gain}}$  is the same as  $\mathbf{w}_{\text{MVDR}}$  if we set  $\alpha = \{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})\}^{-1}$ . Further, to derive the weights we can use the cross-spectral matrix of the outputs of the receivers (which we *can estimate*) rather than the cross-spectral matrix of the noise (which we *cannot*).

It is assumed in (10.21) that the signal wavevector  $\mathbf{v}(\mathbf{k})$  is known exactly. Often it is not, in which case we may plot the power out of the beamformer as a function of  $\mathbf{k}$  – or equivalently, the steering directions  $(\theta, \phi)$ <sup>2</sup>. We term this the *steered beamformer output*.

The mean output power,  $p_{\text{MVDR}}(\mathbf{k})$ , is given by  $E\{|\tilde{y}(\mathbf{k})|^2\}$ :

$$\begin{aligned} p_{\text{MVDR}}(\mathbf{k}) &= E\{|\tilde{y}(\mathbf{k})|^2\} \\ &= E\{\mathbf{w}^H(\mathbf{k}) \tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}^H(f) \mathbf{w}(\mathbf{k})\} \\ &= \mathbf{w}^H(\mathbf{k}) E\{\tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}^H(f)\} \mathbf{w}(\mathbf{k}) \\ &= \mathbf{w}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{w}(\mathbf{k}) \\ &= \frac{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{R}_x(f) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})}{(\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k}))^2} \\ &= \frac{1}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})} \end{aligned} \quad (10.22)$$

The above expression is the *output power of the MVDR beamformer*. The processor represented by (10.22) often is called the *adaptive beamformer*<sup>3</sup>.

## 10.6. Minimum mean-square error

In this approach we assume that we have a “desired signal”,  $d(f)$  to which we wish to approximate; we then select receiver weights so as to minimise the mean-square

<sup>2</sup>In effect, we use the beamformer as an *estimator* of the direction of arrival of the signal.

<sup>3</sup>The term was coined because the processor weights were determined not only by the desired steering direction but also by the noise cross-spectral matrix and were thus adapting to the noise field.



error between  $d(f)$  and the sum of the weighted receiver outputs  $\mathbf{w}^H(\mathbf{k})\tilde{\mathbf{x}}(f)$ <sup>4</sup>. The error signal is

$$\zeta = d(f) - \mathbf{w}^H(\mathbf{k})\tilde{\mathbf{x}}(f). \quad (10.23)$$

The problem then is to minimise  $E\{|\zeta(\mathbf{w}(\mathbf{k}))|^2\}$  with respect to  $\mathbf{w}(\mathbf{k})$ . When signals and noises are all stationary stochastic processes,

$$\begin{aligned} E\{|\zeta(\mathbf{w})|^2\} &= E|d(f)|^2 - 2\Re[E\{d^*(f)\mathbf{w}^H(\mathbf{k})\tilde{\mathbf{x}}(f)\}] \\ &\quad + \mathbf{w}^H(\mathbf{k})E\{\tilde{\mathbf{x}}(f)\tilde{\mathbf{x}}^H(f)\}\mathbf{w}(\mathbf{k}) \\ &= E\{|d(f)|^2\} - 2\Re\{\mathbf{w}^H(\mathbf{k})\mathbf{R}_{xd}\} + \mathbf{w}^H(\mathbf{k})\mathbf{R}_x(f)\mathbf{w}(\mathbf{k}), \end{aligned} \quad (10.24)$$

where

$$\mathbf{R}_{xd} = E\{d^*(f)\tilde{\mathbf{x}}(f)\}. \quad (10.25)$$

The solution is straightforward; setting

$$\nabla_{\mathbf{w}} E\{|\zeta(\mathbf{w})|^2\} = 0 = \nabla_{\mathbf{w}^*} E\{|\zeta(\mathbf{w})|^2\} \quad (10.26)$$

$$\text{we obtain } \mathbf{w}_{MMSE}(\mathbf{k}) = \mathbf{R}_x^{-1}(f)\mathbf{R}_{xd}. \quad (10.27)$$

In the case in which  $\mathbf{R}_{xd} \propto \mathbf{v}(\mathbf{k})$ , this processor takes the same form as the MVDR processor, except for a scaling coefficient.

## 10.7. Maximum likelihood estimator

Here we derive an optimal processor based on maximum likelihood estimation<sup>5</sup>. As before, we operate in the frequency domain but omit the explicit dependence on  $f$ .

We make the following assumptions:

- the desired plane-wave signal is *deterministic*,
- the noises are zero-mean *gaussian* variables with the same variance,
- signal and noises are complex variables,
- we know the cross-spectral matrix  $\mathbf{R}_n(f)$  of the noise.

The vector of receiver outputs can then be written as

$$\tilde{\mathbf{x}}(f) = s_0 \mathbf{v}(\mathbf{k}) + \mathbf{n}(f), \quad (10.28)$$

<sup>4</sup>This processor is the spatial version of the discrete-time finite interval Wiener filter. Knowledge of the ‘desired’ signal can be obtained in a number of ways. There could be prior knowledge of the signal – for example, there could be a coded component in part of the signal waveform, deliberately inserted to facilitate processing – or the ‘desired’ signal could be estimated by pointing a beam in its direction.

<sup>5</sup>Revision notes are provided in Appendix A

where  $s_0$  is a scalar *unknown* parameter which we wish to estimate. For brevity, as before, we omit the explicit dependence on frequency  $f$ .

The noise  $\mathbf{n}$  has a zero-mean multivariate complex normal distribution<sup>6</sup>:

$$p(\mathbf{n}) = \frac{1}{\pi^K |\mathbf{R}_n|} \exp\{-\mathbf{n}^H \mathbf{R}_n^{-1} \mathbf{n}\}. \quad (10.29)$$

Then we have

$$p(\tilde{\mathbf{x}}|s_0) = \frac{1}{\pi^K |\mathbf{R}_n|} \exp\left[-(\tilde{\mathbf{x}} - s_0 \mathbf{v}(\mathbf{k}))^H \mathbf{R}_n^{-1} (\tilde{\mathbf{x}} - s_0 \mathbf{v}(\mathbf{k}))\right] \quad (10.30)$$

and the log-likelihood function for an observation vector  $\tilde{\mathbf{x}}$  is

$$L(s_0, \tilde{\mathbf{x}}) = -K \ln(\pi) - \ln |\mathbf{R}_n| - (\tilde{\mathbf{x}} - s_0 \mathbf{v}(\mathbf{k}))^H \mathbf{R}_n^{-1} (\tilde{\mathbf{x}} - s_0 \mathbf{v}(\mathbf{k})). \quad (10.31)$$

Using the rules of differentiation of Appendix A Section A.4.1, the conditions for a maximum are

$$0 = \nabla_{s_0} L(s_0, \tilde{\mathbf{x}}) = 2 \mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1} (\tilde{\mathbf{x}} - s_0 \mathbf{v}(\mathbf{k})). \quad (10.32)$$

$$\text{and } 0 = \nabla_{s_0} L(s_0, \tilde{\mathbf{x}}) = 2 \mathbf{v}^T(\mathbf{k}) \mathbf{R}_n^{*-1} (\tilde{\mathbf{x}}^* - s_0^* \mathbf{v}^*(\mathbf{k})), \quad (10.33)$$

so the estimated value is<sup>7</sup>

$$\hat{s}_0 = \frac{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \tilde{\mathbf{x}}}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})} = \left( \frac{\mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})} \right)^H \tilde{\mathbf{x}}. \quad (10.34)$$

This estimator is just a beamformer with weights

$$\mathbf{w}_{\text{Capon}}(\mathbf{k}) = \frac{\mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})} \quad (10.35)$$

Note that the maximum likelihood estimate given by (10.34) is identical to the MVDR beamformer of (10.22) for plane-wave signals. It is remarkable that we have obtained the same result using a totally different approach.

---

<sup>6</sup>Many readers might be more familiar with the multivariate *real* normal distribution:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{K/2} |\mathbf{R}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \mathbf{R}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right\}.$$

<sup>7</sup>The approach is due to Capon ([6]) and the resulting processor often called the Capon estimator. We use  $\hat{\cdot}$  to denote the estimated value of a variable.

### 10.8. Summary of optimal processors

Each of the processors derived in this Chapter is a weighted beamformer.

Maximum Gain	$\mathbf{w}_{\max \text{ gain}}(\mathbf{k}) = \alpha \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})$
Minimum power with linear constraint	$\mathbf{w}_{\text{MVDR}}(\mathbf{k}) = \frac{\mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})}$
	$\mathbf{w}_{\text{MVDR}}(\mathbf{k}) = \frac{\mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})}$
Maximum likelihood estimator	$\mathbf{w}_{\text{Capon}}(\mathbf{k}) = \frac{\mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k})}$
Minimum mean-square error	$\hat{\mathbf{w}}_{\text{MMSE}}(\mathbf{k}) = \mathbf{R}_x^{-1}(f) \mathbf{R}_{xd}$

All involve the inverse of a cross-spectral matrix: in the case of the MVDR and Maximum Likelihood processors, the result is the same whether using the cross-spectral matrix of the noise  $\mathbf{R}_n(f)$  or of the observation vector  $\tilde{\mathbf{x}}(f)$ . The former,  $\mathbf{R}_n(f)$ , is usually unknown, but it is generally possible to estimate  $\mathbf{R}_x(f)$ .

## 10.9. Examples – Steered beamformer output

### 10.9.1. Independent receiver noises.

Consider the case in which there is no external signal and the noises from the receivers are independent of one another. In this case  $\mathbf{R}_x(f) = \sigma_n^2 \mathbf{I}$ ,

$$\mathbf{w}(\mathbf{k}) = \frac{\sigma_n^{-2} \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \sigma_n^{-2} \mathbf{v}(\mathbf{k})} = \frac{\mathbf{v}(\mathbf{k})}{K} \quad (10.36)$$

and

$$\begin{aligned} p_{\text{MVDR}}(\mathbf{k}) &= \frac{1}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})} \\ &= \sigma_n^2 / K \\ &= p_{\text{conv}}(\mathbf{k}). \end{aligned} \quad (10.37)$$

As would be expected, the optimum and conventional processors give identical results.

### 10.9.2. Single arrival, independent noises.

Consider the case in which the noise is due to independent receiver noise and a single plane-wave interference from a direction whose wavevector is  $\mathbf{k}_i$ . Using (7.15) (but substituting  $\sigma_i^2$  for  $\sigma_s^2$  and  $\sigma_n^2 \mathbf{I}$  for  $\mathbf{R}_n$ ), the cross-spectral matrix of the receiver outputs is seen to be

$$\mathbf{R}_x(f) = \sigma_i^2 \mathbf{v}(\mathbf{k}_i) \mathbf{v}^H(\mathbf{k}_i) + \sigma_n^2 \mathbf{I}, \quad (10.38)$$

and the output power from the conventional beamformer is

$$\begin{aligned}
p_{\text{conv}}(\mathbf{k}) &= \frac{1}{K^2} \{ \mathbf{v}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{v}(\mathbf{k}) \} \\
&= \frac{\sigma_n^2}{K} + \sigma_i^2 P(\mathbf{k}, \mathbf{k}_i),
\end{aligned} \tag{10.39}$$

where  $P(\mathbf{k}, \mathbf{k}_i)$  is the beam pattern of the array (see (3.55)). As discussed earlier, power from the signal incident upon the array from direction  $\mathbf{k}_i$  ‘leaks’ into the beam steered in direction  $\mathbf{k}$ , by an amount equal to the value of the beam pattern multiplied by  $\sigma_i^2$ .

From Woodbury’s identity (10.12), the inverse of  $\mathbf{R}_x(f)$  is

$$\mathbf{R}_x^{-1}(f) = \frac{1}{\sigma_n^2} \left( \mathbf{I} - \frac{\beta \mathbf{v}(\mathbf{k}_i) \mathbf{v}^H(\mathbf{k}_i)}{1 + \beta \mathbf{v}^H(\mathbf{k}_i) \mathbf{v}(\mathbf{k}_i)} \right) \tag{10.40}$$

where  $\beta \triangleq \sigma_i^2 / \sigma_n^2$  is the ratio of the interference to receiver noise. Using (10.40) it can be shown that the mean output power of the optimum beamformer is given by

$$p_{\text{MVDR}}(\mathbf{k}) = \frac{1}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})} \tag{10.41}$$

$$= \frac{\sigma_n^2 (1 + \beta K)}{(K + \beta K^2 - \beta |\mathbf{v}^H(\mathbf{k}_i) \mathbf{v}(\mathbf{k})|^2)} \tag{10.42}$$

Three special cases of this equation are of interest; in each of these, the optimal processor is the same as the conventional.

(i) As  $\beta \rightarrow 0$ ,  $p_{\text{MVDR}}(\mathbf{k}) \rightarrow p_{\text{conv}}(\mathbf{k})$ .

(ii) When  $\mathbf{k} = \mathbf{k}_i$

$$p_{\text{MVDR}}(\mathbf{k}) = \frac{\sigma_n^2 (1 + \beta K)}{K} = \frac{\sigma_n^2}{K} + \sigma_i^2 = p_{\text{conv}}(\mathbf{k}).$$

(iii) When  $\mathbf{v}^H(\mathbf{k}_i) \mathbf{v}(\mathbf{k}) = 0$ , i.e., at the nulls of the beam pattern,

$$p_{\text{MVDR}}(\mathbf{k}_i) = \frac{\sigma_n^2 (1 + \beta K)}{K(1 + \beta K)} = \frac{\sigma_n^2}{K} = p_{\text{conv}}(\mathbf{k}_i).$$

### 10.9.3. Multiple arrivals, independent noises.

The beamformer output power can be used as an estimator. If we plot  $p_{\text{conv}}(\theta)$  or  $p_{\text{MVDR}}(\theta)$ , the positions and amplitudes of the peaks in the plot provide an indication of direction of arrival and strength of the signals<sup>8</sup>.

Figure 10.3 shows a plot of the steered beamformer output from a linear array with 15 equally spaced receivers. There are three arrivals, with strengths of 0dB, –10dB and –15dB. Results are displayed using conventional and optimal (MVDR) processing.

<sup>8</sup>We only consider azimuthal angles  $\theta$  but the technique can readily be applied to both azimuth and elevation  $\phi$

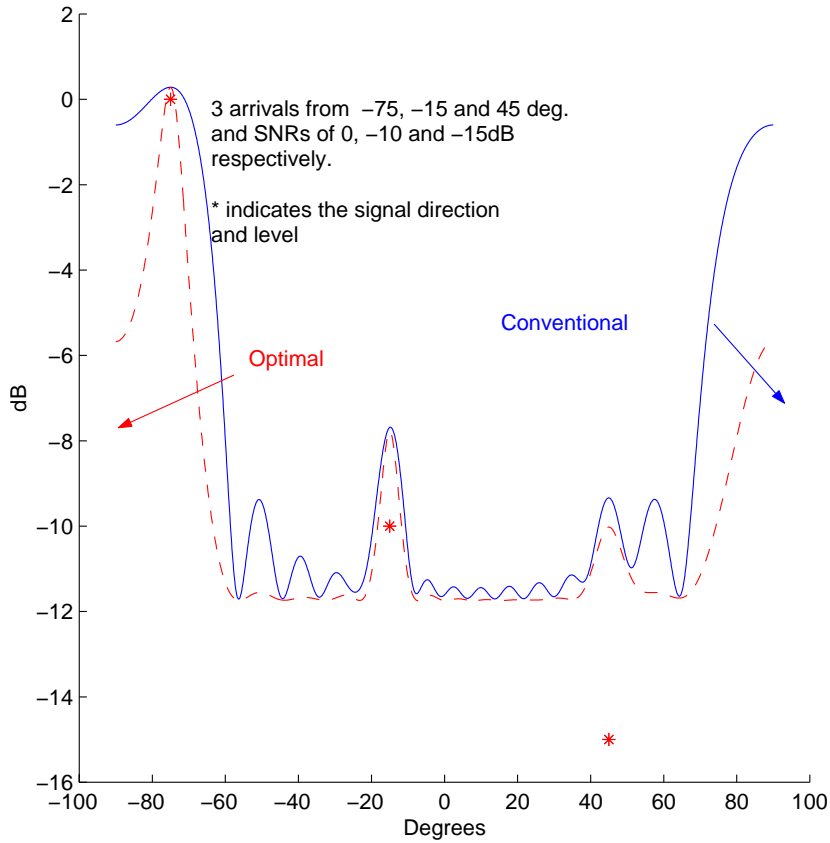


FIGURE 10.3. Steered beamformer output for 15-receiver linear array with conventional and optimal (MVDR) processing. There are three plane-wave arrivals, all with  $\theta_s = 90^\circ$ , from  $\theta_s = -75^\circ$ ,  $-15^\circ$  and  $45^\circ$ , and with SNRs of 0,  $-10$  and  $-15$  dB respectively.

The conventional beamformer can just detect the  $-10$  dB arrival; the MVDR processor can distinguish the  $-15$  dB signal.

The peaks in the steered beamformer output plot give a good indication of direction of arrival. However, for low signal-to-noise ratios a correction would have to be made for the power due to noise.

## 10.10. Discussion

### 10.10.1. Condition of the matrix $\mathbf{R}$ .

The expressions for the mean power of both the conventional and optimum beamformers are Hermitian forms in  $\mathbf{R}_x(f)$  and its inverse, respectively. Since  $\mathbf{R}_x(f)$  is a cross-spectral matrix, it is non-negative definite. However, for the optimum beamformer we require also that  $\mathbf{R}_x(f)$  be of full rank (positive definite) and

hence invertible. Many of the cross-spectral matrices encountered in beamforming are singular or ill-conditioned – indeed, it was proved earlier that the optimal beamformer cannot achieve a gain much higher than that of the conventional beamformer *unless* the cross-spectral matrix is ill-conditioned.

A simple way of ensuring that the matrix will be non-singular is to sacrifice some of the performance of the system and add a little uncorrelated receiver noise which is modelled as  $\mathbf{R}_x(f) \rightarrow \mathbf{R}_x(f) + \sigma_n^2(f)\mathbf{I}$ . This process is called *regularisation*.

### 10.10.2. Sensitivity to system parameters.

The optimal processors are very efficient in suppressing any interference in a direction different from that of the desired signal (the “target”). However, the very effectiveness of this suppression can create a problem: it was pointed out (see Section 10.4.3) that, if the beamsteered direction is not exactly directed towards the desired signal, that signal is itself considered to be an interference and will be – at least partially – suppressed.

One optimisation criterion is to minimise the output power so we would expect the desired signal to be rejected in proportion to its strength. This characteristic is illustrated in Figure (10.2).

This rejection highlights a difficulty with optimal processors; although they do often give large improvements over the conventional beamformers (shaded or unshaded), the greater the improvement the greater the sensitivity to minor errors in the system.

▷ Problem E.31

### 10.10.3. Practical implementation.

Because the weights for the optimal processors all involve the inverse of a matrix, direct calculation of weights is computationally intensive. In some practical cases it may be possible to implement the optimal processor. In other cases it may be too expensive to do so and some *suboptimal* system must be used. Much of the effort in signal processing has been devoted to devising suboptimal algorithms and techniques that are fast, affordable and robust, and yet do not sacrifice too much performance. It is however important to know what the theoretical optimum performance is, so that a benchmark may be set for any proposed suboptimal implementation.

In practice for plane-wave arrivals we can work with either  $\mathbf{R}_x^{-1}(f)$  and  $\mathbf{v}(\mathbf{k})$  or  $\mathbf{w}(\mathbf{k})$  and  $\tilde{\mathbf{x}}(f)$ . The choice is usually determined by the ratio of the number of beams to be formed to the number of receivers. When the number of beams is larger than the number of receivers then using the cross-spectral matrix directly is usually more computationally efficient.

The development in this Chapter has focused on beamforming at a single frequency. Generalisation of the expressions derived to narrowband signals in the time domain is straightforward.

A few examples are listed below of matters that the system designer may need to consider in practice.

- What happens if a receiver fails or if its response characteristics drift?
- What if the array geometry is inadvertently distorted or receiver outputs are subjected to phase errors?
- What are the effects of sampling and quantisation on system performance?
- What if the actual arrival direction is not what we have assumed?
- What if the noise field is not what we have assumed?
- What if the signals do not arrive as plane wavefronts?
- What if there are correlated arrivals – for example, due to multi-path propagation?

### **10.11. Overview**

In this Chapter we have shown how beamforming can be improved by optimising performance according to some criterion, and have considered four criteria:

- Maximum Gain
- Minimum Power with Linear Constraint (or MVDR)
- Minimum Mean-Square Error
- Maximum Likelihood

Although the approaches are very different, the resulting beamformers are similar or identical.

### Summary

- (1) The Maximum Gain beamformer has the disadvantage that it requires knowledge of the cross-spectral matrix of the *noise* which is not generally known.
- (2) The MVDR and the Maximum Likelihood processors can use the cross-spectral matrix of *signal-plus-noise*, which can be estimated.
- (3) The gain of the optimal processor is at least as good as that of the conventional processor.
- (4) In a number of cases – notably, in the presence of uncorrelated noise, or when the beam is steered directly at a strong interference – there is no improvement in using the optimal processor.
- (5) The optimal processor can give a great improvement over the conventional – but only if the cross-spectral matrix is ill-conditioned.
- (6) The condition of a matrix can be improved, often with only a small sacrifice in performance, by ‘regularisation’ (adding a small amount of uncorrelated noise).
- (7) The greater the improvement over the conventional processor, the less robust is the optimal processor (i.e., the greater is its sensitivity to small errors in the system).
- (8) In particular, if there is any difference between the assumed and actual direction of arrival of the wanted signal, the wanted signal will tend to be rejected by the optimal processor.
- (9) In practice, there are numerous system errors (quantisation, sampling, etc.) that must be taken into account when designing an optimal processor.
- (10) Often the optimal processor is too expensive to implement, and suboptimal processors must be found to approximate to the performance of the optimal.



## CHAPTER 11

# SAMPLE MATRIX BEAMFORMING

### 11.1. Introduction

In previous chapters the cross-spectral matrix of the receiver outputs,  $\mathbf{R}_x(f) = E\{\mathbf{x}\mathbf{x}^H\}$ , appeared in various expressions such as:

*Mean output of the conventional beamformer:*

$$p_{\text{conv}}(\mathbf{k}) = \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{v}(\mathbf{k}) \quad (11.1)$$

*Mean output of the MVDR beamformer:*

$$p_{\text{MVDR}}(\mathbf{k}) = \frac{1}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})} \quad (11.2)$$

*Weight vector of the optimal beamformer:*

$$\mathbf{w}_{\text{MVDR}}(\mathbf{k}) = \frac{\mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k})}, \quad (11.3)$$

A common practice is to *estimate*  $\mathbf{R}_x(f)$  and then to *substitute* it in to the relevant expression (a technique called *estimate and plug*).

In this chapter we consider estimation of the cross-spectral matrix,  $\mathbf{R}_x(f)$ , and of functions of  $\mathbf{R}_x(f)$ , again restricting attention to the frequency domain.  $\mathbf{R}_x(f)$  has to be estimated from a finite number of observations of the receiver output, and we examine some consequences of doing so.

An **estimator**, denoted by  $\hat{\cdot}$ , is a function<sup>1</sup> of observations ('snapshots') of a random process and the result is the **estimated value**, often colloquially referred to as '*an estimate of*'.

Estimators are random and their statistical properties, particularly the mean and variance, have a significant influence on system performance.

Appendix B presents some revision notes on statistics.

---

<sup>1</sup>For example, a common estimator for the mean of a random process is

$$\hat{\mu} = (1/N) \sum_{n=1}^N x_n,$$

where  $\{x_n\}$  is a particular set of observations of the random process.

### 11.2. Estimation of the cross-spectral matrix

We begin by considering the case in which we have sampled receiver outputs and the transformation to the frequency domain is effected by a discrete Fourier transform on a block of  $N$  data points. The output of the  $k^{\text{th}}$  receiver is denoted by  $\tilde{x}_k(f)$  and the output vector of the array by  $\tilde{\mathbf{x}}(f)$ <sup>2</sup>:

$$\tilde{x}_k(f) = \sum_{n=0}^{N-1} x_k(n\Delta T) e^{-2\pi i f n \Delta T}, \quad (11.4)$$

where  $\Delta T$  is the sampling interval. If the  $\{x_k(n\Delta T)\}$  are observations of a random process,  $\tilde{x}_k(f)$  will be a random function.

We shall assume that we are dealing with stationary random processes so we can invoke the ergodic theorem to replace the ensemble average by a time average. Implementing a time average in the frequency domain appears contradictory but commonly is effected by segmenting the data into  $M$  blocks each of  $N$  data points and carrying out, for each receiver, an  $N$ -point discrete Fourier transform within each block, as illustrated in Figure 11.1.<sup>3</sup>

---

<sup>2</sup>In practice the frequency variable is usually only evaluated at multiples of  $1/(N\Delta T)$  and  $N$  is chosen to be a power of 2 so that the standard form of the Fast Fourier Transform (FFT) can be used.

<sup>3</sup>Note that the blocks may be well separated or contiguous.

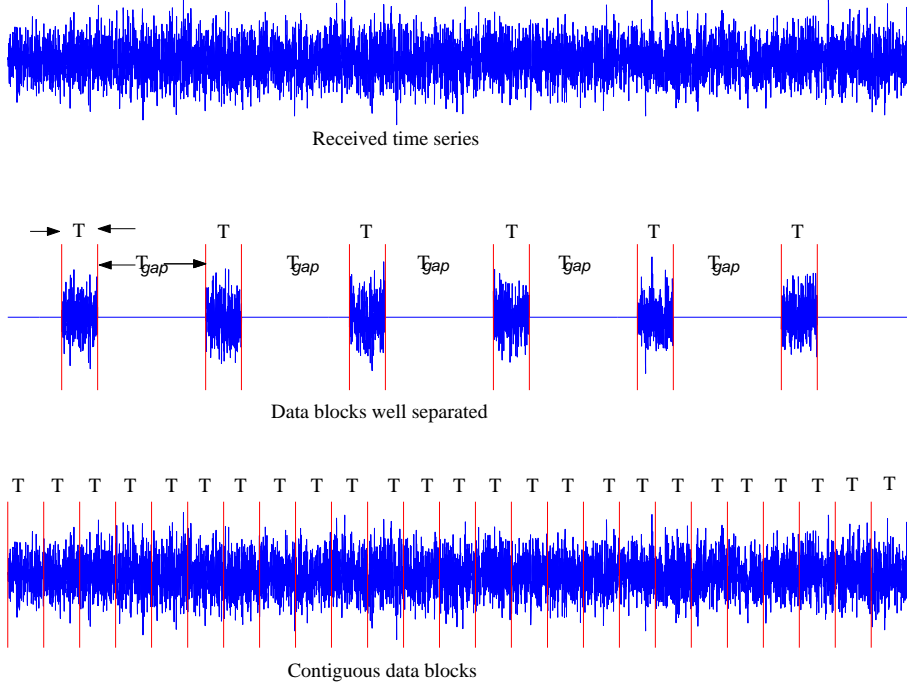


FIGURE 11.1. Blocks of sampled data

Let  $\tilde{x}_k^{(m)}(f)$  denote the Fourier transformed output from the  $m^{\text{th}}$  block of the  $k^{\text{th}}$  receiver, and  $\tilde{\mathbf{x}}^{(m)}(f) = \{\tilde{x}_k^{(m)}(f)\}$  the corresponding vector:

$$\tilde{x}_k^{(m)}(f) = \sum_{n=0}^{N-1} x_k(n \Delta T + m(T_{\text{gap}} + N \Delta T)) e^{-i2\pi f n \Delta T}, \quad (11.5)$$

where  $T_{\text{gap}}$  is the interval between blocks of data.

If the data blocks are well separated in time, as illustrated in the middle plot of Figure 11.1, it is reasonable to expect that they will be independent of one another. However, usually one takes contiguous blocks as illustrated in the bottom plot of the figure ( $T_{\text{gap}} = 0$ ) and we have

$$\tilde{x}_k^{(m)}(f) = \sum_{n=0}^{N-1} x_k(n \Delta T + m N \Delta T) e^{-i2\pi f n \Delta T} \quad (11.6)$$

In this case the observations  $\{\tilde{x}_k^{(m)}(f)\}$  are not necessarily independent of one another, but are often assumed to be so.

Recall the definition of the cross-spectral matrix as an ensemble average:

$$\mathbf{R}_x(f) = E \{ \tilde{\mathbf{x}}(f) \tilde{\mathbf{x}}^H(f) \}.$$

and consider the following estimator:

$$\hat{\mathbf{R}}_x(f) = \frac{1}{M} \sum_{m=1}^M \tilde{x}^{(m)}(f) \tilde{x}^{(m)H}(f), \quad (11.7)$$

If signals and noises are ergodic, the time average will converge to the ensemble average:

$$E\{\cdot\} \rightarrow \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{m=1}^M \{\cdot\}$$

### 11.3. Properties of the estimated cross-spectral matrix

We state without proof some important properties of the estimator,  $\hat{\mathbf{R}}_x(f)$ , given by (11.7) above.

- (a)  $\hat{\mathbf{R}}_x(f)$  is unbiased, i.e.,

$$E\{\hat{\mathbf{R}}_x(f)\} = \mathbf{R}_x(f). \quad (11.8)$$

- (b) If  $M > K$  and if there is some uncorrelated noise present,  $\hat{\mathbf{R}}_x(f)$  is positive definite (i.e., all its eigenvalues are positive) and hence non-singular.
- (c) If  $M < K$  (i.e., if the number of observations is less than the number of receivers),  $\hat{\mathbf{R}}_x(f)$  is positive semi-definite (i.e., at least one of its eigenvalues is zero and all the others are positive). This means  $\hat{\mathbf{R}}_x(f)$  is singular and cannot be inverted. We cannot therefore directly substitute it in the expressions (11.2) and (11.3) for the optimal beamformer<sup>4</sup>.
- (d) If the  $\{\mathbf{x}^{(m)}\}$  have a normal distribution and are statistically independent,  $\hat{\mathbf{R}}_x(f)$  is a maximum likelihood estimator of the cross-spectral matrix – i.e., it maximises the log of the likelihood function

$$p(x^{(1)}, x^{(2)}, \dots, x^{(m)} | \mathbf{R}_x).$$

- (e) Formally,  $\hat{\mathbf{R}}_x(f)$  has a complex Wishart distribution,  $CW(M; K: \mathbf{R}_x(f))$ , which is a matrix generalisation of a Chi-square distribution with  $2M$  degrees of freedom.
- (f) Equation (11.7) defines an *unstructured estimate* of the cross-spectral matrix. However, in a given noise field the exact cross-spectral matrix has a definite structure. Estimators that use prior knowledge of the cross-spectral matrix are termed *structured estimators*<sup>5</sup>.

<sup>4</sup>However, we may employ a technique, known as *regularisation* or *diagonal loading*, which replaces  $\hat{\mathbf{R}}_x(f)$  by  $\hat{\mathbf{R}}_x(f) + \varepsilon \mathbf{I}$  where  $\varepsilon$  is small compared to the noise power ( $\varepsilon \ll \text{Tr}(\mathbf{R}_x(f))/K$ ). This makes  $\hat{\mathbf{R}}_x(f)$  non-singular and allows us to benefit from using the optimal weights, but with some degradation in performance.

<sup>5</sup>See, for example, [9].

### 11.4. Conventional beamforming using the estimated cross-spectral matrix

Consider the output of the conventional beamformer:

$$y(\mathbf{k}) = \frac{1}{K} \mathbf{v}^H(\mathbf{k}) \tilde{\mathbf{x}}(f).$$

Because  $\mathbf{x}(f)$  is random,  $y(\mathbf{k}, f)$  is, for a given value of  $f$ , a complex random variable. To estimate the mean power we can square and average over  $M$  blocks to obtain a smoothed estimate:

$$\begin{aligned} \hat{p}_{\text{conv}}(\mathbf{k}) &= \frac{1}{M} \sum_{m=1}^M \left| \frac{\mathbf{v}^H(\mathbf{k}) \tilde{\mathbf{x}}^{(m)}(f)}{K} \right|^2 \\ &= \frac{1}{M} \sum_{m=1}^M \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \tilde{\mathbf{x}}^{(m)}(f) \tilde{\mathbf{x}}^{(m)H}(f) \mathbf{v}(\mathbf{k}) \\ &= \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \left( \frac{1}{M} \sum_{m=1}^M \tilde{\mathbf{x}}^{(m)}(f) \tilde{\mathbf{x}}^{(m)H}(f) \right) \mathbf{v}(\mathbf{k}) \end{aligned} \quad (11.9)$$

Comparing (11.9) with (11.1) justifies the use of

$$\hat{\mathbf{R}}_x(f) = \frac{1}{M} \left( \sum_{m=1}^M \tilde{\mathbf{x}}^{(m)}(f) \tilde{\mathbf{x}}^{(m)H}(f) \right) \quad (11.10)$$

as an estimator of the cross-spectral matrix.

$\hat{p}_{\text{conv}}(\mathbf{k})$  has a statistical distribution which we now consider.

#### 11.4.1. Properties of the Estimator.

- (a)  $\hat{p}_{\text{conv}}(\mathbf{k})$  is an unbiased estimator of  $p_{\text{conv}}(\mathbf{k})$ .

Proof follows directly from (11.8):

$$\begin{aligned} E\{\hat{p}_{\text{conv}}(\mathbf{k})\} &= \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) E\{\hat{\mathbf{R}}_x(f)\} \mathbf{v}(\mathbf{k}) \\ &= \frac{1}{K^2} \mathbf{v}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{v}(\mathbf{k}) \\ &= p_{\text{conv}}(\mathbf{k}) \end{aligned}$$

- (b) Under the assumption that the  $\{\tilde{\mathbf{x}}^{(m)}(f)\}$  are, for different  $m$ , statistically independent, the probability distribution of  $\hat{p}_{\text{conv}}(\mathbf{k})$  is Chi-square with  $2M$  degrees of freedom:  $\hat{p}_{\text{conv}}(\mathbf{k}) \sim \chi_{2M}^2$ .<sup>6</sup>

<sup>6</sup> The probability density function of a Chi-square variable with  $\nu$  degrees of freedom is given by

$$p_{\chi_\nu^2}(u) = \frac{u^{(\nu/2-1)} e^{-u/2}}{2^{\nu/2} \Gamma(\nu/2)},$$

(c) The variance of a Chi-square random variable is related to the mean by

$$\sigma^2(\chi_\nu^2) = \frac{2\mu^2(\chi_\nu^2)}{\nu}$$

$$\sigma^2(\chi_{2M}^2) = \frac{2\mu^2(\chi_{2M}^2)}{2M}$$

and thus

$$\sigma(\hat{p}_{\text{conv}}(\mathbf{k})) = \frac{E\{\hat{p}_{\text{conv}}(\mathbf{k})\}}{\sqrt{M}}$$

$$= \frac{p_{\text{conv}}(\mathbf{k})}{\sqrt{M}}.$$

Averaging over  $M$  blocks of data therefore reduces the standard deviation of the estimator by a factor of  $\sqrt{M}$ <sup>7</sup>.

### 11.5. Optimal beamforming using the estimated cross-spectral matrix

The output power of the optimal beamformer given in (11.2) involves  $\mathbf{R}_x^{-1}$ . We can apply (11.7) to estimate it<sup>8</sup>:

$$\hat{p}_{\text{MVDR}}(\mathbf{k}) = \frac{1}{\mathbf{v}^H(\mathbf{k}) \hat{\mathbf{R}}_x^{-1}(f) \mathbf{v}(\mathbf{k})} \quad (11.11)$$

We state the following results without proof:

(1)  $\hat{p}_{\text{MVDR}}(\mathbf{k})$  is a biased estimator of  $p_{\text{MVDR}}(\mathbf{k})$  :

$$E \left\{ \left( \mathbf{v}^H(\mathbf{k}) \hat{\mathbf{R}}_x^{-1}(f) \mathbf{v}(\mathbf{k}) \right)^{-1} \right\} = \left( \frac{M - K + 1}{M} \right) \left( \mathbf{v}^H(\mathbf{k}) \mathbf{R}_x^{-1}(f) \mathbf{v}(\mathbf{k}) \right)^{-1}$$

$$= \left( \frac{M - K + 1}{M} \right) p_{\text{MVDR}}(\mathbf{k}) \leq p_{\text{MVDR}}(\mathbf{k}) \quad (11.12)$$

In practice, therefore, the optimal beamformer will *underestimate* the output power— but for large  $M$  the bias becomes insignificant.

(2) The probability distribution of this estimator, under the same assumptions as in section 11.4, is again Chi-square but with  $2(M - K + 1)$  degrees of freedom:

$$\hat{p}_{\text{MVDR}}(\mathbf{k}) \sim \chi_{2(M-K+1)}^2.$$

Some consequences are listed below.

---

where  $\Gamma(x)$  is the Gamma function, defined recursively by  $\Gamma(x) = x\Gamma(x - 1)$ . Plots of the Chi-square probability density function can be found in standard statistical texts.

<sup>7</sup>The factor  $\sqrt{M}$ , expressed in decibels as  $5 \log_{10} M$  dB, is often called the *integration gain*.

<sup>8</sup>When  $M \geq K$  the estimated cross-spectral matrix is almost always non-singular.

- (a) Compared with the conventional estimator for the same number of observations ( $M$ ), the estimate of the beamformer output will have less degrees of freedom and the variance will therefore be greater.
- (b) Heuristically, the  $2(K - 1)$  degrees of freedom lost in the optimal estimator may be attributed to the steering of  $(K - 1)$  nulls.
- (c) From a detection point of view the increased variability will result in a loss in detection performance. For small  $M$ , this loss may be unacceptable and may negate the gains due to optimal beamforming.
- (d) If (assuming that no signal is present) the estimate of the cross-spectral matrix,  $\hat{\mathbf{R}}_x$ , is used to calculate array gain :

$$\hat{G}_{\max} = \left( \frac{\text{Tr}(\hat{\mathbf{R}}_n(f))}{K} \right) \left( \mathbf{v}^H(\mathbf{k}) \hat{\mathbf{R}}_n^{-1}(f) \mathbf{v}(\mathbf{k}) \right),$$

the result will be overstated.

- (e) Using  $\hat{\mathbf{R}}_x$  to calculate the optimal beamformer weights will give biased results.

▷ Problem E.32

## 11.6. Sample matrix inverse update

### 11.6.1. Matrix accumulation.

For real-time applications, it is possible to estimate the optimal weights using the sample matrix inverse techniques discussed above, in a recursive manner. The process uses the Woodbury matrix inversion lemma to estimate the optimal weights using the  $M$  vectors

$$\{\tilde{\mathbf{x}}^{(1)}, \tilde{\mathbf{x}}^{(2)}, \dots, \tilde{\mathbf{x}}^{(M-1)}, \tilde{\mathbf{x}}^{(M)}\}$$

by updating the estimate derived using just the  $M - 1$  vectors

$$\{\tilde{\mathbf{x}}^{(1)}, \tilde{\mathbf{x}}^{(2)}, \dots, \tilde{\mathbf{x}}^{(M-1)}\}.$$

In what follows, we show explicitly the dependence of the estimated cross-spectral matrix on the number of observations, but for brevity omit the dependence on frequency  $f$ . For convenience we define

$$\boldsymbol{\xi}(M) \triangleq \sum_{m=1}^M \tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{(m)H} \quad (11.13)$$

Apart from the factor  $1/M$ , this is an estimator of the cross-spectral matrix:

$$\begin{aligned} \hat{\mathbf{R}}_x^{(M)} &\triangleq \frac{1}{M} \left( \sum_{m=1}^M \tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{(m)H} \right) \\ &= \frac{1}{M} \boldsymbol{\xi}(M) \end{aligned} \quad (11.14)$$

Using the matrix inversion lemma

$$\begin{aligned} (bA + c\mathbf{u}\mathbf{v}^H)^{-1} &= bA^{-1} - \frac{cbA^{-1}\mathbf{u}\mathbf{v}^H A^{-1}}{1 + c\mathbf{v}^H bA^{-1}\mathbf{u}}, \\ \boldsymbol{\xi}(M) &= \boldsymbol{\xi}(M-1)^{-1} - \frac{\boldsymbol{\xi}(M-1)^{-1}\tilde{\mathbf{x}}^{(M)}\tilde{\mathbf{x}}^{(M)H}\boldsymbol{\xi}(M-1)^{-1}}{1 + \tilde{\mathbf{x}}^{(M)H}\boldsymbol{\xi}(M-1)^{-1}\tilde{\mathbf{x}}^{(M)}} \end{aligned} \quad (11.15)$$

From (11.3), the estimated optimal weight vector, based on all  $M$  observations, is given by

$$\hat{\mathbf{w}}_{MVD R}(M) = \frac{\hat{\mathbf{R}}_x^{-1(M)}\mathbf{v}(\mathbf{k})}{\mathbf{v}^H(\mathbf{k})\hat{\mathbf{R}}_x^{-1(M)}\mathbf{v}(\mathbf{k})} \quad (11.16)$$

Let us define<sup>9</sup>

$$\mu(M) \triangleq \tilde{\mathbf{x}}^{(M)H}\boldsymbol{\xi}^{-1}(M-1)\tilde{\mathbf{x}}^{(M)}, \quad (11.17)$$

$$\boldsymbol{\alpha}(M) \triangleq \boldsymbol{\xi}(M)^{-1}\mathbf{v}(\mathbf{k}) \quad (11.18)$$

$$\beta(M) \triangleq \mathbf{v}^H(\mathbf{k})\boldsymbol{\xi}(M)^{-1}\mathbf{v}(\mathbf{k}) \quad (11.19)$$

$$\gamma(M) \triangleq \hat{\mathbf{w}}_{opt}^H(M-1)\tilde{\mathbf{x}}^{(M)} \quad (11.20)$$

$$\boldsymbol{\eta}(M) \triangleq \frac{\boldsymbol{\xi}^{-1}(M-1)\tilde{\mathbf{x}}^{(M)}}{1 + \mu(M)}. \quad (11.21)$$

The estimated optimal weight vector is then

$$\hat{\mathbf{w}}_{MVD R}(M) = \frac{\boldsymbol{\alpha}(M)}{\beta(M)}, \quad (11.22)$$

which is calculated by updating the numerator and denominator of (11.22) as follows:

$$\boldsymbol{\alpha}(M) = \boldsymbol{\alpha}(M-1) - \gamma(M)^*\beta(M-1)\boldsymbol{\eta}(M) \quad (11.23)$$

$$\beta(M) = \beta(M-1) \left( 1 - \frac{\beta(M-1)|\gamma(M)|^2}{1 + \mu(M)} \right). \quad (11.24)$$

The beamformer output is updated using

$$\begin{aligned} y(M) &= \hat{\mathbf{w}}_{opt}^H(M)\tilde{\mathbf{x}}^{(M)} \\ &= \frac{\beta(M-1)\gamma(M)}{\beta(M)(1 + \mu(M))} \end{aligned} \quad (11.25)$$

---

<sup>9</sup>Note that  $\mu(M)$ ,  $\gamma(M)$  and  $\boldsymbol{\eta}(M)$  use the *previous* estimate  $\boldsymbol{\xi}^{-1}(M-1)$  with the *latest* observation  $\mathbf{x}^{(M)}$ .



and the power output using

$$\begin{aligned}\hat{p}_{MVDR}(M) &= \frac{1}{\mathbf{v}^H(\mathbf{k})\boldsymbol{\xi}^{-1}(M-1)\mathbf{v}(\mathbf{k})} \\ &= \frac{1}{\beta(M)}\end{aligned}\quad (11.26)$$

The recursion is initiated by taking  $\boldsymbol{\xi}^{-1}(0) = \varepsilon^{-1}\mathbf{I}$ , where  $0 < \varepsilon \ll 1$ . At each step we obtain the exact sample matrix inverse solution, apart from some transient terms from the initialisation of  $\boldsymbol{\xi}^{-1}(0)$ .

▷Ex 8

### 11.6.2. Forgetting Factor.

In the preceding section, every observation is given the same weight. However, in practice there is advantage in giving more weight to recent observations, and less to earlier ones, to allow for slow variations in the environment. If the noise field were to change, the processing would be able to adapt to the changes.

Consider an estimator of the cross-spectral matrix of the form:

$$\begin{aligned}\hat{\mathbf{R}}_x^{(M)} &= \left( \frac{1-\kappa}{1-\kappa^M} \right) \sum_{m=1}^M \tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{(m)H} \kappa^{M-m} \quad 0 \leq \kappa \leq 1 \\ &= \left( \frac{1-\kappa}{1-\kappa^M} \right) \left( \tilde{\mathbf{x}}^{(M)} \tilde{\mathbf{x}}^{(M)H} + \kappa \hat{\mathbf{R}}_x^{(M-1)} \right).\end{aligned}\quad (11.27)$$

The most recent observation,  $\tilde{\mathbf{x}}^{(M)} \tilde{\mathbf{x}}^{(M)H}$ , is given the most weight and previous observations are weighted by a multiple of a *forgetting factor*  $\kappa$ . Let

$$\begin{aligned}\psi(M) &\triangleq \sum_{m=1}^M \tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{(m)H} \kappa^{M-m} \\ &= \kappa \left( \psi(M-1) + \frac{1}{\kappa} \tilde{\mathbf{x}}^{(M)} \tilde{\mathbf{x}}^{(M)H} \right)\end{aligned}\quad (11.28)$$

Using the same notation as before ((11.17)–(11.21)) for  $\mu(M)$ ,  $\boldsymbol{\alpha}(M)$ ,  $\beta(M)$ ,  $\gamma(M)$  and  $\boldsymbol{\eta}(M)$ , we have

$$\hat{\mathbf{R}}_x^{(M)} = \left( \frac{1-\kappa}{1-\kappa^M} \right) \psi(M) \quad \text{and} \quad (11.29)$$

$$\hat{\mathbf{w}}_{MVDR}(M) = \frac{\boldsymbol{\alpha}(M)}{\beta(M)}, \quad (11.30)$$

We state without proof the following recursive process:

$$\boldsymbol{\alpha}(M) = \frac{1}{\kappa} \left( \boldsymbol{\alpha}(M-1) - \beta(M-1) \gamma^*(M) \boldsymbol{\eta}(M) \right), \quad (11.31)$$

$$\beta(M) = \frac{1}{\kappa} \left( \beta(M-1) - \frac{\beta(M-1)^2 |\gamma(M)|^2}{\kappa + \mu(M)} \right), \quad (11.32)$$

the updated output of the beamformer is

$$\begin{aligned} y(M) &= \hat{\mathbf{w}}^{(M)H} \tilde{\mathbf{x}}^{(M)} \\ &= \frac{\gamma(M)}{\kappa + \mu(M)} \end{aligned} \quad (11.33)$$

and the output power is

$$\hat{p}_{MVDR}(M) = \frac{1}{\beta(M)}. \quad (11.34)$$

## 11.7. Overview

In this Chapter we have described how the cross-spectral matrix of the receiver outputs can be estimated and stated some of the properties of the estimator.

The properties of the conventional and optimal beamforming estimators were then considered.

Finally, the Sample Matrix Inverse Update technique was described, in which the optimal weights are calculated recursively.

### Summary

- (1) The estimated cross-spectral matrix is unbiased.
- (2) If the number of observations ( $M$ ) is less than the number of receivers ( $K$ ) (and in the absence of any other noise) the estimated cross-spectral matrix is non-singular.
- (3) If some uncorrelated noise is present, it becomes non-singular.
- (4) The conventional beamformer is an unbiased estimator.
- (5) Averaging over  $M$  observations reduces the standard deviation of the output of the conventional beamformer by  $\sqrt{M}$ .
- (6) The optimal beamformer is a biased estimator.
- (7) The standard deviation of the optimal beamformer is higher than that of the conventional.
- (8) For small  $M$  the optimal beamformer may actually give worse performance than the conventional.
- (9) The optimal beamforming weights can be calculated recursively using the Sample Matrix Inverse Update technique, and incorporating a forgetting factor if desired.

## CHAPTER 12

# GRADIENT DESCENT ALGORITHMS

### 12.1. Introduction

In Chapter 10 the optimal weights were derived by directly minimising the output power given by

$$\mathbf{w}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{w}(\mathbf{k}) \quad (12.1)$$

subject to the linear constraint

$$\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}) = 1 \quad (12.2)$$

where  $\mathbf{k}$  specified a variable steering direction and the assumption was made that the cross-spectral matrix,  $\mathbf{R}_x(f)$  was positive definite and known or could be estimated from the data. That solution required the inversion of the cross-spectral matrix  $\mathbf{R}_x(f)$  – a computationally intensive operation.

Here we derive *Gradient Descent* algorithms<sup>a</sup> to minimise (12.1) subject to (12.2), which do not involve direct inversion of the matrix.

Recall that for two dimensions the Hermitian form for the mean output power is shaped like a bowl, as illustrated in Figure 12.1. For the case of an array of  $K$ , receivers, the Hermitian form is a convex surface. The constraint (12.2) is a plane, and the solution lies on its intersection with the convex surface.

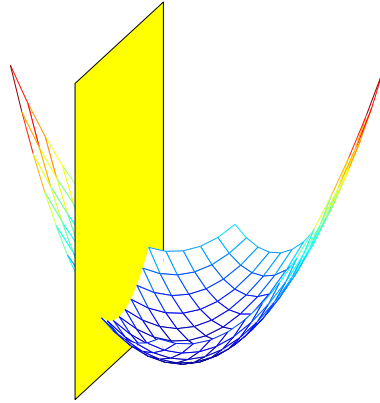


FIGURE 12.1. Bowl-shaped surface

<sup>a</sup>See Appendix A for revision notes on optimisation techniques.

The gradient descent algorithm moves in steps in the direction of the steepest slope of the constrained convex surface, adjusting the weight vector progressively to approximate more closely to the optimal weight vector.

## 12.2. Deterministic Gradient Descent

In this Section, we devise a gradient descent algorithm under the assumption that the cross-spectral matrix of the receiver outputs,  $\mathbf{R}_x(f)$  is *known*.

### 12.2.1. Minimising output power subject to a linear constraint.

As in Chapter 9, we define a cost function comprising the output power and the constraint surface:

$$\begin{aligned} \zeta(\mathbf{w}(\mathbf{k}), \mathbf{w}^*(\mathbf{k})) &= \mathbf{w}^H(\mathbf{k}) \mathbf{R}_x(f) \mathbf{w}(\mathbf{k}) - \lambda [\mathbf{w}^H(\mathbf{k}) \mathbf{v}(\mathbf{k}) - 1] \\ &\quad - \lambda^* [\mathbf{v}^H(\mathbf{k}) \mathbf{w}(\mathbf{k}) - 1]. \end{aligned}$$

In the following, for brevity we omit the explicit dependence on  $\mathbf{k}$  and  $f$  and write

$$\mathbf{w} \equiv \mathbf{w}(\mathbf{k})$$

$$\mathbf{v} \equiv \mathbf{v}(\mathbf{k})$$

$$\mathbf{R}_x \equiv \mathbf{R}_x(f)$$

$$\zeta(\mathbf{w}, \mathbf{w}^*) \equiv \mathbf{w}^H \mathbf{R}_x \mathbf{w} - \lambda [\mathbf{w}^H \mathbf{v} - 1] - \lambda^* [\mathbf{v}^H \mathbf{w} - 1]. \quad (12.3)$$

In [Appendix A](#) it was shown that the gradient descent algorithm for a real function  $\zeta(\mathbf{w}, \mathbf{w}^*)$  of a complex variable and of its conjugate is

$$\mathbf{w}(m+1) = \mathbf{w}(m) - \mu \nabla_{\mathbf{w}^*} \zeta(\mathbf{w}, \mathbf{w}^*)|_{\mathbf{w}=\mathbf{w}(m)}, \quad (12.4)$$

where  $\mathbf{w}(m)$  is the vector of weights at the  $m^{\text{th}}$  iteration, and that

$$\nabla_{\mathbf{w}^*} (\mathbf{w}^H \mathbf{v}) = \mathbf{v}, \quad (12.5)$$

$$\nabla_{\mathbf{w}} (\mathbf{v}^H \mathbf{w}) = \mathbf{v}^*, \quad (12.6)$$

$$\nabla_{\mathbf{w}^*} (\mathbf{v}^H \mathbf{w}) = \mathbf{0} = \nabla_{\mathbf{w}} (\mathbf{w}^H \mathbf{v}), \text{ and } \quad (12.7)$$

$$\nabla_{\mathbf{w}^*} (\mathbf{w}^H \mathbf{R}_x \mathbf{w}) = \mathbf{R}_x \mathbf{w}. \quad (12.8)$$

Let  $\lambda(m)$  be a Lagrange multiplier<sup>1</sup>. Applying (12.5)–(12.8) to (12.3),

$$\nabla_{\mathbf{w}^*} \zeta(\mathbf{w}, \mathbf{w}^*) = \mathbf{R}_x \mathbf{w} - \lambda \mathbf{v}, \quad (12.9)$$

and the gradient descent algorithm is

$$\mathbf{w}(m+1) = \mathbf{w}(m) - \mu [\mathbf{R}_x \mathbf{w}(m) - \lambda(m) \mathbf{v}]. \quad (12.10)$$

We now impose the constraint

$$\mathbf{v}^H \mathbf{w}(m+1) = 1 = \mathbf{w}^H(m+1) \mathbf{v}. \quad (12.11)$$

to obtain

$$\lambda(m) = \frac{\mathbf{v}^H \mathbf{R}_x \mathbf{w}(m)}{K} - \frac{\mathbf{v}^H \mathbf{w}(m) - 1}{\mu K}. \quad (12.12)$$

The last term in equation (12.12) would be zero when the constraint was imposed at the previous iteration (i.e.,  $\mathbf{v}^H \mathbf{w}(m) = 1$ ) and could therefore be ignored – in

<sup>1</sup>See [Appendix A](#) for an introduction to Lagrange multipliers.

theory at least. However, it has been reported [14] and [19] that retaining that term in the algorithm prevents the accumulation of arithmetic errors.

Substituting (12.12) in (12.10) yields the updated vector of weights:

$$\begin{aligned}
 \mathbf{w}(m+1) &= \mathbf{w}(m) - \mu \mathbf{R}_x \mathbf{w}(m) + \frac{\mu (\mathbf{v}^H \mathbf{R}_x \mathbf{w}(m)) \mathbf{v}}{K} + \frac{(1 - \mathbf{v}^H \mathbf{w}(m)) \mathbf{v}}{K} \\
 &= \left( \mathbf{I} - \frac{\mathbf{v} \mathbf{v}^H}{K} \right) \mathbf{w}(m) - \mu \left( \mathbf{I} - \frac{\mathbf{v} \mathbf{v}^H}{K} \right) \mathbf{R}_x \mathbf{w}(m) + \frac{\mathbf{v}}{K} \\
 &= \mathbf{P} \mathbf{w}(m) - \mu \mathbf{P} \mathbf{R}_x \mathbf{w}(m) + \frac{\mathbf{v}}{K} \\
 &= \mathbf{P} (\mathbf{I} - \mu \mathbf{R}_x) \mathbf{w}(m) + \frac{\mathbf{v}}{K}, \tag{12.13}
 \end{aligned}$$

where

$$\mathbf{P} \triangleq \mathbf{I} - \frac{\mathbf{v} \mathbf{v}^H}{K}. \tag{12.14}$$

(12.13) defines a *deterministic* algorithm which assumes knowledge of  $\mathbf{R}_x$ <sup>2</sup>.

$\mathbf{P} (\mathbf{I} - \mu \mathbf{R}_x)$  does not depend on the  $\mathbf{w}(m)$  and can be pre-calculated.

Note the term  $(\mathbf{I} - \mu \mathbf{R}_x)$  in (12.13): the cross-spectral matrix mixes the components of the weight vector during each iteration. Later we shall carry out transformations of variables to diagonalise the matrix so that, during iterations, each component of the weight vector is decoupled from every other.

A suitable initial weight vector which satisfies the constraint is that of the conventional beamformer:

$$\mathbf{w}(0) = \frac{\mathbf{v}}{K}. \tag{12.15}$$

Substituting (12.15) in (12.12) gives

$$\lambda(0) = \frac{\mathbf{v}^H \mathbf{R}_x \mathbf{v}}{K},$$

which is the power output of the conventional beamformer.

With appropriate choice of  $\mu$ , the weights will converge to those of the optimal beamformer:

$$\mathbf{w}(\infty) = \frac{\mathbf{R}_x^{-1} \mathbf{v}}{\mathbf{v}^H \mathbf{R}_x^{-1} \mathbf{v}}$$

and the Lagrange multiplier will converge to the power output of the optimal beamformer:

$$\lambda(\infty) = \frac{1}{(\mathbf{v}^H \mathbf{R}_x^{-1} \mathbf{v})},$$

The operator  $\mathbf{P}$  has some important properties that will be used to study the convergence of this algorithm.

---

<sup>2</sup>Later we shall describe a processor which does not rely on such knowledge.

- (i)  $P$  is *idempotent*, i.e.,  $P^2 = P$  and, since  $Pv = 0$ , is the projection operator onto the subspace orthogonal to  $v$ . It is of rank  $K - 1$ .

(ii)

$$PR_x w_{\text{opt}} = \frac{\left(I - \frac{vv^H}{K}\right) R_x R_x^{-1} v}{v^H R_x^{-1} v} = \frac{\left(I - \frac{vv^H}{K}\right) v}{v^H R_x^{-1} v} = 0 \quad (12.16)$$

(ii)

$$Pw_{\text{opt}} = \frac{\left(I - \frac{vv^H}{K}\right) R_x^{-1} v}{v^H R_x^{-1} v} = w_{\text{opt}} - \frac{v}{K}. \quad (12.17)$$

### 12.2.2. Convergence.

We now derive conditions that guarantee the convergence of the iteration described above. We transform the variables by a translation and subsequently a rotation of axes.

We first shift the origin by the *translation*  $w \rightarrow t = w - w_{\text{opt}}$  and use (12.16) and (12.17):

$$\begin{aligned} t(m+1) &= w(m+1) - w_{\text{opt}} \\ &= P[w(m) - \mu R_x w(m)] + \frac{v}{K} - w_{\text{opt}} \\ &= P\{w(m) - w_{\text{opt}}\} + \left\{Pw_{\text{opt}} - w_{\text{opt}} + \frac{v}{K}\right\} \\ &\quad - \mu\{PR_x w(m) - PR_x w_{\text{opt}}\} \\ &= Pt(m) - \mu PR_x t(m) \\ &= P(I - \mu R_x)t(m). \end{aligned}$$

Pre-multiplying by  $P$  and recalling that  $P^2 = P$ :

$$\begin{aligned} Pt(m+1) &= P^2(I - \mu R_x)t(m) \\ &= P(I - \mu R_x)t(m) \\ &= t(m+1) \quad \forall m. \end{aligned} \quad (12.18)$$

Hence

$$\begin{aligned} t(m+1) &= Pt(m+1) \\ &= Pt(m) - \mu PR_x t(m) \\ &= t(m) - \mu PR_x Pt(m) \\ &= (I - \mu PR_x P)t(m) \end{aligned} \quad (12.19)$$

We now derive conditions for convergence of the algorithm, by considering the eigenvalues of  $PR_x P$ . Since both  $P$  and  $R_x$  are Hermitian, so is  $PR_x P$ .

Let

$$\{d_{\max} \equiv d_1, d_2, \dots, d_K \equiv d_{\min} = 0\}$$

be the ordered eigenvalues of  $\mathbf{P} \mathbf{R}_x \mathbf{P}$ , with corresponding eigenvectors  $\{\mathbf{Q}_i\}$ , let  $\mathbf{Q}$  be the unitary matrix<sup>3</sup>:

$$\mathbf{Q} \triangleq [\mathbf{Q}_1 : \mathbf{Q}_2 : \cdots : \mathbf{Q}_K],$$

and let  $\mathbf{D}$  be the diagonal matrix

$$\mathbf{D} = \text{Diag}(d_1, \dots, d_K).$$

$\mathbf{P} \mathbf{R}_x \mathbf{P}$  then has an eigen-decomposition of the form

$$\mathbf{P} \mathbf{R}_x \mathbf{P} = \mathbf{Q} \mathbf{D} \mathbf{Q}^H \quad (12.20)$$

Substituting (12.20) in (12.19),

$$\begin{aligned} \mathbf{t}(m+1) &= \{\mathbf{I} - \mu \mathbf{Q} \mathbf{D} \mathbf{Q}^H\} \mathbf{t}(m) \\ &= \{\mathbf{Q} \mathbf{Q}^H - \mu \mathbf{Q} \mathbf{D} \mathbf{Q}^H\} \mathbf{t}(m) \\ &= \mathbf{Q} \{\mathbf{I} - \mu \mathbf{D}\} \mathbf{Q}^H \mathbf{t}(m). \end{aligned} \quad (12.21)$$

Next we *rotate* the axes by pre-multiplying both sides of (12.21) by  $\mathbf{Q}^H$ . Defining  $\mathbf{R}(m) \triangleq \mathbf{Q}^H \mathbf{t}(m)$ , we have

$$\begin{aligned} \mathbf{R}(m+1) &= (\mathbf{I} - \mu \mathbf{D}) \mathbf{R}(m) \\ &= (\mathbf{I} - \mu \mathbf{D})^2 \mathbf{R}(m-1) \\ &= \cdots \\ &= (\mathbf{I} - \mu \mathbf{D})^{m+1} \mathbf{R}(0). \end{aligned}$$

Since  $\{\mathbf{I} - \mu \mathbf{D}\}$  is diagonal, we can write

$$r_k(m+1) = (1 - \mu d_k) r_k(m), \quad (12.22)$$

so it is evident that, at each iteration, the components of  $\mathbf{R}(m)$  are *decoupled* from one another. We term each component a *mode* and by virtue of the decoupling can consider the convergence properties of each mode separately.

Convergence is guaranteed if

$$|1 - \mu d_k| < 1 \iff 0 < \mu d_k < 2 \quad \forall i. \quad (12.23)$$

If the above inequality holds for  $d_{\max}$  then it holds for all  $d_i$ . Thus the convergence condition is

$$\mu < 2/d_{\max}. \quad (12.24)$$

---

<sup>3</sup>A  $(K \times K)$  unitary matrix  $\mathbf{Q}$  has the property  $\mathbf{Q} \mathbf{Q}^H = \mathbf{I} = \mathbf{Q}^H \mathbf{Q}$ . Multiplying a vector by a unitary matrix therefore does not change the norm of the vector, but rotates the vector.



### 12.2.3. Convergence Properties.

*Choice of  $\mu$*

In practice we do not know, or may find it difficult to estimate,  $d_{\max}$ . It is more convenient to work instead with  $\lambda_{\max}$ . In the following we use

$$\begin{aligned}\text{Tr}(\mathbf{A}\mathbf{B}) &= \text{Tr}(\mathbf{B}\mathbf{A}) \text{ and} \\ \mathbf{P}^2 &= \mathbf{P}.\end{aligned}$$

In addition, if

$$\{\lambda_{\max} \equiv \lambda_1, \lambda_2, \dots, \lambda_{K-1}, \lambda_K \equiv \lambda_{\min}\}$$

are the ordered eigenvalues of  $\mathbf{R}_x$ , it can be shown that

$$\lambda_{\max} \geq d_{\max} \geq d_{\min} \geq \lambda_{\min}. \quad (12.25)$$

Then

$$\begin{aligned}d_{\max} &< \text{Tr}\{\mathbf{P}\mathbf{R}_x\mathbf{P}\} \\ &= \text{Tr}(\mathbf{R}_x\mathbf{P}) \\ &= \text{Tr}\left(\mathbf{R}_x\left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^H}{K}\right)\right) \\ &= \text{Tr}\left(\mathbf{R}_x - \frac{\mathbf{v}^H\mathbf{R}_x\mathbf{v}}{K}\right) \\ &\leq \text{Tr}(\mathbf{R}_x) \\ &= E\left\{\sum_{k=1}^K |\tilde{x}_k|^2\right\} \\ &\approx \sum_{k=1}^K |\tilde{x}_k|^2.\end{aligned} \quad (12.26)$$

$\sum_{k=1}^K |\tilde{x}_k|^2$  is the sum of the instantaneous powers of the receiver outputs (at frequency  $f$ ). A suitable (but conservative) choice of  $\mu$  is therefore

$$\mu < \frac{2}{\sum_{k=1}^K |\tilde{x}_k|^2} \quad (12.27)$$

*Convergence in terms of weight vectors*

Since  $\mathbf{R}(m) = \mathbf{Q}^H(\mathbf{w}(m) - \mathbf{w}_{\text{opt}})$  it follows that

$$\begin{aligned}
 \mathbf{w}(m) &= \mathbf{w}_{\text{opt}} + \mathbf{Q} \mathbf{R}(m) \\
 &= \mathbf{w}_{\text{opt}} + [\mathbf{Q}_1 \quad \mathbf{Q}_2 \quad \cdots \quad \mathbf{Q}_K] \begin{bmatrix} r_1(m) \\ r_2(m) \\ \vdots \\ r_K(m) \end{bmatrix} \\
 &= \mathbf{w}_{\text{opt}} + \sum_{k=1}^K r_k(m) \mathbf{Q}_k \\
 &= \mathbf{w}_{\text{opt}} + \sum_{k=1}^K r_k(0) (1 - \mu d_k)^m \mathbf{Q}_k, \\
 &= \mathbf{w}_{\text{opt}} + \sum_{k=1}^K r_k(0) \exp(-m/\tau_k) \mathbf{Q}_k
 \end{aligned} \tag{12.28}$$

where

$$\tau_k \triangleq -\frac{1}{\ln |1 - \mu d_k|} \tag{12.29}$$

is the *time constant* for convergence of each mode.

Provided  $\mu d_k \ll 1$  (i.e., the convergence is sufficiently slow),

$$\tau_k \approx \frac{1}{\mu d_k}. \tag{12.30}$$

*Rate of convergence*

From (12.29) it can be seen that bounds on  $\mu$  are set by the largest and smallest eigenvalues ( $d_{\text{max}}$  and  $d_{\text{min}}$ ):

$$\begin{aligned}
 (1 - \mu d_{\text{max}}) &> -1 \quad \text{and} \\
 (1 - \mu d_{\text{min}}) &< +1
 \end{aligned}$$

The overall speed of convergence is thus determined, as illustrated in Figure 12.2, by how close  $(1 - \mu d_{\text{min}})$  is to +1 and  $(1 - \mu d_{\text{max}})$  is to -1.

The ‘best’ choice of  $\mu$  – a compromise between the two rates of convergence – is given by

$$\begin{aligned}
 1 - \mu_1 d_{\text{min}} &= -(1 - \mu_1 d_{\text{max}}) \\
 \text{or } \mu_1 &= \frac{2}{d_{\text{max}} + d_{\text{min}}}.
 \end{aligned} \tag{12.31}$$

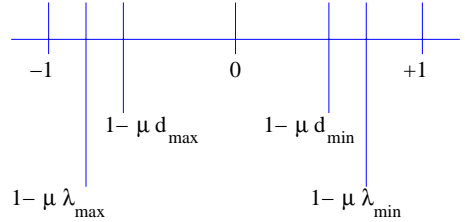


FIGURE 12.2. Illustrating factors that determine convergence rate

Instead of using the eigenvalues of  $\mathbf{P}\mathbf{R}_x\mathbf{P}$ , which depend on  $\mathbf{P}$  (and hence on the steering vector  $\mathbf{v}$ ), it is more convenient to use the eigenvalues of  $\mathbf{R}_x$ :

$$\mu_2 = \frac{2}{\lambda_{\max} + \lambda_{\min}}. \quad (12.32)$$

The maximum rate of convergence is then determined by how small is

$$\begin{aligned} |1 - \mu_2\lambda_{\min}| &= \left| 1 - \frac{2\lambda_{\min}}{\lambda_{\min} + \lambda_{\max}} \right| \\ &= \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \\ &= \frac{C - 1}{C + 1}, \end{aligned} \quad (12.33)$$

where  $C = \lambda_{\max}/\lambda_{\min}$  is the **condition number** of the matrix  $\mathbf{R}_x$ .

*Plane wave in uncorrelated receiver noise*

We consider here the example of a plane wave in uncorrelated noise:

$$\mathbf{R}_x = \sigma_n^2 \mathbf{I} + \sigma_s^2 \mathbf{v}_s \mathbf{v}_s^H,$$

so we have

$$\begin{aligned} \lambda_{\max} &= \sigma_n^2 + K\sigma_s^2, \\ \lambda_{\min} &= \sigma_n^2, \\ \mu_2 &= \frac{K\sigma_s^2}{2\sigma_n^2 + K\sigma_s^2}. \end{aligned}$$

### 12.3. Least Means Squares—Stochastic Gradient Descent Algorithm

**12.3.1. Introduction.** In Section 12.1 an iterative scheme was derived that converged to the optimum weight vector when the cross-spectral matrix was known exactly. Unfortunately the cross-spectral matrix is not usually known and has to be estimated. This can be done, as in Chapter 10, by taking a number of observations of the output vector from the receivers  $\{\mathbf{x}^{(m)}, m = 1, \dots, M\}$  and calculating:

$$\hat{\mathbf{R}}_x = \sum_{m=1}^M \mathbf{x}^{(m)} \mathbf{x}^{(m)H}.$$

In this section we consider the so-called least means-squares (*LMS*) algorithm, which is a stochastic gradient descent technique in which a rank-one estimate of the cross-spectral matrix is used at each step of the iteration. The LMS algorithm is widely used because of its simplicity and low computational complexity.

Throughout this section, we work in the frequency domain. For convenience we again omit the explicit dependence of the variables on frequency  $f$  and wavevector  $\mathbf{k}$ .

### 12.3.2. Stochastic Gradients.

The LMS algorithm replaces the cross-spectral matrix,  $\mathbf{R}_x$ , at the  $m$ -th stage in the iteration by its rank-one estimate

$$\hat{\mathbf{R}}_x(m) = \tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{(m)H}. \quad (12.34)$$

Substituting (12.34) in (12.13), we have the constrained LMS algorithm, often known as the Frost algorithm:

$$\hat{\mathbf{w}}(m+1) = \mathbf{P} \left( \mathbf{I} - \mu \tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{(m)H} \right) \hat{\mathbf{w}}(m) + \frac{\mathbf{v}}{K} \quad (12.35)$$

where, as before,

$$\mathbf{P} = \mathbf{I} - \frac{\mathbf{v} \mathbf{v}^H}{K}$$

Noting that the output of the beamformer at the  $m$ -th stage of the iteration,  $\hat{y}(m)$ , is given by

$$\hat{y}(m) = \hat{\mathbf{w}}^H(m) \tilde{\mathbf{x}}^{(m)}, \quad (12.36)$$

(12.35) can be expressed simply as <sup>4</sup>

$$\hat{\mathbf{w}}(m+1) = \mathbf{P} \left[ \hat{\mathbf{w}}(m) - \mu \hat{y}^*(m) \tilde{\mathbf{x}}^{(m)} \right] + \frac{\mathbf{v}}{K}. \quad (12.37)$$

As in the case of the deterministic gradient descent algorithm, a suitable initial weight vector is that of the conventional beamformer:

$$\hat{\mathbf{w}}(0) = \mathbf{v}/K. \quad (12.38)$$

The computational efficiency of the algorithm results from the simplicity of (12.35) and (12.36).

### 12.3.3. Convergence in the Mean.

As discussed in earlier chapters, using an estimated cross-spectral matrix introduces a statistical variability. For gradient descent algorithms this is manifested through *gradient noise* whereby each component of the estimated weight vector follows a noisy trajectory as illustrated in Figure 12.3.

Since we are dealing with statistical quantities we need to define convergence criteria that reflect the statistical variability of the noise in the weights. We shall consider **convergence in the mean** and **mean square convergence**.

Convergence in the mean is defined as

$$\lim_{m \rightarrow \infty} E\{\hat{\mathbf{w}}(m)\} = \mathbf{w}_{\text{opt}}.$$

If this condition is met, the mean value of the weight vector will converge to the true value. However, it is important to note that it does not necessarily ensure good performance, as will be discussed shortly.

To proceed further we need to make the simplifying assumption that the  $\{\tilde{\mathbf{x}}^{(m)}\}$  are, for different  $m$ , independent. In beamforming applications this is often valid.

---

<sup>4</sup>Note we have, somewhat innocuously, used  $m$  – formerly the iteration index – to denote which block of finite Fourier transformed receiver outputs is used in the estimation.

Observe in (12.34) that  $\hat{\mathbf{w}}(m)$  is a function only of  $\{\hat{\mathbf{x}}^{(m-1)}, \hat{\mathbf{x}}^{(m-2)}, \hat{\mathbf{x}}^{(m-3)}, \dots\}$ , and not of  $\hat{\mathbf{x}}^{(m)}$ . It follows that  $\hat{\mathbf{w}}(m)$  and  $\hat{\mathbf{x}}^{(m)}$  are independent. Taking the expectation of both sides of (12.34), we have

$$\begin{aligned} E\{\hat{\mathbf{w}}(m+1)\} &= \mathbf{P} \left[ E\{\hat{\mathbf{w}}(m)\} - \mu E\{\tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{H(m)} \hat{\mathbf{w}}(m)\} \right] + \frac{\mathbf{v}}{K} \\ &= \mathbf{P} \left[ E\{\hat{\mathbf{w}}(m)\} - \mu E\{\tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{H(m)}\} E\{\hat{\mathbf{w}}(m)\} \right] + \frac{\mathbf{v}}{K} \\ &= \mathbf{P} [E\{\hat{\mathbf{w}}(m)\} - \mu \mathbf{R}_x E\{\hat{\mathbf{w}}(m)\}] + \frac{\mathbf{v}}{K}, \end{aligned} \quad (12.39)$$

since by definition  $\mathbf{R}_x = E\{\tilde{\mathbf{x}}^{(m)} \tilde{\mathbf{x}}^{(m)H}\}$ .

Note that (12.36) takes the same form as (12.13) when the cross-spectral matrix is known. If  $E\{\hat{\mathbf{w}}(m)\} = \mathbf{w}(m)$  then  $E\{\hat{\mathbf{w}}(m+1)\} = \mathbf{w}(m+1)$ , and the convergence conditions are exactly the same as those derived in Section 12.1, viz.,

$$\mu < 2/d_{\max}. \quad (12.40)$$

Let us define the vector of *weight vector noise* by

$$\boldsymbol{\varepsilon}(m) \triangleq \hat{\mathbf{w}}(m) - \mathbf{w}_{\text{opt}}; \quad (12.41)$$

if  $\mu$  satisfies (12.37), then  $\lim_{m \rightarrow \infty} \boldsymbol{\varepsilon}(m) = \mathbf{0}$  and the weights will converge in the mean.

#### 12.3.4. Mean-square Convergence.

Convergence in the mean is not sufficient as it guarantees nothing about the variability of the weight noise. Consider the two graphs in Figure 12.3. Both exhibit convergence in the mean.

The first shows the desired performance, where the mean-square amplitude of the noise decays to zero.

However, it can be shown that this does not occur with the LMS algorithm: the variance of the weight noise will asymptote to a fixed level, and the weights will have fluctuations called *misadjustment noise*. The result is that the output power from the LMS beamformer is greater than that of the optimal.

The second plot in Figure 12.3 illustrates an example of noise in the weights that converges to zero but has an undesirably high variance.

The level of misadjustment noise will depend not only on the step size  $\mu$  but also on the eigenvalues of  $\mathbf{P} \mathbf{R}_x \mathbf{P}$ . We define the covariance matrix of the weight noise vector :

$$\boldsymbol{\Sigma}(m) \triangleq E\{\boldsymbol{\varepsilon}(m) \boldsymbol{\varepsilon}^H(m)\}$$

and then use (12.13) to derive a recursive equation for  $\boldsymbol{\Sigma}(m)$ . Proceeding much as before, we find that convergence of  $\boldsymbol{\Sigma}(m)$  is guaranteed by setting

$$\mu = \frac{1}{\text{Tr}(\mathbf{R}_x)}. \quad (12.42)$$

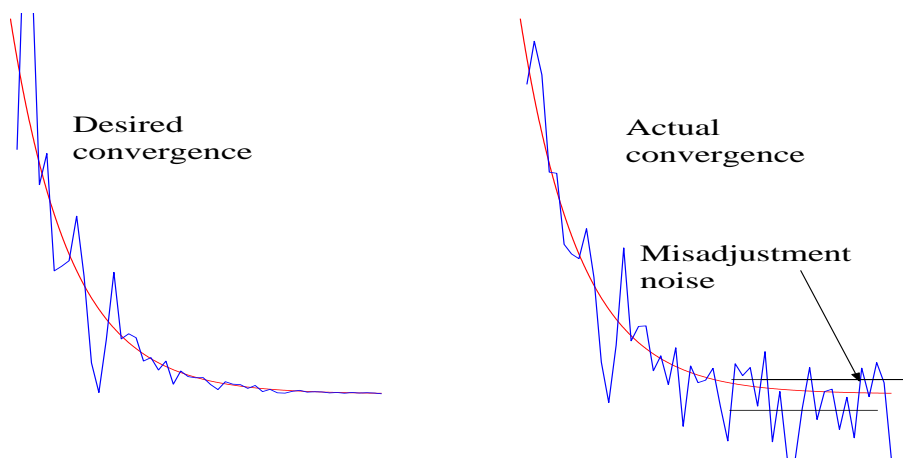


FIGURE 12.3. Illustrating convergence in the mean

However, in general with this value of  $\mu$  the level of misadjustment noise will not be acceptable. The misadjustment noise decreases as  $\mu$  is decreased. To make it small enough, a practical choice might be

$$\mu < \frac{1}{10 \operatorname{Tr}(\mathbf{R}_x)}.$$

Note that decreasing the step size  $\mu$  reduces the rate of convergence, so in choosing  $\mu$  we have to compromise between convergence rate and misadjustment noise.

▷Ex 9

## 12.4. Overview

In this Chapter we derived the deterministic gradient descent algorithm for the (largely theoretical) case in which the cross-spectral matrix of the receiver outputs,  $\mathbf{R}_x(f)$ , is known, and the conditions for the convergence of the algorithm.

Then we derived the stochastic gradient descent (LMS) algorithm.

### Summary

- (1) Convergence rates for the deterministic gradient descent algorithm are determined by the eigenvalues of a certain matrix product which is inconvenient to calculate.
- (2) It is possible to use a conservative approach, which uses only the eigenvalues of the cross-spectral matrix and which is more easily calculated.
- (3) The stochastic gradient descent (LMS) algorithm uses simply a rank-one estimate of the cross-spectral matrix.
- (4) This algorithm is simple and fast but, because of the crude approximation to the cross-spectral matrix, suffers from misadjustment noise.
- (5) If the step size is made sufficiently small, the misadjustment noise can be reduced to a tolerable level. Thus there is a tradeoff between the rate of convergence and the error.

## CHAPTER 13

# SPACE-TIME ADAPTIVE PROCESSING

### 13.1. Introduction

In Chapter 12 we derived the (deterministic) gradient descent and the stochastic constrained LMS algorithms for narrowband signals. Here we extend the results to the broadband case and introduce what has become known as *space-time adaptive processing (STAP)*. As before, we begin with the optimal processor which minimises output power subject to a set of linear constraints<sup>1</sup>; next we derive the gradient descent algorithm and finally we consider the stochastic constrained LMS algorithm. In this Chapter, we only have real quantities.

Recall that in the narrow-band case there was only one constraint: the response in the beamsteered direction was unity ( $\mathbf{w}^H \mathbf{v} = 1$ .) In the broadband case, because we are processing in both space and time, it is possible to constrain not only the angular (wavevector) but also the frequency response of the processor.

We assume that we have a linear array of  $K$  receivers and that the desired signal arrives from the broadside direction<sup>2</sup>. STAP processing operates in the space-time domain and uses a bank of transversal filters as illustrated in Figure 13.1. The outputs of the receivers are sampled synchronously at times  $\Delta T, 2\Delta T, \dots$  and are fed into a bank of transversal filters, each with  $J$  tap points, so there are  $KJ$  unknowns. We use the following notation:

$k$  denotes the  $k^{\text{th}}$  receiver,  
 $j$  denotes the  $j^{\text{th}}$  tap point,  
 $n$  denotes the  $n^{\text{th}}$  time sample.

Occasionally, where it will make matters clearer, we show the dimension of the matrix or vector explicitly, e.g.,  $\mathbf{x}_{[K]}$  is a  $K$ -dimensional vector.

---

<sup>1</sup>Here we follow Frost's work [14], but with a change in notation to be consistent with ours.

<sup>2</sup>To cater for the general case of an array with any known geometry, we simply add a set of beamsteering delays before this processor, so as to bring all the signals into phase ("*pre-steering*").



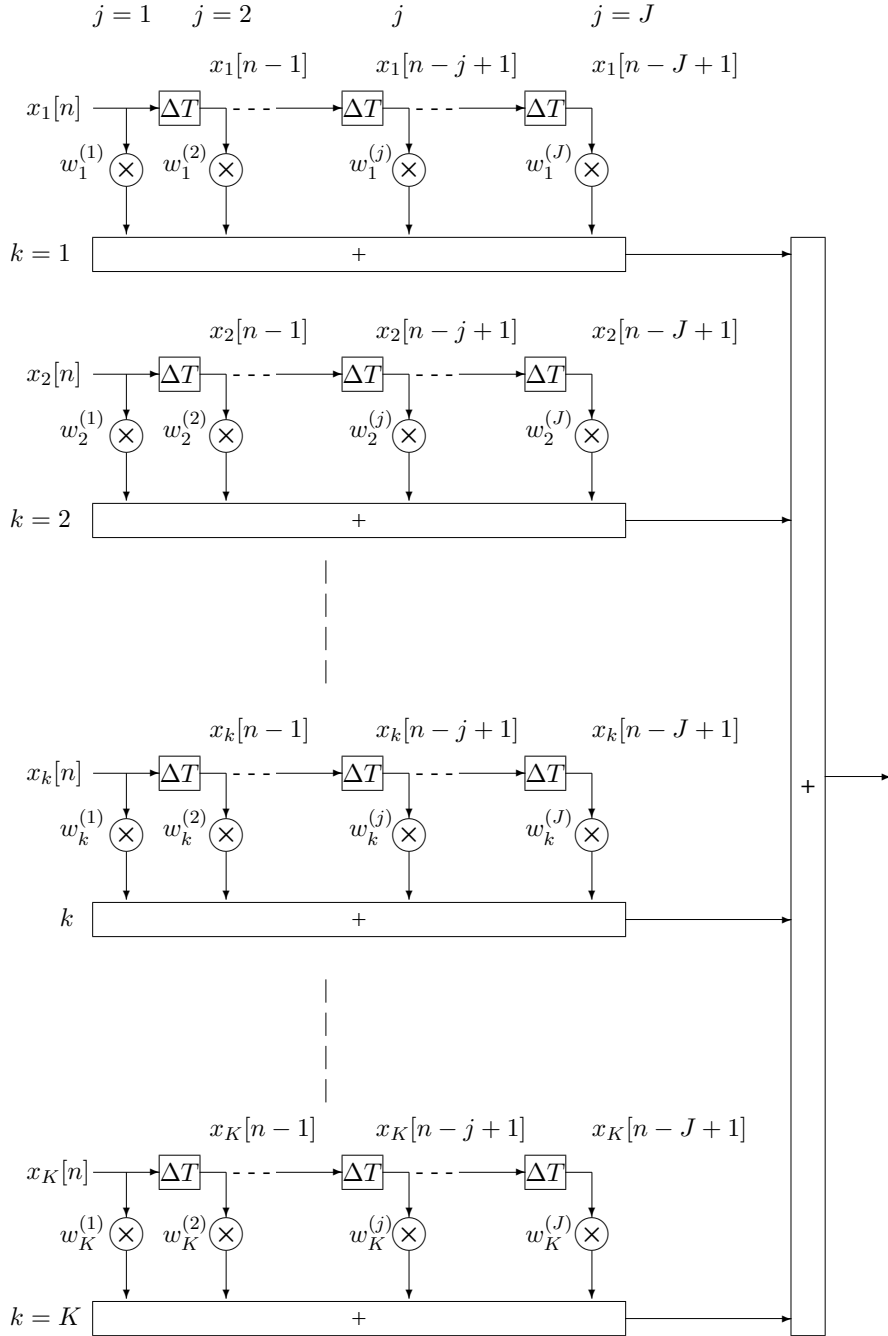
**13.2. Optimal broadband processor**

FIGURE 13.1. Adaptive constrained broadband LMS processor—broadside case

We now derive an expression for the optimal (minimum power with linear constraint) processor. Let  $\mathbf{x}[n] \equiv \mathbf{x}(n\Delta T)$  be the  $K$ -dimensional vector of receiver outputs at time  $t = n\Delta T$ :

$$\mathbf{x}[n] = \begin{bmatrix} x_1[n] \\ \vdots \\ x_K[n] \end{bmatrix}; \quad (13.1)$$

at the  $j^{\text{th}}$  tap point it becomes

$$\mathbf{x}[n-j+1] = \begin{bmatrix} x_1[n-j+1] \\ \vdots \\ x_K[n-j+1] \end{bmatrix}. \quad (13.2)$$

Let us stack all these  $J$  vectors to create a composite  $KJ$ -dimensional vector; we use a prime ( $'$ ) to designate such large vectors and matrices, :

$$\mathbf{x}'[n] = \begin{bmatrix} \mathbf{x}[n] \\ \mathbf{x}[n-1] \\ \vdots \\ \mathbf{x}[n-j+1] \\ \vdots \\ \mathbf{x}[n-J+1] \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} x_1[n] \\ \vdots \\ x_K[n] \end{bmatrix} \\ \begin{bmatrix} x_1[n-1] \\ \vdots \\ x_K[n-1] \end{bmatrix} \\ \vdots \\ \begin{bmatrix} x_1[n-j+1] \\ \vdots \\ x_K[n-j+1] \end{bmatrix} \\ \vdots \\ \begin{bmatrix} x_1[n-J+1] \\ \vdots \\ x_K[n-J+1] \end{bmatrix} \end{bmatrix}. \quad (13.3)$$

$$\mathbf{x}'[n] = \begin{bmatrix} \mathbf{x}[n] \\ \mathbf{x}[n-1] \\ \vdots \\ \mathbf{x}[n-j+1] \\ \vdots \\ \mathbf{x}[n-J+1] \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} x_1[n] \\ \vdots \\ x_K[n] \end{bmatrix} \\ \begin{bmatrix} x_1[n-1] \\ \vdots \\ x_K[n-1] \end{bmatrix} \\ \vdots \\ \begin{bmatrix} x_1[n-j+1] \\ \vdots \\ x_K[n-j+1] \end{bmatrix} \\ \vdots \\ \begin{bmatrix} x_1[n-J+1] \\ \vdots \\ x_K[n-J+1] \end{bmatrix} \end{bmatrix}. \quad (13.4)$$

We use a similar representation for the noises and signal:

$$\mathbf{n}'[n] = \begin{bmatrix} \mathbf{n}[n] \\ \mathbf{n}[n-1] \\ \vdots \\ \mathbf{n}[n-j+1] \\ \vdots \\ \mathbf{n}[n-J+1] \end{bmatrix}, \quad (13.5)$$

where  $\mathbf{n}[n-j+1]$  is the noise  $K$ -vector at the  $j^{\text{th}}$  tap point at time  $t = (n-j+1)\Delta T$ .

Because the signal from the broadside direction arrives at all the receivers simultaneously, the  $K$ -vector of signals at the  $j^{\text{th}}$  tap point takes the form:

$$\begin{bmatrix} s[n-j+1] \\ \vdots \\ s[n-j+1] \end{bmatrix} = s[n-j+1] \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = s[n-j+1]\mathbf{1}, \quad (13.6)$$

and hence

$$\mathbf{s}'[n] = \begin{bmatrix} s[n]\mathbf{1} \\ s[n-1]\mathbf{1} \\ \vdots \\ s[n-j+1]\mathbf{1} \\ \vdots \\ s[n-J+1]\mathbf{1} \end{bmatrix}. \quad (13.7)$$

The  $KJ$ -vector of the outputs at all the tap points is the sum of signal plus noise:

$$\mathbf{x}'[n] = \mathbf{s}'[n] + \mathbf{n}'[n]. \quad (13.8)$$

This  $KJ$  vector contains the  $K$  spatial samples and the  $J$  time samples of information from the receivers.

The signal and the noises are assumed to be real, zero-mean, stationary, random processes, and the signal is assumed to be independent of the noises:

$$E\{\mathbf{s}'[n]\mathbf{n}'^T[n]\} = \mathbf{0}'. \quad (13.9)$$

Their (unknown)  $[KJ \times KJ]$  covariance matrices are denoted by:

$$E\{\mathbf{x}'[n]\mathbf{x}'^T[n]\} \triangleq \mathbf{R}'_x, \quad (13.10)$$

$$E\{\mathbf{n}'[n]\mathbf{n}'^T[n]\} \triangleq \mathbf{R}'_n, \quad (13.11)$$

$$E\{\mathbf{s}'[n]\mathbf{s}'^T[n]\} \triangleq \mathbf{R}'_s, \quad (13.12)$$

$$(13.13)$$

Further, it is assumed that  $\mathbf{R}'_x$  and  $\mathbf{R}'_n$  are nonsingular (and hence positive definite).

Writing the  $K$  weights at the  $j^{\text{th}}$  tap point as  $\mathbf{w}^{(j)} = \begin{bmatrix} w_1^{(j)} \\ \vdots \\ w_K^{(j)} \end{bmatrix}$ , and stacking them so as to create a  $KJ$ -vector, we have:

$$\mathbf{w}' = \begin{bmatrix} \mathbf{w}^{(1)} \\ \vdots \\ \mathbf{w}^{(J)} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} w_1^{(1)} \\ \vdots \\ w_K^{(1)} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} w_1^{(J)} \\ \vdots \\ w_K^{(J)} \end{bmatrix} \end{bmatrix}; \quad (13.14)$$

$\mathbf{w}'_{[KJ \times 1]}$  is the vector of weights which we wish to derive.

The output of the processor at the  $m^{\text{th}}$  time sample is

$$y[n] = \mathbf{w}'^T \mathbf{x}'[n] \quad (13.15)$$

and the mean output power—which we wish to minimise—is

$$E\{y^2[n]\} = E\{\mathbf{w}'^T \mathbf{x}'[n] \mathbf{x}'[n] \mathbf{w}'^H\} = \mathbf{w}'^T \mathbf{R}'_x \mathbf{w}'. \quad (13.16)$$

We now consider the constraints. In the narrow-band case of Chapter 11 we imposed a single constraint—viz, that the beam response in the beamsteered direction must be unity ( $\mathbf{w}^H \mathbf{v} = 1$ ).

In the broadband case we can specify as well the *frequency response* in the beamsteered direction. Let us refer to Figure 13.1. Because of the pre-steering, signals appear synchronously at the inputs to the  $K$  transversal filters as  $s[n]$  and propagate in step down the tapped delay lines and are then summed. As far as the signal  $s[n]$  is concerned, we can collapse all the  $K$  receivers into one, and all the weights as well, so that we have a single-channel receiver and tapped delay line, as shown in Figure 13.2. Thus the signal is filtered by a transversal filter with weights  $g_j = \sum_k w_k^{(j)}$ , or

$$\mathbf{g} = \begin{bmatrix} g_1 \\ \vdots \\ g_J \end{bmatrix} = \begin{bmatrix} \sum_k w_k^1 \\ \vdots \\ \sum_k w_k^J \end{bmatrix}. \quad (13.17)$$

Note that the vector  $\mathbf{g}$  is the impulse response of the filter. We need to select that set of weights which gives the frequency response of the array in the beamsteered direction.

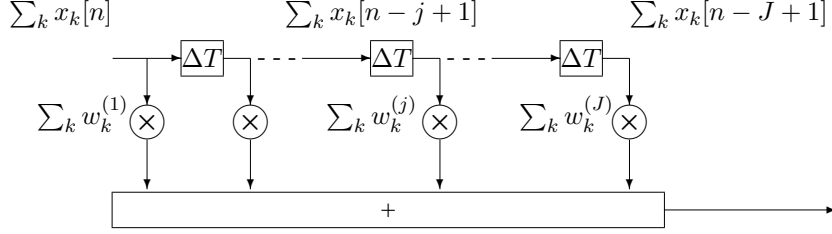


FIGURE 13.2. Equivalent delay line for the beamsteered direction

Let us define  $\mathbf{C}'^{(j)}$  to be the  $KJ$ -vector which is comprised of a stack of  $K$ -vectors, all of them  $\mathbf{0}$  except the  $j^{\text{th}}$  which is  $\mathbf{1}$ :

$$\mathbf{C}'^{(j)}_{[KJ]} = \begin{bmatrix} \mathbf{0}_{[K]} \\ \vdots \\ \mathbf{0}_{[K]} \\ \mathbf{1}_{[K]} \\ \mathbf{0}_{[K]} \\ \vdots \\ \mathbf{0}_{[K]} \end{bmatrix} j^{\text{th}} \text{ element} \quad (13.18)$$

The sum of the weights of the  $j^{\text{th}}$  tap point is then  $\sum_k w_k^{(j)} = \mathbf{w}'^T \mathbf{C}'^{(j)}$ ; to attain the desired frequency response we require that

$$\mathbf{w}'^T_{[1 \times KJ]} \mathbf{C}'^{(j)}_{[KJ \times 1]} = g_j, \quad j = 1, \dots, J. \quad (13.19)$$

These are the  $J$  constraints which we wish to impose. Defining the  $KJ \times J$  matrix<sup>3</sup>:

$$\mathbf{C}'_{[KJ \times J]} = \begin{bmatrix} \mathbf{C}'^{(1)}_{[KJ]} & \dots & \mathbf{C}'^{(J)}_{[KJ]} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_{[K]} & \mathbf{0}_{[K]} & \mathbf{0}_{[K]} & \cdot & \mathbf{0}_{[K]} \\ \mathbf{0}_{[K]} & \mathbf{1}_{[K]} & \cdot & \cdot & \cdot \\ \mathbf{0}_{[K]} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \mathbf{1}_{[K]} & \mathbf{0}_{[K]} \\ \mathbf{0}_{[K]} & \cdot & \cdot & \mathbf{0}_{[K]} & \mathbf{1}_{[K]} \end{bmatrix}, \quad (13.20)$$

the  $J$  constraints may be represented concisely by

$$\mathbf{C}'^T \mathbf{w}' = \mathbf{g}. \quad (13.21)$$

The optimisation problem is to minimise the output power subject to these constraints:

$$\underset{\mathbf{w}}{\text{Minimise}} \quad \mathbf{w}'^T \mathbf{R}'_x \mathbf{w}' \quad \text{subject to} \quad \mathbf{C}'^T \mathbf{w}' = \mathbf{g}.$$

The solution is obtained as before but here we use  $J$  Lagrange multipliers. Define the cost function:

$$\zeta = \mathbf{w}'^T \mathbf{R}'_x \mathbf{w}' + \lambda^T_{[J \times 1]} (\mathbf{C}'^T \mathbf{w}' - \mathbf{g}), \quad (13.22)$$

<sup>3</sup>Note that  $\mathbf{C}'$  is of full rank.  $\mathbf{C}'$  can be written concisely as  $\mathbf{C}' = \mathbf{I}_{[J]} \otimes \mathbf{1}_{[K]}$ , where  $\otimes$  denotes the Kronecker product (see [Appendix C](#))

where  $\lambda_{[J]}$ , is an unknown  $J$ -vector of Lagrange multipliers. Proceeding exactly as in Chapter 11 (but working with larger – and real – vectors and matrices), we have

$$2\mathbf{R}'_x \mathbf{w}' + \mathbf{C}'^T \lambda_{[J]} = \mathbf{0}', \quad (13.23)$$

$$\text{whence } \mathbf{w}'_{\text{STAP}} = -\frac{1}{2} \mathbf{R}'_x{}^{-1} \mathbf{C}' \lambda_{[J]}. \quad (13.24)$$

Substitution in equation (13.21) gives<sup>4</sup>

$$\lambda_{[J]} = 2 (\mathbf{C}'^T \mathbf{R}'_x \mathbf{C}')^{-1} \mathbf{g}, \quad (13.25)$$

and hence the solution to the optimisation problem is

$$\mathbf{w}'_{\text{STAP}} = \mathbf{R}'_x{}^{-1} \mathbf{C}' (\mathbf{C}'^T \mathbf{R}'_x \mathbf{C}')^{-1} \mathbf{g}. \quad (13.26)$$

For the case in which  $J = 1$ , (13.26) reduces to the narrowband result derived in Chapter 4.

### 13.3. Constrained broadband gradient descent algorithm

We next consider the case in which the covariance matrix  $\mathbf{R}'_x$  is known and derive the constrained gradient descent algorithm for the broadband case. The procedure follows the same steps as for the narrowband case in Chapter 11, but here we have  $KJ$  unknown *real* weights and  $J$  constraints). The gradient descent algorithm for the  $m$ -th iteration is as usual

$$\mathbf{w}'(m+1) = \mathbf{w}'(m) - \mu \left. \frac{\partial \zeta}{\partial \mathbf{w}'} \right|_{\mathbf{w}'=\mathbf{w}'(m)}. \quad (13.27)$$

where  $\zeta$  is given by (13.22):

$$\zeta = \mathbf{w}'^T \mathbf{R}'_x \mathbf{w}' + \lambda_{[J]}^T (\mathbf{C}'^T \mathbf{w}' - \mathbf{g}).$$

Recalling that for real variables

$$\frac{\partial}{\partial \mathbf{w}} \mathbf{w}^T \mathbf{R} \mathbf{w} = 2 \mathbf{R} \mathbf{w} \quad \text{and} \quad (13.28)$$

$$\frac{\partial}{\partial \mathbf{w}} \mathbf{w}^T \mathbf{A} \mathbf{z} = \mathbf{A} \mathbf{z} \quad (13.29)$$

(13.27) becomes

$$\mathbf{w}'(m+1) = \mathbf{w}'(m) - \mu [2\mathbf{R}'_x \mathbf{w}'(m) + \mathbf{C}' \lambda_{[J]}(m)]. \quad (13.30)$$

We now impose the constraints  $\mathbf{C}'^T \mathbf{w}'(m+1) = \mathbf{g}$  to obtain

$$\lambda(m) = -2(\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{C}'^T \mathbf{R}'_x \mathbf{w}'(m) + \frac{1}{\mu} (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{C}'^T \mathbf{w}'(m) - (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{g}. \quad (13.31)$$

Substituting equation (13.31) in (13.27):

$$\mathbf{w}'(m+1) = \mathbf{w}'(m) - \mu \mathbf{P}' \mathbf{R}'_x \mathbf{w}'(m) + \mathbf{C}' (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{g}, \quad (13.32)$$

<sup>4</sup>Because  $\mathbf{C}'$  and  $\mathbf{R}'_x$  are (by assumption) of full rank, the inverse exists.

where  $\mathbf{P}'_{[KJ \times KJ]} = \mathbf{I}' - \mathbf{C}'(\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{C}'^T$  is a projection matrix and is idempotent (i.e.,  $\mathbf{P}'^2 = \mathbf{P}'$ ).

### 13.4. Stochastic constrained broadband LMS algorithm

(13.32) is the gradient descent algorithm which assumes that  $\mathbf{R}'_x$ , is known and invariant. In practice we do not normally know  $\mathbf{R}'_x$ , need to estimate it. Proceeding along the same path as in Chapter 12 we derive the *broadband stochastic constrained LMS algorithm*. We begin by replacing  $\mathbf{R}'_x$  in (13.30) by  $\hat{\mathbf{R}}'_x \sim \mathbf{x}'(m)\mathbf{x}'^T(m)$  in (13.32):

$$\hat{\mathbf{w}}'(m+1) = \mathbf{P}' \hat{\mathbf{w}}'(m) - \mu \mathbf{P}' \mathbf{x}'(m) \mathbf{x}'^T(m) \hat{\mathbf{w}}'(m) + \mathbf{C}'^T (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{g}. \quad (13.33)$$

Writing the beamformer output as  $\hat{y}(m) = \hat{\mathbf{w}}'^T(m) \mathbf{x}'(m)$ , the algorithm is

$$\hat{\mathbf{w}}'(m+1) = \mathbf{P}' [\hat{\mathbf{w}}'(m) - \mu \hat{y}(m) \mathbf{x}'(m)] + \mathbf{C}' (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{g}. \quad (13.34)$$

The initial weight vector is taken to be<sup>5</sup>

$$\hat{\mathbf{w}}'(0) = \mathbf{C}' (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{g}. \quad (13.35)$$

As in the narrowband case, some parameters—in this case,  $\mathbf{P}'$  and  $\mathbf{C}' (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{g}$ —can be computed beforehand. We stress again that the stochastic constrained LMS algorithm introduces noise into the system.

## 13.5. Overview

In this Chapter we have considered broadband signals and space-time adaptive processing and derived the optimal broadband processor, the (deterministic) gradient descent algorithm and the stochastic constrained LMS algorithm.

---

<sup>5</sup> $\mathbf{w}(0)$  satisfies the constraint  $\mathbf{C}'^T \mathbf{w}(0) = \mathbf{g}$ .

## Summary

### Narrowband

### Broadband

Optimum weight vector

$$\begin{aligned}
 & \min_{\mathbf{w}_{[K]}} \mathbf{w}^H \mathbf{R}_x \mathbf{w} \\
 & \text{subject to } \mathbf{w}^H \mathbf{v} = 1 \\
 & \zeta = \mathbf{w}^H \mathbf{R}_x \mathbf{w} - \lambda(\mathbf{w}^H \mathbf{v}_s - 1) \\
 & \quad - \lambda^*(\mathbf{v}_s^H \mathbf{w} - 1) \\
 & \mathbf{w}_{\text{opt}} = \frac{\mathbf{R}_x^{-1} \mathbf{v}_s}{\mathbf{v}_s^H \mathbf{R}_x^{-1} \mathbf{v}_s} \\
 & y_{\text{opt}} = \mathbf{x}^H \mathbf{w}_{\text{opt}}
 \end{aligned}$$

$$\begin{aligned}
 & \min_{\mathbf{w}'_{[K]}} \mathbf{w}'^H \mathbf{R}'_x \mathbf{w}' \\
 & \text{subject to } \mathbf{C}'^T_{[J \times KJ]} \mathbf{w}'_{[KJ]} = \mathbf{g}_{[J]} \\
 & \zeta = \mathbf{w}'^H \mathbf{R}'_x \mathbf{w}' \\
 & \quad + \lambda^T_{[J]} (\mathbf{C}'^T \mathbf{w}' - \mathbf{g}) \\
 & \mathbf{w}'_{\text{opt}} = \mathbf{R}'_x \mathbf{C}' (\mathbf{C}'^T \mathbf{R}'_x \mathbf{C}')^{-1} \mathbf{g} \\
 & y_{\text{opt}} = \mathbf{x}'^T \mathbf{w}'_{\text{opt}}
 \end{aligned}$$

Gradient descent

$$\begin{aligned}
 \mathbf{w}(m+1) &= \mathbf{w}(m) - \mu \mathbf{g}(\mathbf{w}, m) \\
 \mathbf{g}(\mathbf{w}, m) &= \mathbf{R}_x \mathbf{w}(m) - \lambda \mathbf{v}_s \\
 \mathbf{w}(m+1) &= \mathbf{P}(\mathbf{I} - \mu \mathbf{R}_x) \mathbf{w}(m) + \frac{\mathbf{v}_s}{K} \\
 \mathbf{P} &= \mathbf{I} - \frac{\mathbf{v}_s \mathbf{v}_s^H}{K}
 \end{aligned}$$

$$\begin{aligned}
 \mathbf{w}'(m+1) &= \mathbf{w}'(m) - \mu \mathbf{g}'(\mathbf{w}', m) \\
 \mathbf{g}'(\mathbf{w}', m) &= \mathbf{R}'_x \mathbf{w}'(m) - \mathbf{C}' \lambda_{[J]}(m) \\
 \mathbf{w}'(m+1) &= \mathbf{P}'(\mathbf{I}' - \mu \mathbf{R}'_x) \mathbf{w}'(m) \\
 & \quad + \mathbf{C}' (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{g} \\
 \mathbf{P}' &= \mathbf{I}' - \frac{\mathbf{C}' \mathbf{C}'^T}{K}
 \end{aligned}$$

Stochastic constrained LMS

$$\begin{aligned}
 \mathbf{w}(m+1) &= \mathbf{w}(m) - \mu \mathbf{g}(\mathbf{w}, m) \\
 \mathbf{g}(\mathbf{w}, m) &= \mathbf{R}_x \mathbf{w}(m) - \lambda \mathbf{v}_s \\
 \mathbf{w}(m+1) &= \mathbf{P}[\mathbf{w}(m) - \mu \mathbf{x}(m) y^*(m)] \\
 & \quad + \frac{\mathbf{v}_s}{K} \\
 \mathbf{P} &= \mathbf{I} - \frac{\mathbf{v}_s \mathbf{v}_s^H}{K}
 \end{aligned}$$

$$\begin{aligned}
 \mathbf{w}'(m+1) &= \mathbf{w}'(m) - \mu \mathbf{g}'(\mathbf{w}', m) \\
 \mathbf{g}'(\mathbf{w}', m) &= \mathbf{R}'_x \mathbf{w}'(m) - \mathbf{C}' \lambda_{[J]}(m) \\
 \mathbf{w}'(m+1) &= \mathbf{P}'[\mathbf{w}'(m) - \mu y(m) \mathbf{x}'(m)] \\
 & \quad + \mathbf{C}' (\mathbf{C}'^T \mathbf{C}')^{-1} \mathbf{g} \\
 \mathbf{P}' &= \mathbf{I}' - \frac{\mathbf{C}' \mathbf{C}'^T}{K}
 \end{aligned}$$



## CHAPTER 14

# ESTIMATION

### 14.1. Introduction

Many applications of array processing require knowledge of signal parameters, such as its direction of arrival (DOA) or its power. Whilst the above are usually the primary parameters to be estimated there are others that can be derived from the beam outputs which can be used in further processing such as classification, tracking and localization. For example, in sonar, lines in the power spectrum of a beam are commonly used to classify the sound source. Usually the signal parameters are unknown and the process of estimating them is the subject of estimation theory. A short review of its fundamentals is provided in Appendix D.

A key requirement of any estimator is that it is accurate and there are theoretical limits to the accuracy with which any parameter can be estimated in practice. Indeed since any practical estimator is a function of the received data which contains noise it follows that any estimator is a random variable. Thus to fully characterize the statistical properties of an estimator its probability density function should be determined. Often this is intractable and only the bias and variance of an estimator can be determined. Typically the variance will vary with the input SNR, array gain, integration times and processing resolution.

For unbiased estimators an important quantity is the Cramer-Rao lower bound. This is the result of a remarkable theorem that allows a lower bound for the variance of any linear unbiased estimator of parameters to be determined. The practical importance of this cannot be over stressed as it is an important design tool that determines, given a particular SNR and set of processing parameters, the lowest possible variance that can be achieved over a wide class of estimators. Knowing this often determines whether the system and processing design will be adequate for the task.

Important estimators are the maximum likelihood and least squares estimators. The application of these to array processing and some properties of the resulting estimators are presented.

### 14.2. Classical DOA Estimation Techniques

In this section some commonly used techniques for estimating the arrival direction of a plane wave incident on an array of receivers are discussed.

### 14.2.1. Steering in Angle.

A common direction finding technique with mechanically steering antenna is to rotate the antenna and to estimate the DOA of the incident signal as the angle at which maximum output power is obtained. This idea can readily be implemented in array beamforming by forming multiple beams (see Chapter 6) and selecting the angle of maximum beam power as the DOA. These beams can either be formed simultaneously or sequentially depending on the computational power available.

One or several DOA's can be estimated by this approach as illustrated in Figure 14.1 below for conventional beamforming. However the approach is not limited to the outputs of a conventional beamformer – optimum beamformers could be used instead. Figure 14.2 illustrates the advantage of using optimal instead of conventional processing. For this example two signals are incident upon an array and the DOA's are such that the difference of the DOA's is less than the beamwidth of a conventional beamforming. As can be seen each peak in the conventional beamformer output power is biased away from the true DOA by the presence of the other signal. However the optimal processor by virtue of its null steering ability can dramatically reduce this bias. Thus the position of peaks in the optimal beamformer outputs experience less bias than the corresponding peaks of the conventional beamformer output power.

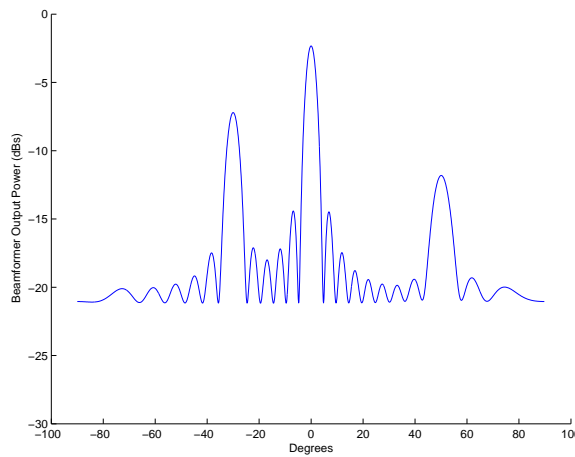


FIGURE 14.1. Output of a conventional beamformer power for a 24 receiver linear array. There are three plane-wave arrivals from  $\theta_s = -30^\circ, 0^\circ$  and  $50^\circ$  with SNRs of 5dB, 0dB and  $-5$ dB respectively.  $d/\lambda = \frac{1}{2}$ .

The accuracy of this approach is ultimately limited by the number and spacing of the beams formed. This is illustrated in Figure 14.1 where the number of beams is increased.

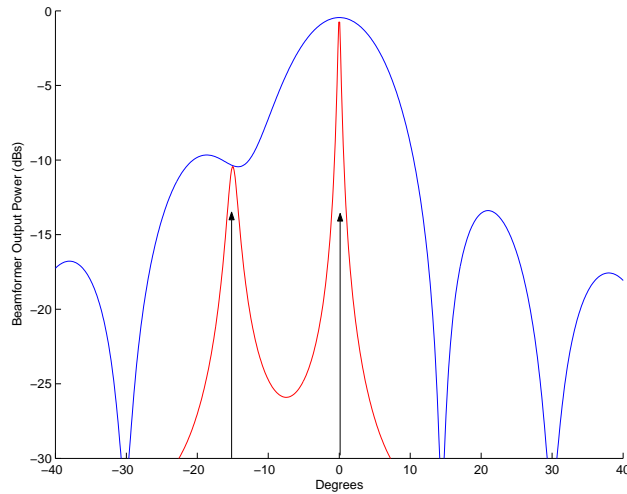


FIGURE 14.2. Output of a conventional and optimum beamformer powers for a 16 receiver linear array. There are two plane-wave arrivals from  $\theta_s = -15^\circ$  and  $0^\circ$  with SNRs of 20 and 10 dB respectively.  $d/\lambda = \frac{1}{4}$ .

### 14.2.2. Beam Interpolation.

In practice only a limited number of beams are formed—often only independent beams are formed. In this situation the separation of beams is usually significantly greater than the desired DOA accuracy.

This problem is commonly solved by interpolating between beams. Several beams around a beam with maximum (local) power are selected and a small curve fitted to their powers—the peak of this curve is then taken as the DOA estimate. One commonly used approach is quadrature interpolation using the beam with peak power and the beam on each side of it. However, this approach can result in biased estimates of the signal DOA.

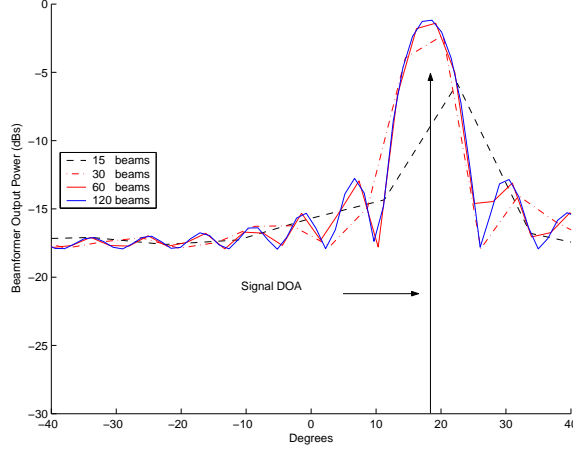


FIGURE 14.3. Output of a conventional beamformer power for a 15 receiver linear array. A single plane-wave of SNR 5dB incident on the array from  $18^\circ$  and  $d/\lambda = \frac{1}{2}$ .

### 14.3. Fundamental Limits to Estimation

There exists a fundamental limit, known as the Cramer-Rao lower bound on the accuracy of **any** unbiased estimator. This bound is derived from the Fisher Information Matrix (FIM). Note that the parameters estimated could be a scalar quantity such as DOA or power or they could be multivariate quantity such as frequency, DOA and signal power or the vector of receiver locations in the case of an array with uncertainty in its receiver positions.

#### 14.3.1. Fisher Information Matrix.

A fundamental quantity in estimation theory is the Fisher Information Matrix,  $\mathbf{J}$ , defined, for a random process with probability density function,  $p(\mathbf{x}; \boldsymbol{\alpha})$ , by

$$(\mathbf{J})_{ij} = J_{ij}(\boldsymbol{\alpha}) \triangleq -E \left\{ \frac{\partial^2 \ln(p(\mathbf{x}; \boldsymbol{\alpha}))}{\partial \alpha_i \partial \alpha_j} \right\} = -E \left\{ \frac{\partial^2 L_x(\boldsymbol{\alpha})}{\partial \alpha_i \partial \alpha_j} \right\} \quad (14.1)$$

where the  $\mathbf{x} = [x_1, x_2, \dots, x_N]^T$  are  $N$  samples of the random process and  $\boldsymbol{\alpha}^T = [\alpha_1 \alpha_2 \dots \alpha_L]$  is the vector of  $L$  parameters characterizing the random process. The function  $l_x(\boldsymbol{\alpha}) = \ln(p(\mathbf{x}; \boldsymbol{\alpha}))$  is called the *likelihood function*, and its logarithm,

$$L_x(\boldsymbol{\alpha}) \triangleq \ln(p(\mathbf{x}; \boldsymbol{\alpha})) \quad (14.2)$$

is called the *log-likelihood function*. Using various identities the Fisher Information Matrix can also be given by

$$J_{ij}(\boldsymbol{\alpha}) = E \left\{ \frac{\partial L_x(\boldsymbol{\alpha})}{\partial \alpha_i} \cdot \frac{\partial L_x(\boldsymbol{\alpha})}{\partial \alpha_j} \right\} \quad (14.3)$$

Note that the FIM is a symmetric, non-negative definite matrix.

For a complex vector  $\boldsymbol{\alpha}$  the Fisher Information Matrix is an Hermitian non-negative definite matrix defined as

$$J_{ij}(\boldsymbol{\alpha}) \triangleq E \left\{ \nabla_{\alpha_i} L_x(\boldsymbol{\alpha}) \nabla_{\alpha_j^*} L_x(\boldsymbol{\alpha}) \right\} \quad (14.4)$$

### 14.3.2. Cramer-Rao Lower Bounds.

Let  $\hat{\boldsymbol{\alpha}}^T = [\hat{\alpha}_1, \hat{\alpha}_2, \dots, \hat{\alpha}_L]$  be any unbiased estimator of the parameters of the random process. Define the covariance matrix,  $\mathbf{C}_{\hat{\boldsymbol{\alpha}}}$ , of this estimator as

$$(\mathbf{C}_{\hat{\boldsymbol{\alpha}}})_{ij} = E\{(\hat{\alpha}_i - \alpha_i)(\hat{\alpha}_j - \alpha_j)\} \quad (14.5)$$

Under a regularity condition the Cramer-Rao lower bound states that covariance matrix,  $\mathbf{C}_{\hat{\boldsymbol{\alpha}}}$ , for **any** unbiased estimator satisfies

$$\mathbf{C}_{\hat{\boldsymbol{\alpha}}} - \mathbf{J}^{-1}(\boldsymbol{\alpha}) \geq 0 \quad (14.6)$$

where  $\geq 0$  means positive definite. (For proof see Appendix D).

When an estimator attains the Cramer-Rao lower bound it is said to be *efficient*.

Each diagonal term is of particular interest as it indicates a lower bound for the variance of any unbiased estimator of that parameter. For the vector,  $\boldsymbol{\alpha}$ , of  $L$  real parameters to be estimated, it follows from Equation (14.6), that the corresponding Cramer-Rao inequality is

$$\text{var}(\hat{\alpha}_\ell) \geq [\mathbf{J}^{-1}]_{\ell\ell} \quad \ell = 1, 2, \dots, L \quad (14.7)$$

For a real scalar,  $\alpha$ , the Cramer-Rao inequality reduces to

$$E\{|\hat{\alpha} - \alpha|^2\} \geq E \left\{ \left| \frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} \right|^2 \right\} \leq 1$$

or

$$\text{var}(\hat{\alpha}) \geq \frac{1}{E \left\{ \left| \frac{\partial L_{\mathbf{x}}(\alpha)}{\partial \alpha} \right|^2 \right\}} \quad (14.8)$$

and for a complex scalar  $\alpha$  it becomes

$$\begin{aligned} \text{var}(\hat{\alpha}) &\geq \frac{1}{E \{ \nabla_{\alpha} L_{\mathbf{x}}(\alpha) \nabla_{\alpha^*} L_{\mathbf{x}}(\alpha) \}} \\ &= - \frac{1}{E \{ \nabla_{\alpha} \nabla_{\alpha^*} L_{\mathbf{x}}(\alpha) \}} \end{aligned} \quad (14.9)$$

An important application in array processing is when the Fourier transformed receiver outputs of an array of receivers can be modelled as correlated complex Gaussian random variables with cross-spectral matrix  $\mathbf{R}_x(f, \alpha)$  - see Chapter 8. Note, for clarity, that the dependence of the cross-spectral matrix on the signal parameters  $\alpha$  has been explicitly shown. In this case the probability density function is given by

$$p(\mathbf{x}; \alpha) = \frac{1}{\pi^K |\mathbf{R}_x(f, \alpha)|} \exp \left( -\mathbf{x}^H(f) \mathbf{R}_x^{-1}(f, \alpha) \mathbf{x}(f) \right) \quad (14.10)$$

for the case where the Fourier transformed receiver outputs have zero mean. For this case the Fisher Information Matrix (see [22]) reduces to

$$\begin{aligned} J_{ij}(\alpha) &= \text{Tr} \left[ \mathbf{R}_x^{-1}(f, \alpha) \frac{\partial \mathbf{R}_x(f, \alpha)}{\partial \alpha_i} \mathbf{R}_x^{-1}(f, \alpha) \frac{\partial \mathbf{R}_x(f, \alpha)}{\partial \alpha_j} \right] \\ &= -\text{Tr} \left[ \frac{\partial \mathbf{R}_x^{-1}(f, \alpha)}{\partial \alpha_i} \frac{\partial \mathbf{R}_x(f, \alpha)}{\partial \alpha_j} \right] \end{aligned} \quad (14.11)$$

In the presence of deterministic signals the Fourier transformed receiver outputs will have non-zero means and the above expression is generalized to

$$\begin{aligned} J_{ij}(\alpha) &= \text{Tr} \left[ \mathbf{R}_x^{-1}(f, \alpha) \frac{\partial \mathbf{R}_x(f, \alpha)}{\partial \alpha_i} \mathbf{R}_x^{-1}(f, \alpha) \frac{\partial \mathbf{R}_x(f, \alpha)}{\partial \alpha_j} \right] \\ &\quad + 2\Re \left[ \frac{\partial \boldsymbol{\mu}^H(f, \alpha)}{\partial \alpha_i} \mathbf{R}_x^{-1}(f, \alpha) \frac{\partial \boldsymbol{\mu}(f, \alpha)}{\partial \alpha_j} \right] \end{aligned} \quad (14.12)$$

where  $\boldsymbol{\mu}(f, \alpha)$  is the vector of means.

### 14.3.3. Application to DOA Estimation.

For  $N$  samples of a plane wave in uncorrelated receiver noise of signal to noise ratio, SNR, incident upon an array of  $K$  receivers and an array length of  $L$ , the CRLB for the variance of any unbiased estimator of  $\beta$ , the cone angle relative to endfire, is given by

$$\sigma_{CRLB}^2(\beta) = \frac{12}{4\pi^2 K SNR \left(\frac{L}{\lambda}\right)^2 \sin^2 \beta} \quad (14.13)$$

\*\*\*Assumptions underlying the above expression needs to be checked thru\*\*\*

### 14.3.4. Bhattacharyya Lower Bound.

When an efficient estimator does not exist, it is possible to improve on the Cramer-Rao bound, using an inequality due to Bhattacharyya [39, Chapter 3]. Consider just the case of a scalar parameter  $\alpha$  and define the  $(M \times M)$  matrix,  $\mathbf{B}$  where

$$b_{jk} = E \left\{ \frac{\partial^j L_x(\alpha)}{\partial \alpha^j} \frac{\partial^k L_x(\alpha)}{\partial \alpha^k} \right\} \quad (14.14)$$

and

$$\mathbf{C} = \mathbf{B}^{-1} \quad (14.15)$$

The Bhattacharyya bound on the error variance for unbiased estimators is

$$\text{var}(\hat{\alpha}) \geq c_{11} \quad (14.16)$$

For  $M = 1$ , the Cramer-Rao bound (14.9) is

$$\text{var}(\hat{\alpha}) \geq \frac{1}{b_{11}} = \frac{1}{E \left\{ \left( \frac{\partial L_x(\alpha)}{\partial \alpha} \right)^2 \right\}} \quad (14.17)$$

For  $M = 2$ ,

$$\text{var}(\hat{\alpha}) \geq \frac{1}{b_{11}} + \frac{b_{12}^2}{b_{11}(b_{11}b_{22} - b_{12}^2)} \quad (14.18)$$

Since the second term is  $\geq 0$  the Bhattacharyya bound is  $\geq$  the Cramer-Rao bound.

## 14.4. Statistical Estimation Techniques

Two statistical approaches to estimation are maximum likelihood and least squares. They are discussed below.

### 14.4.1. Maximum Likelihood Estimation.

The maximum likelihood estimator is based on the following notion. Consider a probability density function  $p(x; \alpha)$  which depends on an unknown deterministic scalar parameter  $\alpha$ . In Figure 14.4.1  $p(x; \alpha)$  is sketched for several values of  $\alpha$ . Given an observed value  $x$  it is required to estimate  $\alpha$ .

It is natural to pick that value of  $\alpha$  for which  $p(x; \alpha)$  is greatest. This is called the *maximum likelihood estimate*  $\hat{\alpha}_{ML}$ . The estimator is

$$\begin{aligned}\hat{\alpha}_{ML} &= \arg \left( \max_{\alpha} p(x; \alpha) \right) \\ &= \arg \left( \max_{\alpha} L_x(\alpha) \right) \quad (14.19)\end{aligned}$$

where

$$L_x(\alpha) \triangleq \ln(p(x; \alpha)) \quad (14.20)$$

is the *log-likelihood function* introduced earlier. When  $L_x(\alpha)$  is differentiable,  $\hat{\alpha}$  is a root of the equation

$$\frac{\partial}{\partial \alpha} L_x(\alpha) = 0 \quad (14.21)$$

As an example of this method consider an array where

- the signal and noise are narrow-band, zero-mean gaussian random processes;
- the cross-spectral matrix of the noise is known;
- the signal arrives as a plane wave-front from a known direction  $\mathbf{k}_s$ .

The problem is to estimate the signal power  $\sigma_s^2(f)$ . In the remainder of this example, the dependence on frequency  $f$  is omitted for convenience. The probability density function of the receiver outputs is

$$p(\mathbf{x}; \sigma_s^2) = \frac{1}{\pi^K \det(\mathbf{R}_x)} \exp(\mathbf{x}^H \mathbf{R}_x^{-1} \mathbf{x}) \quad (14.22)$$

where

$$\mathbf{R}_x = \mathbf{R}_n + \sigma_s^2 \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s) \quad (14.23)$$

The log-likelihood function is

$$L_x(\sigma_s^2) = K \ln \pi - \ln(\det(\mathbf{R}_x)) - \mathbf{x}^H \mathbf{R}_x^{-1} \mathbf{x} \quad (14.24)$$

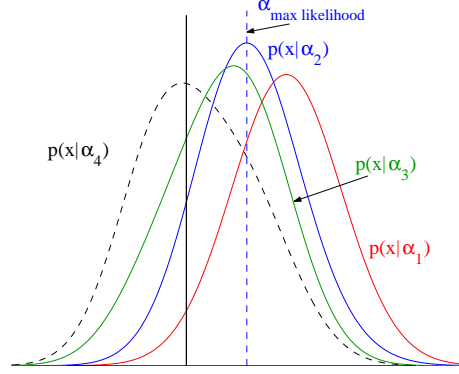


FIGURE 14.4. Illustrating maximum likelihood



The derivatives of the determinant and of the inverse of a matrix  $\mathbf{R}$  are given by (see (C.31) and (C.32))

$$\frac{\partial \ln \det(\mathbf{R})}{\partial \alpha} = \text{Tr} \left( \mathbf{R}^{-1} \frac{\partial \mathbf{R}}{\partial \alpha} \right)$$

and

$$\frac{\partial \mathbf{R}^{-1}}{\partial \alpha} = -\mathbf{R}^{-1} \frac{\partial \mathbf{R}}{\partial \alpha} \mathbf{R}^{-1}.$$

Setting  $\partial L_x(\sigma_s^2)/\partial \sigma_s^2 = 0$ , gives

$$\text{Tr}(\mathbf{R}_x^{-1} \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s)) + |\mathbf{x}^H \mathbf{R}_x^{-1} \mathbf{v}(\mathbf{k}_s)|^2 = 0$$

or

$$\mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_x^{-1} \mathbf{v}(\mathbf{k}_s) = |\mathbf{x}^H \mathbf{R}_x^{-1} \mathbf{v}(\mathbf{k}_s)|^2 \quad (14.25)$$

Using (C.9) for the determinant of a partitioned matrix:

$$\det(\mathbf{R} + \alpha \mathbf{u} \mathbf{u}^H) = \alpha \mathbf{u}^H \mathbf{R}^{-1} \mathbf{u} \det(\mathbf{R}),$$

and Woodbury's identity (C.14) gives

$$\hat{\sigma}_s^2 = \frac{|\mathbf{x}^H \mathbf{R}_n^{-1} \mathbf{v}(\mathbf{k}_s)|^2}{(\mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_n^{-1} \mathbf{v}(\mathbf{k}_s))^2} - \frac{1}{\mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_x^{-1} \mathbf{v}(\mathbf{k}_s)} \quad (14.26)$$

Writing

$$\gamma \triangleq \mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_n^{-1} \mathbf{v}(\mathbf{k}_s) \quad (14.27)$$

the expectation is

$$\begin{aligned} E\{\hat{\sigma}_s^2\} &= \frac{\mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_n^{-1} E\{\mathbf{x} \mathbf{x}^H\} \mathbf{R}_n^{-1} \mathbf{v}(\mathbf{k}_s)}{\gamma^2} - \frac{1}{\gamma} \\ &= \frac{\mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_n^{-1} (\mathbf{R}_n + \sigma_s^2 \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s)) \mathbf{R}_n^{-1} \mathbf{v}(\mathbf{k}_s)}{\gamma^2} - \frac{1}{\gamma} \\ &= \sigma_s^2 \end{aligned} \quad (14.28)$$

so the estimator is unbiased.

Using (B.19), it can be shown that

$$E\left\{(\mathbf{a}^H \mathbf{x} \mathbf{x}^H \mathbf{a})^2\right\} = 2(\mathbf{a}^H \mathbf{R}_x \mathbf{a})^2 \quad (14.29)$$

and hence

$$\begin{aligned} \text{var}(\hat{\sigma}_s^2) &= E\{\hat{\sigma}_s^4\} - (E\{\hat{\sigma}_s^2\})^2 \\ &= \frac{(1 + \sigma_s^2 \gamma^2)^2}{\gamma^2} \end{aligned} \quad (14.30)$$

Differentiating with respect to  $\sigma_s^2$  the Cramer-Rao lower bound can be derived:

$$\begin{aligned} \frac{\partial L_x(\sigma_s^2)}{\partial \sigma_s^2} &= -\text{Tr} \left( \mathbf{R}_x^{-1} \frac{\partial \mathbf{R}_x}{\partial \sigma_s^2} \right) + \mathbf{x}^H \mathbf{R}_x^{-1} \frac{\partial \mathbf{R}_x}{\partial \sigma_s^2} \mathbf{R}_x^{-1} \mathbf{x} \\ &= \mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_x^{-1} \mathbf{v}(\mathbf{k}_s) - |\mathbf{x}^H \mathbf{R}_x^{-1} \mathbf{v}(\mathbf{k}_s)|^2 \\ &= \frac{1}{1 + \sigma_s^2 \gamma} \left( \gamma - \frac{\mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_n^{-1} \mathbf{x} \mathbf{x}^H \mathbf{v}(\mathbf{k}_s)}{1 + \sigma_s^2 \gamma} \right) \end{aligned} \quad (14.31)$$

Using (14.29) and after a little manipulation it can be shown that

$$E \left( \frac{\partial L_x(\sigma_s^2)}{\partial \sigma_s^2} \right)^2 = \frac{\gamma^2}{(1 + \sigma_s^2 \gamma)^2} \quad (14.32)$$

so the Cramer-Rao bound is

$$\text{var}(\sigma_s^2) \geq \frac{(1 + \sigma_s^2 \gamma)^2}{\gamma^2} \quad (14.33)$$

Referring to (14.30) it can be seen that the ML estimator attains the Cramer-Rao lower bound and hence it is efficient.

#### 14.4.2. Least Squares Estimation.

Least squares<sup>1</sup> is a useful estimation technique when there is a linear relationship between the unknown signal parameters  $\alpha$  and the receiver outputs. This occurs in array processing when  $L$  plane waves are incident upon an array from known directions but their amplitudes are unknown. Considering the frequency domain as discussed in Chapter 8.4, the receiver output vector can be written as

$$\tilde{\mathbf{x}}(f) = \mathbf{V}(f) \tilde{\mathbf{s}}(f) + \tilde{\mathbf{n}}(f) \quad (14.34)$$

where the  $(K \times L)$  array manifold matrix  $\mathbf{V}$  is given by

$$\mathbf{V}(f) = \begin{bmatrix} \mathbf{v}(\mathbf{k}_1) & \vdots & \mathbf{v}(\mathbf{k}_2) & \vdots & \cdots & \vdots & \mathbf{v}(\mathbf{k}_L) \end{bmatrix} \quad (14.35)$$

and the  $L$ -vector  $\tilde{\mathbf{s}}(f)$  is given by

$$\tilde{\mathbf{s}}(f) = [s(\mathbf{k}_1) s(\mathbf{k}_2) \cdots s(\mathbf{k}_L)]^T \quad (14.36)$$

where  $\tilde{\mathbf{n}}(f)$  is additive noise with cross-spectral matrix  $\mathbf{R}_n(f)$ .

Now consider where, in a similar manner to Chapter 11.2, we have  $M$  independent realisations of the receiver outputs,  $\tilde{\mathbf{x}}^{(m)}(f)$ , for  $m = 1, 2, \dots, M$ . Thus (1.34)

<sup>1</sup>For a brief review of least squares estimation refer to Appendix D.4

generalizes to

$$\tilde{\mathbf{x}}^{(m)} = \mathbf{V} \tilde{\mathbf{s}} + \tilde{\mathbf{n}}^{(m)} \quad (14.37)$$

where, again, for consistency, the frequency dependence has been omitted.

The unknown parameters  $\alpha$  are estimated by minimising the total weighted error power for the block of data, i.e., by minimising

$$\sum_{m=1}^M \tilde{\mathbf{n}}^{(m)H} \mathbf{R}_n^{-1} \tilde{\mathbf{n}}^{(m)} = \sum_{m=1}^M \left( \tilde{\mathbf{x}}^{(m)H} - \tilde{\mathbf{s}}^H \mathbf{V}^H \right) \mathbf{R}_n^{-1} \left( \tilde{\mathbf{x}}^{(m)} - \mathbf{V} \tilde{\mathbf{s}} \right) \quad (14.38)$$

Differentiating with respect to the  $s(k_j)$  and equating to zero results in the following estimator

$$\hat{\mathbf{s}}_{\text{LS}} = (\mathbf{V}^H \mathbf{V})^{-1} \sum_{m=1}^M \mathbf{V}^H \mathbf{R}_n^{-1} \tilde{\mathbf{x}}^{(m)} \quad (14.39)$$

As shown in appendix D, the LS estimator is unbiased and the covariance of its estimates is given by

$$E \left\{ \left( \hat{\mathbf{s}}_{\text{LS}} - \tilde{\mathbf{s}} \right) \left( \hat{\mathbf{s}}_{\text{LS}} - \tilde{\mathbf{s}} \right)^H \right\} = \left( \sum_{m=1}^M \mathbf{V}^H \mathbf{V} \right)^{-1} = \frac{1}{M} (\mathbf{V}^H \mathbf{V})^{-1} \quad (14.40)$$

for the complex case. Note the typical  $\frac{1}{M}$  reduction in variance from inefficient averaging over  $M$  blocks of data.

Consider a special case of the above, i.e. estimating the amplitude of a deterministic plane wave in uncorrelated Gaussian receiver noise. The vector of receiver outputs is given by

$$\tilde{\mathbf{x}}^{(m)} = s_0 \mathbf{v}(\mathbf{k}_s) + \tilde{\mathbf{n}}^{(m)} \quad (14.41)$$

where  $s_0 \equiv s_0(f)$  is a scalar *unknown* parameter to be estimated.

In this case the LS optimization function becomes

$$\frac{1}{M} \sum_{m=1}^M \left( \tilde{\mathbf{x}}^{(m)} - s_0 \mathbf{v}(\mathbf{k}_s) \right)^H \mathbf{R}_n^{-1} \left( \tilde{\mathbf{x}}^{(m)} - s_0 \mathbf{v}(\mathbf{k}_s) \right) \quad (14.42)$$

In Appendix A Section A.4.1, the condition <sup>2</sup> for the extremum of a real function of complex variables is

$$\nabla_{s_0^*} L_{\tilde{\mathbf{x}}}(s_0) = 2\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}^{-1}\frac{1}{M}\sum_{m=1}^M\left(\tilde{\mathbf{x}}^{(m)} - s_0\mathbf{v}(\mathbf{k}_s)\right) = 0 \quad (14.43)$$

so the estimate is <sup>3</sup>

$$\hat{s}_0 = \frac{\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}(f)\tilde{\mathbf{x}}'}{\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}(f)\mathbf{v}(\mathbf{k}_s)} = \left(\frac{\mathbf{R}_n^{-1}(f)\mathbf{v}(\mathbf{k}_s)}{\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}(f)\mathbf{v}(\mathbf{k}_s)}\right)^H \tilde{\mathbf{x}}' \quad (14.44)$$

where

$$\tilde{\mathbf{x}}' = \frac{1}{M}\sum_{m=1}^M\tilde{\mathbf{x}}^{(m)} \quad (14.45)$$

This estimator is just a beamformer with weights

$$\mathbf{w}_{\text{Capon}}(\mathbf{k}_s) = \frac{\mathbf{R}_n^{-1}(f)\mathbf{v}(\mathbf{k}_s)}{\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}(f)\mathbf{v}(\mathbf{k}_s)} \quad (14.46)$$

Note that the maximum likelihood estimator given by (14.46) is identical to the MVDR beamformer of (10.22) for plane-wave signals. It is remarkable that the same result using a totally different approach is obtained. Observing that  $E\{\tilde{\mathbf{x}}'\} = s_0\mathbf{v}(\mathbf{k}_s)$ , the expected value of  $\hat{s}_0$  is

$$\begin{aligned} E\{\hat{s}_0\} &= \left(\frac{\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}}{\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}\mathbf{v}(\mathbf{k}_s)}\right) s_0\mathbf{v}(\mathbf{k}_s) \\ &= s_0. \end{aligned} \quad (14.47)$$

Hence this estimator is unbiased. The variance of the estimate is

$$\begin{aligned} \text{var}(\hat{s}_0) &= E\{|\hat{s}_0|^2\} - |E\{\hat{s}_0\}|^2 \\ &= E\left\{\frac{\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}\tilde{\mathbf{x}}'\tilde{\mathbf{x}}'^H\mathbf{R}_n^{-1}\mathbf{v}(\mathbf{k}_s)}{(\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}\mathbf{v}(\mathbf{k}_s))^2a}\right\} - |s_0|^2 \\ &= \frac{\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}E\{\tilde{\mathbf{x}}'\tilde{\mathbf{x}}'^H\}\mathbf{R}_n^{-1}\mathbf{v}(\mathbf{k}_s)}{(\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}\mathbf{v}(\mathbf{k}_s))^2a} - |s_0|^2 \end{aligned} \quad (14.48)$$

On the assumption that the noise is uncorrelated from block to block, i.e.,

$$E\left\{\tilde{\mathbf{n}}^{(m)}\tilde{\mathbf{n}}^{(m)H}\right\} = \mathbf{R}_n\delta_{mn} \quad (14.49)$$

it follows that

$$E\{\tilde{\mathbf{x}}'\tilde{\mathbf{x}}'^H\} = \frac{1}{M}\mathbf{R}_n + |s_0|^2\mathbf{v}(\mathbf{k}_s)\mathbf{v}^H(\mathbf{k}_s), \quad (14.50)$$

and thus the variance is given by

$$\text{var}(\hat{s}_0) = \frac{1}{M\mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_n^{-1}\mathbf{v}(\mathbf{k}_s)}. \quad (14.51)$$

<sup>2</sup>This is a necessary but not sufficient condition.

<sup>3</sup>The approach is due to Capon ([6]) and the resulting processor often called the Capon estimator. See Chapter 10.7

Now consider the Cramer-Rao bound. The log-likelihood function is

$$L_{\tilde{\mathbf{x}}}(s_0) = -K \ln(\pi) - \ln(\det(\mathbf{R}_n)) - \sum_{m=1}^M \left( \tilde{\mathbf{x}}^{(m)} - s_0 \mathbf{v}(\mathbf{k}_s) \right)^H \mathbf{R}_n^{-1}(f) \left( \tilde{\mathbf{x}}^{(m)} - s_0 \mathbf{v}(\mathbf{k}_s) \right) \quad (14.52)$$

so

$$\nabla_{s_0} \nabla_{s_0^*} L_{\tilde{\mathbf{x}}}(s_0) = -M \mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k}_s) \quad (14.53)$$

and the Cramer-Rao bound is

$$E\{|\mathbf{s}_0|^2\} - |E\{\mathbf{s}_0\}|^2 \geq \frac{1}{M \mathbf{v}^H(\mathbf{k}_s) \mathbf{R}_n^{-1}(f) \mathbf{v}(\mathbf{k}_s)} \quad (14.54)$$

The right-hand side of (14.54) is identical to that of (14.51), so in this case the least squares estimator attains the Cramer-Rao lower bound.

Note that assuming the noise  $\mathbf{n}$  has a zero-mean multivariate complex normal distribution

$$p(\mathbf{n}) = \frac{1}{\pi^K \det(\mathbf{R}_n)} \exp\{-\mathbf{n}^H \mathbf{R}_n^{-1} \mathbf{n}\} \quad (14.55)$$

then

$$p(\tilde{\mathbf{x}}^{(m)}; s_0) = \frac{1}{\pi^K \det(\mathbf{R}_n)} \exp\left\{-\left(\tilde{\mathbf{x}}^{(m)} - s_0 \mathbf{v}(\mathbf{k}_s)\right)^H \mathbf{R}_n^{-1} \left(\tilde{\mathbf{x}}^{(m)} - s_0 \mathbf{v}(\mathbf{k}_s)\right)\right\} \quad (14.56)$$

Assuming statistical independence of the noise from observation to observation, the probability density function of the block of data is given by

$$\begin{aligned} p\left(\tilde{\mathbf{x}}^{(1)}, \tilde{\mathbf{x}}^{(2)}, \dots, \tilde{\mathbf{x}}^{(M)}; \boldsymbol{\alpha}\right) &= \prod_{m=1}^M p(\tilde{\mathbf{x}}^{(m)}; \boldsymbol{\alpha}) \\ &= \frac{1}{\pi^K \det(\mathbf{R}_n)} \exp\left\{-\sum_{m=1}^M \left(\tilde{\mathbf{x}}^{(m)} - s_0 \mathbf{v}(\mathbf{k}_s)\right)^H \mathbf{R}_n^{-1} \left(\tilde{\mathbf{x}}^{(m)} - s_0 \mathbf{v}(\mathbf{k}_s)\right)\right\} \end{aligned} \quad (14.57)$$

Thus maximizing the log-likelihood function in this case is identical to minimizing the least squares, i.e.,

$$\hat{\mathbf{s}}_{\text{OML}} = \hat{\mathbf{s}}_{\text{OLS}} \quad (14.58)$$

### 14.5. Crossed dipoles

Here a rather unusual array is considered: it has three receivers, effectively collocated at the one point.<sup>a</sup> One of the receivers is omnidirectional; the other two are directional, with responses proportional to  $\sin \theta_s$  and  $\cos \theta_s$  respectively, where  $\theta_s$  is the angle of arrival of a plane-wave signal in the horizontal plane; these two directional receivers are named 'N-S' and 'E-W', respectively.

This type of array has been used in radio direction-finding and also in sonar for submarine detection and location. The directional receivers can be realised physically in a number of ways; for this example, it suffices that they be modelled by closely spaced dipoles.

<sup>a</sup>That is to say, the phase centres of the receivers are at the same point.

A dipole can be considered to comprise two receivers whose outputs are subtracted, as shown in Figure 14.5.

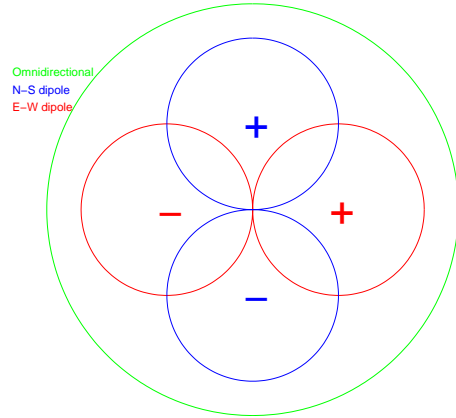


FIGURE 14.5. Polar responses

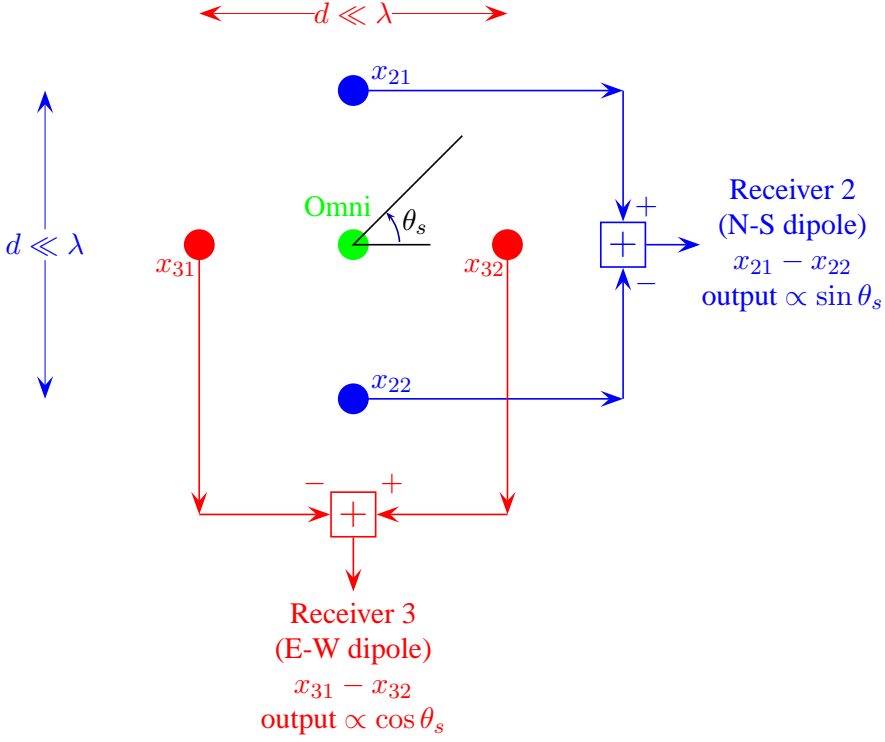


FIGURE 14.6. Crossed dipoles

Again in this example the dependence on frequency  $f$  shall be omitted. For a single signal in the absence of noise, the vector of receiver outputs takes the form

$$\mathbf{x} \equiv \tilde{\mathbf{x}}(f) = s \begin{bmatrix} 1 \\ \alpha(f) \sin \theta_s \\ \alpha(f) \cos \theta_s \end{bmatrix}, \quad (14.59)$$

$$= s \mathbf{v}(f, \theta_s),$$

where  $s \equiv s(f)$  is the spectral component of the signal at the omnidirectional receiver,  $\alpha \equiv \alpha(f)$  is the sensitivity of the directional receivers at frequency  $f$ , and

$$\mathbf{v} \equiv \mathbf{v}(f, \theta_s) = \begin{bmatrix} 1 \\ \alpha \sin \theta_s \\ \alpha \cos \theta_s \end{bmatrix}. \quad (14.60)$$

For the N-S dipole, when only signal is present, define

$$\begin{aligned} x_2 &= x_{21} - x_{22} \\ &\propto s \left( e^{i\pi d \sin \theta_s / \lambda} - e^{-i\pi d \sin \theta_s / \lambda} \right) \\ &\propto \sin \theta_s \quad \text{for } d \ll \lambda. \end{aligned} \quad (14.61)$$

Similarly,  $x_3 \propto \cos \theta_s$  for  $d \ll \lambda$ .

Next consider the case in which

- the ambient noise is isotropic in the horizontal plane,
- receiver noise is uncorrelated,
- signals and noises are narrow-band zero-mean gaussian random variables, and
- signal and noises are independent of one another.

Using the same subscript notation as before, the outputs of the two dipoles due to noise alone are

$$n_2 = n_{21} - n_{22}$$

and

$$n_3 = n_{31} - n_{32} \quad (14.62)$$

then

$$E\{n_1 n_2^*\} = E\{n_1 n_{21}^*\} - E\{n_1 n_{22}^*\} = 0$$

and

$$E\{n_2 n_3^*\} = E\{n_{21} n_{31}^*\} - E\{n_{21} n_{32}^*\} - E\{n_{22} n_{31}^*\} + E\{n_{22} n_{32}^*\} \quad (14.63)$$

From the symmetry of the element positions, it is evident that, for isotropic and receiver noise,

$$E\{n_{21} n_{31}^*\} = E\{n_{21} n_{32}^*\} = E\{n_{22} n_{31}^*\} = E\{n_{22} n_{32}^*\}$$

so

$$E\{n_2 n_3^*\} = 0 \quad (14.64)$$

Similarly  $E\{n_1 n_2^*\} = 0 = E\{n_1 n_3^*\}$  so the cross-spectral matrix for isotropic ambient noise plus receiver noise is diagonal:

$$\mathbf{R}_n = \sigma_n^2 \text{diag}\{1 \quad \zeta^2 \quad \zeta^2\}, \quad (14.65)$$

where  $\sigma_n^2 \triangleq (\sigma_a^2 + \sigma_r^2)$  and  $\sigma_a^2$  and  $\sigma_r^2$  are the ambient and receiver noise levels, respectively, and

$$\zeta^2 = \frac{\sigma_r^2 + \alpha^2 \sigma_a^2}{\sigma_r^2 + \sigma_a^2}. \quad (14.66)$$

The signal-to-noise ratios from the omnidirectional and directional receivers are, respectively

$$\mu_o = \frac{\sigma_s^2}{\sigma_n^2} \quad (14.67)$$

and

$$\mu_d = \frac{\alpha^2 \sigma_s^2}{\zeta^2 \sigma_n^2} \quad (14.68)$$



Noting that

$$\begin{aligned} E\{x_1 x_2^*\} &= \alpha \sigma_s^2 \sin \theta_s \\ E\{x_1 x_3^*\} &= \alpha \sigma_s^2 \cos \theta_s \\ E\{|x_1|^2\} &= \sigma_s^2 + \sigma_n^2 \end{aligned} \quad (14.69)$$

one is led intuitively to the following approach in estimating  $\theta_s$  when  $\sigma_s^2 \gg \sigma_n^2$  :

- collect a large number  $M$  of receiver samples  $\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(M)}\}$ ,
- estimate the cross-spectral matrix  $\hat{\mathbf{R}}_x = \frac{1}{M} \sum_{m=1}^M \mathbf{x}^{(m)} \mathbf{x}^{(m)H}$ ,
- take  $\alpha \sigma_s^2 \sin \hat{\theta}_s \approx \frac{[\hat{\mathbf{R}}_x]_{12}}{[\hat{\mathbf{R}}_x]_{11}}$  and  $\alpha \sigma_s^2 \cos \hat{\theta}_s \approx \frac{[\hat{\mathbf{R}}_x]_{13}}{[\hat{\mathbf{R}}_x]_{11}}$ , respectively,
- from their ratio calculate  $\tan \hat{\theta}_s$  and hence  $\hat{\theta}_s$ . (The quadrant for  $\tan \hat{\theta}_s$  is resolved from the signs of  $\sin \hat{\theta}_s$  and  $\cos \hat{\theta}_s$ .)

Next expressions for the maximum likelihood estimator and for the Cramer-Rao lower bound are derived.

The cross-spectral matrix of the receiver outputs is then

$$\mathbf{R}_x = \mathbf{R}_n + \sigma_s^2 \mathbf{v} \mathbf{v}^T \quad (14.70)$$

The probability density function of the receiver outputs is

$$p(\mathbf{x}) = \frac{1}{\pi^3 \det(\mathbf{R}_x)} \exp\{-\mathbf{x}^H \mathbf{R}_x^{-1} \mathbf{x}\} \quad (14.71)$$

and the log-likelihood function is

$$L_x(\theta_s) = -3 \ln \pi - \ln \det(\mathbf{R}_x) - \mathbf{x}^H \mathbf{R}_x^{-1} \mathbf{x}. \quad (14.72)$$

In what follows primes ( $'$ ) are used to denote derivatives  $\frac{\partial}{\partial \theta_s}$ :

$$\begin{aligned} \mathbf{v}' &= \frac{\partial \mathbf{v}}{\partial \theta_s}, \\ \mathbf{v}'' &= \frac{\partial^2 \mathbf{v}}{\partial \theta_s^2}, \text{ etc} \end{aligned} \quad (14.73)$$

$$\frac{\partial L_x(\theta_s)}{\partial \theta_s} = -\text{Tr}(\mathbf{R}_x^{-1} \mathbf{R}_x') + \mathbf{x}^H \mathbf{R}_x^{-1} \mathbf{R}_x' \mathbf{R}_x^{-1} \mathbf{x} \quad (14.74)$$

It is not difficult to show that

$$\begin{aligned} \mathbf{v}^T \mathbf{R}_n^{-1} \mathbf{v} &= \frac{1}{\sigma_s^2} (\mu_o + \mu_d) \\ \mathbf{v}^T \mathbf{R}_n^{-1} \mathbf{v}' &= 0, \end{aligned} \quad (14.75)$$

and hence that

$$\frac{\partial L_x(\theta_s)}{\partial \theta_s} = \frac{1}{1 + \sigma_s^2 (\mu_o + \mu_d)} \mathbf{x}^H (\mathbf{R}_n^{-1} \mathbf{v}' \mathbf{v}^T \mathbf{R}_n^{-1} + \mathbf{R}_n^{-1} \mathbf{v} \mathbf{v}^T \mathbf{R}_n^{-1}) \mathbf{x} \quad (14.76)$$

To obtain the maximum likelihood estimator, set  $\frac{\partial L_x(\theta_s)}{\partial \theta_s} = 0$ . The resulting equation can be expressed as a quartic in  $\tan \frac{\theta_s}{2}$  but does not appear to have a solution in closed form, so a numeric approach would be needed to find its roots. The maximum likelihood estimator would be the root corresponding the largest maximum of  $L_x$ .

For the Cramer-Rao lower bound use the following relationships which can be readily derived from (14.72):

$$\begin{aligned} \mathbf{v}^T \mathbf{R}_n^{-1} \mathbf{v}' &= \mathbf{v}'^T \mathbf{R}_n^{-1} \mathbf{v} = 0 \\ \mathbf{v}^T \mathbf{R}_n^{-1} \mathbf{v}'' &= \mathbf{v}''^T \mathbf{R}_n^{-1} \mathbf{v} = -\sigma_s^2 \mu_d \\ \mathbf{v}'^T \mathbf{R}_n^{-1} \mathbf{v}' &= \sigma_s^2 \mu_d \end{aligned} \tag{14.77}$$

these are then used to give

$$\begin{aligned} \text{var}(\hat{\theta}) &\geq \left( E \left\{ \frac{\partial^2 L_x(\theta_s)}{\partial \theta_s^2} \right\} \right)^{-1} \\ &\geq \sigma_s^2 \left( \frac{1 + \mu_o + \mu_d}{2\mu_d(\mu_o + \mu_d)} \right) \end{aligned} \tag{14.78}$$

## CHAPTER 15

# SUBSPACE METHODS

### 15.1. Introduction

Eigenvalues and eigenvectors are important in many physical systems. Typically, an eigenvector corresponds to a natural mode of oscillation and its corresponding eigenvalue gives the intensity of that oscillation. Not surprisingly, they also play an important part in beamforming.

In this Chapter we work in the frequency domain and introduce beamforming using the so-called *subspace* or *eigenanalysis* methods. These are based on the notion that signals and noise are usually independent of one another, and so span different subspaces. If we can find these subspaces, then it is possible to distinguish clearly between the two.

Some revision notes on eigenvalues and eigenvectors are provided in [Appendix C](#); here we briefly highlight some essential points.

As in earlier Chapters, the cross-spectral matrix of the receiver outputs,  $\mathbf{R}_x$ , plays a central role. We shall assume that  $\mathbf{R}_x$  is non-singular, and as a consequence its eigenvalues will all be positive. We shall arrange the eigenvalues in descending order:

$$\lambda_{\min} \equiv \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_K \equiv \lambda_{\max}. \quad (15.1)$$

This ordered set of eigenvalues of a cross-spectral matrix is called its *eigenspectrum*. Let  $\{\mathbf{q}_{\max} \equiv \mathbf{q}_1, \cdots, \mathbf{q}_K \equiv \mathbf{q}_{\min}\}$  be the corresponding eigenvectors, and let us arrange these eigenvectors as the columns of the  $K \times K$  matrix:

$$\mathbf{Q} = [\mathbf{q}_1 : \mathbf{q}_2 : \cdots : \mathbf{q}_K]. \quad (15.2)$$

The eigenvectors are by convention scaled to be orthonormal:

$$\begin{aligned} \mathbf{q}_i^H \mathbf{q}_j &= \delta_{ij}, \\ \delta_{ij} &= 1, \quad i = j, \\ &= 0, \quad i \neq j. \end{aligned} \quad (15.3)$$

$\mathbf{Q}$  is a *unitary* matrix (i.e.,  $\mathbf{Q}\mathbf{Q}^H = \mathbf{I}$ ), and it is readily seen that

$$\mathbf{R}_x = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^H, \quad (15.4)$$

where  $\mathbf{\Lambda}$  is the diagonal matrix:

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_K \end{bmatrix} \quad (15.5)$$

Any subset of the eigenvectors spans a **subspace** which is orthonormal to the subspace spanned by the remaining eigenvectors.

### 15.2. Uncorrelated noise only

Let us begin by considering the simple case in which the noises are all independent and identically distributed. The cross-spectral matrix then takes the form:

$$\mathbf{R}_n(f) = \sigma_n^2(f) \mathbf{I}. \quad (15.6)$$

In this case, all the eigenvalues are the same ( $\lambda_k = \sigma_n^2 \forall k$ ) and the eigenvectors are not unique ( $\mathbf{R}_n(f) = \sigma_n^2(f) \mathbf{Q} \mathbf{Q}^H$  for *any* unitary matrix  $\mathbf{Q}$ ).

### 15.3. Single signal and uncorrelated noise

Next take the case in which there is a single signal plane-wave arrival with power  $\sigma_s^2(f)$  and arriving from a direction corresponding to the steering vector  $\mathbf{v}(\mathbf{k}_s)$ , with noises again independent and identically distributed; the cross-spectral matrix is then:

$$\mathbf{R}_x(f) = \sigma_s^2(f) \mathbf{v}(\mathbf{k}_s) \mathbf{v}^H(\mathbf{k}_s) + \sigma_n^2(f) \mathbf{I}. \quad (15.7)$$

It can easily be shown<sup>1</sup> that

$$\det(\mathbf{R}_x(f) - (K\sigma_s^2(f) + \sigma_n^2(f))\mathbf{I}) = 0. \quad (15.8)$$

We conclude that one eigenvalue is  $K\sigma_s^2(f) + \sigma_n^2(f)$ . It can readily be ascertained that the corresponding eigenvector is<sup>2</sup>

$$\mathbf{q}_s = \frac{\mathbf{v}(\mathbf{k}_s)}{\sqrt{K}}. \quad (15.9)$$

Similarly, since  $\det(\mathbf{R}_x(f) - \sigma_n^2(f)\mathbf{I}) = 0$ , all the remaining eigenvalues are equal to  $\sigma_n^2(f)$ . Hence

$$\begin{aligned} \lambda_1 &= K\sigma_s^2(f) + \sigma_n^2(f) \quad \text{and} \\ \lambda_k &= \sigma_n^2(f), \quad k = 2, \dots, K. \end{aligned}$$

Thus we can conveniently express  $\mathbf{R}_x(f)$  in terms of partitioned matrices:

<sup>1</sup>The determinant of a nonsingular matrix plus a dyad is given by (see (C.9)):

$$\det(\mathbf{R} - c\mathbf{v}\mathbf{v}^H) = \det(\mathbf{R})(1 - c\mathbf{v}^H\mathbf{R}^{-1}\mathbf{v})$$

<sup>2</sup>In general, if the plane-wave signals are strong compared to other forms of noise, the eigenvectors of the signal subspace will be approximately proportional to the signal wavevectors.

$$\mathbf{R}_x = \begin{bmatrix} \frac{\mathbf{v}(\mathbf{k}_s)}{\sqrt{K}} & \vdots & \mathbf{F} \end{bmatrix} \begin{bmatrix} K\sigma_s^2 + \sigma_n^2 & \vdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \vdots & \sigma_n^2 \mathbf{I} \end{bmatrix} \begin{bmatrix} \frac{\mathbf{v}^H(\mathbf{k}_s)}{\sqrt{K}} \\ \vdots \\ \mathbf{F}^H \end{bmatrix}, \quad (15.10)$$

where  $\mathbf{F}$  is some  $[K \times (K-1)]$  matrix whose columns are orthonormal:

$$\mathbf{F}^H \mathbf{F} = \mathbf{I}_{[(K-1) \times (K-1)]}. \quad (15.11)$$

Note that

- only the first (the largest) eigenvalue is related to the signal;
- its eigenvector is proportional to the signal arrival vector  $\mathbf{v}(\mathbf{k}_s)$ ;
- the single signal vector defines a one-dimensional subspace called the *signal subspace*;
- the remaining eigenvectors, which constitute  $\mathbf{F}$ , span the *noise subspace*;
- the signal arrival vector  $\mathbf{v}(\mathbf{k}_s)$  is orthogonal to the noise subspace:

$$\mathbf{F}^H \mathbf{v}(\mathbf{k}_s) = \mathbf{0}_{[K-1]}, \quad (15.12)$$

where  $\mathbf{0}_{[K-1]}$  denotes a  $(K-1)$ -vector of zeros.

#### 15.4. $L$ signals in uncorrelated noise

We can extend these results to the case in which there are  $L < K$  independent signal sources and uncorrelated additive noise  $\sigma_n^2(f)\mathbf{I}$ .

Recall from (B.22) that the cross-spectral matrix of the receiver outputs can be expressed as

$$\mathbf{R}_x(f) = \mathbf{V} \mathbf{S}(f) \mathbf{V}^H + \sigma_n^2(f) \mathbf{I}, \quad (15.13)$$

$$\text{where } [\mathbf{S}]_{i,j} = E \{ \tilde{s}_i(f) \tilde{s}_j^*(f) \}$$

and the matrix of steering vectors

$$\mathbf{V} = [\mathbf{v}(\mathbf{k}_1) \quad \mathbf{v}(\mathbf{k}_2) \quad \cdots \quad \mathbf{v}(\mathbf{k}_L)]. \quad (15.14)$$

In general these signals need not be independent, but we shall consider them to be so, in which case  $\mathbf{S}$  is diagonal:

$$\mathbf{S}(f) = \begin{bmatrix} \sigma_1^2(f) & 0 & \cdots & 0 \\ 0 & \sigma_2^2(f) & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_L^2(f) \end{bmatrix} \quad (15.15)$$

$\mathbf{V} \mathbf{S}(f) \mathbf{V}^H$  is of rank  $L < K$ ; it can be expressed in terms of its non-zero eigenvalues as follows:

$$[\mathbf{V} \mathbf{S}(f) \mathbf{V}^H]_{[K \times K]} = \mathbf{E}_{[K \times L]} \mathbf{\Lambda}_s [L \times L] \mathbf{E}_{[L \times K]}^H \quad (15.16)$$

It is easy to show that the cross-spectral matrix  $\mathbf{R}_x(f)$  then has the following eigen-decomposition:

$$\mathbf{R}_x(f) = \begin{bmatrix} \mathbf{E} & \vdots & \mathbf{F} \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_s + \sigma_n^2 \mathbf{I} & \vdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \vdots & \sigma_n^2 \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{E}^H \\ \vdots \\ \mathbf{F}^H \end{bmatrix}, \quad (15.17)$$

where the columns of  $\mathbf{E}_{[K \times L]}$  and  $\mathbf{F}_{[K \times (K-L)]}$  are the eigenvectors of  $\mathbf{R}_x(f)$ .

In this more general case, again we again have a signal subspace defined by the signal vectors  $\mathbf{V}$  and a noise subspace defined by the eigenvectors  $\mathbf{F}$ .

This is illustrated in the adjacent sketch for the simple case of a linear array with three receivers and two independent signal arrivals. The signal arrival vectors,  $\mathbf{v}_1$  and  $\mathbf{v}_2$ , define the signal subspace which is orthogonal to the noise subspace  $\mathbf{q}_3$ .

We introduce here the concept of the **array manifold** which is defined as the locus of all steering vectors as  $\{\theta, \phi\}$  is varied. For a linear array it is a curve in  $2K$ -dimensional space but in general it is a surface.

The array manifold for this linear array intersects the signal subspace at the points  $\mathbf{v}_1$  and  $\mathbf{v}_2$  as sketched.

Note that:

- the  $L$  largest eigenvalues can be used to determine the signal subspace;
- the remaining  $K - L$  eigenvalues define the noise subspace;
- $\mathbf{E}_{[K \times L]}$  and  $\mathbf{F}_{[K \times (K-L)]}$  are orthogonal:  $\mathbf{E}^H \mathbf{F} = \mathbf{0}$ ;
- $\mathbf{E} \mathbf{E}^H + \mathbf{F} \mathbf{F}^H = \mathbf{I}$ .

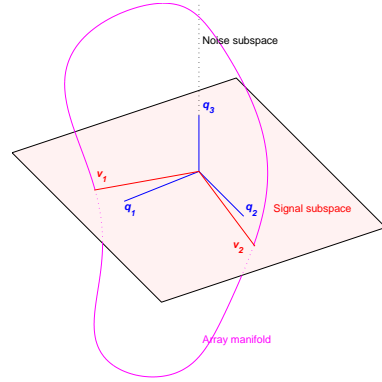


FIGURE 15.1. Illustrating subspaces and manifold

Let us define matrix

$$\mathbf{P} \triangleq \mathbf{F} \mathbf{F}^H = \sum_{\ell=L+1}^K \mathbf{q}_\ell \mathbf{q}_\ell^H$$

$\mathbf{P}$  is a projection operator: for any vector  $\mathbf{x}$ , the projected vector  $\mathbf{P} \mathbf{x}$  lies in the noise subspace.

We can express  $\mathbf{P}$  in terms of the  $[K \times L]$  matrix  $\mathbf{V}$  of steering vectors (15.14):

$$\mathbf{P} = \mathbf{I} - \mathbf{V} \mathbf{V}^\dagger \quad (15.18)$$

where the  $[(K - L) \times K]$  matrix  $\mathbf{V}^\dagger$  is the Moore-Penrose pseudo-inverse of  $\mathbf{V}$  (see (C.10.3))<sup>3</sup>.

$\mathbf{P}$  is Hermitian symmetric ( $\mathbf{P}^H = \mathbf{P}$ ) and *idempotent* (i.e.,  $\mathbf{P}^2 = \mathbf{P}$ ).

If  $\mathbf{V}$  is of full rank,

$$\mathbf{V}^\dagger = (\mathbf{V}^H \mathbf{V})^{-1} \mathbf{V}^H \quad (15.19)$$

$$\text{and } \mathbf{P} = \mathbf{I} - \mathbf{V} (\mathbf{V}^H \mathbf{V})^{-1} \mathbf{V}^H. \quad (15.20)$$

We shall use these results in following sections.

## 15.5. Estimating the direction of arrival

### 15.5.1. Introduction.

Subspace techniques can be used to estimate the directions of arrival of  $L$  plane waves incident on the array. Such *eigenanalysis methods* are based on estimating the number of signal sources and then partitioning the cross-spectral matrix into signal and noise subspaces.

It was mentioned in Section 15.4 that, when the noises are independent of one another, the largest  $L$  eigenvalues correspond to the signal subspace.

---

<sup>3</sup>In fact,  $(\mathbf{I} - \mathbf{V} \mathbf{V}^-)$  is also a projection operator onto the noise subspace, where  $\mathbf{V}^-$  is any generalised inverse of  $\mathbf{V}$ .

$$\mathbf{R}_x(f) = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^H, \quad (15.21)$$

$$\text{where } \mathbf{\Lambda} = \left[ \begin{array}{ccc|ccc} \lambda_1 & \cdots & 0 & & & \\ \vdots & \ddots & 0 & & \mathbf{0} & \\ 0 & & \lambda_L & & & \\ \hline & & & \lambda_{L+1} & \cdots & 0 \\ & \mathbf{0} & & & \ddots & \\ & & & 0 & & \lambda_K \end{array} \right] \quad (15.22)$$

$$= \left[ \begin{array}{ccc} \mathbf{\Lambda}_{\text{Signal subspace}} & \vdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \vdots & \mathbf{\Lambda}_{\text{Noise subspace}} \end{array} \right],$$

$$\text{and } \mathbf{Q} = \left[ \begin{array}{ccc|ccc} \mathbf{q}_1 & \cdots & \mathbf{q}_L & \vdots & \mathbf{q}_{L+1} & \cdots & \mathbf{q}_K \\ \hline & \text{Signal subspace} & & & \text{Noise subspace} & & \end{array} \right] \quad (15.23)$$

$$= \left[ \begin{array}{ccc} \underbrace{\mathbf{E}_{[K \times L]}}_{\text{Signal subspace}} & \vdots & \underbrace{\mathbf{F}_{[K \times (K-L)]}}_{\text{Noise subspace}} \end{array} \right].$$

Once the partitioning has been effected, we can apply one of several methods of estimating the arrival vectors of the signal sources<sup>4</sup>.

### 15.6. Pisarenko's method

Pisarenko's approach[32], although not particularly useful in practice, is nonetheless interesting because it introduces other, more practical, subspace methods.

It is assumed here that there are  $L = K - 1$  arrivals and that their steering vectors  $\mathbf{v}(\theta_j)$ ,  $j = 1, \dots, K - 1$  are linearly independent<sup>5</sup>. There will then be only one 'noise eigenvector',  $\mathbf{q}_K \equiv \mathbf{q}_{\min}$ . From (15.12),

$$\mathbf{v}^H(\theta_j) \mathbf{q}_K = 0, \quad j = 1, \dots, K - 1.$$

A plot of the function

$$p_{\text{Pisarenko}} = \frac{1}{|\mathbf{v}^H(\theta) \mathbf{q}_K|^2} \quad (15.24)$$

<sup>4</sup>Here we do not distinguish between wanted signals and unwanted interference: we only seek to separate signal sources from receiver and ambient noise.

<sup>5</sup>We confine ourselves for simplicity to azimuthal angles  $\theta_j$  but, as pointed out earlier, we can simply extend the process to  $(\theta_j, \phi_j)$ .



will exhibit spikes, theoretically infinite, whenever  $\theta$  equals one of the signal directions  $\theta_j$ . This function will thus serve as an estimator of the directions of arrival of the signals<sup>6</sup>. In practice  $L$  will usually not equal  $K - 1$  and the Pisarenko method then will give false peaks.

### 15.7. Multiple Signal Classification (MUSIC)

A popular technique, called *Multiple Signal Classification (MUSIC)*[36], is a major improvement on the Pisarenko method.

Recall that the array manifold was defined as the locus of  $\mathbf{v}(\theta)$  as  $\theta$  is varied. If we project  $\mathbf{v}(\theta)$  onto the noise subspace, its vector component in that subspace is given by

$$\mathbf{P}\mathbf{v}(\theta) = \sum_{\ell=L+1}^K \mathbf{q}_\ell \mathbf{q}_\ell^H \mathbf{v}(\theta), \quad (15.25)$$

which is zero when  $\theta = \theta_\ell$ ,  $\ell = L + 1, \dots, K$ .

Using the fact that  $\mathbf{P}$  is Hermitian symmetric and idempotent, the Euclidean norm of the projected vector in the noise subspace is

$$\mathbf{v}^H(\theta) \mathbf{P}^H \mathbf{P} \mathbf{v}(\theta) = \mathbf{v}^H(\theta) \mathbf{P} \mathbf{v}(\theta) \quad (15.26)$$

$$= \sum_{\ell=L+1}^K \mathbf{v}^H(\theta) \mathbf{q}_\ell \mathbf{q}_\ell^H \mathbf{v}(\theta) \quad (15.27)$$

$$= \sum_{\ell=L+1}^K |\mathbf{v}^H(\theta) \mathbf{q}_\ell|^2. \quad (15.28)$$

The MUSIC algorithm plots the reciprocal of the Euclidean norm as a function of  $\theta$  – i.e., it traverses the array manifold.

$$p_{\text{MUSIC}} = \frac{1}{\sum_{\ell=L+1}^K |\mathbf{v}^H(\theta) \mathbf{q}_\ell|^2} \quad (15.29)$$

The peaks of  $p_{\text{MUSIC}}$ , theoretically infinite, indicate the signal arrival directions  $\{\theta_\ell\}$ .

---

<sup>6</sup>The heights of the peaks do *not* indicate signal power.

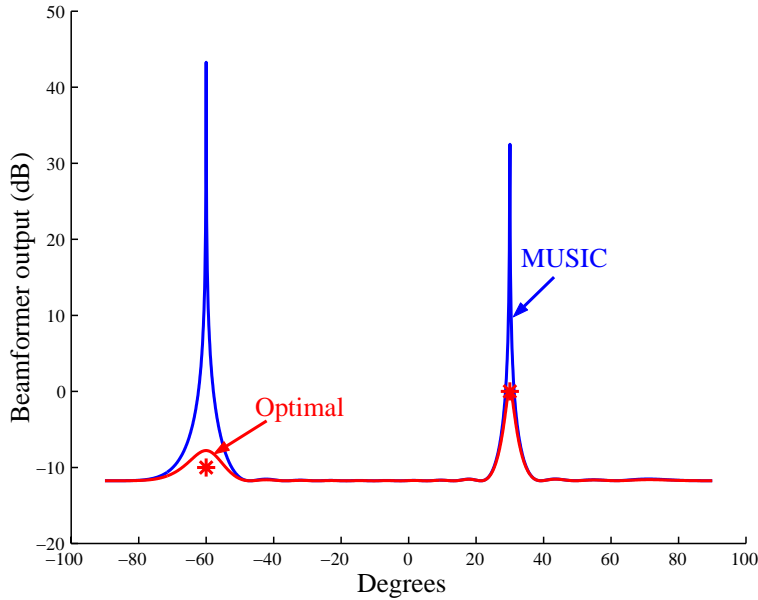


FIGURE 15.2. Steered beamformer outputs for MUSIC and Optimal processors, for a linear array of 15 receivers spaced half a wavelength apart. There are two arrivals, both at  $\phi_s = 90^\circ$ , and from horizontal angles  $\theta_s = -60^\circ$  and  $\theta_s = +30^\circ$ , and SNRs of  $-10\text{dB}$  and  $0\text{dB}$  respectively.

This function is shown plotted in Figure 15.7 for the case of two signals, at  $-60^\circ$  and  $30^\circ$ , with amplitudes of  $-10\text{dB}$  and  $0\text{dB}$  respectively, and uncorrelated noise of amplitude  $0\text{dB}$ . For comparison, the output of the optimal (MVDR) beamformer is also shown.

As expected, the output of the MUSIC processor shows large, precise peaks in the directions of the signals<sup>7</sup>. When the model assumed for the noise accurately reflects the physical world, use of the MUSIC method (see (15.7)) gives a more precise estimate of position than does the optimal processor (15.21).

It is stressed that this subspace processor by itself only provides an estimate of signal direction but – unlike the conventional and optimal beamformers – not of signal strength. However, signal strength can be estimated in a straightforward manner using one of several techniques.

<sup>7</sup>It is important not to confuse *precision* with *accuracy*: an estimate can be very precise – defined to many significant figures – but inaccurate.

### 15.8. Eigenvector method

Recall (see 10.9.3) that the output of the MVDR beamformer can be used as an estimator of the direction of signal arrivals by calculating

$$p_{\text{MVDR}}(\theta) = \frac{1}{\mathbf{v}^H(\theta) \mathbf{R}_x^{-1}(f) \mathbf{v}(\theta)}. \quad (15.30)$$

The inverse of the matrix  $\mathbf{R}_x(f)$  of (15.21) is<sup>8</sup>

$$\mathbf{R}_x^{-1} = \mathbf{Q} \begin{bmatrix} \lambda_1^{-1} & 0 & \cdots & 0 \\ 0 & \lambda_2^{-1} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_K^{-1} \end{bmatrix} \mathbf{Q}^H. \quad (15.31)$$

Substituting (15.31) in (15.30) gives:

$$\begin{aligned} p_{\text{MVDR}}(\theta) &= \frac{1}{\sum_{i=1}^K \lambda_i^{-1} |\mathbf{v}^H(\theta) \mathbf{q}_i|^2} \\ &= \frac{1}{\underbrace{\sum_{i=1}^L \lambda_i^{-1} |\mathbf{v}^H(\theta) \mathbf{q}_i|^2}_{\text{SIGNAL SUBSPACE}} + \underbrace{\sum_{i=L+1}^K \lambda_i^{-1} |\mathbf{v}^H(\theta) \mathbf{q}_i|^2}_{\text{NOISE SUBSPACE}}}. \end{aligned} \quad (15.32)$$

When the noise is uncorrelated,  $\lambda_i = \sigma_n^2$ ,  $i = L + 1, \dots, K$ , and the second term in the denominator of (15.32) would, except for a scale factor of  $\sigma_n^2$ , be the same as that of the MUSIC estimator.

This has led to the suggestion[21] to use just the second term in the denominator:

$$p_{\text{EigenMethod}}(\theta) = \frac{1}{\sum_{i=L+1}^K \lambda_i^{-1} |\mathbf{v}^H(\theta) \mathbf{q}_i|^2}. \quad (15.33)$$

This is called the *Eigenvector method*. It is similar to MUSIC:

- the directions of arrival are indicated by very sharp peaks;
- there is no indication of the levels of the signals;
- it relies on being able to separate signal and noise subspaces.

▷Ex 10

<sup>8</sup>This follows directly from the fact that the matrix of eigenvectors  $\mathbf{Q} = [\mathbf{E}|\mathbf{F}]$  is unitary:  $\mathbf{Q}\mathbf{Q}^H = \mathbf{I}$ .

## 15.9. Estimating the number of signals

### 15.9.1. Introduction.

To apply these subspace techniques successfully, it is necessary to decide how many signals are present in order to be able to partition the cross-spectral matrix correctly.

Figure (15.9.1) shows a plot of the 15 eigenvalues, in rank order, for the cross-spectral matrix of the output of a linear array of 15 equi-spaced receivers. There are four signals present, each of strength 0dB, and uncorrelated receiver noise of strength 0dB.

In this rather idealised case, the eigenvalues associated with the noise can easily be distinguished from those associated with signal.

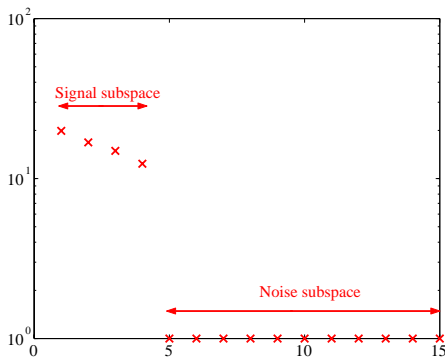


FIGURE 15.3. Linear array, 15 receivers uniform spacing  $d = \lambda/2$ , 4 arrivals all with  $SNR = 0\text{dB}$ . Noise and signal subspaces are easily distinguishable

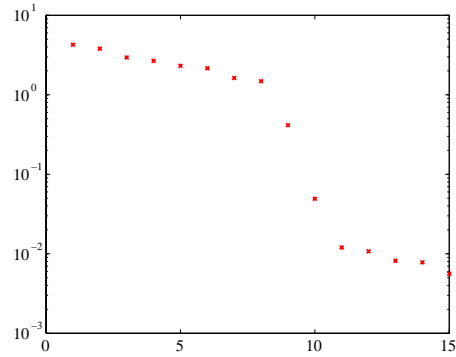


FIGURE 15.4. Linear array, 15 receivers uniform spacing  $d = \lambda/4$ , 4 arrivals all with strength  $-10\text{dB}$ , receiver noise  $0\text{dB}$ , spherically isotropic ambient noise  $0\text{dB}$ . Noise and signal subspaces difficult to separate

Figure (15.9.1) shows a similar plot for the same array, with receivers spaced  $\lambda/4$  apart. There is uncorrelated receiver noise of strength  $-20\text{dB}$  and again there are four signals present, each of level  $-10\text{dB}$ . In addition, there is spherically isotropic ambient noise of strength  $0\text{dB}$ . The estimated cross-spectral matrix  $\hat{\mathbf{R}}_x$  is the average of 100 samples.

The signal and noise subspaces are now difficult to separate. By eye, one might decide that there are present 8 or more signals, when in reality there are only 4.

It is not satisfactory to rely solely on human judgement to give estimates of the number of signals present, so efforts have been made to develop mathematical techniques to assist in the decision. Two have been shown to be effective: the  $A$

*Information Criterion* (AIC) and *Minimum Description Length* (MDL) criterion. These take into account the distribution of noise and the number of samples used to estimate the cross-spectral matrix. The results are stated below without proof.

### 15.9.2. AIC technique.

For normal distributions the AIC technique calculates a function  $AIC(L)$  :

$$AIC(L) = -M(K - L) \ln \left( \frac{\left( \prod_{\ell=L+1}^K \hat{\lambda}_{\ell} \right)^{1/(K-L)}}{\frac{1}{(K-L)} \sum_{\ell=L+1}^K \hat{\lambda}_{\ell}} \right) + L(2K - L + 1) \quad (15.34)$$

The number of sources is taken to be that value of  $L$  that minimises  $AIC(L)$ .

In (15.34),  $\{\hat{\lambda}_{\ell}, \ell = L + 1, \dots, K\}$  are the  $K - L$  smallest eigenvalues of the estimated cross-spectral matrix  $\hat{\mathbf{R}}_x$  and  $M$  is the number of samples used in making that estimate. As before,  $K$  is the number of receivers. The term in large parentheses is the ratio of the geometric and arithmetic means of the  $K - L$  smallest eigenvalues.

### 15.9.3. MDL technique.

The function to be minimised in the case of the MDL criterion is very similar:

$$MDL(L) = -M(K - L) \ln \left( \frac{\left( \prod_{\ell=L+1}^K \hat{\lambda}_{\ell} \right)^{1/(K-L)}}{\frac{1}{(K-L)} \sum_{\ell=L+1}^K \hat{\lambda}_{\ell}} \right) + \frac{L(2K - L + 1)}{2} \ln(M) \quad (15.35)$$

Each of these techniques has advantages and disadvantages, depending on the actual application.

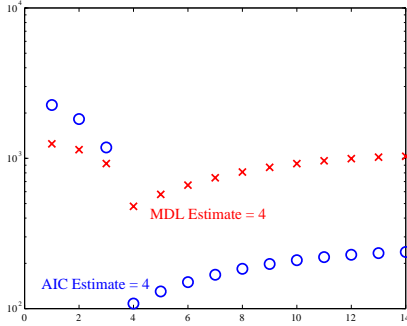


FIGURE 15.5. Estimating the number of sources using AIC and MDL techniques. Linear array, 15 receivers uniform spacing  $d = \lambda/2$ , 4 arrivals all with  $SNR = 0\text{dB}$ .

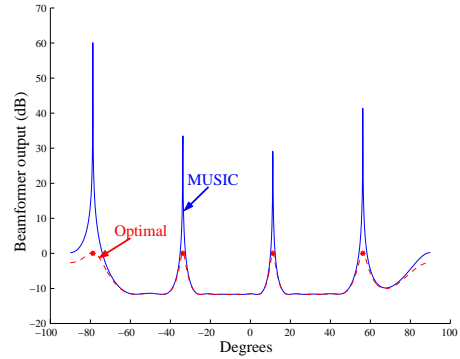


FIGURE 15.6. Steered beamformer output for MUSIC and optimal processors.

#### 15.9.4. Some results.

Figure 15.9.3 shows a plot of  $AIC(L)$  and  $MDL(L)$  for the example of Figure 15.9.1 where the signal and noise subspaces are easily separable. Both techniques correctly give the number of sources as 4.

For this same case, Figure 15.9.3 shows the steered beamformer output for MUSIC and for the optimal processor. Again we find that, when the model accurately represents the real situation, subspace methods perform well.

However, when the number of signal sources is not estimated correctly, these two subspace methods give misleading results.

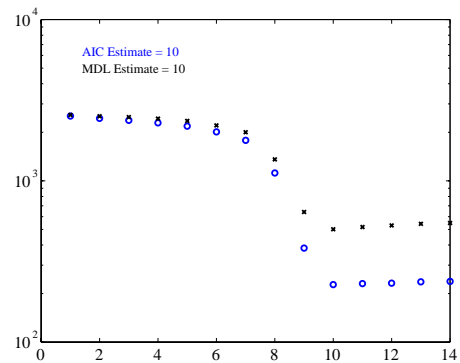


FIGURE 15.7. Plots of  $AIC(L)$  and  $MDL(L)$ . Linear array, 15 receivers uniform spacing  $d = \lambda/4$ , 4 arrivals all with strength  $-10\text{dB}$ , receiver noise  $0\text{dB}$ , spherically isotropic ambient noise  $0\text{dB}$ .

Consider the case corresponding to Figure 15.9.1. This is a situation in which the noise and signal subspaces are not easily separable, and both the AIC and MDL methods overestimate the number of sources, picking 10 as shown in Figure 15.9.4.

If we accept that estimate of 10, quite poor results follow. The resulting steered beamformer output for MUSIC is shown in Figure 15.9.4, together with the corresponding plot for the optimal beamformer. Also shown is what would have resulted if somehow the MUSIC algorithm could have been used with the true number of sources (4). The true positions and levels of the signals are indicated by \*. It is noteworthy that, even when the number of signals is correctly chosen, there is a significant error in the estimate of position for one of the signals (and the optimal beamformer makes a similar error).

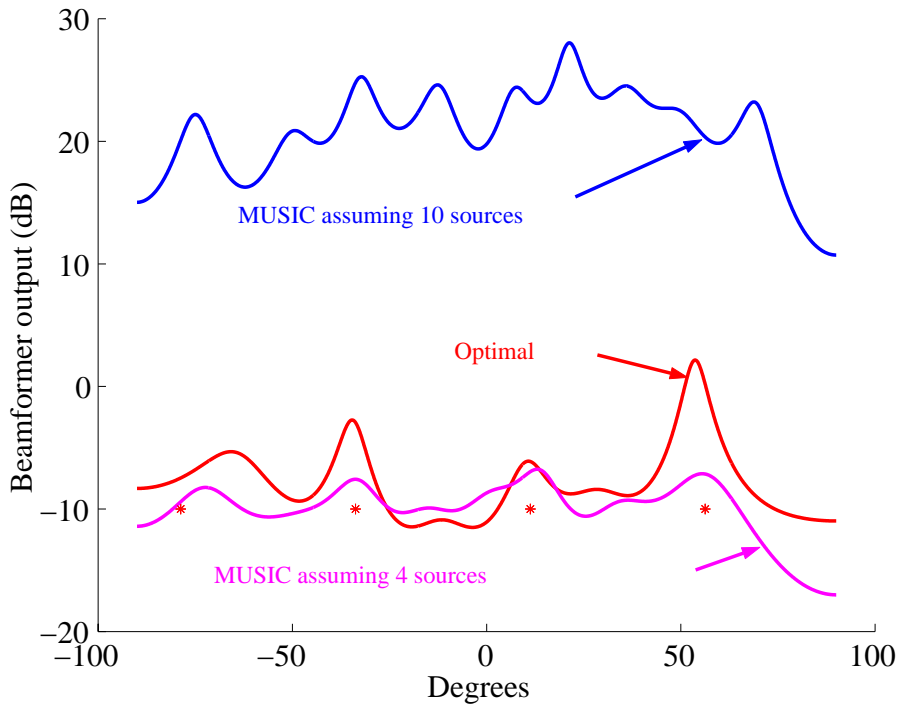


FIGURE 15.8. Steered beamformer output for MUSIC with correct and incorrect estimate of number of sources. Linear array, 15 receivers uniform spacing  $d = \lambda/4$ , 4 arrivals all with strength  $-10\text{dB}$ , receiver noise  $0\text{dB}$ , spherically isotropic ambient noise  $0\text{dB}$ .

## 15.10. ESPRIT

### 15.10.1. Concept.

This algorithm, originally proposed by Roy [30], is an acronym for *Estimation of Signal Parameters via Rotational Invariance Principles*. As the name suggests,

it estimates signal arrival direction by exploiting the rotational invariance of the signal subspaces of subsets of the array receivers. There are several versions of ESPRIT. Here we consider two: Least-squares (LS) and Total least-squares (TLS).

To illustrate the concept, consider a uniform linear array of  $K$  receivers spaced  $d$  apart. We can select any pair of identical sub-arrays such as the two examples shown in Figure 15.9, with the second sub-array shifted  $k'$  receivers to the right.

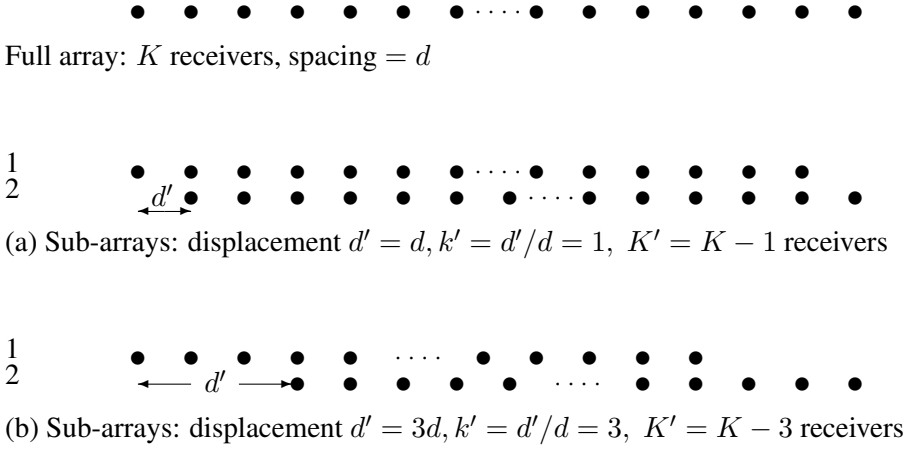


FIGURE 15.9. Illustrating matched sub-arrays

Recall from 8.19 that the receiver output can be modelled as follows:<sup>9</sup>

$$\mathbf{x}(\mathbf{k}_s) = \mathbf{V} \mathbf{s}(\mathbf{k}_s) + \mathbf{n}. \quad (15.36)$$

Consider a uniform linear array with  $L$  signals incident upon it from directions  $\theta_\ell, \ell = 1, \dots, L < K'$ .  $\mathbf{V}$  then takes the form<sup>10</sup>

$$\mathbf{V} = [\mathbf{v}_1 : \mathbf{v}_2 : \dots : \mathbf{v}_L] = \begin{bmatrix} 1 & 1 & \dots & 1 \\ z_1 & z_2 & \dots & z_L \\ z_1^2 & z_2^2 & \dots & z_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{K'-1} & z_2^{K'-1} & \dots & z_L^{K'-1} \end{bmatrix},$$

where  $z_\ell = \exp(2\pi i d \sin \theta_\ell / \lambda)$ .

In what follows, we assume that the columns of  $\mathbf{V}$  are independent.

<sup>9</sup>For brevity we omit the explicit dependence on frequency  $f$ .

<sup>10</sup>For a uniform linear array,  $\mathbf{V}$  is a so-called Vandermonde matrix.



Let  $\mathbf{B}_1$  and  $\mathbf{B}_2$  be the selection matrices to generate the sub-arrays from the full array. For the examples shown in Figure 15.9, we have

$$\begin{aligned}
 \text{(a)} \quad \mathbf{B}_1 &= \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & & 0 & 0 \\ \vdots & & \ddots & \vdots & \vdots \\ 0 & 0 & \ddots & 1 & 0 \end{bmatrix} = \left[ \mathbf{I}_{(K-k') \times (K-k')} \vdots \mathbf{0}_{(K-k') \times L} \right] \\
 \mathbf{B}_2 &= \begin{bmatrix} 0 & 1 & & \cdots & 0 \\ 0 & & 1 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & & \ddots & 1 \end{bmatrix} = \left[ \mathbf{0}_{(K-k') \times L} \vdots \mathbf{I}_{(K-k') \times (K-k')} \right] \\
 \text{(b)} \quad \mathbf{B}_1 &= \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & & 0 & 0 & 0 & 0 \\ \vdots & & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ddots & 1 & 0 & 0 & 0 \end{bmatrix} = \left[ \mathbf{I}_{(K-3) \times (K-3)} \vdots \mathbf{0}_{(K-3) \times L} \right] \\
 \mathbf{B}_2 &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 1 & & 0 \\ \vdots & \vdots & \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \ddots & 1 \end{bmatrix} = \left[ \mathbf{0}_{(K-3) \times L} \vdots \mathbf{I}_{(K-3) \times (K-3)} \right]
 \end{aligned} \tag{15.37}$$

The outputs of the sub-arrays are

$$\mathbf{x}_1 = \mathbf{V}_1 \mathbf{s} + \mathbf{n}_1, \tag{15.38}$$

$$\text{and} \quad \mathbf{x}_2 = \mathbf{V}_2 \mathbf{s} + \mathbf{n}_2, \tag{15.39}$$

where  $\mathbf{V}_j = \mathbf{B}_j \mathbf{V}$ ,  $\mathbf{n}_j = \mathbf{B}_j \mathbf{n}$ ,  $j = 1, 2$ ,

with cross-spectral matrices

$$\begin{aligned}
 \mathbf{R}_{x_1 x_1} &= E\{\mathbf{x}_1 \mathbf{x}_1^H\} \\
 \text{and} \quad \mathbf{R}_{x_2 x_2} &= E\{\mathbf{x}_2 \mathbf{x}_2^H\}.
 \end{aligned} \tag{15.40}$$

Consider the sub-arrays in Figure 15.9 (in which the second sub-array is displaced by  $k'$  receivers to the right relative to the first):

$$\begin{aligned}
 \mathbf{V}_1 &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_L \\ z_1^2 & z_2^2 & \cdots & z_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{K-k'-1} & z_2^{K-k'-1} & \cdots & z_L^{K-k'-1} \end{bmatrix} \\
 \text{and } \mathbf{V}_2 &= \begin{bmatrix} z_1^{k'} & z_2^{k'} & \cdots & z_L^{k'} \\ z_1^{k'+1} & z_2^{k'+1} & \cdots & z_L^{k'+1} \\ z_1^{k'+2} & z_2^{k'+2} & \cdots & z_L^{k'+2} \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{K-1} & z_2^{K-1} & \cdots & z_L^{K-1} \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_L \\ z_1^2 & z_2^2 & \cdots & z_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{K-1} & z_2^{K-1} & \cdots & z_L^{K-1} \end{bmatrix} \begin{bmatrix} z_1^{k'} & 0 & 0 & \cdots & 0 \\ 0 & z_2^{k'} & 0 & \cdots & 0 \\ 0 & 0 & z_3^{k'} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & z_L^{k'} \end{bmatrix} \\
 &= \mathbf{V}_1 \mathbf{\Phi},
 \end{aligned}$$

where  $\mathbf{\Phi} \triangleq \text{diag}(z_\ell^{k'})$  contains all the information on the directions of arrival  $\{\theta_1, \theta_2, \dots, \theta_L\}$ . This result can be generalised to any pair of matched sub-arrays, with appropriate definition of the diagonal matrix  $\mathbf{\Phi}$ . In its most general form the elements of  $\mathbf{\Phi}$  are

$$\begin{aligned}
 z_\ell &= \exp \left( \frac{2\pi i}{\lambda} (d'_x \sin \theta_\ell \sin \phi_\ell + d'_y \cos \theta_\ell \sin \phi_\ell + d'_z \cos \theta_\ell) \right), \\
 &= \exp(i \mathbf{k}_\ell^T \mathbf{d}')
 \end{aligned} \tag{15.41}$$

where  $\mathbf{d}' = [d'_x \ d'_y \ d'_z]^T$  is the displacement vector between the phase centres of the two sub-arrays.

Let  $\mathbf{E}_1$  and  $\mathbf{E}_2$  be the  $(K' \times L)$  matrices whose column vectors are the eigenvectors of  $\mathbf{R}_{\mathbf{x}_1 \mathbf{x}_1}$  and  $\mathbf{R}_{\mathbf{x}_2 \mathbf{x}_2}$  respectively. Because  $\mathbf{E}_1$ ,  $\mathbf{E}_2$  and  $\mathbf{V}$  all span the same signal subspace, there exists an orthonormal  $(L \times L)$  matrix  $\mathbf{T}$  such that

$$\begin{aligned}
 \mathbf{E}_1 &= \mathbf{V}_1 \mathbf{T} \\
 \text{and } \mathbf{E}_2 &= \mathbf{V}_2 \mathbf{T} = \mathbf{V}_1 \mathbf{\Phi} \mathbf{T}.
 \end{aligned} \tag{15.42}$$

### 15.10.2. Least-squares ESPRIT.

Defining  $\Psi_{\text{LS}} \triangleq T^{-1}\Phi T$ , we have

$$\mathbf{E}_2 = \mathbf{E}_1 T^{-1} \Phi T = \mathbf{E}_1 \Psi_{\text{LS}}. \quad (15.43)$$

Note that the eigenvalues of  $\Psi$  are  $\{z_\ell\}$  which would yield  $\{\theta_\ell\}$ . The Least-squares ESPRIT algorithm estimates  $\hat{\mathbf{E}}_1$  and  $\hat{\mathbf{E}}_2$  and then solves

$$\hat{\mathbf{E}}_2 = \hat{\mathbf{E}}_1 \hat{\Psi}_{\text{LS}} \quad (15.44)$$

for  $\hat{\Psi}_{\text{LS}}$ .

(15.44) is overdetermined; the Least-squares (LS) version of ESPRIT minimises  $\varepsilon = \|\hat{\mathbf{E}}_2 - \hat{\mathbf{E}}_1 \hat{\Psi}\|^2$ . Using (A.29) we have

$$\hat{\Psi} = \left[ \hat{\mathbf{E}}_1^H \hat{\mathbf{E}}_1 \right]^{-1} \hat{\mathbf{E}}_1^H \hat{\mathbf{E}}_2, \quad (15.45)$$

from which are calculated its eigenvalues  $\{\hat{z}_\ell\}$  and hence  $\{\hat{\theta}_\ell\}$ .

The LS Esprit algorithm when the sub-arrays are selected from a uniform linear array with spacing  $d$  (as illustrated in Figure 15.9) is outlined below.

- (1) Decide on the displacement  $d'$  between the matched pair of sub-arrays and calculate their selection matrices  $\mathbf{B}_1, \mathbf{B}_2$ .
- (2) From the receiver outputs, estimate the cross-spectral matrix  $\hat{\mathbf{R}}_x$  of the full array.
- (3) Calculate the eigendecomposition  $\hat{\mathbf{R}} = \hat{\mathbf{P}} \hat{\Lambda} \hat{\mathbf{P}}^H$ , where the eigenvalues (and, of course, the corresponding eigenvectors) are arranged in descending order:  $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_K$ .
- (4) Estimate the number of plane-wave arrivals  $\hat{L}$  (using, for example, the AIC (15.34) or MDL (15.35) criterion).
- (5) Eliminate the last  $\hat{L}$  columns of  $\hat{\mathbf{P}}$  (corresponding to the  $\hat{L}$  smallest eigenvalues) to obtain the eigenvectors of the signal subspace  $\hat{\mathbf{E}}$ .
- (6) Calculate  $\hat{\mathbf{E}}_1 = \mathbf{B}_1 \hat{\mathbf{E}}$ , and  $\hat{\mathbf{E}}_2 = \mathbf{B}_2 \hat{\mathbf{E}}$ .
- (7) Calculate  $\hat{\Psi} = (\hat{\mathbf{E}}_1^H \hat{\mathbf{E}}_1)^{-1} \hat{\mathbf{E}}_1^H \hat{\mathbf{E}}_2$ .
- (8) Calculate the eigenvalues  $\{\hat{\lambda}_\ell, \ell = 1, \dots, \hat{L}\}$  of  $\hat{\Psi}$ .
- (9) Calculate the angles of arrival  $\{\hat{\theta}_\ell\}$  from  $\hat{\lambda}_\ell = \exp(2\pi i d' \sin \hat{\theta}_\ell / \lambda)$ .

### 15.10.3. Total least-squares ESPRIT.

Let us consider the following  $(2L \times 2L)$  matrix formed from the eigenvectors  $\mathbf{E}_1$  and  $\mathbf{E}_2$  of the signal space of the sub-arrays:

$$\mathbf{C} = \begin{bmatrix} \mathbf{E}_1^H \\ \mathbf{E}_2^H \end{bmatrix} \begin{bmatrix} \mathbf{E}_1 & \mathbf{E}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \Psi \\ \Psi^H & \mathbf{I} \end{bmatrix} \quad (15.46)$$

It is not difficult to show that  $C$  has multiple eigenvalues  $\lambda_{\max} = 2$  and  $\lambda_{\min} = 0$  and the eigendecomposition

$$\begin{aligned} C &= \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} \Lambda_1 & \mathbf{0} \\ \mathbf{0} & \Lambda_2 \end{bmatrix} \begin{bmatrix} Q_{11}^H & Q_{21}^H \\ Q_{12}^H & Q_{22}^H \end{bmatrix} \\ &= \frac{1}{\sqrt{2}} \begin{bmatrix} I & -\Psi^H \\ \Psi^H & I \end{bmatrix} \begin{bmatrix} 2I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} I & \Psi \\ -\Psi & I \end{bmatrix}, \end{aligned} \quad (15.47)$$

where the eigenvalues are arranged in descending order.

The presence of the term  $\Psi$  indicates that the signal arrival direction could be extracted from the eigenvectors  $\hat{Q}_{12}$  and  $\hat{Q}_{22}$  of  $C$ . When using estimated values, we write

$$\hat{C} = \begin{bmatrix} \hat{Q}_{11} & \hat{Q}_{12} \\ \hat{Q}_{21} & \hat{Q}_{22} \end{bmatrix} \begin{bmatrix} \hat{\Lambda}_1 & \mathbf{0} \\ \mathbf{0} & \hat{\Lambda}_2 \end{bmatrix} \begin{bmatrix} \hat{Q}_{11}^H & \hat{Q}_{21}^H \\ \hat{Q}_{12}^H & \hat{Q}_{22}^H \end{bmatrix}, \quad (15.48)$$

Golub and Van Loan [[15]] reason that, because both  $\hat{E}_1$  and  $\hat{E}_2$  contain errors, a total least-squares approach should be used, in which case the estimate is

$$\hat{\Psi}_{\text{TLS}} = -\hat{Q}_{12} \hat{Q}_{22}^{-1}. \quad (15.49)$$

The first 6 steps of the TLS ESPRIT algorithm follows that of the LS ESPRIT above; the remaining steps are listed below.

(7) Create the augmented matrix

$$\hat{C} = \begin{bmatrix} \hat{E}_1^H \hat{E}_1 & \hat{E}_1^H \hat{E}_2 \\ \hat{E}_2^H \hat{E}_1 & \hat{E}_2^H \hat{E}_2 \end{bmatrix}$$

(8) Calculate the eigendecomposition of  $\hat{C}$  using (15.48).

(9) Calculate  $\hat{\Psi}$  using (15.49).

(10) Calculate the eigenvalues  $\{\hat{\lambda}_\ell, \ell = 1, \dots, \hat{L}\}$  of  $\hat{\Psi}$ .

(11) Calculate the angles of arrival  $\{\hat{\theta}_\ell\}$  from  $\hat{\lambda}_\ell = \exp(2\pi i d' \sin \hat{\theta}_\ell / \lambda)$ .

#### 15.10.4. Ambiguities.

It is important to note that, when the displacement  $d' > 0.5\lambda$ , the ESPRIT algorithm gives ambiguous results, analogous to grating lobes in beamforming. For example, if  $d'/\lambda = 1$ , and  $\hat{\lambda} = 1$ ,  $\hat{\theta} = 0^\circ, \pm 30^\circ$  or  $\pm 90^\circ$ . There is no ambiguity for  $d'/\lambda \leq 0.5$ .

### 15.11. Overview

In this Chapter we have introduced subspace methods, in which the cross-spectral matrix is divided into parts that correspond to signal and noise subspaces. Subspace methods can give a precise estimate of the directions of arrival of signals. The Pisarenko, MUSIC, Eigenvector and ESPRIT methods were described.

With subspace methods, the number of sources has to be estimated. This can be done using either the AIC or MDL techniques.

### Summary

- (1) Subspace methods can be very effective in estimating the direction of arrival of signal sources.
- (2) None of the methods described directly give information on the strengths of the arrivals.
- (3) Their performance depends on whether the noise eigenvalues are distinguishable from the (signal+noise) eigenvalues.
- (4) To estimate the number of arrivals, one can use the AIC or MDL techniques.
- (5) When the signal and noise subspaces are clearly separable (as in the ideal case of independent and identically distributed noises) subspace methods work well.
- (6) When the noise eigenvalues are not clustered around one value, the techniques are prone to serious error. Under those circumstances, the optimal beamformer often gives more reliable results.
- (7) ESPRIT gives ambiguous results when the displacement of the sub-arrays  $d' > 0.5\lambda$ .

## APPENDIX A

# OPTIMISATION

### A.1. Introduction

The following are some brief revision notes on optimisation. We begin with maximisation (or minimisation) of a scalar function  $f(x)$  of a single real variables  $x$ . If the function is differentiable, the condition for an extremum (maximum or minimum) is

$$\frac{df}{dx} = 0.$$

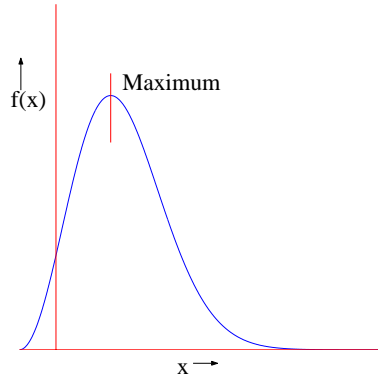


FIGURE A.1. Maximisation of function of one real variable

We can visualise maximisation of a real function of two real variables. If  $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ , then the condition for an extremum is

$$\frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial x_2} = 0.$$

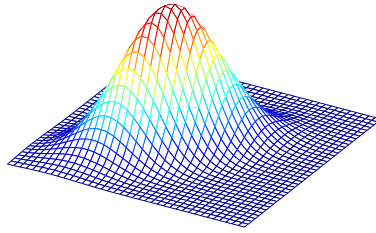


FIGURE A.2. Maximum of function of two real variables

### A.2. Extrema of functions of $K$ real variables

We can readily extend this concept to a real function of  $K$  real variables  $x_1 x_2 \cdots x_K$ .

The condition for an extremum is that each of the  $K$  partial derivatives must be zero:

$$0 = \frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial x_2} = \cdots = \frac{\partial f}{\partial x_K}.$$

Let us define the operator

$$\frac{\partial}{\partial \mathbf{x}} = \begin{bmatrix} \partial/\partial x_1 \\ \partial/\partial x_2 \\ \vdots \\ \partial/\partial x_K \end{bmatrix}. \quad (\text{A.1})$$

Then the condition for an extremum is

$$\frac{\partial f}{\partial \mathbf{x}} = \mathbf{0}_{[K \times 1]}. \quad (\text{A.2})$$

We state here two useful expressions for derivatives; if  $\mathbf{x}$ ,  $\mathbf{y}$  are real vectors,  $\mathbf{R}$  is a  $K \times K$  real symmetric matrix, and  $\mathbf{A}$  is a real  $L \times K$  matrix, then

$$\frac{\partial}{\partial \mathbf{x}} (\mathbf{x}^T \mathbf{R} \mathbf{x}) = 2\mathbf{R} \mathbf{x} \quad (\text{A.3})$$

and

$$\frac{\partial}{\partial \mathbf{x}} (\mathbf{A} \mathbf{x}) = \mathbf{A}^T. \quad (\text{A.4})$$

The proof of (A.3) is straightforward; noting that  $r_{jk} = r_{kj}$ ,

$$\begin{aligned} \frac{\partial}{\partial x_m} (\mathbf{x}^T \mathbf{R} \mathbf{x}) &= \frac{\partial}{\partial x_m} \sum_{j=1}^K \sum_{k=1}^K x_j r_{jk} x_k \\ &= \sum_{j \neq m} x_j r_{jm} + \sum_{k \neq m} r_{mk} x_k + 2r_{mm} x_m \\ &= 2 \sum_{j \neq m} x_j r_{jm} + 2r_{mm} x_m \\ &= 2 \sum_{j=m}^K x_j r_{jm} \end{aligned} \quad (\text{A.5})$$

and hence  $\partial (\mathbf{x}^T \mathbf{R} \mathbf{x}) / \partial \mathbf{x} = 2\mathbf{R} \mathbf{x}$ .

(A.4) is derived in a similar manner.

### A.3. Constrained optimisation

In the previous section the multidimensional independent variable  $\mathbf{x}$  could take any value. However, often there are constraints on the independent variable  $\mathbf{x}$ . Here we consider minimising or maximising a real function  $f(\mathbf{x})$  subject to linear constraints of the form

$$\mathbf{c}_{[L \times K]}^T \mathbf{x}_{[K]} = \alpha_{[L \times 1]} \quad (\text{A.6})$$

where  $\mathbf{c}_{[K \times L]}$  is a  $K \times L$  matrix,  $L < K$ , and  $\alpha_{[L \times 1]}$  is a vector of dimension  $L$ .

This is illustrated for the case in which  $\mathbf{x}$  is a two-dimensional vector and  $\alpha$  is a scalar; in this case, the constraint is a plane. For the unconstrained case, the variable  $\mathbf{x}$  lies on a quadratic “bowl”. With the constraint,  $\mathbf{x}$  lies on the intersection of the plane and the bowl.

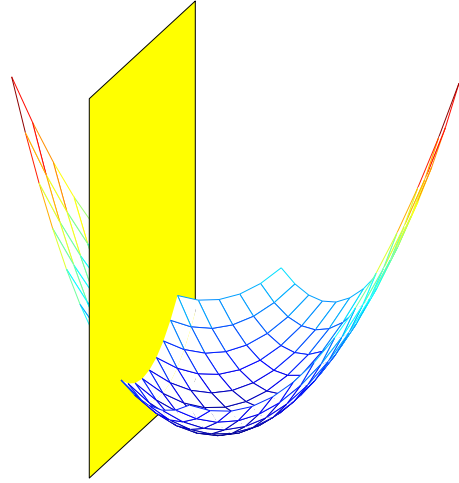


FIGURE A.3. Minimum of function of two variables, with constraint plane

To solve such a problem we employ a mathematical technique called the method of *undetermined multipliers*, which is described below.

We first combine the function to be minimised and the constraint into a single equation (called the *Lagrangian*):

$$\epsilon = f(\mathbf{x}) - \boldsymbol{\mu}^T (\mathbf{c}^T \mathbf{x} - \alpha). \quad (\text{A.7})$$

The last term vanishes when  $\mathbf{c}^T \mathbf{x} = \alpha$ , but for the time being we proceed as if we did not have the constraint.  $\boldsymbol{\mu}$ , an  $L$ -vector, is called a *Lagrange multiplier* which for the present is unknown.

To find the extremum as a function of  $\boldsymbol{\mu}$ , we differentiate  $\epsilon$  with respect to  $\mathbf{x}$  and then solve for  $\boldsymbol{\mu}$  by imposing the constraint. Finally we substitute  $\boldsymbol{\mu}$  in the expression for  $\mathbf{x}$ .

As an example, let us address the following problem:

$$\begin{aligned} &\text{Minimise } \mathbf{x}^T \mathbf{A} \mathbf{x} \\ &\text{subject to } \mathbf{c}^T \mathbf{x} = 1 \end{aligned}$$

where  $\mathbf{A}$  is a real symmetric non-singular matrix. In this example,  $\mathbf{c}$  is a  $K$ -vector and  $\alpha = 1$ .

Following the procedure described above, we first combine the function to be minimised and the constraint into a single equation:



$$\epsilon = \mathbf{x}^T \mathbf{A} \mathbf{x} - \mu(\mathbf{c}^T \mathbf{x} - 1), \quad (\text{A.8})$$

where in this example  $\mu$  is a scalar. Using (A.3) and (A.4) we set  $d\epsilon/d\mathbf{x} = 0$  to obtain:

$$\mathbf{x} = \frac{\mu}{2} \mathbf{A}^{-1} \mathbf{c}. \quad (\text{A.9})$$

Now imposing the constraint:

$$\mathbf{c}^T \mathbf{x} = 1, \quad (\text{A.10})$$

$$\mu = \frac{2}{\mathbf{c}^T \mathbf{A}^{-1} \mathbf{c}} \quad (\text{A.11})$$

$$\mathbf{x}_{\min} = \frac{1}{\mathbf{c}^T \mathbf{A}^{-1} \mathbf{c}} \mathbf{A}^{-1} \mathbf{c}, \quad (\text{A.12})$$

and the desired minimum is

$$\epsilon_{\min} = \mathbf{x}_{\min}^T \mathbf{A} \mathbf{x}_{\min} \quad (\text{A.13})$$

$$= \frac{1}{\mathbf{c}^T \mathbf{A}^{-1} \mathbf{c}} \quad (\text{A.14})$$

## A.4. Real functions of complex variables

### A.4.1. Rules of differentiation.

So far we have been dealing with real variables but in array signal processing we frequently encounter complex variables. In general, we may not take derivatives with respect to complex variables<sup>1</sup>. For example, the function  $\zeta(z) = z^*$  is not analytic and hence is not differentiable. In all the cases we consider here it is possible to express them as functions of real variables, but the expressions are rather clumsy).

In the following, we introduce a convenient approach for handling real functions  $\zeta(\mathbf{w}, \mathbf{w}^*)$  of complex variables, such as:

$$\begin{aligned} \zeta(\mathbf{w}, \mathbf{w}^*) &= \mathbf{w}^H \mathbf{R} \mathbf{w}, \text{ or} \\ \zeta(\mathbf{w}, \mathbf{w}^*) &= \frac{|\mathbf{w}^H \mathbf{v}|^2}{\mathbf{w}^H \mathbf{R} \mathbf{w}}, \end{aligned}$$

where  $\mathbf{R}$  is Hermitian<sup>2</sup>, and we wish to maximise (or minimise)  $\zeta$  with respect to  $\mathbf{w}$ .

Let us express  $\mathbf{w}$  in terms of its real and imaginary components:  $\mathbf{w} = \mathbf{x} + i\mathbf{y}$ , and define the operator  $\nabla_{\mathbf{w}}$  as follows<sup>3</sup>:

<sup>1</sup>The exception is for *analytic* functions, i.e., those that satisfy the Cauchy-Riemann equations. Unfortunately, the functions in array processing generally *are not analytic*.

<sup>2</sup>A (square) Hermitian matrix  $\mathbf{R}$  has the property  $\mathbf{R}^H \equiv \mathbf{R}^{*T} = \mathbf{R}$

<sup>3</sup>We need to stress that  $\nabla_{\mathbf{w}}$  is *not* a gradient; it is only a useful shorthand. Nevertheless, under certain quite specific conditions it bears a similarity to a gradient, as will be made apparent later.

$$\nabla_{\mathbf{w}} \triangleq \frac{1}{2} \left( \frac{\partial}{\partial \mathbf{x}} - i \frac{\partial}{\partial \mathbf{y}} \right), \quad (\text{A.15})$$

$$\nabla_{\mathbf{w}^*} = \frac{1}{2} \left( \frac{\partial}{\partial \mathbf{x}} + i \frac{\partial}{\partial \mathbf{y}} \right) = (\nabla_{\mathbf{w}})^*, \quad (\text{A.16})$$

where

$$\frac{\partial}{\partial \mathbf{x}} = \begin{bmatrix} \partial/\partial x_1 \\ \partial/\partial x_2 \\ \vdots \\ \partial/\partial x_K \end{bmatrix}$$

The condition for an extremum (maximum or minimum) is

$$\frac{\partial \zeta}{\partial \mathbf{x}} = 0 = \frac{\partial \zeta}{\partial \mathbf{y}} \quad (\text{A.17})$$

or equivalently,

$$\nabla_{\mathbf{w}}(\zeta) = 0 = \nabla_{\mathbf{w}^*}(\zeta). \quad (\text{A.18})$$

We state the following useful principle[5].

*If the real-valued function  $\zeta(\mathbf{w}, \mathbf{w}^H)$  is differentiable by  $\mathbf{w}$  and  $\mathbf{w}^H$  independently, then the normal rules of differentiation of real-valued functions apply to  $\nabla_{\mathbf{w}}(\zeta)$  and  $\nabla_{\mathbf{w}^*}(\zeta)$ .*

For example, we could obtain the following results either by applying the above principle, or by using (A.15) and (A.16):

$$\nabla_{\mathbf{w}^*} |\mathbf{w}^H \mathbf{v}|^2 = (\mathbf{v}^H \mathbf{w}) \mathbf{v}, \quad (\text{A.19})$$

$$\nabla_{\mathbf{w}} |\mathbf{w}^H \mathbf{v}|^2 = (\mathbf{w}^H \mathbf{v}) \mathbf{v}^*, \quad (\text{A.20})$$

$$\nabla_{\mathbf{w}^*} (\mathbf{w}^H \mathbf{v}) = \mathbf{v}, \quad (\text{A.21})$$

$$\nabla_{\mathbf{w}} (\mathbf{v}^H \mathbf{w}) = \mathbf{v}^* \quad (\text{A.22})$$

$$\nabla_{\mathbf{w}^*} (\mathbf{v}^H \mathbf{w}) = \mathbf{0} = \nabla_{\mathbf{w}} (\mathbf{w}^H \mathbf{v}), \quad (\text{A.23})$$

$$\nabla_{\mathbf{w}} (\mathbf{w}^H \mathbf{R} \mathbf{w}) = \mathbf{R} \mathbf{w}, \quad (\text{A.24})$$

#### A.4.2. Least-squares.

Let  $\mathbf{A}$  and  $\mathbf{B}$  be  $(K \times L)$  complex matrices and  $\mathbf{X}$  a hermitian symmetric  $(L \times L)$  matrix. We wish to select  $\mathbf{X}$  so as to minimise the Euclidean norm

$$\begin{aligned} \varepsilon &= \|\mathbf{A} - \mathbf{B} \mathbf{X}\|_2 \\ &= \text{Tr}((\mathbf{A} - \mathbf{B} \mathbf{X})^H (\mathbf{A} - \mathbf{B} \mathbf{X})) \\ &= \sum_j |a_{ji}|^2 - \sum_{j,k} a_{ji}^* b_{jk} x_{ki} - \sum_{j,k} x_{ji}^* b_{kj}^* a_{ki} + \sum_{j,k,\ell} x_{ji}^* b_{kj}^* b_{k\ell} x_{\ell i}. \end{aligned} \quad (\text{A.25})$$

Setting  $\nabla_{x_{rs}^*} \varepsilon = 0$  : we have

$$-\sum_k b_{kr}^* a_{ks} + \sum_{k,\ell} b_{kr}^* b_{k\ell} x_{\ell s} = 0$$

$$\mathbf{B}^H \mathbf{B} \mathbf{X} = \mathbf{B}^H \mathbf{A}$$

so, if the inverse exists,

$$\mathbf{X} = (\mathbf{B}^H \mathbf{B})^{-1} \mathbf{B}^H \mathbf{A}. \quad (\text{A.26})$$

#### A.4.3. Maximising gain.

Applying the same principle, we readily obtain

$$\nabla_{\mathbf{w}} \frac{|\mathbf{w}^H \mathbf{v}|^2}{(\mathbf{w}^H \mathbf{R} \mathbf{w})} = \frac{(\mathbf{w}^H \mathbf{R} \mathbf{w})(\mathbf{v}^H \mathbf{w}) \mathbf{v} - |\mathbf{w}^H \mathbf{v}|^2 \mathbf{R} \mathbf{w}}{(\mathbf{w}^H \mathbf{R} \mathbf{w})^2}. \quad (\text{A.27})$$

Hence to maximise  $G = \frac{|\mathbf{w}^H \mathbf{v}|^2}{(\mathbf{w}^H \mathbf{R} \mathbf{w})}$ , for example, we set

$$\nabla_{\mathbf{w}} G = \mathbf{0} = \nabla_{\mathbf{w}^*} G \text{ and, using (A.27),} \quad (\text{A.28})$$

$$\nabla_{\mathbf{w}} G = \frac{(\mathbf{w}^H \mathbf{R} \mathbf{w})(\mathbf{v}^H \mathbf{w}) \mathbf{v} - |\mathbf{w}^H \mathbf{v}|^2 \mathbf{R} \mathbf{w}}{(\mathbf{w}^H \mathbf{R} \mathbf{w})^2} \quad (\text{A.29})$$

$$\nabla_{\mathbf{w}^*} G = \frac{(\mathbf{w}^H \mathbf{R} \mathbf{w})(\mathbf{w}^H \mathbf{v}) \mathbf{v}^* - |\mathbf{w}^H \mathbf{v}|^2 \mathbf{R}^* \mathbf{w}^*}{(\mathbf{w}^H \mathbf{R} \mathbf{w})^2}, \quad (\text{A.30})$$

whence

$$\mathbf{R} \mathbf{w} = \left( \frac{\mathbf{w}^H \mathbf{v}}{\mathbf{w}^H \mathbf{R} \mathbf{w}} \right) \mathbf{v} \quad (\text{A.31})$$

and

$$\mathbf{R}^* \mathbf{w}^* = \left( \frac{\mathbf{v}^H \mathbf{w}}{\mathbf{w}^H \mathbf{R} \mathbf{w}} \right) \mathbf{v}^* \quad (\text{A.32})$$

*Note that, for this symmetrical function  $G$ , the second equation is simply the conjugate of the first.*

So, if the inverse exists, the maximum is attained when

$$\mathbf{w}_{\max} = \alpha \mathbf{R}^{-1} \mathbf{v} \quad (\text{A.33})$$

where  $\alpha$  is some scaling factor.

These results are used in Chapter 10.

#### A.4.4. Minimising quadratic form subject to linear constraint.

In Chapter 10 we consider minimising a quadratic form  $\mathbf{w}^H \mathbf{R} \mathbf{w}$  with respect to  $\mathbf{w}$  subject to the constraint  $\mathbf{w}^H \mathbf{v} = 1 = \mathbf{v}^H \mathbf{w}$ .

To solve it we use Lagrange multipliers  $\mu$  and  $\nu^*$ <sup>4</sup> and combine both the function to be minimised and the constraints into a single equation:

$$\zeta = \mathbf{w}^H \mathbf{R} \mathbf{w} - \mu [\mathbf{w}^H \mathbf{v} - 1] - \nu^* [\mathbf{v}^H \mathbf{w} - 1]. \quad (\text{A.34})$$

The condition for a minimum is

$$\nabla_{\mathbf{w}^*} \zeta = 0 = \nabla_{\mathbf{w}} \zeta. \quad (\text{A.35})$$

Using (A.21), (A.22) and (A.24)–(A.29), it can readily be shown that the two conditions of (A.35) become, if the inverse exists,

$$\mathbf{R} \mathbf{w} - \mu \mathbf{v} = 0 \quad \Rightarrow \quad \mathbf{w} = \mu \mathbf{R}^{-1} \mathbf{v} \quad (\text{A.36})$$

$$\text{and } \mathbf{R}^* \mathbf{w}^* - \nu^* \mathbf{v}^* = 0 \quad \Rightarrow \quad \mathbf{w} = \nu \mathbf{R}^{-1} \mathbf{v}. \quad (\text{A.37})$$

Applying the constraint  $\mathbf{w}^H \mathbf{v} = 1 = \mathbf{v}^H \mathbf{w}$  we have

$$\mu = \frac{1}{\mathbf{v}^H \mathbf{R}^{-1} \mathbf{v}} = \nu, \quad (\text{A.38})$$

and hence

$$\mathbf{w}_{\text{solution}} = \frac{\mathbf{R}^{-1} \mathbf{v}}{\mathbf{v}^H \mathbf{R}^{-1} \mathbf{v}} \quad (\text{A.39})$$

### A.5. Gradient descent

#### A.5.1. Real variables.

In Section A we found the stationary point(s) of a function directly by setting the derivative to zero. Here we develop an iterative technique for arriving at the solution.

---

<sup>4</sup>We use, without loss of generality,  $\nu^*$  instead of  $\nu$  for convenience.

Consider the simple quadratic function

$$y \equiv y(x) = y_0 + \gamma(x - x_0)^2, \quad \gamma \geq 0 \quad (\text{A.40})$$

We can approach the minimum  $(x_0, y_0)$  iteratively by moving in the direction of the gradient<sup>a</sup>, using the *gradient descent* method, illustrated in Figure A.5.1.

Given some approximation,  $x(m)$ , at the  $m^{\text{th}}$  iteration, an improvement is made by taking a step in the direction of the minimum:

$$\begin{aligned} x(m+1) &= x(m) - \mu \left( \frac{dy}{dx} \right) \Big|_{x=x(m)} \\ &= x(m) - 2\mu\gamma(x(m) - x_0), \end{aligned} \quad (\text{A.41})$$

where  $\mu > 0$ .

<sup>a</sup>A gravitational analogy of the gradient is the direction in which an inertia-free ball would roll.

The rationale for using the gradient is that, the further away we are from the minimum, the greater the gradient and so also the larger the step size.

To investigate the convergence of this process, we perform a shift in axes of (A.41),

$$\begin{aligned} x(m+1) - x_{opt} &= x(m) - x_{opt} - 2\mu\gamma(x(m) - x_{opt}), \\ &= (1 - 2\mu\gamma)(x(m) - x_{opt}) = \dots \\ &= (1 - 2\mu\gamma)^{m+1}(x(0) - x_{opt}), \end{aligned} \quad (\text{A.42})$$

where  $x(0)$  is the initial approximation. Thus

$$\lim_{m \rightarrow \infty} x(m) = x_{opt} \quad \text{if and only if} \quad |1 - 2\mu\gamma| < 1 \quad \Longleftrightarrow \quad 0 < \mu < 1/\gamma. \quad (\text{A.43})$$

The technique can readily be extended to multivariate optimisation.

### A.5.2. Gradient of a real function of a complex variable and its conjugate.

In many array beamforming algorithms we need to use the gradient of the real scalar-valued function of a complex vector variable and of its complex conjugate,  $\zeta(\mathbf{w}, \mathbf{w}^*)$ . It was pointed out earlier that we cannot in general take the derivative with respect to complex variables. However, by treating  $\mathbf{w}$  and  $\mathbf{w}^*$  as independent variables, we can determine the direction of the maximum rate of change of  $\zeta(\mathbf{w}, \mathbf{w}^*)$ , as follows.

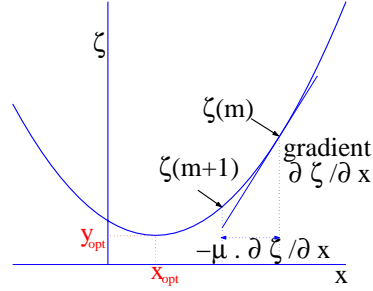


FIGURE A.4. Illustrating gradient descent – 1 dimension

Consider the variation<sup>5</sup>:

$$\begin{aligned}
 \delta\zeta &\triangleq 2 \sum_k \left( \frac{\partial\zeta}{\partial x_k} \delta x_k + \frac{\partial\zeta}{\partial y_k} \delta y_k \right) \\
 &= \sum_k \left( \frac{\partial\zeta}{\partial x_k} - i \frac{\partial\zeta}{\partial y_k} \right) (\delta x_k + i \delta y_k) + \left( \frac{\partial\zeta}{\partial x_k} + i \frac{\partial\zeta}{\partial y_k} \right) (\delta x_k - i \delta y_k) \\
 &= (\nabla_{\mathbf{w}} \zeta)^T \delta \mathbf{w} + (\nabla_{\mathbf{w}^*} \zeta)^T \delta \mathbf{w}^* \\
 &= (\nabla_{\mathbf{w}^*} \zeta)^H \delta \mathbf{w} + (\nabla_{\mathbf{w}} \zeta)^H \delta \mathbf{w}^* \\
 &= \text{Re} \left\{ (\nabla_{\mathbf{w}^*} \zeta)^H \delta \mathbf{w} \right\}
 \end{aligned} \tag{A.44}$$

From the Cauchy-Schwarz inequality<sup>6</sup>, this variation is a maximum when and only when  $\delta \mathbf{w}$  is in the same direction as  $\nabla_{\mathbf{w}^*} \zeta$ . Thus we have the following result.

*The greatest change in the function  $\zeta(\mathbf{w}, \mathbf{w}^*)$  is in the direction of its gradient with respect to  $\mathbf{w}^*$ .*

If  $\mathbf{w}$  is a complex variable, the gradient descent algorithm to minimise a real function  $\zeta(\mathbf{w}, \mathbf{w}^*)$  is then:

$$\mathbf{w}(m+1) = \mathbf{w}(m) - \mu \nabla_{\mathbf{w}^*} \zeta(\mathbf{w}, \mathbf{w}^*)|_{\mathbf{w}=\mathbf{w}(m)} \tag{A.45}$$

This result is used in Chapter 11.

---

<sup>5</sup>The factor of 2 is introduced for convenience.

<sup>6</sup>See [Appendix C](#).

## A.6. Maximum likelihood estimator

### A.6.1. Principle.

The maximum likelihood estimator is based on the following notion. Consider a probability density function  $p(x|a)$  of a variable  $x$ , which depends on an unknown parameter  $\alpha$ . In Figure A.6 we have sketched  $p(x|\alpha)$  for several values of  $\alpha$ . Now suppose that we observe a value  $x$  from which we wish to estimate  $\alpha$ .

It is natural to pick that value of  $\alpha$  for which  $p(x|\alpha)$  is greatest. This is called the *maximum likelihood estimate*  $\hat{\alpha}$ .

In other words, we select:

$$\hat{\alpha} = \arg(\max_{\alpha} p(x|\alpha)) \quad (\text{A.46})$$

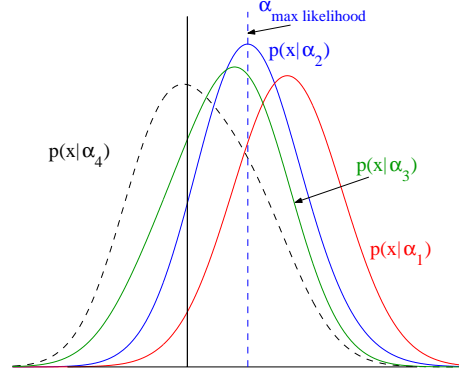


FIGURE A.5. Illustrating maximum likelihood

The function  $l(\alpha, x) = p(x|\alpha)$  is called the *likelihood function*, and its logarithm,

$$L(\alpha, x) = \ln(p(x|\alpha)) \quad (\text{A.47})$$

is called the *log-likelihood function*. When  $L$  is differentiable,  $\hat{\alpha}$  is a root of the equation

$$\frac{\partial}{\partial \alpha} L(\alpha, x) = 0. \quad (\text{A.48})$$

Take the simple case in which we have a sample of gaussian noise  $x$  of known variance  $\sigma^2$  and unknown mean  $\mu$ . The distribution of  $x$  will be gaussian with mean  $\mu$  and variance  $\sigma^2$ :

$$p(x|\mu) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad (\text{A.49})$$

so

$$L(\mu, x) = -\frac{(x-\mu)^2}{2\sigma^2} - \text{constant} \quad (\text{A.50})$$

and the solution is simply

$$\hat{\mu} = x. \quad (\text{A.51})$$

The maximum likelihood estimator is particularly useful when we have a deterministic unknown parameter which we wish to estimate.

The concept can readily be generalised to the multivariate case in which the unknown scalar parameter and the observation vectors are complex. These results are applied in Chapter 9.

## APPENDIX B

# RANDOM SIGNALS AND NOISE

### B.1. Introduction

We present here some brief revision notes on random signals and noise.

In Chapters 1 – 6 we considered beamforming for deterministic signals. However in practice there are present internal and external noise sources with random properties. For example, in sonar, random noise generated by the wind on the surface of the sea is a common source of external noise, whilst in radar and communications systems it is often internal receiver thermal noise that limits system performance. Furthermore the desired signals may themselves exhibit properties similar to random noise. For example, the pseudorandom signals used in spread spectrum communications and GPS signal can conveniently be represented as a random process.

Here we consider representing the receiver outputs as random processes; to do so we first need to consider some preliminaries from the theory of random processes.

### B.2. Random Variable and Probability Density Functions

A **random variable**,  $X$ , is a function that maps events, i.e., the outcomes of experiments, into numbers. It is measurable in the sense that we can associate a **probability density function (pdf)**, denoted by  $p_X(x)$ , with it .

We shall often use the shorthand notation  $p(x)$ . For example, the Gaussian random variable  $X$  has a domain  $-\infty$  to  $\infty$  and a probability density function given by

$$p_X(x) \equiv p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left\{ -\frac{(x - \mu)^2}{2\sigma^2} \right\}, \quad (\text{B.1})$$

where  $\mu$  is the **mean** and  $\sigma$  is the **standard deviation**. We shall denote this special type of random variable by  $X \sim N(\mu, \sigma^2)$ , where  $\sim$  means ‘distributed as’.

The probability density function often supplies more information on the probability properties of a random variable than we need. In particular, we often only need to know certain moments of the random variable, where we define the  $j^{\text{th}}$  moment,  $m_j$  by

$$m_j = \int_{-\infty}^{\infty} x^j p_X(x) dx.$$



Some important moments of random variables are defined below:

(1) **mean**,  $\mu$ :

$$m_1 \equiv \mu \triangleq \int x p_X(x) dx. \quad (\text{B.2})$$

(2) **variance**,  $\sigma^2$ :

$$m_2 \equiv \sigma^2 \triangleq \int x^2 p_X(x) dx. \quad (\text{B.3})$$

(3) **entropy**<sup>1</sup>,  $H$ :

$$H = \int p_X(x) \log\{p_X(x)\} dx. \quad (\text{B.4})$$

### B.3. Multiple Random Variables

$K$  multiple real random variables,  $\{X_1, X_2, \dots, X_K\}$ , can be represented as a real *random vector*,  $\mathbf{X}$

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_K \end{bmatrix}$$

defined over some sample space and characterised by a joint probability density function, i.e.,

$$P_{(X_1, X_2, \dots, X_K)}(X_1 = x_1, X_2 = x_2, \dots, X_K = x_K) = p_{\mathbf{X}}(\mathbf{X}). \quad (\text{B.5})$$

Two important quantities that often suffice for our statistical description are

(1) the **mean vector**,  $\boldsymbol{\mu}$ , defined by<sup>2</sup>

$$\boldsymbol{\mu} = \int dx_1 \int dx_2 \cdots \int dx_K \mathbf{X} p_{\mathbf{X}}(\mathbf{X}). \quad (\text{B.6})$$

(2) For zero-mean random variables, the **covariance** between the random variables  $X_i$  and  $X_j$  is defined by

$$R_{X_i, X_j} = \int x_i x_j p_{X_i X_j}(x_i, x_j) dx_i dx_j. \quad (\text{B.7})$$

We now consider the important concept of statistical independence.

DEFINITION. Two random variables  $X_i$  and  $X_j$  are **statistically independent** if

$$p_{X_i X_j}(x_i, x_j) = p_{X_i}(x_i) p_{X_j}(x_j).$$

Loosely, this states that knowledge of the distribution of  $X_j$  does not allow us to infer anything about the distribution of  $X_i$ .<sup>3</sup>

<sup>1</sup>Entropy has a deep and fundamental link with both information theory and linear prediction.

<sup>2</sup>Exercise: Show that  $(\boldsymbol{\mu})_j = \int dx_j p_{X_j}(x_j)$ .

<sup>3</sup>Exercise: Show that if  $X_i$  and  $X_j$  are statistically independent and zero mean, then  $R_{X_i, X_j} = 0$ .

**Example 2.0.** Consider the bivariate Gaussian density when  $X_1$  and  $X_2$  are statistically independent and with zero mean. In this case,

$$p(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2} \exp\left(-\frac{x_1^2}{2\sigma_1^2} - \frac{x_2^2}{2\sigma_2^2}\right). \quad (\text{B.8})$$

**Example 2.1.** If  $X_1$  and  $X_2$  are zero mean but are not statistically independent, then

$$p(x_1, x_2) = \frac{1}{2\pi(\sigma_1^2\sigma_2^2 - \sigma_{12}^2)^{1/2}} \exp\left(-\frac{\sigma_1^2x_1^2 - 2\sigma_{12}x_1x_2 + \sigma_2^2x_2^2}{2(\sigma_1^2\sigma_2^2 - \sigma_{12}^2)}\right), \quad (\text{B.9})$$

where  $\sigma_{12} = \iint x_1x_2p(x_1, x_2) dx_1 dx_2$ .

Note that a complex random variable can be treated as a two-dimensional random vector. The probability density  $p(z)$  of complex random variable  $z = u + iv$  is defined by  $p(u, v)$  which denotes the probability of the complex part of the random variable  $z$  lying close to  $u$  and the imaginary part lying close to  $v$ .

#### B.4. Ensemble Averaging

Given a random variable and functions of that random variable, we want to define quantities that measure properties that are independent of the outcome of a particular experiment but reflect the outcome of all possible experiments. The collection of the outcomes of an infinite number of experiments we term the **ensemble** and when we average a quantity over all possible outcomes we form the **ensemble average** or **expectation** of a random variable. If  $f(X)$  is some function of the random variable  $X$  then we formally define the expectation to be

$$E_X\{f(X)\} = \int f(x)p(x) dx. \quad (\text{B.10})$$

In practice, when there is no ambiguity as to the values of the random variable over which we are integrating we usually omit the subscript  $X$ . Note the strong intuitive concept of linear averaging in the above definition. Some examples are

- (1) The **mean**,  $\mu$ , of a random variable is defined by  $\mu = E\{X\}$ .
- (2) The **variance**,  $\sigma^2$ , of a random variable is defined by  $\sigma^2 = E\{(X - \mu)^2\}$ .
- (3) The **entropy**,  $H$ , of a random variable is defined by  $H = E\{\log(p(X))\}$ .

Some important properties of ensemble averages are listed below.

- (1) If  $X$  is a random and  $c$  is not, then  $E\{cX\} = cE\{X\}$ .
- (2) If  $X$  and  $Y$  are random and  $c$  and  $d$  are not, then<sup>4</sup>  $E\{cf(X) + dg(Y)\} = cE\{f(X)\} + dE\{g(Y)\}$ .

In particular,  $E\{X + Y\} = E\{X\} + E\{Y\}$ .

---

<sup>4</sup>This result holds whether or not  $X$  and  $Y$  are independent.

(3) If  $X$  and  $Y$  are statistically independent, then

$$E_{X,Y}\{f(X)g(Y)\} = E_X\{f(X)\}E_Y\{g(Y)\}.$$

$$\text{As an example } E\{XY^2\} = E\{X\}E\{Y^2\}.$$

N.B. For zero-mean variables, we say  $X$  and  $Y$  are uncorrelated if  $E\{XY\} = 0$ . Thus statistical independence  $\implies$  uncorrelated but not vice-versa.

The above properties hold for random variables  $X$  and  $Y$  and can readily be generalised for random vectors, i.e.,

$$E_{(X_1, X_2, \dots, X_K)}\{f(X_1, X_2, \dots, X_K)\} = \int dx_1 \int dx_2 \cdots \int dx_K f(x_1, x_2, \dots, x_K) \times p_{(X_1, X_2, \dots, X_K)}(x_1, x_2, \dots, x_K). \quad (\text{B.11})$$

### B.5. Covariance Matrix

We now consider the  $K \times K$  matrix formed by taking all possible pairs of covariances between the  $i^{\text{th}}$  and  $j^{\text{th}}$  elements of the random vector. We shall denote this matrix by  $\mathbf{R}_X$ , and from its definition (B.7) and the definition of the ensemble average (B.11) the  $(i, j)^{\text{th}}$  element is given by

$$(\mathbf{R}_X)_{i,j} = \mathbf{R}_{X_i, X_j} = E(X_i, X_j). \quad (\text{B.12})$$

In matrix form, this becomes<sup>5</sup>

$$\mathbf{R}_X = \begin{bmatrix} E\{X_1^2\} & E\{X_1X_2\} & \cdots & E\{X_1X_K\} \\ E\{X_2X_1\} & E\{X_2^2\} & \cdots & E\{X_2X_K\} \\ \vdots & \vdots & \ddots & \vdots \\ E\{X_KX_1\} & \cdots & \cdots & E\{X_K^2\} \end{bmatrix} = E\{\mathbf{X} \mathbf{X}^T\} \quad (\text{B.13})$$

and is termed the **covariance matrix**. The covariance matrix satisfies two important properties.

(a)  $\mathbf{R}_X$  is non-negative definite.<sup>6</sup> It follows from (B.13) that

$$\mathbf{w}^T \mathbf{R}_X \mathbf{w} = E\{\mathbf{w}^T \mathbf{X} \mathbf{X}^T \mathbf{w}\} = E\{|\mathbf{w}^T \mathbf{X}|^2\} \geq 0. \quad (\text{B.14})$$

(b)  $\mathbf{R}_X$  is a symmetric matrix (i.e.,  $\mathbf{R}_X^T = \mathbf{R}_X$ ).

As an example we consider  $K = 2$  and the zero-mean bivariate Gaussian distribution. In this case,<sup>7</sup>

$$\mathbf{R}_X = \mathbf{R}_2 = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{bmatrix}. \quad (\text{B.15})$$

<sup>5</sup>Exercise: Verify that  $\mathbf{R}_X = E\{\mathbf{X} \mathbf{X}^T\}$ .

<sup>6</sup>Recall that a matrix  $\mathbf{R}$  is non-negative definite if  $\mathbf{w}^T \mathbf{R} \mathbf{w} \geq 0 \quad \forall \mathbf{w}$ .

<sup>7</sup>Exercise: Show that  $p(\mathbf{X}) = p(x_1, x_2) = \frac{1}{2\pi|\mathbf{R}_2|^{1/2}} \exp[-(\mathbf{X}^T \mathbf{R}_2^{-1} \mathbf{X}/2)]$ .

Finally we give expressions for the joint *pdf* of the vector,  $\mathbf{X}$ , of  $K$  Gaussian random variables. For real variables we have, with some liberty of notation,

$$p_{\mathbf{X}}(\mathbf{x}) = \frac{1}{(2\pi)^{K/2} |\mathbf{R}_{\mathbf{x}}|^{1/2}} \exp \left( -\frac{\mathbf{x}^T \mathbf{R}_{\mathbf{x}}^{-1} \mathbf{x}}{2} \right). \quad (\text{B.16})$$

For complex  $\mathbf{X}$ ,

$$p_{\mathbf{X}}(\mathbf{x}) = \frac{1}{(\pi)^K |\mathbf{R}_{\mathbf{x}}|^{1/2}} \exp \left( -\mathbf{x}^T \mathbf{R}_{\mathbf{x}}^{-1} \mathbf{x} \right). \quad (\text{B.17})$$

We state two useful results for the expectation of the product of four Gaussian random variables. For real variables,

$$\begin{aligned} E\{X_1 X_2 X_3 X_4\} = & E\{X_1 X_2\} E\{X_3 X_4\} + E\{X_1 X_3\} E\{X_2 X_4\} + \\ & E\{X_1 X_4\} E\{X_2 X_3\}; \end{aligned} \quad (\text{B.18})$$

and for complex variables

$$E\{X_1 X_2^* X_3 X_4^*\} = E\{X_1 X_2^*\} E\{X_3 X_4^*\} + E\{X_1 X_4^*\} E\{X_2^* X_3\}. \quad (\text{B.19})$$

## B.6. Complex random vectors

In the above we have considered only real random variables and random vectors. However, in many array processing problems we have complex variables – for example, in frequency-domain beamforming when we consider the complex Fourier coefficients of the receiver outputs, which can model as complex random variables.

There are two ways to handle complex signals. The first is to consider, for  $K$  receivers, the  $2K$  vector consisting of the real and imaginary components; i.e.,  $\mathbf{X}$  becomes

$$\mathbf{X} = [\Re(X_1) \quad \Re(X_2) \quad \cdots \quad \Re(X_K) \quad \Im(X_1) \quad \Im(X_2) \quad \cdots \quad \Im(X_K)]^T. \quad (\text{B.20})$$

To describe the statistical properties of  $\mathbf{X}$  we need a joint probability density function with  $2K$  real variables. However, with this approach the algebra tends to be cumbersome.

A more elegant mathematical formulation is to let  $\mathbf{X}$  be a  $K$ -vector with complex components and to generalise the definitions in an appropriate fashion. In particular, the covariance between  $X_i$  and  $X_j$  is defined as

$$R_{X_i, X_j} = E\{X_i X_j^*\}$$

and the covariance matrix definition is generalised to

$$\mathbf{R}_{\mathbf{X}} = E\{\mathbf{X} \mathbf{X}^H\},$$

where  $\cdot^H \equiv \cdot^{*T}$  denotes the Hermitian (complex conjugate) transpose.

Of course the two approaches are equivalent but we should always be on guard that the complex notation does not lead us astray – occasionally we may need to go back to the  $2K$ -dimensional real representation to clarify physical concepts.

For the complex Gaussian case we have

$$p(\mathbf{X}) = \frac{1}{(\pi)^K |\mathbf{R}_{\mathbf{X}}|} \exp(-\mathbf{X}^H \mathbf{R}_{\mathbf{X}}^{-1} \mathbf{X}), \quad (\text{B.21})$$

where  $\mathbf{R}_{\mathbf{X}}$  is now the  $K \times K$  complex covariance matrix.

### B.7. Stationary Random Processes

So far in considering random variables and random vectors we have said nothing about time. Now consider a time-varying random process from a single receiver as illustrated below for four realisations.

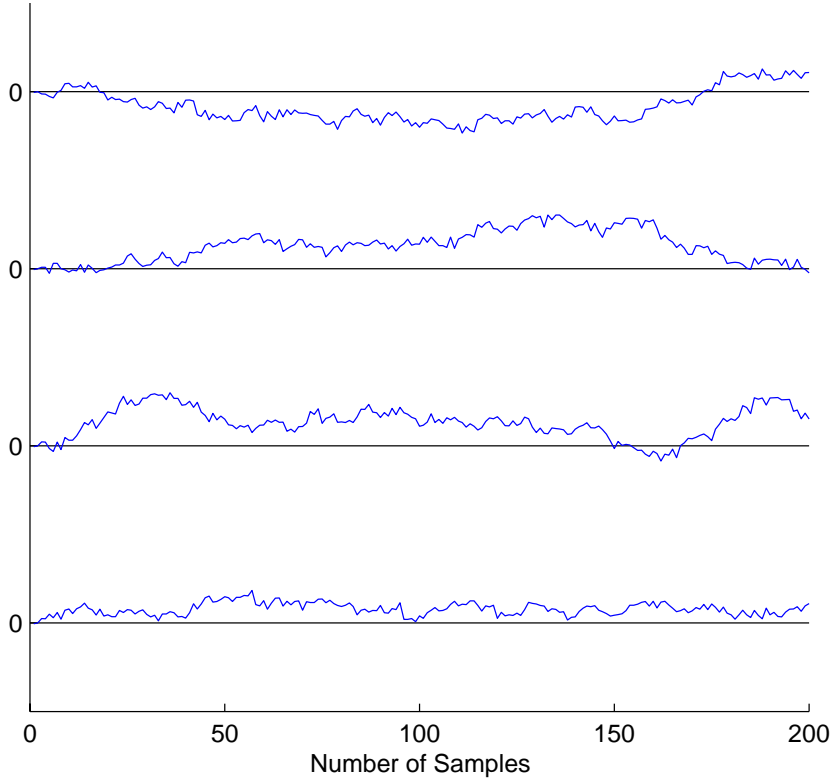


FIGURE B.1. Realisations of a random process

Each of the above signals we term a *sample path* or *realisation* of our random process. Each realisation may be thought of as the outcome of an experiment and the set of an infinite number of experiments is termed the ensemble of the random process.

Considering the random process at a finite set of times  $t_1, t_2, \dots$  allows a rapprochement with our earlier ideas of a random variable and the formal introduction of probabilistic concepts. At time  $t_1$  consider an equivalent random variable  $x_1 = x(t_1)$  which takes all the values that the random process could take and has

the same probability density function say  $p(x_1)$ . Similarly we do so at time  $t_2$ , etc. Now in order to capture the temporal properties of the random time series,  $x(t)$ , we need to say something about how the probabilities of particular events (i.e., the sample values  $x_1, x_2, \dots$  etc. of the random process at times  $t_1, t_2, \dots$  etc.) at different time are related. Again because we are considering random processes we can only say things about the temporal relationships in a probabilistic sense and the necessary tool for doing this is the joint probability density function of the samples  $x_1, x_2, \dots$  etc. Thus we need to specify  $p(x_1)$ ,  $p(x_1, x_2)$ ,  $p(x_1, x_2, x_3)$ , etc. at all times. Note that these can all be obtained from  $p(x_1, x_2, \dots)$  as marginals. Stationarity implies things not changing with time. For most of the random processes that we will consider the sample values won't be fixed in time for a particular realisation so we need to come up with an appropriate definition of stationarity. Again we use the joint probability density function of our sampled random process and we define a random process to be stationary if all joint probability density function's are independent of time. Formally this is defined as

$$p(x(t_1) = u_1, x(t_2) = u_2, \dots) = p(x(t_1 + t_0) = u_1, x(t_2 + t_0) = u_2, \dots). \quad (\text{B.22})$$

Note this has to hold for all  $N$  and for all  $t_0$ .

In a similar way to that considered for random variables the ensemble average of some function  $f(x(t_1), x(t_2), \dots)$  of the samples of a random process is formally defined by

$$\begin{aligned} E\{f(x(t_1), x(t_2), \dots, x(t_N))\} \\ = \iint \cdots \int f(x(t_1), x(t_2), \dots, x(t_N)) \\ \times p(x(t_1) = u_1, x(t_2) = u_2, \dots, x(t_N) = u_N) du_1 du_2 \cdots du_N. \end{aligned} \quad (\text{B.23})$$

Since the joint probability density function of the samples of a stationary random process are independent of time it follows that ensemble averages of functions of a stationary random process will be independent of time. For example the mean of a stationary random process is independent of time and the covariance function depends only on time separation.

## APPENDIX C

# MATRIX REVISION

### C.1. Introduction

We shall use bold-face characters to denote *matrices* and *column vectors*: lower-case for vectors and upper-case for matrices. The size of the matrix may be indicated specifically by subscripts; for example,  $\mathbf{A}_{[M \times N]}$  is an  $M \times N$  rectangular matrix, and  $\mathbf{x}_K$  is a  $K$ -dimensional column vector, as shown below.

$$\mathbf{A}_{[M \times N]} \equiv \mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & \cdot & \cdot & \cdot \\ \vdots & & & \vdots \\ a_{M1} & \cdot & \cdots & a_{MN} \end{bmatrix} = [a_{mn}]$$

$$\mathbf{x}_K \equiv \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_K \end{bmatrix}.$$

$\mathbf{A}_{[K \times K]}$  is a *square* matrix.

The *transpose* of a matrix is denoted by  $.^T$  and the *Hermitian* (i.e., complex conjugate transpose) by  $.^H$ :

$$\mathbf{A}^T = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{M1} \\ a_{12} & \cdot & \cdot & \cdot \\ \vdots & & & \vdots \\ a_{1N} & \cdot & \cdots & a_{NM} \end{bmatrix}.$$

$$\mathbf{A}^H = \begin{bmatrix} a_{11}^* & a_{21}^* & \cdots & a_{M1}^* \\ a_{12}^* & \cdot & \cdot & \cdot \\ \vdots & & & \vdots \\ a_{1N}^* & \cdot & \cdots & a_{NM}^* \end{bmatrix}.$$

$$\mathbf{x}^T = [x_1 \quad x_2 \quad \cdots \quad x_K] \quad (\text{a row vector}),$$

$$\mathbf{x}^H = [x_1^* \quad x_2^* \quad \cdots \quad x_K^*] \quad (\text{a row vector}).$$

The *identity* matrix is denoted by  $\mathbf{I}$ , the *unit* matrix by  $\mathbf{1}$ , and the *null* vector by  $\mathbf{0}$ :

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \quad \mathbf{1} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad \mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

A *diagonal* matrix is one with all of its off-diagonal elements equal to zero:

$$\mathbf{D} = \text{diag}(\mathbf{d}) = \text{diag}(d_k; k = 1, \dots, K) = \begin{bmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_K \end{bmatrix}.$$

The *determinant* of a square matrix  $\mathbf{A}$  is denoted by  $\det(\mathbf{A})$ .

## C.2. Definitions

For a square matrix  $\mathbf{A}$  we have the following definitions and relationships.

*Determinant*

$$\det(\mathbf{AB}) = \det(\mathbf{BA}) \quad (\text{C.1})$$

*Symmetric matrix*

$$\mathbf{A} = \mathbf{A}^T, \quad a_{jk} = a_{kj} \quad \forall j, k$$

*Skew symmetric (anti-symmetric) matrix*

$$\mathbf{A} = -\mathbf{A}^T, \quad a_{jk} = -a_{kj} \quad \forall j, k$$

*Hermitian symmetric matrix*

$$\mathbf{A} = \mathbf{A}^{*T} \equiv \mathbf{A}^H$$

*Trace*

$$\text{Tr}(\mathbf{A}_{[K \times K]}) = \sum_{k=1}^K a_{kk} = \text{Tr}(\mathbf{A}^T) \quad (\text{C.2})$$

$$\text{Tr}(\mathbf{A} + \mathbf{B}) = \text{Tr}(\mathbf{A}) + \text{Tr}(\mathbf{B}) \quad (\text{C.3})$$

$$\text{Tr}(\mathbf{AB}) = \text{Tr}(\mathbf{BA}) \quad (\text{C.4})$$

*Singular*

$$\det(\mathbf{A}) = 0$$

*Nonsingular*

$$\det(\mathbf{A}) \neq 0$$

## C.3. Matrix operations

*Addition of matrices is defined by*

$$\mathbf{A}_{[M \times N]} + \mathbf{B}_{[M \times N]} = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1N} + b_{1N} \\ a_{21} + b_{21} & \cdot & \cdot & \cdot \\ \vdots & \cdot & \ddots & \vdots \\ a_{M1} + b_{M1} & \cdot & \cdots & a_{MN} + b_{MN} \end{bmatrix} = [a_{mn} + b_{mn}]$$



The *product* of a matrix  $\mathbf{A}$  by a scalar  $\alpha$  is

$$\alpha \mathbf{A}_{[M \times N]} = \begin{bmatrix} \alpha a_{11} & \alpha a_{12} & \cdots & \alpha a_{1N} \\ \alpha a_{21} & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \alpha a_{M1} & \vdots & \cdots & \alpha a_{MN} \end{bmatrix} = [\alpha a_{mn}]$$

The *product of two matrices* is defined by

$$\mathbf{C}_{[K \times N]} = \mathbf{A}_{[K \times M]} \mathbf{B}_{[M \times N]} = \sum_{m=1}^M a_{km} b_{mn}$$

The *transpose* of a product is

$$(\mathbf{A}_{[K \times M]} \mathbf{B}_{[M \times N]})^T = (\mathbf{B}^T)_{[N \times M]} (\mathbf{A}^T)_{[M \times K]}$$

The *inverse*  $\mathbf{A}^{-1}$  of a nonsingular matrix  $\mathbf{A}$  is defined by

$$\mathbf{A}^{-1} \mathbf{A} = \mathbf{I} = \mathbf{A} \mathbf{A}^{-1}$$

For nonsingular matrices  $\mathbf{A}$  and  $\mathbf{B}$ , the inverse of the product,  $(\mathbf{A} \mathbf{B})^{-1}$ , is

$$(\mathbf{A} \mathbf{B})^{-1} = \mathbf{B}^{-1} \mathbf{A}^{-1} \quad (\text{C.5})$$

The determinant of the product of a matrix  $\mathbf{A}$  by a scalar  $\alpha$  is given by

$$\det(\alpha \mathbf{A}_{[K \times K]}) = \alpha^K \det(\mathbf{A}) \quad (\text{C.6})$$

## C.4. Special matrices

**Idempotent matrix**

$$\mathbf{H}^2 = \mathbf{H}$$

**Nilpotent matrix**

$$\mathbf{A}^k = \mathbf{0} \quad \text{for some integer } k$$

**Orthogonal** (for a real matrix  $\mathbf{A}$ )

$$\mathbf{A} \mathbf{A}^T = \mathbf{A}^T \mathbf{A} = \mathbf{D}, \quad \text{where } \mathbf{D} \text{ is diagonal.}$$

**Orthonormal** (for a real matrix  $\mathbf{A}$ )

$$\mathbf{A} \mathbf{A}^T = \mathbf{A}^T \mathbf{A} = \mathbf{I}$$

**Unitary**

$$\mathbf{A} \mathbf{A}^H = \mathbf{A}^H \mathbf{A} = \mathbf{I}$$

**Toeplitz**

$$\mathbf{A}_{[K \times K]} = [a_{jk} : a_{jk} = a_{(j+1)(k+1)}]$$

A Toeplitz matrix takes the form:

$$\mathbf{A}_{[K \times K]} = \begin{bmatrix} r_0 & r_1 & \cdots & r_{K-1} \\ r_{-1} & r_0 & \ddots & \\ \vdots & \ddots & \ddots & r_0 \\ r_{-K+1} & \cdots & r_{-1} & r_0 \end{bmatrix}$$

### C.5. Quadratic forms

For a square real symmetric matrix  $\mathbf{A}$  and a real column vector  $\mathbf{x}$ , we have the non-negative (real scalar) *quadratic form*:

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = \sum_{i,j=1}^K x_i a_{ij} x_j \geq 0 \quad \forall \mathbf{x}.$$

Similarly, for a Hermitian symmetric matrix  $\mathbf{A}$  and a complex column vector  $\mathbf{x}$ , we have the non-negative (real scalar) *Hermitian form*

$$\mathbf{x}^H \mathbf{A} \mathbf{x} = \sum_{i,j=1}^K x_i^* a_{ij} x_j \geq 0 \quad \forall \mathbf{x}.$$

We use the following definitions<sup>1</sup>:

**Positive definite (pd) matrix**

$$\mathbf{x}^H \mathbf{A} \mathbf{x} > 0 \quad \forall \mathbf{x} \neq \mathbf{0}.$$

**Nonnegative definite (nnd) matrix**

$$\mathbf{x}^H \mathbf{A} \mathbf{x} \geq 0 \quad \forall \mathbf{x}.$$

**Positive semi-definite (psd) matrix**

$$\mathbf{x}^H \mathbf{A} \mathbf{x} \geq 0 \quad \forall \mathbf{x} \quad \text{and}$$

$$\mathbf{x}^H \mathbf{A} \mathbf{x} = 0 \quad \text{for some } \mathbf{x} \neq \mathbf{0} \quad \text{and}$$

### C.6. Partitioned matrices

It is often useful to partition matrices into two or more sub-matrices; with appropriate attention to the dimensions of the matrices, operations can be carried out as if the sub-matrices were elements.

For rectangular matrices  $\mathbf{A}_{[(J+M) \times (K+L)]}$  and  $\mathbf{B}_{[(K+L) \times (N)]}$ ,

$$\mathbf{A}_{[(J+M) \times (K+L)]} = \begin{bmatrix} \mathbf{P}_{[J \times K]} & \mathbf{Q}_{[J \times L]} \\ \mathbf{R}_{[M \times K]} & \mathbf{S}_{[M \times L]} \end{bmatrix}$$

$$\mathbf{B}_{[(K+L) \times (N)]} = \begin{bmatrix} \mathbf{C}_{[K \times N]} \\ \mathbf{D}_{[L \times N]} \end{bmatrix}$$

---

<sup>1</sup>There is not consistency in the definition of positive semi-definite matrices: some writers do not distinguish between positive semi-definite and non-negative definite.

then

$$\left( \mathbf{A}_{[(K+L) \times (J+M)]} \right)^T = \begin{bmatrix} \mathbf{P}^T & \mathbf{R}^T \\ \mathbf{Q}^T & \mathbf{S}^T \end{bmatrix}$$

and

$$(\mathbf{A}\mathbf{B})_{[(J+M) \times (N)]} = \begin{bmatrix} (\mathbf{P}\mathbf{C} + \mathbf{Q}\mathbf{D})_{[J \times N]} \\ (\mathbf{R}\mathbf{C} + \mathbf{S}\mathbf{D})_{[M \times N]} \end{bmatrix}$$

## C.7. Useful formulae for determinants

### C.7.1. Partitioned matrix.

$$\begin{aligned} \det \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} &= \det(\mathbf{A}) \det(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}) \\ &= \det(\mathbf{D}) \det(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}) \end{aligned} \quad (\text{C.7})$$

### C.7.2. Bordered matrix.

$$\det \begin{bmatrix} \mathbf{A} & \mathbf{x} \\ \mathbf{x}^H & c \end{bmatrix} = \det(\mathbf{A}) (c - \mathbf{x}^H \mathbf{A}^{-1} \mathbf{x}) \quad (\text{C.8})$$

$$\det(\mathbf{A} - c\mathbf{x}\mathbf{x}^H) = \det(\mathbf{A}) (1 - c\mathbf{x}^H \mathbf{A}^{-1} \mathbf{x}) \quad (\text{C.9})$$

$$\det(\mathbf{A}) = \frac{1}{2} (\det(\mathbf{A} + \mathbf{x}\mathbf{x}^H) + \det(\mathbf{A} - \mathbf{x}\mathbf{x}^H)) \quad (\text{C.10})$$

## C.8. Useful formulae for matrix inverses

### C.8.1. Partitioned Hermitian symmetric matrix.

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} + \mathbf{F}\mathbf{E}^{-1}\mathbf{F}^H & -\mathbf{F}\mathbf{E}^{-1} \\ -\mathbf{E}^{-1}\mathbf{F}^H & \mathbf{E}^{-1} \end{bmatrix} \quad (\text{C.11})$$

$$= \begin{bmatrix} (\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{B}^H)^{-1} & -\mathbf{F}\mathbf{E}^{-1} \\ -\mathbf{E}^{-1}\mathbf{F}^H & \mathbf{E}^{-1} \end{bmatrix} \quad (\text{C.12})$$

where  $\mathbf{E} = \mathbf{D} - \mathbf{B}^H \mathbf{A}^{-1} \mathbf{B}$ , and  $\mathbf{F} = \mathbf{A}^{-1} \mathbf{B}$ .

**C.8.2. Bordered Hermitian symmetric matrix.**

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^H & d \end{bmatrix}^{-1} = \begin{bmatrix} \left( \mathbf{A} - \frac{\mathbf{B}\mathbf{B}^H}{d} \right)^{-1} & -\frac{\mathbf{A}^{-1}\mathbf{B}}{d - \mathbf{B}^H\mathbf{A}^{-1}\mathbf{B}} \\ -\frac{\mathbf{B}^H\mathbf{A}^{-1}}{d - \mathbf{B}^H\mathbf{A}^{-1}\mathbf{B}} & \frac{1}{d - \mathbf{B}^H\mathbf{A}^{-1}\mathbf{B}} \end{bmatrix} \quad (\text{C.13})$$

**C.8.3. Matrix plus dyad (Woodbury's identity).**

$$(\mathbf{A} + c\mathbf{u}\mathbf{v}^H)^{-1} = \mathbf{A}^{-1} - \frac{c\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^H\mathbf{A}^{-1}}{1 + c\mathbf{v}^H\mathbf{A}^{-1}\mathbf{u}} \quad (\text{C.14})$$

**C.8.4. Inverse of sum of matrices.**

For  $\mathbf{A}_{[M \times M]}$ ,  $\mathbf{D}_{[N \times N]}$ ,  $\mathbf{B}_{[M \times N]}$ ,  $\mathbf{A}$  and  $\mathbf{D}$  nonsingular,  $M < N$ ,

$$(\mathbf{A} + \mathbf{B}\mathbf{D}\mathbf{B}^H)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{B}^H\mathbf{A}^{-1}\mathbf{B} + \mathbf{D}^{-1})^{-1}\mathbf{B}^H\mathbf{A}^{-1} \quad (\text{C.15})$$

**C.8.5. Elements of the inverse.**

Let  $\mathbf{P} = [p_{ij}]$  be a nonsingular matrix,  
 $\mathbf{Q}$  be  $\mathbf{P}$  but with the  $(m, n)^{th}$  element reduced by unity,  
 $\mathbf{R}$  be  $\mathbf{P}$  but with every element increased by unity, and  
 $\mathbf{P}^{-1} \equiv [p^{(ij)}]$ .

Then

$$P^{(mn)} = 1 - \frac{|Q|}{|P|} \quad \text{and} \quad (\text{C.16})$$

$$\sum_{i,j} p^{(ij)} = 1 - \frac{|R|}{|P|} \quad (\text{C.17})$$

**C.9. Inequalities****C.9.1. Cauchy-Schwarz inequality.**

For any two column vectors  $\mathbf{x}$ ,  $\mathbf{y}$ ,

$$|\mathbf{x}^H \mathbf{y}|^2 \leq (\mathbf{x}^H \mathbf{x})(\mathbf{y}^H \mathbf{y}), \quad (\text{C.18})$$

with equality when and only when  $\mathbf{x} \propto \mathbf{y}$

### C.9.2. Diagonal elements of the inverse.

*Lemma C.1:*

If  $\mathbf{A}_{[K \times K]}$  is positive definite and  $\mathbf{A}^{-1} = [a^{ij}]$ , then

- (a)  $a^{ii} \geq 1/a_{ii}$ , with equality if and only if  $a_{ij} = 0$  for  $j = 1, \dots, i-1, i+1, \dots, K$ ;
- (b)  $a^{ii} = 1/a_{ii} \quad \forall i \Rightarrow a_{ij} = 0, \quad i \neq j$ ;
- (c)  $|\mathbf{A}| \leq a_{11} \dots a_{KK}$  with equality if and only if  $a_{ij} = 0, \quad i \neq j$ .

### C.9.3. Power of a matrix.

*Lemma C.2:*

Let  $\mathbf{A}$  be a square, real, symmetric matrix with all elements non-negative. Then for any integer  $n$ ,

$$(\mathbf{x}^H \mathbf{A}^n \mathbf{x})(\mathbf{x}^H \mathbf{x}) \geq (\mathbf{x}^H \mathbf{A} \mathbf{x})^n,$$

with equality if and only if  $\mathbf{x}$  is an eigenvector of  $\mathbf{A}$ .

### C.9.4. Kantorovich inequality.

If  $\mathbf{A}$  is positive definite Hermitian with eigenvalues

$$\{\lambda_i : \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K > 0\}$$

and corresponding eigenvectors  $\{\mathbf{p}_i\}$ , then

$$1 \leq \frac{(\mathbf{x}^H \mathbf{A} \mathbf{x})(\mathbf{x}^H \mathbf{A}^{-1} \mathbf{x})}{(\mathbf{x}^H \mathbf{x})^2} \leq \frac{1}{4} \left( \left( \frac{\lambda_1}{\lambda_K} \right)^{\frac{1}{2}} + \left( \frac{\lambda_K}{\lambda_1} \right)^{\frac{1}{2}} \right)^2. \quad (\text{C.19})$$

The lower bound is attained when  $\mathbf{x}$  is an eigenvector of  $\mathbf{A}$  and the upper bound when

$$\mathbf{x}(\mathbf{p}_1 + \mathbf{p}_K)/\sqrt{2}.$$

### C.9.5. Inequality involving the trace.

Let  $\mathbf{A}_{[K \times K]}$  be positive definite Hermitian symmetric,  $\mathbf{B}_{[K \times K]}$  be nonnegative definite Hermitian symmetric, with eigenvalues

$$\{\lambda_i : \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K > 0\}$$

then

$$\lambda_1 \text{Tr}(\mathbf{B}) \text{Tr}(\mathbf{A} \mathbf{B}) \leq \lambda_K \text{Tr}(\mathbf{A}) \quad (\text{C.20})$$

### C.10. Generalised inverses

#### C.10.1. Definition.

Let  $\mathbf{A}$  be an  $[M \times N]$  matrix. A *generalised inverse* (or *g-inverse*) of  $\mathbf{A}$  is an  $[M \times N]$  matrix  $\mathbf{A}^-$  such that  $\mathbf{x} = \mathbf{A}^- \mathbf{y}$  is a solution of  $\mathbf{A} \mathbf{x} = \mathbf{y}$  for any  $\mathbf{y}$  which makes the equation consistent.

#### C.10.2. Properties.

- (1)  $\mathbf{A}^-$  exists  $\iff \mathbf{A} \mathbf{A}^- \mathbf{A} = \mathbf{A}$ .
- (2)  $\mathbf{A}^-$  exists  $\iff \mathbf{H} = \mathbf{A}^- \mathbf{A}$  is idempotent and  $\text{rank}(\mathbf{H}) = \text{Tr}(\mathbf{H}) = \text{rank}(\mathbf{A})$ .
- (3) If  $\mathbf{A}^-$  exists, a general solution to the generalised inverse of the matrix  $\mathbf{A}$  is

$$\mathbf{A}^- + \mathbf{Z} - \mathbf{A}^- \mathbf{A} \mathbf{Z} \mathbf{A} \mathbf{A}^-, \quad (\text{C.21})$$

where  $\mathbf{Z}$  is an arbitrary matrix.

#### C.10.3. Moore-Penrose g-inverse.

The Moore-Penrose g-inverse, or pseudo-inverse of a matrix  $\mathbf{A}$  is denoted by  $\mathbf{A}^\dagger$  and has the following properties:

$$\begin{aligned} \mathbf{A} \mathbf{A}^\dagger \mathbf{A} &= \mathbf{A} \\ \mathbf{A}^\dagger \mathbf{A} \mathbf{A}^\dagger &= \mathbf{A}^\dagger \\ (\mathbf{A} \mathbf{A}^\dagger)^H &= \mathbf{A} \mathbf{A}^\dagger \\ (\mathbf{A}^\dagger \mathbf{A})^H &= \mathbf{A}^\dagger \mathbf{A}. \end{aligned} \quad (\text{C.22})$$

If  $\mathbf{A}$  is of full rank,

$$\mathbf{A}^\dagger = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H. \quad (\text{C.23})$$

### C.11. Kronecker product

If  $\mathbf{A}_{[K \times L]} = [a_{kl}]$  and  $\mathbf{B}_{[M \times N]} = [b_{mn}]$ , their Kronecker product is defined by:

$$(\mathbf{A} \otimes \mathbf{B})_{[(KM) \times (LN)]} = \begin{bmatrix} a_{11} \mathbf{B} & a_{12} \mathbf{B} & \cdots & a_{1L} \mathbf{B} \\ a_{21} \mathbf{B} & \cdot & \cdot & \cdot \\ \vdots & \cdot & \ddots & \cdot \\ a_{K1} \mathbf{B} & \cdot & \cdots & a_{KL} \mathbf{B} \end{bmatrix}. \quad (\text{C.24})$$

The following relationships apply:

$$\begin{aligned}
 \mathbf{0} \otimes \mathbf{A} &= \mathbf{A} \otimes \mathbf{0} = \mathbf{0}, \\
 (\alpha \mathbf{A}) \otimes (\beta \mathbf{B}) &= \alpha\beta(\mathbf{A} \otimes \mathbf{B}), \\
 (\mathbf{A}_1 \mathbf{A}_2) \otimes \mathbf{B} &= \mathbf{A}_1 \otimes \mathbf{B} + \mathbf{A}_2 \otimes \mathbf{B}, \\
 \mathbf{A} \otimes (\mathbf{B}_1 \mathbf{B}_2) &= (\mathbf{A} \otimes \mathbf{B}_1)(\mathbf{A} \otimes \mathbf{B}_2), \\
 (\mathbf{A}_1 \mathbf{A}_2) \otimes (\mathbf{B}_1 \mathbf{B}_2) &= (\mathbf{A}_1 \mathbf{B}_1)(\mathbf{A}_2 \mathbf{B}_2).
 \end{aligned} \tag{C.25}$$

If  $\mathbf{A}$  and  $\mathbf{B}$  are of full rank, so is  $\mathbf{A} \otimes \mathbf{B}$ , and

$$(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1} \tag{C.26}$$

For any  $\mathbf{A}, \mathbf{B}$ ,

$$(\mathbf{A} \otimes \mathbf{B})^{-} = \mathbf{A}^{-} \otimes \mathbf{B}^{-} \tag{C.27}$$

## C.12. Calculus

### C.12.1. Notation.

We often want to find the extremum (maximum or minimum) of a function, so we need to set its derivative to zero. The derivative of a matrix  $\mathbf{A}$  or a vector  $\mathbf{A}$  with respect to some real scalar variable  $x$  is defined as follows.

$$\begin{aligned}
 \frac{\partial \mathbf{A}}{\partial x} &= \frac{\partial}{\partial x} \begin{bmatrix} a_{11} & \cdots & a_{1K} \\ \vdots & \ddots & \vdots \\ a_{K1} & \cdots & a_{KK} \end{bmatrix} = \begin{bmatrix} \frac{\partial a_{11}}{\partial x} & \cdots & \frac{\partial a_{1K}}{\partial x} \\ \vdots & \ddots & \vdots \\ \frac{\partial a_{K1}}{\partial x} & \cdots & \frac{\partial a_{KK}}{\partial x} \end{bmatrix} \\
 \frac{\partial \mathbf{a}}{\partial x} &= \frac{\partial}{\partial x} \begin{bmatrix} a_{11} \\ \vdots \\ a_{K1} \end{bmatrix} = \begin{bmatrix} \frac{\partial a_{11}}{\partial x} \\ \vdots \\ \frac{\partial a_{K1}}{\partial x} \end{bmatrix}
 \end{aligned}$$

We can also define derivatives of a scalar function  $f(\mathbf{x})$  with respect to a real vector variable  $\mathbf{x}$  as follows:

$$\frac{\partial}{\partial \mathbf{x}} f(\mathbf{x}) = \begin{bmatrix} \frac{\partial}{\partial x_1} f(\mathbf{x}) \\ \vdots \\ \frac{\partial}{\partial x_K} f(\mathbf{x}) \end{bmatrix}$$

### C.12.2. Derivative of products.

We frequently want to differentiate products of the type  $\mathbf{x}^T \mathbf{A} \mathbf{y}$  and quadratic forms  $\mathbf{x}^T \mathbf{A} \mathbf{x}$  with respect to the real vector  $\mathbf{x}$ . It is not difficult to show that

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{x}^T \mathbf{A} \mathbf{y} = \mathbf{A} \mathbf{y} \quad (\text{C.28})$$

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{x}^T \mathbf{x} = 2 \mathbf{x} \quad (\text{C.29})$$

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{x}^T \mathbf{A} \mathbf{x} = 2 \mathbf{x} \quad (\text{C.30})$$

### C.12.3. Derivative of determinant.

$$\frac{\partial \ln \det(\mathbf{A})}{\partial \alpha} = \text{Tr} \left( \mathbf{A}^{-1} \frac{\partial \mathbf{A}}{\partial \alpha} \right) \quad (\text{C.31})$$

### C.12.4. Derivative of inverse of matrix.

$$\frac{\partial \mathbf{A}^{-1}}{\partial \alpha} = -\mathbf{A}^{-1} \frac{\partial \mathbf{A}}{\partial \alpha} \mathbf{A}^{-1} \quad (\text{C.32})$$

## C.13. Eigenvalues and eigenvectors

Here we concern ourselves only with Hermitian symmetric matrices.

For a  $[K \times K]$  matrix  $\mathbf{R}$ , the eigenvalues  $\{\lambda_k, k = 1, \dots, K\}$  are the solutions to the set of  $K$  linear equations:

$$\det(\mathbf{R} - \lambda \mathbf{I}) = 0. \quad (\text{C.33})$$

Eigenvectors  $\{\mathbf{P}_k, k = 1, \dots, K\}$  are solutions to the equation:

$$\mathbf{R} \mathbf{P}_k = \lambda_k \mathbf{P}_k. \quad (\text{C.34})$$

Note that if we multiply  $\mathbf{P}$  by any scalar, the product would still satisfy (C.34); by convention,  $\mathbf{P}$  is normalised so that its Euclidean (or  $L_2$ ) norm

$$\|\mathbf{P}\|_2 = \sum_k |p_k|^2 = \mathbf{P}^H \mathbf{P} = 1. \quad (\text{C.35})$$

The eigenvectors are *orthonormal*:

$$\mathbf{P}_i^H \mathbf{P}_j = \delta_{ij}, \quad (\text{C.36})$$

where  $\delta_{ij}$  is the Kronecker product

$$\begin{aligned} \delta_{ij} &= 1, i = j \\ &= 0, i \neq j. \end{aligned} \quad (\text{C.37})$$

If the  $K$  eigenvalues are distinct (i.e., no two are the same), then the corresponding  $K$  eigenvectors are unique.



All the eigenvalues of a Hermitian matrix are non-negative ( $\lambda_i \geq 0 \forall i$ ).

The eigenvalues give valuable information about a matrix. We may arrange them in rank order:  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K \geq 0$  and, of course, we have to arrange their corresponding eigenvectors  $\mathbf{P}_i$  in the same order.

Let us assemble the eigenvalues and eigenvectors to form  $[K \times K]$  matrices as follows:

$$\mathbf{P} = [\mathbf{P}_1 : \mathbf{P}_2 : \dots : \mathbf{P}_K], \quad \text{and} \quad (\text{C.38})$$

$$\begin{aligned} \mathbf{\Lambda}_{[K]} &= \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_K \end{bmatrix} \\ &= \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_K) \end{aligned} \quad (\text{C.39})$$

$\mathbf{P}$  is *unitary*:

$$\mathbf{P}\mathbf{P}^H = \mathbf{P}^H\mathbf{P} = \mathbf{I}. \quad (\text{C.40})$$

It then follows from (C.34) that

$$\begin{aligned} \mathbf{R}\mathbf{P} &= \mathbf{P}\mathbf{\Lambda}_{[K]}, \quad \text{whence} \\ \mathbf{R} &= \mathbf{P}\mathbf{\Lambda}_{[K]}\mathbf{P}^H. \end{aligned} \quad (\text{C.41})$$

This expression of a Hermitian matrix  $\mathbf{R}$  in terms of its eigenvectors is termed *eigen-decomposition*. We can use a set of  $K$  such orthonormal vectors  $\{p_i : i = 1, \dots, K\}$  as the basis to decompose any  $K$ -vector  $\mathbf{A}$ :

$$\mathbf{A} = \sum_{k=1}^K \alpha_k p_k, \quad (\text{C.42})$$

where  $\{\alpha_k, k = 1, \dots, K\}$  is the set of appropriate scalar weights.

If  $\lambda_k > 0 \quad \forall k$ , the matrix is nonsingular with inverse:

$$\mathbf{R}^{-1} = \mathbf{P}\mathbf{\Lambda}^{-1}\mathbf{P}^H, \quad (\text{C.43})$$

$$\mathbf{\Lambda}^{-1} = \text{diag}(\lambda_1^{-1}, \lambda_2^{-1}, \dots, \lambda_K^{-1}). \quad (\text{C.44})$$

If any of the eigenvalues is zero, the matrix is singular. If  $K - L$  of the eigenvalues are zero, the rank of the matrix is  $L$ . If all the eigenvalues are positive, the matrix is of *full rank* (nonsingular).

## APPENDIX D

### FUNDAMENTALS OF ESTIMATION THEORY

Let  $\mathbf{x} = [x_1 x_2 \dots x_N]^H$  be a vector of  $N$  observations of random variables with a probability density function

$$p(\mathbf{x}; \boldsymbol{\alpha}) \tag{D.1}$$

where  $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \dots, \alpha_L]^H$  are a set of parameters that characterize the random process. These parameters are generally unknown in value and need be estimated. An example is where the  $x_1, x_2, \dots, x_N$  are samples of a zero mean Gaussian white noise process of power  $\sigma_x^2$ . Then the probability density function,  $p(\mathbf{x}; \boldsymbol{\alpha})$ , is given by

$$p(\mathbf{x}; \alpha_1 = \sigma_n^2) = \frac{1}{\sqrt{2\pi\sigma_n^2}} \exp \left[ -\frac{1}{2\sigma_n^2} \sum_{n=1}^N x_n^2 \right] \tag{D.2}$$

Statistical estimation theory is concerned with the problem of estimating the parameters  $\boldsymbol{\alpha}$  from the given observations (data) i.e.,  $\mathbf{x}$ . In general this will be of the form

$$\hat{\alpha}_j = g(x_1, x_2, \dots, x_N) \tag{D.3}$$

Note that  $\hat{\alpha}_j$  is a particular function of the observations,  $x_j$ , and will take on a certain value for a given set of  $x_j$ 's. Since the  $x_j$  are random it follows that  $\hat{\alpha}_j$  is a random variable; it is termed an **estimator** of  $\alpha_j$ . Any particular value it takes on is termed an **estimate**. Thus taking another different set of  $x_j$ 's, i.e., a different set of observations, results in a different value for  $\hat{\alpha}_j$ . Since  $\hat{\alpha}_j$  is a random variable it can be described by a probability density function and in some cases it is important to know that. However for many purposes it suffices to be able to calculate the mean value and the variance of  $\hat{\alpha}_j$ . Generalising, knowing the second order statistics of the estimators, i.e., the covariances of the total parameter set  $\hat{\alpha}_1, \hat{\alpha}_2, \dots, \hat{\alpha}_L$  often suffices for many applications.

As an example the power of a zero mean stationary random process is given by

$$\sigma_x^2 = E \{x^2[n]\} \quad (\text{D.4})$$

### D.1. Mean Value of an Estimator

The mean value of an estimator, defined as  $E\{\hat{\alpha}_j\}$  determines whether, on average, the value of the estimate,  $\hat{\alpha}_j$ , and that of the unknown parameter,  $\alpha_j$ , are equal. If they are the estimator is said to be unbiased. The bias,  $b$ , of an estimator is defined as

$$b \triangleq E\{\hat{x} - x\} \quad (\text{D.5})$$

For example the estimator

$$\hat{\sigma}_x^2 = \frac{1}{N} \sum_{j=1}^N x_j^2 \quad (\text{D.6})$$

is an unbiased estimator of the power if the  $x_j$  are known to have zero mean.

### D.2. Variance of an Estimator

The variance of an estimator tells us how tightly the estimate is clustered around its mean value. The tighter the clustering the more likely it will be that the estimate, derived from any set of data, will be close to the true value. For example, the estimator of the mean of a random process

$$\hat{\mu}_x = \frac{1}{N} \sum_{n=1}^N x_n \quad (\text{D.7})$$

where  $E\{x_n\} = 0$  will obviously be better clustered around zero if  $N$  is large. To see this it can be shown that the variance of  $\hat{\mu}_x$  is given by

$$\sigma^2(\hat{\mu}_x) = E\{\hat{\mu}_x^2\} = \frac{1}{N^2} \sum_n \sum_m E\{x_n x_m\} \quad (\text{D.8})$$

which for white noise reduces to

$$\sigma^2(\hat{\mu}_x) = \frac{1}{N^2} \sum_n \sum_m \sigma_n^2 \delta_{mn} = \frac{1}{N} \sigma_n^2 \quad (\text{D.9})$$

indicating that increasing  $N$  decreases the standard deviation of the estimate.

### D.3. Proof of CRLB

#### D.3.1. Scalar non-random parameter.

Here a lower bound on the variance of an unbiased estimator is derived.

Let  $\mathbf{x} = [x_1 \ x_2 \ \cdots \ x_N]$  be a vector of observations of random variables with differentiable probability density function  $p(\mathbf{x}; \alpha)$ , where  $\alpha$  is an unknown real parameter, and let  $\hat{\alpha} = f(\mathbf{x})$  be some unbiased estimator of  $\alpha$ . It follows that

$$E\{\hat{\alpha} - \alpha\} = 0 \quad (\text{D.10})$$

or

$$\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (\hat{\alpha} - \alpha) p(\mathbf{x}; \alpha) dx_1 \cdots dx_N = \int_{-\infty}^{\infty} (\hat{\alpha} - \alpha) p(\mathbf{x}; \alpha) d\mathbf{x} = 0. \quad (\text{D.11})$$

Differentiating with respect to  $\alpha$  gives:

$$\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (\hat{\alpha} - \alpha) \frac{\partial p(\mathbf{x}; \alpha)}{\partial \alpha} d\mathbf{x} - 1 = 0. \quad (\text{D.12})$$

Noting that

$$\frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} = \frac{1}{p(\mathbf{x}; \alpha)} \frac{\partial p(\mathbf{x}; \alpha)}{\partial \alpha} \quad (\text{D.13})$$

gives

$$\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \left( (\hat{\alpha} - \alpha) \frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} \right) p(\mathbf{x}; \alpha) d\mathbf{x} = 1$$

or

$$E \left\{ (\hat{\alpha} - \alpha) \frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} \right\} = 1. \quad (\text{D.14})$$

Taking the square of both sides and applying the Cauchy-Schwarz inequality (C.18):

$$E\{|\hat{\alpha} - \alpha|^2\} E \left\{ \left| \frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} \right|^2 \right\} \leq 1$$

or

$$\text{var}(\hat{\alpha}) \geq \frac{1}{E \left\{ \left| \frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} \right|^2 \right\}} \quad (\text{D.15})$$

where as before  $L_{\mathbf{x}}(\alpha)$  is the log-likelihood function. This is the Cramer-Rao lower bound, a most useful inequality, which applies to *any* unbiased estimator of a single real parameter  $\alpha$ .

If the second derivative of  $p(\mathbf{x}; \alpha)$  exists, an alternative useful form can be obtained as follows. Recall that

$$\int_{-\infty}^{\infty} p(\mathbf{x}; \alpha) d\mathbf{x} = 1. \quad (\text{D.16})$$

Differentiating twice with respect to  $\alpha$  gives

$$\int_{-\infty}^{\infty} \frac{\partial p(\mathbf{x}; \alpha)}{\partial \alpha} d\mathbf{x} = \int_{-\infty}^{\infty} \frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} p(\mathbf{x}; \alpha) d\mathbf{x} = 0 \quad (\text{D.17})$$

Differentiating again with respect to  $\alpha$  gives

$$\int_{-\infty}^{\infty} \frac{\partial^2 \ln p(\mathbf{x}; \alpha)}{\partial \alpha^2} p(\mathbf{x}; \alpha) d\mathbf{x} + \int_{-\infty}^{\infty} \left( \frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} \right)^2 p(\mathbf{x}; \alpha) d\mathbf{x} = 0$$

or

$$E \left\{ \frac{\partial^2 \ln p(\mathbf{x}; \alpha)}{\partial \alpha^2} \right\} = -E \left\{ \left( \frac{\partial \ln p(\mathbf{x}; \alpha)}{\partial \alpha} \right)^2 \right\}. \quad (\text{D.18})$$

Hence from (D.15)

$$\text{var}(\hat{\alpha}) \geq \frac{1}{E \left\{ \left| \frac{\partial L_{\mathbf{x}}(\alpha)}{\partial \alpha} \right|^2 \right\}} = -\frac{1}{E \left\{ \frac{\partial^2 L_{\mathbf{x}}(\alpha)}{\partial \alpha^2} \right\}}. \quad (\text{D.19})$$

### D.3.2. Estimation of the mean.

Take the simple case of  $N$  independent random variables, all with the same known variance  $\sigma^2$  and the same unknown mean  $\mu$  which is to be estimated. It follows that

$$E\{x_n x_m\} = \delta_{nm}(\sigma^2 + \mu^2). \quad (\text{D.20})$$

Their joint probability density function is:

$$\begin{aligned}
p(\mathbf{x}|\mu) &= \prod_{n=1}^N \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-(x_n - \mu)^2/(2\sigma^2)\right) \\
L_x(\mu) &= \sum_{n=1}^N -\frac{(x_n - \mu)^2}{2\sigma^2} - \text{constant}
\end{aligned} \tag{D.21}$$

Setting  $\partial L_x(\mu)/\partial\mu = 0$ , the solution is simply

$$\hat{\mu} = \frac{1}{N} \sum_{n=1}^N x_n \tag{D.22}$$

The expectation is

$$E\{\hat{\mu}\} = \mu \tag{D.23}$$

so the estimator is unbiased. The variance of  $\hat{\mu}$  is

$$\begin{aligned}
\text{var}(\hat{\mu}) &= E\{\hat{\mu}^2\} - (E\{\hat{\mu}\})^2 \\
&= \frac{\sigma^2}{N}
\end{aligned} \tag{D.24}$$

Next the Cramer-Rao lower bound is derived.

$$\left(\frac{\partial L_x(\mu)}{\partial\mu}\right)^2 = \sum_n \sum_m \frac{(x_n - \mu)(x_m - \mu)}{\sigma^4}$$

so

$$\text{var}(\hat{\mu}) \geq \frac{1}{E\left\{\left(\frac{\partial L_x(\mu)}{\partial\mu}\right)^2\right\}} = \frac{\sigma^2}{N}. \tag{D.25}$$

In this case the variance attains the lower limit. Such an estimator is said to be *efficient*.

#### D.4. Least Squares Estimation

Least squares estimation is useful in situations where there is a linear relationship between the unknown signal parameters  $\boldsymbol{\alpha}$  and the observations, i.e.,

$$\mathbf{x}_k = \mathbf{A}_k \boldsymbol{\alpha} + \mathbf{n}_k \tag{D.26}$$

where  $\mathbf{x}_k$  is the vector of derivatives at time  $k$ , for  $k = 1, 2, \dots, K$ , and  $\mathbf{n}_k$  is additive measurement noise. Often the  $\mathbf{A}_k$  are independent of  $k$ .

Minimizing the quadratic

$$\sum_{k=1}^K (\mathbf{x}_k - \mathbf{A}_k \boldsymbol{\alpha})^H \mathbf{R}_n^{-1} (\mathbf{x}_k - \mathbf{A}_k \boldsymbol{\alpha}) \quad (\text{D.27})$$

with respect to the unknown  $\alpha_j$ 's gives the following estimator

$$\hat{\boldsymbol{\alpha}}_{\text{LS}} = \left( \sum_{k=1}^K \mathbf{A}_k^H \mathbf{A}_k \right)^{-1} \left( \sum_{k=1}^K \mathbf{A}_k^H \mathbf{x}_k \right) \quad (\text{D.28})$$

In the above  $\mathbf{R}_n = E\{\mathbf{n}_k \mathbf{n}_k^H\}$ .

When the  $\mathbf{A}_k$  are independent of  $k$  the above reduces to

$$\hat{\boldsymbol{\alpha}}_{\text{LS}} = \frac{1}{K} (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H \sum_{k=1}^K \mathbf{x}_k \quad (\text{D.29})$$

### D.5. Bias of LS Estimator

Substituting the above equation for the  $\mathbf{x}_k$  in the expression for  $\hat{\boldsymbol{\alpha}}_{\text{LS}}$  gives

$$\begin{aligned} \hat{\boldsymbol{\alpha}}_{\text{LS}} &= \left( \sum_{k=1}^T \mathbf{A}_k^H \mathbf{A}_k \right)^{-1} \sum_{k=1}^T \mathbf{A}_k^H (\mathbf{A}_k \boldsymbol{\alpha} + \mathbf{n}_k) \\ &= \left( \sum_{k=1}^T \mathbf{A}_k^H \mathbf{A}_k \right)^{-1} \left( \sum_{k=1}^T \mathbf{A}_k^H \mathbf{A}_k \right) \boldsymbol{\alpha} + \left( \sum_{k=1}^T \mathbf{A}_k^H \mathbf{A}_k \right)^{-1} \sum_{k=1}^T \mathbf{A}_k^H \mathbf{n}_k \\ &= \boldsymbol{\alpha} + \left( \sum_{k=1}^T \mathbf{A}_k^H \mathbf{A}_k \right)^{-1} \sum_{k=1}^T \mathbf{A}_k^H \mathbf{n}_k \end{aligned} \quad (\text{D.30})$$

Now taking an ensemble average over the  $\mathbf{n}_k$ , gives

$$\begin{aligned} E\{\hat{\boldsymbol{\alpha}}_{\text{LS}}\} &= \boldsymbol{\alpha} + \left( \sum_{k=1}^T \mathbf{A}_k^H \mathbf{A}_k \right)^{-1} \sum_{k=1}^T \mathbf{A}_k^H E\{\mathbf{n}_k\} \\ &= \boldsymbol{\alpha} \end{aligned} \quad (\text{D.31})$$

Thus the LS estimator is unbiased.

### D.6. Covariance of the LS Estimator

From (D.30)

$$\hat{\alpha}_{LS} - \alpha = \left( \sum_{k=1}^T A_k^H A_k \right)^{-1} \sum_{k=1}^T A_k^H E\{\mathbf{n}_k\} \quad (\text{D.32})$$

Thus

$$\begin{aligned} & E \left\{ (\hat{\alpha}_{LS} - \alpha) (\hat{\alpha}_{LS} - \alpha)^H \right\} \\ &= E \left\{ \left( \sum_{k=1}^T A_k^H A_k \right)^{-1} \left( \sum_{k=1}^T A_k^H \mathbf{n}_k \right) \left( \sum_{k=1}^T \mathbf{n}_k^H A_k \right) \left( \sum_{k=1}^T A_k^H A_k \right)^{-1} \right\} \\ &= \left\{ \left( \sum_{k=1}^T A_k^H A_k \right)^{-1} \sum_{k'=1}^T \sum_{k=1}^T A_k^H E\{\mathbf{n}_k \mathbf{n}_{k'}^H\} A_{k'} \left( \sum_{k=1}^T A_k^H A_k \right)^{-1} \right\} \\ &= \left\{ \left( \sum_{k=1}^T A_k^H A_k \right)^{-1} \left( \sum_{k=1}^T A_k^H A_k \right) \left( \sum_{k=1}^T A_k^H A_k \right)^{-1} \right\} \\ &= \left( \sum_{k=1}^T A_k^H A_k \right)^{-1} \quad (\text{D.33}) \end{aligned}$$



## APPENDIX E

# PROBLEMS

### E.1. Mechanical vs Electronic steering

Discuss the difference between a mechanically steered and an electronically steered array.

[▷ Back to Course Notes](#)

### E.2. Beamwidth

Consider a linear array of  $K \gg 1$  equi-spaced receivers, with its beam steered in the horizontal plane ( $\phi = 90^\circ$ .)

- (1) Derive an expression for the beamwidth of the main lobe as a function of the horizontal steer angle  $\theta$ . (Hint: consider separately the beamwidth in the vicinity of  $90^\circ$ .)
- (2) Write Matlab code for the beamwidth and plot as a function of  $\theta$ .

[▷ Back to Course Notes](#)

### E.3. Beampattern nulls

Consider an array of 8 receivers equally spaced  $\lambda/2$  apart.

- (1) Determine the positions of nulls of the beampattern of a beam steered at broadside.
- (2) How many beams overlapping at  $\sim -3.9\text{dB}$  points are required for complete angular coverage over  $360^\circ$ ?
- (3) What are the steering directions of these beams?
- (4) Write down the steering vector in each case.

Re-do the question for the case in which the spacing is  $\lambda/4$ .

[▷ Back to Course Notes](#)

### E.4. Interferences

Consider a linear array of equally spaced receivers. There is uncorrelated receiver noise present. A beam is steered at broadside in the direction of a plane-wave signal whose with a signal-to-(receiver) noise ratio of 0dB. An interfering plane-wave signal is incident upon the array from some angle  $\theta$  and the interference-to-receiver noise ratio is 10dB.

- (1) Power from the interference will leak into the broadside beam through its sidelobes. What is the minimum signal-to-interference ratio provided that the interference is not within the main beamwidth?
- (2) What is the maximum signal-to-interference ratio and when does it occur?
- (3) What happens when the interference-to-receiver noise ratio is increased to 30dB?

▷ Back to Course Notes

### E.5. Beampattern vs frequency

Consider a linear array of  $K$  equally spaced receivers, where  $K$  is large ( $> 10$ , say), and use as a reference the beampattern when the receivers are spaced half a wavelength apart.

- (1) What would you expect to happen to the beamwidth and the amplitude of the first sidelobe if the frequency is now reduced?
- (2) Briefly give reasons for your answer.

▷ Back to Course Notes

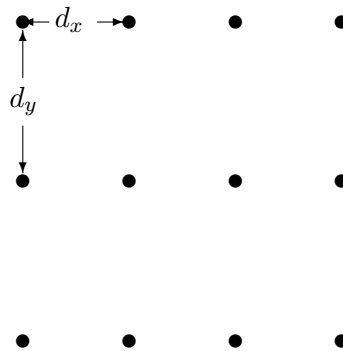
### E.6. Beamwidth and array geometry

Consider two arrays, each with adjacent receivers spaced a distance  $d$  apart: one is linear and with a beam steered broadside, and the second is circular. Which array would have the narrower beamwidth and why?

▷ Back to Course Notes

### E.7. Beampattern of rectangular array

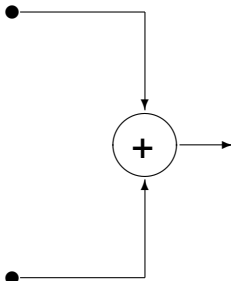
Consider a  $3 \times 4$  rectangular array located in the  $(x, y)$  plane with spacings  $d_x$  and  $d_y$  between columns and rows, respectively.



- (1) Write down an expression for the steering vector of this array.
- (2) Hence, or otherwise, derive an expression for the beampattern of this array

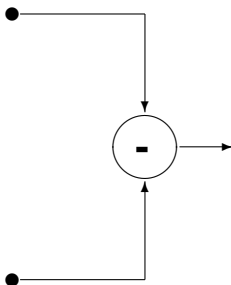
▷ Back to Course Notes

### E.8. 2-receiver array



- (1) Write down an expression for the beampattern of a two-element array steered at broadside (as illustrated).

- (2) Plot this as a function of the incidence angle when  $d/\lambda = 1, 1/2, 1/4, 1/8$ .



Repeat for the two-receiver array (but with sign change) as shown, for  $d/\lambda = .5$ .

[▷ Back to Course Notes](#)

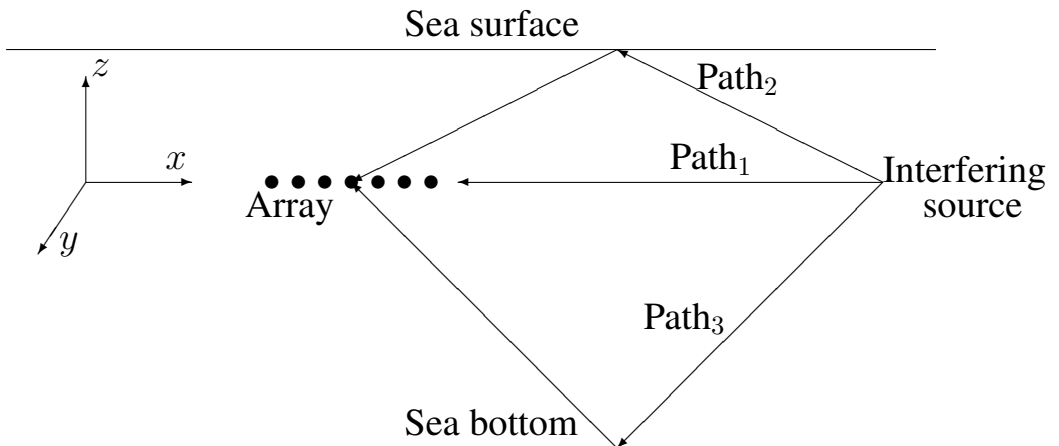
### E.9. Grating lobes

Consider a linear array of equally spaced receivers.

- (1) Derive an expression for the positions of the grating lobes when the array is steered in an arbitrary direction  $\theta$ .
- (2) When the receivers are spaced half a wavelength apart, derive the positions of the grating lobes of a beam steered at broadside.

[▷ Back to Course Notes](#)

### E.10. Multipath arrivals



Consider an array of hydrophone (underwater acoustic receivers) towed as a linear array, as illustrated in the sketch showing a section in the vertical ( $y - z$ ) plane. Wanted signals are in the horizontal ( $x - y$ ) plane of the array and beams are steered over angles  $0 \leq \theta \leq 180^\circ$  in the expectation that signals will only arrive from sources in the horizontal plane.

However, there is an interfering signal source which arrives at the array via three paths (all in the  $y - z$  plane). The arrivals via the three paths are assumed to be independent.

Path<sub>1</sub> will appear in the beam of the array steered at end-fire.

- (1) Where will the signals from Path<sub>2</sub> and Path<sub>3</sub> appear?
- (2) Discuss in terms of the cones of symmetry for a linear array.

[▷ Back to Course Notes](#)

### E.11. Towed array shape errors

You have a linear array of hydrophones (acoustic receivers) in a flexible hose (called a ‘streamer’) towed behind a ship in the  $x$ -direction as illustrated. The weight of the tow cable keeps the array at its operating depth. The streamer is made neutrally buoyant (i.e., with the same density as sea water) so that it tends to stream horizontally.

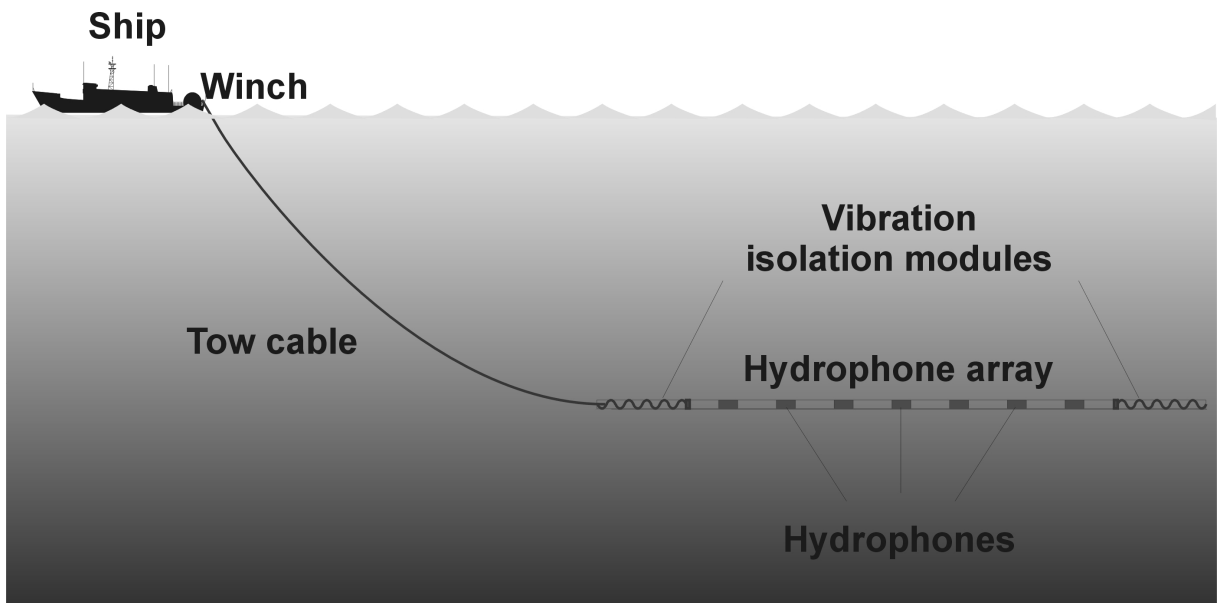
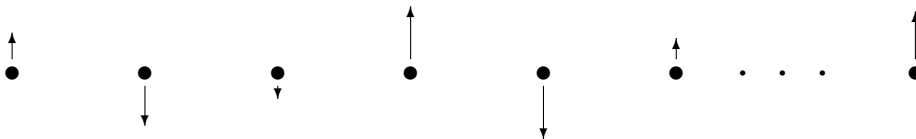


FIGURE E.1. Array towed behind ship



The array is meant to be straight. However, because of hydrodynamic forces, the receivers are slightly displaced in the  $y$ - and  $z$ - directions. Assume that these displacements are random and unknown, and that there are no displacements in the  $x$ - direction.

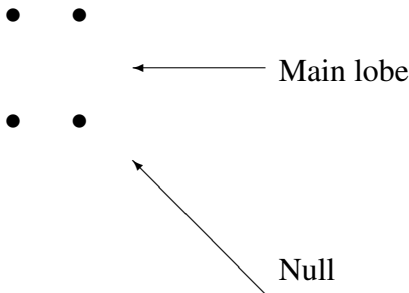
- (1) Consider the pattern of a beam steered at broadside. What would you expect to be the effect of the displacements on:
  - (a) the main lobe,
  - (b) the first sidelobe,

- (c) the sidelobe furthest from the main lobe?  
In each case, briefly give your reasons.

- (2) Consider next the pattern of a beam steered at end fire. What would you expect to be the effect of the displacements on:
- (a) the main lobe,
  - (b) the first sidelobe,
  - (c) the sidelobe furthest from the main lobe,
- In each case, briefly give your reasons.

[▷ Back to Course Notes](#)

### E.12. Null-steering 1



Consider the array shown. Write down the  $2 \times 4$  constraint matrix for steering a main lobe in the  $x$ -direction and a null in the direction  $-45^\circ$ . Find the weight vectors using the Moore-Penrose pseudo-inverse.

[▷ Back to Course Notes](#)

### E.13. Steering multiple nulls

To derive an expression for the null-steering beamformer we seek a solution to

$$\mathbf{A} \mathbf{w} = \delta_1^{L+1},$$

where

$$\mathbf{A} = \begin{bmatrix} \mathbf{v}^H(\mathbf{k}) \\ \mathbf{v}^H(\mathbf{k}_1) \\ \vdots \\ \mathbf{v}^H(\mathbf{k}_L) \end{bmatrix},$$

$\mathbf{k}$  is the beamsteering wavevector and  $\mathbf{k}_1, \mathbf{k}_2, \dots, \mathbf{k}_L$  are the wavevectors corresponding to  $L$  null directions, and

$$\delta_1^{(L+1)} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

A general solution is

$$\mathbf{w} = \mathbf{A}^- \delta_1^{(L+1)},$$

where  $\mathbf{A}^-$  is a generalised inverse of  $\mathbf{A}$ .

A special case is the Moore-Penrose generalised inverse denoted by  $\mathbf{A}^\dagger$ ; when  $(\mathbf{A}\mathbf{A}^H)$  is non-singular,

$$\mathbf{A}^\dagger = \mathbf{A}^H (\mathbf{A}\mathbf{A}^H)^{-1}$$

and the so-called minimum-norm solution is

$$\mathbf{w} = \mathbf{A}^H (\mathbf{A}\mathbf{A}^H)^{-1} \delta_1^{(L+1)}.$$

Verify by direct multiplication that a general form for  $\mathbf{w}$  is

$$\mathbf{w} = \mathbf{A}^\dagger \delta_1^{(L+1)} + (\mathbf{I} - \mathbf{A}^\dagger \mathbf{A}) \mathbf{z},$$

where  $\mathbf{z}$  is an arbitrary vector.

▷ Back to Course Notes

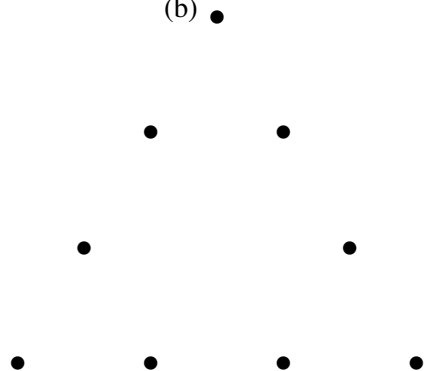
### E.14. Symmetries

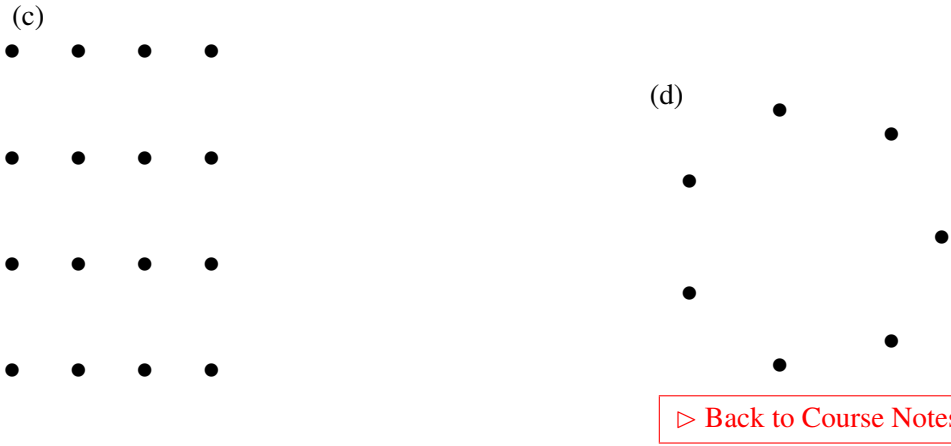
What are the symmetries of the following arrays?

(a)



(b)





### E.15. Beampattern of staggered array

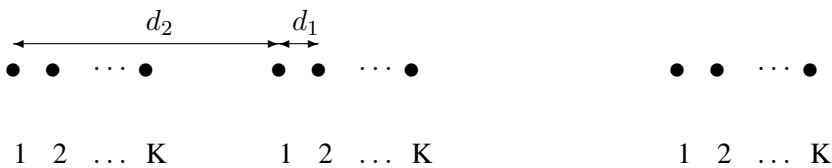


Derive an expression for the beampattern of the staggered array of  $2K$  receivers as shown.

▷ Back to Course Notes

### E.16. $N$ sub-arrays of $K$ receivers

Consider an array of  $N$  equi-spaced sub-arrays as illustrated below, but aligned along the  $y$ -axis .



Each sub-array is a linear array of  $K$  equi-spaced receivers spaced a distance  $d_1$  apart. There are  $N$  sub-arrays and the first receiver of adjacent sub-arrays are  $d_2$  apart. Consider the situation where all the receivers are summed to steer a beam in the broadside direction.

- (1) Using the Principle of Beampattern Multiplication, derive an expression for the beampattern of the full array.
- (2) Sketch on the same axis (do not use Matlab) the beampattern of a sub-array and that of the an array of  $K$  receivers spaced  $d_2$  apart as a function



of  $(\mathbf{k}_s)_y$  for the following cases.

(a)  $d_2 = 4Kd_1$

(b)  $d_2 = 2Kd_1$

(c)  $d_2 = Kd_1$

[▷ Back to Course Notes](#)

### E.17. Time vs frequency domain beamforming

Discuss the differences between time domain and frequency domain beamforming. Discuss under what conditions one would use one in preference to the other.

[▷ Back to Course Notes](#)

### E.18. Time delays

•

$d$

•

$d$

•

Consider a 7-element receiver array aligned along the  $y$ -axis as illustrated.

$2d$

(1) Calculate the receiver time delays required to steer a beam in direction  $\theta$ .

•

(2) Write down an expression for the steering vector of phase delays.

$2d$

(3) Evaluate this steering vector for the broadside and end-fire directions when  $d/\lambda = 1/2$ , where  $\lambda$  is the wavelength.

•

$d$

•

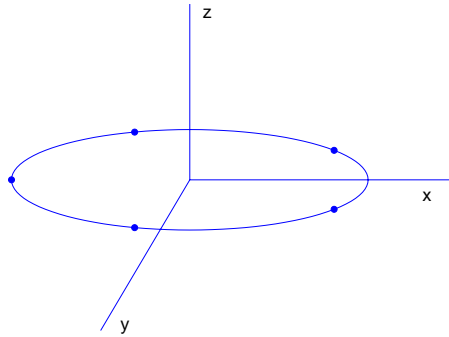
$d$

•

[▷ Back to Course Notes](#)

### E.19. Time delays for circular array

Consider a circular array of  $K$  receivers in the  $x - y$  plane.



Derive the time delays for a signal incident on the array from direction  $\theta, \phi$ .

▷ [Back to Course Notes](#)

### E.20. Circular array

Consider an array of seven omnidirectional receivers equally spaced around the circumference of a circle of radius  $r$  as illustrated below. If a plane wave is incident upon the array from a direction  $\theta_s$  within the plane of the array, derive expressions for the time delays required to add, in phase, the receiver outputs from this plane wave.

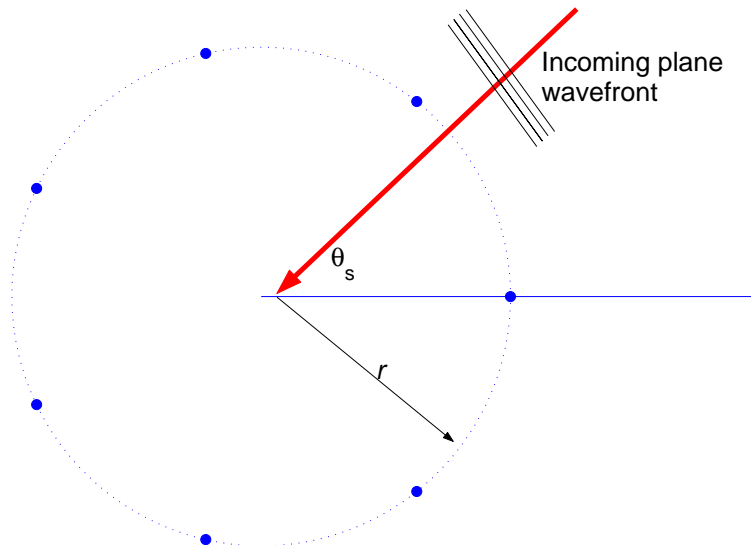


FIGURE E.2. Conventional beamformer – steered broadside

- (1) Write down the steering vector as a function of  $\theta_s$  and  $r/\lambda$ .

- (2) Derive an analytic expression for the beampattern of this array.

▷ [Back to Course Notes](#)

### E.21. Time sampling errors

Consider a linear array of  $K$  receivers equally spaced by a distance  $d$ . The receiver outputs are sampled simultaneously every  $T$  seconds. Beams are formed by time delay and summing (i.e., in the time domain).

- (1) Derive an expression for the steering angles at which no degradation of the beams occurs.
- (2) Find a relationship between  $d$  and  $T$  such that no spatial or temporal aliasing occurs, and discuss your result in the context of (2) above.

▷ [Back to Course Notes](#)

### E.22. Wavenumber

•  
 $2d$

•

$3d$

•

$2d$

•

$3d$

- (1) Plot the  $k_y$  (wavenumber) responses of the staggered array shown for  $d/\lambda = 1, 0.5$  and  $0.25$ .

•

- (2) Interpret and discuss their features.

$2d$

•

$3d$

•

$2d$

•

▷ Back to Course Notes

### E.23. Cross-spectral matrix

Consider the array shown.

(1) **Single arrival**

- $2d$

•

$3d$

•

$d$

•

(a) Write down the cross-spectral matrix for a narrow-band complex plane wave arriving from direction  $\theta_s$  and power at the receivers of  $\sigma_s^2$ . (Hint: consider the phase vector and the expression for the cross-spectral matrix in terms of this phase vector.)
- (b) Modify the cross-spectral matrix to incorporate identically distributed uncorrelated receiver noise of power  $\sigma_n^2$ .
- (c) Write down the expression for the steered beam-former mean output power as a function of the steering angle  $\theta$ .
- (d) For  $\theta_s = -45^\circ$  and  $d/\lambda = 0.5$ , plot this as a function of  $\theta$ ,  $-90^\circ \leq \theta \leq 90^\circ$ , and for signal-to-noise ratios  $(\sigma_s^2/\sigma_n^2)$  of  $-10, 0$  and  $+10\text{dB}$ .

(2) **Two arrivals.**

- (a) Repeat the above but with two incident plane waves which are statistically independent of one another, arriving from  $\theta_1 = 0^\circ$  and  $\theta_2 = 45^\circ$ .
- (b) Keeping the signal-to-noise ratios of both arrivals constant at  $+10\text{dB}$ , investigate and discuss the effect of varying  $\theta_2$ .
- (c) Keeping the signal strength of the first arrival  $\sigma_0^2$  constant at  $+10\text{dB}$ , and the direction of the second arrival  $\theta_2$  constant at  $45^\circ$ , investigate and discuss the effect of varying the signal-to-noise ratio  $\sigma_{45}^2$  of the second arrival.

▷ Back to Course Notes

### E.24. Gain of 3-receiver array

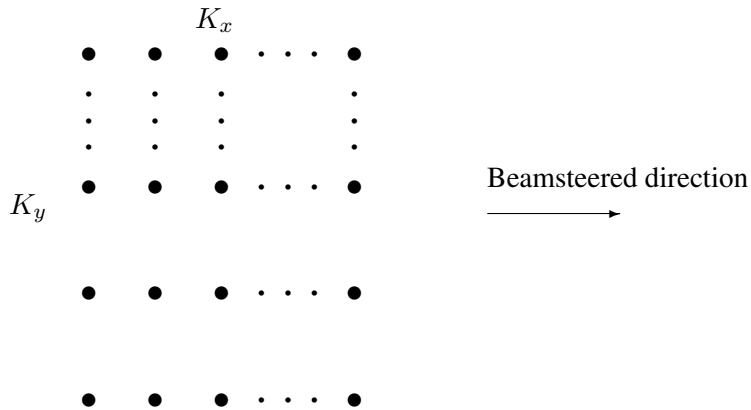
Consider a linear array with 3 receivers spaced  $d$  apart; a beam is steered at broad-side using conventional processing.

- (1) Derive an expression for the gain of the array in uncorrelated noise of power  $\sigma_n^2$  and a single interference from  $45^\circ$  with a power of  $\sigma_i^2$ .
- (2) Set the uncorrelated noise power to 0dB and the interference power to 10dB and plot the gain as a function of  $d/\lambda$  for a sensible range of  $d/\lambda$ .
- (3) Interpret and discuss your results.

▷ Back to Course Notes

### E.25. Array gain of rectangular array

Consider a rectangular array of  $K_y \times K_x$  receivers as shown, in the presence of uncorrelated receiver noise. There is no other noise present. A beam is steered in the horizontal direction.



- (1) Derive an expression for array gain as a function of the number of receivers  $K = K_x \times K_y$ .
- (2) Does varying the steering angle affect this result?

▷ Back to Course Notes

### E.26. Gain in isotropic noise

Consider the case of a linear array with  $K$  equally spaced receivers in a spherically isotropic noise field and steered end-fire using conventional beamforming. All other noise sources may be neglected. If the spacing between adjacent receivers is  $d = n\lambda/4$  where  $\lambda$  is the wavelength, show that the array gain is  $K$ .

▷ Back to Course Notes

### E.27. Null-steering 2

Consider an arbitrary array of  $K$  receivers operating in the frequency domain. Let  $P(\mathbf{k}, \mathbf{k}_s)$  be the beampattern of this array for conventional processing, where  $\mathbf{k}_s$  is the wavevector of the signal and  $\mathbf{k}$  is that of the beamsteered direction.

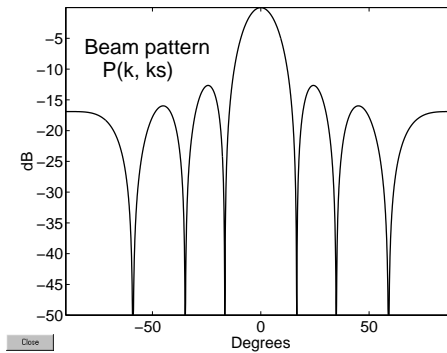
- (1) Write down *without proof* an expression for the gain of the array for a plane-wave signal with wavevector  $\mathbf{k}_s$ , a noise cross-spectral matrix  $\mathbf{R}_n$ , and a vector of beamformer weights  $\mathbf{w}(\mathbf{k}_s)$ .

---

Consider now the case in which all the noises are *independent* and the beamformer is required to steer a single null in a direction with wavevector  $\mathbf{k}_1$ .

- (2) Write down an expression for the weight vector  $\mathbf{w}(\mathbf{k})$  of this null-steering beamformer, and hence *prove* that the gain is

$$G_{nullsteer} = K(1 - P(\mathbf{k}_1, \mathbf{k}_s))$$



Beampattern of  
conventional beamformer  $P(\mathbf{k}, \mathbf{k}_s)$

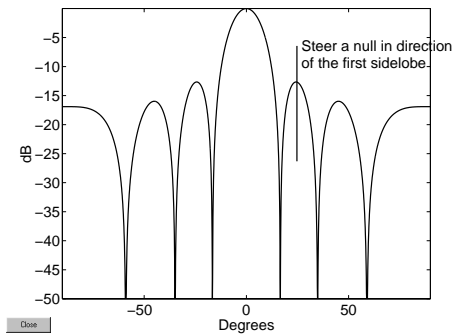
---

- (3) If the first sidelobe of  $P(\mathbf{k}, \mathbf{k}_s)$  has an amplitude of -10dB at  $\mathbf{k}_{sidelobe}$ , what are the array gains of

(a) the conventional beamformer and

(b) the null-steering beamformer

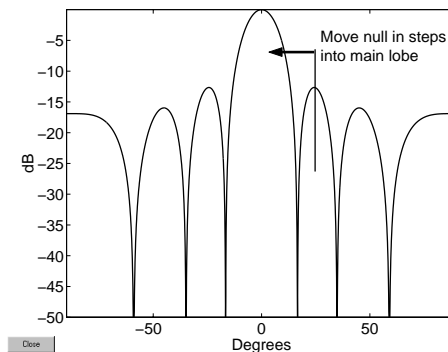
when the null is placed at the peak of that sidelobe ( $\mathbf{k}_1 = \mathbf{k}_{sidelobe}$ )?



Steering a single null at the first sidelobe of the conventional beamformer

---

- (4) *Without detailed derivation*, discuss what would be the effect on the gain of moving  $\mathbf{k}_1$  in successive steps into the main beam of  $P(\mathbf{k}, \mathbf{k}_s)$ ?



Moving the null towards and into the main lobe

---

- (5) If there is an interference of amplitude +30dB at  $\mathbf{k}_i = \mathbf{k}_{sidelobe}$  and a null is placed at  $\mathbf{k}_1 = \mathbf{k}_{sidelobe}$ , what would the gains of

- (a) the conventional and
- (b) null-steering beamformers

then be?

### E.28. Kantorovich inequality

The Kantorovich inequality states that, for a positive definite Hermitian symmetric  $[K \times K]$  matrix  $\mathbf{R}$  with eigenvalues

$$\{\lambda_{\max} \equiv \lambda_1 \geq \cdots \lambda_k \geq \lambda_K \equiv \lambda_{\min}\},$$

$$(\mathbf{x}^H \mathbf{x})^2 \leq (\mathbf{x}^H \mathbf{R} \mathbf{x})(\mathbf{x}^H \mathbf{R}^{-1} \mathbf{x}) \leq \frac{(\mathbf{x}^H \mathbf{x})^2}{4} \left\{ \left( \frac{\lambda_{\max}}{\lambda_{\min}} \right)^{1/2} + \left( \frac{\lambda_{\min}}{\lambda_{\max}} \right)^{1/2} \right\}^2. \quad (\text{E.1})$$

- (1) Show that the lower bound is attained when  $\mathbf{x} = \mathbf{p}_k$ , where  $\mathbf{p}_k$  is any eigenvector of  $\mathbf{R}$ .
- (2) Show that the upper bound is attained when

$$\mathbf{x} = \frac{(\mathbf{p}_{\max} + \mathbf{p}_{\min})}{\sqrt{2}},$$

where  $\mathbf{p}_{\max}$  and  $\mathbf{p}_{\min}$  are the eigenvectors of  $\mathbf{R}$  corresponding to the largest and smallest eigenvalues  $\lambda_{\max}$  and  $\lambda_{\min}$ , respectively.

- (3) Comment on whether in general  $\mathbf{p}_k$  can correspond to a realisable signal wavevector and take the form

$$\mathbf{p}_k \propto \mathbf{v}(\mathbf{k}_s) = [v_j(\mathbf{k}_s)] = [\exp i2\pi(\mathbf{k}_s^T \mathbf{u}_j)],$$

where  $\mathbf{u}_j$  is the vector of coordinates of the  $j^{\text{th}}$  receiver and  $\mathbf{k}_s$  is the signal wavevector.

▷ Back to Course Notes

### E.29. Spherically isotropic noise

Consider the case of a **linear** array of receivers equally spaced  $d$  apart, the ambient noise is spherically isotropic (and there are no interferences present) and the receiver noise is uncorrelated.

- (1) Write down an expression for the cross-spectral matrix of the noise  $\mathbf{R}_n(f)$  as a function of  $d/\lambda$ , where  $\lambda$  is the wavelength.



- (2) Give a situation in which the performance of the optimal beamformer will be identical to that of the conventional when the beamsteered direction is broadside.
- (3) In the situation you have just given above, is there any difference when the beamsteered direction is end-fire.

▷ Back to Course Notes

### E.30. Correlated noise field

Consider a linear array of  $K$  equi-spaced receivers in a noise field with correlation function of the form  $r_n(s) = \exp -(|\alpha s|)$ , where  $s$  is the distance between any two points in the array[?].

- (1) Show that the cross-spectral matrix takes the form

$$\mathbf{R}_n = \sigma_n^2 \begin{bmatrix} 1 & a & a^2 & \dots & a^{K-1} \\ a & 1 & a & & \vdots \\ a^2 & a & 1 & & \\ \vdots & & & \ddots & a \\ a^{K-1} & \dots & & a & 1 \end{bmatrix}.$$

- (2) Demonstrate that for  $K = 4$  the inverse is

$$\mathbf{R}_n^{-1} = \frac{1}{\sigma_n^2(1 - a^2)} \begin{bmatrix} 1 & -a & 0 & 0 \\ -a & 1 + a^2 & -a & 0 \\ 0 & -a & 1 + a^2 & -a \\ 0 & 0 & -a & 1 \end{bmatrix}.$$

- (3) What are the optimal (MVDR) weights when the beamsteered direction is broadside and
- (a) the noise field is spatially uncorrelated ( $a = 0$ ),
  - (b) partially correlated ( $a = 0.5$ ) and
  - (c) highly correlated ( $a = 1 - \epsilon, \epsilon \ll 1$ )?

- (4) What is the optimal array gain as a function of  $a$ ?

▷ Back to Course Notes

### E.31. Array position errors

You have a linear array of receivers, equally spaced half a wavelength apart,

There is a strong interfering signal arriving as a plane wavefront from a distant source.

You have designed two processors, both intended to detect a signal arriving from broadside:

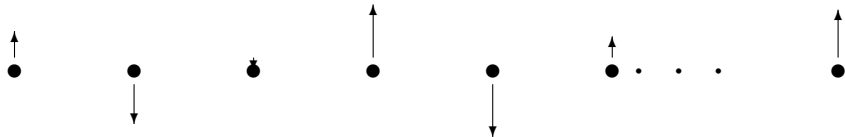
- a conventional, unshaded beamformer, and
- an optimal (MVDR) processor.

- (1) First consider the situation in which the desired signal is in the far field and arrives as a plane wavefront. Unknown to you, some receivers are slightly displaced from their correct positions, all randomly left or right *along the axis of the array* as illustrated below.



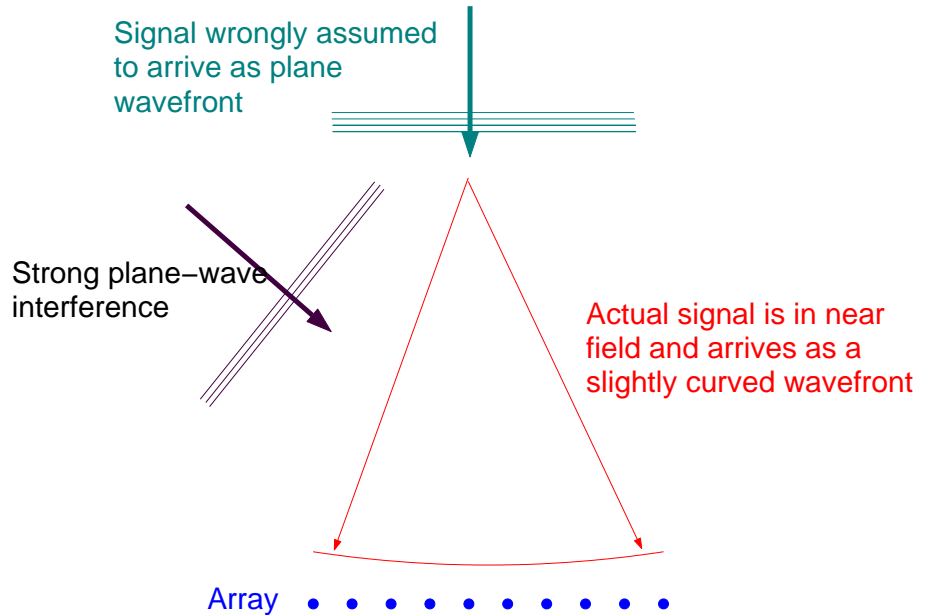
What would you expect the effect of these displacements to be on the array gain for the conventional and optimal beamformers? Briefly explain your reasoning.

- (2) Next take the case in which you again have random displacements of the receivers, but all up or down as illustrated below.



Again, what would you expect the effect of these displacements to be on the array gain for the conventional and optimal beamformers? Briefly explain your reasoning.

- (3) Finally, consider the situation in which you thought the desired signal was from a distant source, arriving as a plane wavefront and designed your processors accordingly. But instead – and unknown to you – it is in fact from a nearby source that arrives as a slightly curved wavefront, as illustrated below.



Which of the processors would be more affected by the curvature of the wavefront, and why?

▷ [Back to Course Notes](#)

### E.32. Optimal beamformer

Consider the steered beamformer output for a *general* array of  $K$  receivers, using the *exact* cross-spectral matrix of the receiver outputs  $\mathbf{R}_x(f)$ .

Take the situation in which there is present uncorrelated receiver noise and several plane-wave interferences, and the level of ambient noise is negligible.

- (1) Give an example in which the optimal beamformer will provide a much larger array gain than the conventional beamformer.
- (2) Comment on the computational precision required in this case.
- (3) How will the gain of the optimal beamformer compare with that of the conventional
  - (a) when the beamsteered direction is pointed directly at a strong interference,
  - (b) when the (uncorrelated) receiver noise is much stronger than the interferences?

- (4) Next consider the practical case in which  $\mathbf{R}_x(f)$  has to be *estimated* from  $M$  observations of the receiver outputs  $\tilde{\mathbf{x}}(f)$ . State a situation in which the optimal beamformer will give little improvement in array gain over the conventional when using the estimated matrix  $\hat{\mathbf{R}}_x(f)$ .

▷ [Back to Course Notes](#)

## Bibliography

- [1] H Akaike. A new look at the statistical model identification problem *IEEE Trans Auto. Control*, AC-19:716, Dec 1974
- [2] V Anderson. DICANNE, a Realizable Adaptive Process *Journal of the Acoustical Society of America* 45:2 1969
- [3] EF Beckenbach and R Bellman. *Inequalities*, Springer-Verlag 1965, p70
- [4] SD Bedrosian. Nonuniform linear arrays: Graph theoretic approach to minimum redundancy, *Proc. IEEE*, 74:1040-1043, July 1986.
- [5] DH Brandwood. A complex gradient operator and its application to adaptive array theory, *Proc IEE* , Vol 130, Pts F & H, No. 1, Feb 1983
- [6] J Capon. High-resolution frequency-wavenumber spectrum analysis. *Proc IEEE*, 57:1408, 1969.
- [7] J Capon and NR Goodman. Probability distributions for estimators of the frequency-wavenumber spectrum. *Proc IEEE* 58:1786, Oct 1970
- [8] GE Carlson. *Signal and Linear Systems Analysis* John Wiley & Sons, 1998
- [9] HA d'Assumpcao. Some New Signal Processors for Arrays of Sensors *Trans IEEE Information Theory*, IT-26.4: 441-453 1980
- [10] DJ Edelblute, JM Fisk and GL Kinnison. Criteria for Optimum-Signal-Detection Theory for Arrays, *Journal of the Acoustical Society of America* 41:1 1967
- [11] RS Elliott. *Antenna theory and design*, Prentice-Hall, Inc., NJ 1981
- [12] A Farina. *Antenna-Based Signal Processing Techniques for Radar Systems* Artech House Books 1992
- [13] N Fourikis. *Advanced Array Systems, Applications and RF Technologies* Academic Press 2000
- [14] OL Frost III. An Algorithm for Linearly Constrained Adaptive Array Processors, *Proc IEEE* 60:926 1972
- [15] GH Golub & CF Van Loan. An analysis of the total least squares problem, *SIAM J. Numerical Analysis* 17:883 1980
- [16] LJ Griffiths. A Simple Adaptive Algorithm for Real-Time Processing in Antenna Arrays, *Proc IEEE* 57:1696-1704 1969
- [17] FJ Harris. On the use of windows for harmonic analysis with the discrete Fourier transform, *Proc IEEE* 66:51, Jan 1978
- [18] RA Haubrich. Array design, *Bull. Seismological Soc. Am.*, 58:977, 1968
- [19] S Haykin. *Array Processing* Dowden, Hutchinson and Ross Inc, 1990
- [20] DH Johnson and DE Dudgeon. *Array Processing* Prentice Hall, 1993
- [21] DH Johnson and S DeGraaf. Improving the resolution of bearing in passive sonar arrays by eigenvalue analysis. *IEEE Trans Acoustics, Speech and Signal Processing*, ASSP-30:638, Aug 1982
- [22] SM Kay. *Fundamentals of Statistical Signal Processing—Estimation Theory*, Prentice Hall 1993
- [23] R Klemm. *Principles of Space-Time Adaptive Processing* IEE-UK 2002
- [24] JD Kraus. *Antennas* McGraw-Hill International Editions, 1988
- [25] J Litva & T Lo. *Digital Beamforming in Wireless Communications* Artech House Books 1996
- [26] RJ Mailloux. *Phased Array Antenna Handbook* Artech House 1993
- [27] RA Monzingo and TW Miller. *Introduction to Adaptive Arrays* Wylie Interscience, New York, 1980

- [28] E Nicolau and D Zaharia. *Adaptive Arrays* Vol 35 of Studies in Electrical and Electronic Engineering, Elsevier Science Publishers, New York, 1989
- [29] A Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 1984
- [30] A Paulraj, R Roy and T Kailath *A subspace rotation approach to signal parameter estimation*. Proc IEEE, 74.7:1044, Jul 1986
- [31] SU Pillai. *Array Signal Processing* Springer-Verlag 1989
- [32] VF Pisarenko. The retrieval of harmonics from a covariance function. Geophys. J. Roy. Astron. Soc., 33:347 1973.
- [33] CR Rao. *Linear Statistical Inference and its Applications*, John Wiley & Sons, 2nd ed., p.74
- [34] CR Rao and Mitra. *Generalized Inverse of Matrices and its Applications* John Wiley 1971
- [35] J Rissanen. Modeling by shortest data description, *Automatica*, 14:465, Sept 1978
- [36] RO Schmidt. Multiple Emitter Location and Signal Parameter Estimation Trans IEEE on Antennas and Propagation, AP-34:3, 1986
- [37] BD Steinberg. *Principles of Aperture and Array System Design* John Wiley & Sons, New York, 1976
- [38] CW Therrien. *Discrete Random Signals and Statistical Signal Processing* Prentice-Hall 1988
- [39] HL Van Trees. *Detection, Estimation and Modulation Theory Part I* John Wiley & Sons 1968
- [40] HL Van Trees. *Optimum Array Processing (Detection, Estimation and Modulation Theory Part IV)* John Wiley & Sons 2002
- [41] M Wax and T Kailath. Detection of Signals by Information Theoretic Criteria Trans IEEE on Acoustics, Speech and Signal Processing, Vol ASSP-33, No. 2, April 1985
- [42] B Widrow, PE Mantey, LJ Griffiths and BB Goode. Adaptive Antenna Systems Proc IEEE 55:2143-2159 (1967)

## List of symbols

Symbol	Definition	Page
$\hat{\cdot}$	estimator	??
$\alpha_j$	weighting coefficients for shading	??
$\alpha(\mathbf{k}, \omega)$	complex amplitude at the origin	??
$\beta$	interference-to-receiver noise ratio	??
$\Delta T$	sampling interval	??
$\delta_1^{(L+1)}$	the $((L + 1) \times 1)$ column vector with all entries equal to zero except the first which is equal to 1	??
$\kappa$	forgetting factor	??
$\lambda$	wavelength	??
$\lambda'$	component of the wavelength of the incoming signal, resolved along the length of the linear array	??
$\theta_s$	horizontal angle of signal	??
$\phi_s$	vertical angle of signal	??
$\sigma_s^2(f)$	power of the incident signal at frequency $f$	??
$\tau_j(\theta)$	time delay for the $j^{\text{th}}$ receiver to steer the beam in the direction $\theta$	??
$\chi_{2M}^2$	Chi-square distribution with $2M$ degrees of freedom	??
$\mathbf{A}$	$2 \times K$ matrix formed from the signal and interference vectors	??
$c$	speed of propagation in the medium	??
$d$	inter-element spacing of a uniform linear array of receivers	??
$E\{\cdot\}$	Expectation	??
$f$	frequency	??
$f(\mathbf{u}, t)$	propagating field	??
$G$	Array gain	??
$G_{\text{conv}}$	Gain of the conventional beamformer	??
$G_{\text{max gain}}$	maximum gain	??

Symbol	Definition	Page
$J_0$	zero-th order spherical Bessel function	??
$K$	number of receivers in the array	29
$\mathbf{k}_s$	signal wavevector	??
$\mathbf{k}_i$	interference wavevector	??
$(\mathbf{k}_s)_y$	component of wave-vector along axis of linear array	??
$k_y$	spatial frequency	91
$L$	number of signal sources	??
$\tilde{\mathbf{n}}(f)$	noise vector	??
$P(\theta, \theta_s)$	beampattern of array steered in direction $\theta$	??
$P(k_y, (\mathbf{k}_s)_y)$	Beampattern of array steered in wavevector direction $k_y$ as a function of wavevector component $(\mathbf{k}_s)_y$	91
$p_{\text{conv}}(\mathbf{k})$	mean output power from the array	??
$p_n$	output power of the array in the absence of signal	??
$p_s$	array output power in the absence of noise	??
$p_{\mathbf{X}}(\mathbf{x}(t))$	joint p.d.f. of receiver outputs	??
$p(x_i(t_1), x_j(t_2))$	joint p.d.f of $x_i(t_1)$ and $x_j(t_2)$	??
$\mathbf{R}_s(f)$	cross-spectral matrix of signal	??
$\mathbf{R}_n(f)$	cross-spectral matrix of noise	105
$\mathbf{R}_{\text{iso}}$	cross-spectral matrix of spherically isotropic noise	??
$\mathbf{R}_{2\text{d-iso}}$	cross-spectral matrix of 2-dimensional isotropic noise	??
$\mathbf{R}_{X_i, X_j}(t_1, t_2)$	cross covariance function	??
$\mathbf{R}_x(t_1, t_2)$	covariance matrix	??
$SNR$	signal-to-noise ratio	??
$SNR_{\text{beam}}$	signal-to-noise ratio out of beamformer	??
$SNR_{\text{receiver}}$	signal-to-noise ratio at a single receiver	??



Symbol	Definition	Page
$\tilde{\mathbf{s}}(\mathbf{k}_s, f)$	signal vector	??
$s_j(t, \theta_s)$	signal at the $j^{\text{th}}$ receiver	??
$s(t)$	output of the receiver located at the origin	??
$\mathbf{s}(t)$	vector of receiver signal outputs	??
$T_{\text{gap}}$	interval between blocks of data	??
$\text{Tr}(\cdot)$	trace of a matrix	??
$\mathbf{u}$	vector of coordinates of a point in space	??
$\mathbf{v}(\theta_s)$	steering vector	??
$\mathbf{w}$	generalised weighting vector	??
$\mathbf{w}_{\text{Capon}}(\mathbf{k})$	weights for maximum likelihood (Capon) estimator	??
$\mathbf{w}_{\text{max gain}}$	weights that maximise gain	??
$\mathbf{w}_{\text{MMSE}}(\mathbf{k})$	weights for minimum mean-square error processor	??
$\mathbf{w}_{\text{MVDR}}(\mathbf{k})$	weights for minimum variance distortionless response (MVDR) processor	??
$x_j(t)$	continuous time output of the $j^{\text{th}}$ receiver	??
$\mathbf{x}(t)$	vector of time output of receivers	??
$\tilde{\mathbf{x}}(f)$	vector of Fourier transformed receiver outputs	??
$y(t)$	time series of the output of the array processor	??
$y(\theta, \theta_s, t)$	output of the array processor	??
$\tilde{y}(\theta, \phi, f)$	continuous Fourier transform of a beam steered in direction $\{\theta, \phi\}$	??
$\tilde{y}(\theta, f)$	output of a frequency-domain beam steered in direction $\theta$	??