

Acoustic Vector-Sensor Array Performance

by

Jonathan Paul Kitchens

B.S., Duke University (2004)

Submitted to the

Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2008

© Massachusetts Institute of Technology 2008. All rights reserved.

Author
Department of Electrical Engineering and Computer Science
May 6, 2008

Certified by
Arthur B. Baggeroer
Ford Professor of Engineering
Secretary of the Navy/Chief of Naval Operations
Chair for Ocean Science
Thesis Supervisor

Accepted by
Professor Terry P. Orlando
Chairman, Department Committee on Graduate Students

Acoustic Vector-Sensor Array Performance

by

Jonathan Paul Kitchens

Submitted to the Department of Electrical Engineering and Computer Science
on May 6, 2008, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

Classical hydrophones measure pressure only, but acoustic vector-sensors also measure particle velocity. Velocity measurements can increase array gain and resolve ambiguities, but make vector-sensor arrays more difficult to analyze. This thesis derives a new set of useful performance measures for acoustic vector-sensor arrays. It characterizes the vector-sensor array beampattern with and without modeling errors, or “mismatch.” It also develops a hybrid Cramér-Rao bound for direction-of-arrival estimation under mismatch. The results are analyzed, compared to Monte-Carlo simulations, and explored for insight.

Thesis Supervisor: Arthur B. Baggeroer
Title: Ford Professor of Engineering
Secretary of the Navy/Chief of Naval Operations
Chair for Ocean Science

Contents

1	Introduction	9
1.1	Background	9
1.2	Motivation	11
1.3	Outline	11
1.4	Sensor and Environment Model	12
1.5	Plane Wave Measurement Model	13
2	Ideal Vector-Sensor Arrays	17
2.1	General Response Pattern	17
2.2	Uniform Linear Array (ULA)	22
2.2.1	Beampattern	22
2.2.2	Conical Angle (Left/Right) Discrimination	26
2.2.3	Spatial Aliasing (Grating Lobe) Discrimination	27
2.2.4	Near-Field Processing	31
3	Mismatched Vector-Sensor Arrays	33
3.1	Perturbed Linear Array	34
3.1.1	Nonuniform Sampling	35
3.1.2	Relation to Perturbed ULA	35
3.1.3	Analysis of the Expected Beampattern	38
3.2	Cramér-Rao Bound	42
3.2.1	Statement of the Hybrid Bound	42
3.2.2	Evaluation of Terms	46

3.2.3	Including Gain and Phase Errors in the CRB	54
3.2.4	The Maximum Likelihood (ML) Estimator	56
3.2.5	Analysis and Examples	59
4	Conclusion	67
4.1	Summary	67
4.2	Future Work	68
A	Nomenclature	69
A.1	Acronyms	69
A.2	Notation	70

List of Figures

2.1.1 Unit vectors \mathbf{u} and $\hat{\mathbf{u}}$	20
2.1.2 Vector-sensor modulation term B_v	21
2.2.1 ULA Beampattern Components: $\hat{\phi} = \pi/2$	23
2.2.2 ULA Beampattern Components: $\hat{\phi} = \pi/4$	24
2.2.3 ULA Beampatterns: $\hat{\phi} = \pi/2$ and $\pi/4$	25
2.2.4 Vector-sensor “Grating Lobe” Height: $f/f_d = 4$	29
2.2.5 Vector-sensor “Grating Lobe” Height: $\hat{\phi} = \pi/2$	30
2.2.6 Vector-sensor “Grating Lobe” Height Contours	30
3.1.1 Illustration of Nonuniform Spatial Sampling	39
3.1.2 Mismatched Beampattern: $f = 2f_d$	40
3.1.3 Mismatched Beampattern: $f = f_d/2$	41
3.2.1 Orthogonal Vectors $\{\mathbf{u}, \boldsymbol{\delta}_\phi, \boldsymbol{\delta}_\psi\}$	48
3.2.2 CRB with Gain and/or Phase Errors	61
3.2.3 CRB with Rotation and/or Position Errors	61
3.2.4 Pressure and Vector-Sensor Arrays: CRB vs. SNR	63
3.2.5 Pressure and Vector-Sensor Arrays: CRB vs. Snapshots	63
3.2.6 Pressure-Sensor Array Algorithm Performance	65
3.2.7 Vector-Sensor Array Algorithm Performance	65

Chapter 1

Introduction

Most introductions to array processing characterize the performance of classical hydrophone arrays. Texts commonly derive and interpret performance measures that prove useful in both theory and practice. Acoustic vector-sensors, however, are more difficult to evaluate because they measure particle velocity in addition to pressure.

This thesis develops and analyzes performance measures for acoustic vector-sensor arrays. It organizes into two logical parts: 1) performance measures and insights under correct modeling and 2) performance analysis and bounds under Gaussian modeling errors. For those unfamiliar with sonar array processing or vector-sensors, the following two sections give background material and motivate this research.

1.1 Background

The principles that have historically driven passive sonar research are the same ones motivating this work. To fully understand the logic behind this thesis, then, one needs some background in undersea surveillance.

For readers new to the ocean environment, I must first explain the preference for passive sonar. Most detection systems in air or free space use electromagnetic waves, but these generally absorb quickly in salt water. Sound waves, however, can travel great distances - sometimes thousands of miles - underwater. They are reflected by objects and produced by machinery, making sound useful for active and passive

detection. Undersea surveillance applications often reject active sonar for two reasons. First, active sonar pulses travel beyond the maximum detection range. Targets can intercept these pulses at great distance and avoid detection. Second, active sonar is not covert. Sources transmitting active sonar may sometimes be located and classified easily.

The most common sensor employed for sonar is the hydrophone. Essentially an underwater microphone, hydrophones measure pressure only. Sound waves passing over a hydrophone introduce changes in pressure that are measured and used for detection. Omnidirectional hydrophones are common because they are easy to build, maintain, and analyze. Decades of experience with hydrophones show they survive well in the corrosive ocean environment and can easily be assembled into arrays.

The most common sensor configuration is the uniformly spaced linear array. Linear arrays may be fixed to the side of a ship, mounted on the sea floor, or towed behind a moving vessel. When a vessel travels in a straight line, drag pulls a towed array into a roughly linear shape. The exact sensor locations and orientations, however, are usually unknown.

Given a configuration of sensors, performance can be readily improved by increasing the information measured by each sensor. For acoustic measurements, particle velocity provides additional information about the direction of sound arrival. Acoustic vector-sensors each contain one omnidirectional hydrophone measuring pressure and three orthogonal geophones measuring the components of particle velocity.¹ One common geophone structure is a tube containing a magnetic mass suspended by springs. Any vibration along the axis of the tube causes the mass to move, inducing a current in a wire coil. This induced current yields a measurement of velocity along the geophone axis.

¹My analysis uses this description although some vector-sensors equivalently use accelerometers or directional hydrophones.

1.2 Motivation

The previous section motivated the use of acoustic vector-sensors for undersea surveillance; this section outlines the need for useful vector-sensor array performance measures. As vector-sensors become more common, the need for useful design and analysis tools increases.

Although geological vector-sensors have existed for decades, recent advances in geophone design have increased their utility in sonar applications. Because they provide more information per sensor, vector-sensor arrays will likely play a larger role in the future of sonar. The ability of engineers to exploit vector-sensors will depend largely on how well their performance is understood.

When designing or analyzing pressure-sensor arrays, engineers use established metrics like response pattern, beamwidth, gain, and design frequency. Such measures are useful because they appear often in theory and are robust in practice. Together with recent work in [1, 2, 3, 4], this thesis provides a set of similar tools for vector-sensor arrays. Such tools give insight into the design and analysis of these arrays both in theory and practice.

Although this thesis should be taken in context with these references, its contributions are unique. Unlike [3], the second chapter of this work studies the vector-sensor beampattern under simple models to gain intuition and insight. Also, the beampattern expression derived in [3] is distinct from the direction-of-arrival performance bound developed here. The Cramér-Rao bounds derived in [1] and [2] apply only to vector-sensor arrays without mismatch and are therefore special cases of the bound presented here.

1.3 Outline

This thesis is a discussion in two logical parts. First, it covers vector-sensor arrays that are correctly modeled, or “ideal.” This topic leads logically into the second discussion on “mismatched” vector-sensor arrays. The document is presented in four

chapters:

1. Chapter 1 forms introductory material. It includes a description of the topic, the relevance of my research, and background material including the measurement model used for the rest of the thesis.
2. Chapter 2 explores the properties of vector-sensor arrays whose position and orientation are known. It derives several formulas relating vector-sensor arrays to classical pressure-sensor arrays, specializing the results for the uniform linear array.
3. Chapter 3 performs a detailed analysis of “mismatched” vector-sensor arrays whose position and orientation are stochastic. First, it considers the simple randomly perturbed linear array. Second, it explores a more complex model with Gaussian position and orientation errors. Using this model, it derives a Cramér-Rao bound for direction-of-arrival estimation.
4. Chapter 4 concludes the thesis, evaluating this work and its contribution. This chapter also discusses implications of the results and additional areas of potential work.

1.4 Sensor and Environment Model

To simplify discussion, this entire document assumes the same basic sensor and environment model. Each section explicitly notes any departures from or extensions to this common model. The subsequent analysis assumes the following sensor model:

1. *Co-located sensor components.* The hydrophone and three geophones of each vector-sensor are located at the same point and observing the same state. In practice, this requires the component spacing to be small compared with the minimum wavelength (set by the highest operating frequency).
2. *Point sensors.* Each vector-sensor is modeled as a single point. In practice, this requires the sensor dimensions to be small compared with the minimum

wavelength.

3. *Geophones with cosine response.* The signal response of each geophone is proportional to the cosine of the angle between the geophone axis and the source. Cosine geophone response results from measuring velocity along only one axis.
4. *Orthogonal geophones.* The axes of the three geophones are orthogonal. In practice, this is true when each vector-sensor is a static unit.

The thesis also assumes the following environment model:

1. *Free-space environment.* Sound waves travel in a quiescent, homogeneous, isotropic fluid wholespace. This implies direct-path propagation only.
2. *Narrowband signals.* The signal is analyzed at a single frequency. In practice, this means the signal is sufficiently band-limited to allow narrowband processing in the frequency domain. Such a band-limited signal may be obtained by pre-filtering or computing the DFT of the measurements.
3. *Plane wave propagation.* The sound waves are planar at each sensor and across the array. This implies the unit vector from each sensor to the “source” is the same, regardless of the sensor location. In practice, it requires far-field sources whose distance is much greater than the length of the array and the maximum wavelength.

The underlying assumptions and notation are similar to those in [1, 2, 5], although this document has a different objective.

1.5 Plane Wave Measurement Model

Under the assumptions in Section 1.4, I consider a plane wave parameterized by azimuth $\phi \in [0, 2\pi)$ and elevation $\psi \in [-\pi/2, \pi/2]$ impinging on an array of M vector sensors. When necessary I use the right-handed coordinate system with $\phi = 0$ as forward endfire, $\phi = \pi/2$ as port broadside, $\psi = 0$ as zero elevation, and $\psi = \pi/2$

as upward. For notational convenience, I group the parameters ϕ and ψ into the vector Θ . Without loss of generality, I assume the geophone axes are the axes of the coordinate system. If this is not the case, the data from each vector sensor may be rotated to match the coordinate axes. I also define

$$\mathbf{u} = [\cos \phi \cos \psi, \sin \phi \cos \psi, \sin \psi]^T \quad (1.5.1)$$

as the unit length vector pointing from the origin to the source (or, opposite the direction of the wave propagation). The following derivations touch only briefly on direct-path acoustic propagation. For a much more detailed study of ocean acoustics, see [6].

I first derive an equation relating pressure and particle velocity. Assuming an inviscid homogeneous fluid, the Navier-Stokes equations become the Euler equations

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}^T \nabla \mathbf{v} = -\frac{\nabla p}{\rho} \quad (1.5.2)$$

where \mathbf{v} is fluid velocity, ρ is density, and p is pressure. For acoustic propagation this equation is linearized, neglecting the convective acceleration term $\mathbf{v}^T \nabla \mathbf{v}$. With a plane wave, the pressure p relates across time t and position \mathbf{x} by the sound speed c :

$$p(\mathbf{x}, t) = f\left(\frac{\mathbf{u}^T \mathbf{x}}{c} + t\right) \quad (1.5.3)$$

$$\therefore \nabla p = \frac{\mathbf{u}}{c} \cdot \frac{\partial p}{\partial t}. \quad (1.5.4)$$

Substituting Equation 1.5.4 into the Euler equations in 1.5.2, it can be shown that under weak initial conditions the pressure and fluid velocity obey the plane wave impedance relation

$$\mathbf{v} = -\frac{\mathbf{u}}{\rho c} p. \quad (1.5.5)$$

Because the geophones are aligned with the coordinate axes, they simply measure the components of the velocity vector \mathbf{v} . The resulting linear relationship between the

pressure and each component of the fluid velocity greatly simplifies the analysis of vector-sensor array performance.

This linear relationship proves most useful by allowing me to express the velocity measurements in terms of pressure and the source unit vector. Returning to the array of M vector-sensors, I now write the measurement of the k th vector sensor in phasor form as

$$e^{j2\pi(\mathbf{r}_k^T \mathbf{u})} \begin{bmatrix} 1 \\ -\mathbf{u}/\rho c \end{bmatrix} \quad (1.5.6)$$

where \mathbf{r}_k is the position of the sensor *in units of wavelengths*. Measuring distance in wavelengths simplifies many expressions and is used often in the following sections. The term outside the vector is the wave phase delay, which factors out because of Equation 1.5.5. In practice, only the gain difference between the pressure sensors and geophones is important. For convenience, I choose a normalization that absorbs that gain difference into the pressure term:

$$e^{j2\pi(\mathbf{r}_k^T \mathbf{u})} \begin{bmatrix} \eta \\ \mathbf{u} \end{bmatrix}. \quad (1.5.7)$$

Although this choice of normalization seems arbitrary, it results in simpler expressions later and is similar to the notation used in [1, 2, 5].² Also note that this choice of normalization requires a factor of $(\rho c)^{-2}$ when comparing beam estimates in units of power.

²My η is not exactly the same as the one used in [1, 2, 5].

Chapter 2

Ideal Vector-Sensor Arrays

This chapter explores the theoretical performance of acoustic vector-sensor arrays whose position and orientation are known. For these idealized arrays, it develops connections to classical pressure-sensor arrays. The goal of this chapter is to develop a design and analysis framework for vector-sensor arrays that mirrors the classical results for pressure-sensor arrays.

When analyzing the behavior of a sensor array, one of the most fundamental expressions is the array directional response or “beam pattern”. In its most general form, the directional response gives the output power of a spatial matched filter or beamformer as a function of source location. The following sections derive the response pattern for an arbitrary array and analyze the resulting expression for the uniform linear array.

2.1 General Response Pattern

Applying the measurement model specified in Chapter 1, I now examine the entire array of M arbitrarily placed vector-sensors. With the measurement from each sensor

given in expression 1.5.7, the measurement from the array is

$$\begin{aligned} \mathbf{v}(\boldsymbol{\Theta}) &= \left[e^{j2\pi(\mathbf{r}_1^T \mathbf{u})}, e^{j2\pi(\mathbf{r}_2^T \mathbf{u})}, \dots, e^{j2\pi(\mathbf{r}_M^T \mathbf{u})} \right]^T \otimes \begin{bmatrix} \eta \\ \mathbf{u} \end{bmatrix} \\ &\triangleq \mathbf{a}_p \otimes \mathbf{h} \end{aligned} \quad (2.1.1)$$

where \otimes represents the Kronecker, or tensor, product. Note again that the sensor positions \mathbf{r}_m are in units of wavelengths. Equation 2.1.1 factors the $4M \times 1$ measurement vector into a $M \times 1$ phase vector \mathbf{a}_p and a 4×1 vector-sensor component vector \mathbf{h} . A few things are worth noting about this expression. First, this factorization is possible because of Equation 1.5.5 and because I have chosen a common orientation for each vector-sensor. Second, the phase vector \mathbf{a}_p is simply the measurement vector for the corresponding pressure-sensor array.

Having specified a measurement vector, I now explore the conventional beamforming (CBF) “beampattern” for this array. Conventional beamforming is essentially spatial matched filtering, in this case with a signal parameterized by $\boldsymbol{\Theta}$ and a hypothesis parameterized by $\hat{\boldsymbol{\Theta}}$. Operating in the frequency domain, CBF computes the normalized inner product

$$y(\hat{\boldsymbol{\Theta}}, \boldsymbol{\Theta}) \triangleq \frac{\mathbf{v}(\hat{\boldsymbol{\Theta}})^H \mathbf{v}(\boldsymbol{\Theta})}{\mathbf{v}(\hat{\boldsymbol{\Theta}})^H \mathbf{v}(\hat{\boldsymbol{\Theta}})}, \quad (2.1.2)$$

the response of a spatial matched filter to the signal $\boldsymbol{\Theta}$. This thesis uses H to denote the Hermitian, or conjugate, transpose. Normalizing the inner product produces a peak value of unity when the hypothesis is exactly correct. In practice, the normalization constant absorbs into the hypothesized measurement replica, defining a weight vector

$$\mathbf{w}(\hat{\boldsymbol{\Theta}}) \triangleq \frac{\mathbf{v}(\hat{\boldsymbol{\Theta}})}{\mathbf{v}(\hat{\boldsymbol{\Theta}})^H \mathbf{v}(\hat{\boldsymbol{\Theta}})}. \quad (2.1.3)$$

For this work, I am only interested in the magnitude squared of the response, $B(\hat{\boldsymbol{\Theta}}, \boldsymbol{\Theta}) \triangleq |y(\hat{\boldsymbol{\Theta}}, \boldsymbol{\Theta})|^2$, called the beampattern. In the following sections, I use the shorter nota-

tion

$$y = \frac{\hat{\mathbf{v}}^H \mathbf{v}}{\hat{\mathbf{v}}^H \hat{\mathbf{v}}}, \quad \hat{\mathbf{w}} = \frac{\hat{\mathbf{v}}^H}{\hat{\mathbf{v}}^H \hat{\mathbf{v}}}, \quad B = |y|^2, \quad \text{etc.} \quad (2.1.4)$$

where the dependence on $\hat{\Theta}$ and Θ is implied.

I now compute the beampattern function from the definitions given above. I first simplify the numerator of y using the mixed-product property of the \otimes operator:

$$\begin{aligned} \hat{\mathbf{v}}^H \mathbf{v} &= (\hat{\mathbf{a}}_p \otimes \hat{\mathbf{h}})^H (\mathbf{a}_p \otimes \mathbf{h}) \\ &= (\hat{\mathbf{a}}_p^H \mathbf{a}_p) \otimes (\hat{\mathbf{h}}^H \mathbf{h}) \\ &= (\hat{\mathbf{a}}_p^H \mathbf{a}_p) \cdot (\hat{\mathbf{h}}^H \mathbf{h}). \end{aligned} \quad (2.1.5)$$

I then substitute this result to write the response expression as

$$\begin{aligned} y &= \frac{(\hat{\mathbf{a}}_p^H \mathbf{a}_p) \cdot (\hat{\mathbf{h}}^H \mathbf{h})}{(\hat{\mathbf{a}}_p^H \hat{\mathbf{a}}_p) \cdot (\hat{\mathbf{h}}^H \hat{\mathbf{h}})} \\ &= y_p \cdot \frac{\hat{\mathbf{h}}^H \mathbf{h}}{\hat{\mathbf{h}}^H \hat{\mathbf{h}}} \end{aligned} \quad (2.1.6)$$

where y_p is the response of the corresponding pressure-sensor array. I further simplify the vector sensor component

$$\begin{aligned} \hat{\mathbf{h}}^H \mathbf{h} &= \begin{bmatrix} \eta \\ \hat{\mathbf{u}} \end{bmatrix}^H \begin{bmatrix} \eta \\ \mathbf{u} \end{bmatrix} \\ &= \eta^2 + \hat{\mathbf{u}}^H \mathbf{u} \\ &= \eta^2 + \cos(\theta) \end{aligned} \quad (2.1.7)$$

$$\begin{aligned} \hat{\mathbf{h}}^H \hat{\mathbf{h}} &= \eta^2 + \hat{\mathbf{u}}^H \hat{\mathbf{u}} \\ &= \eta^2 + 1 \end{aligned} \quad (2.1.8)$$

where θ is the angle, not necessarily in the x - y plane, between the vectors specified

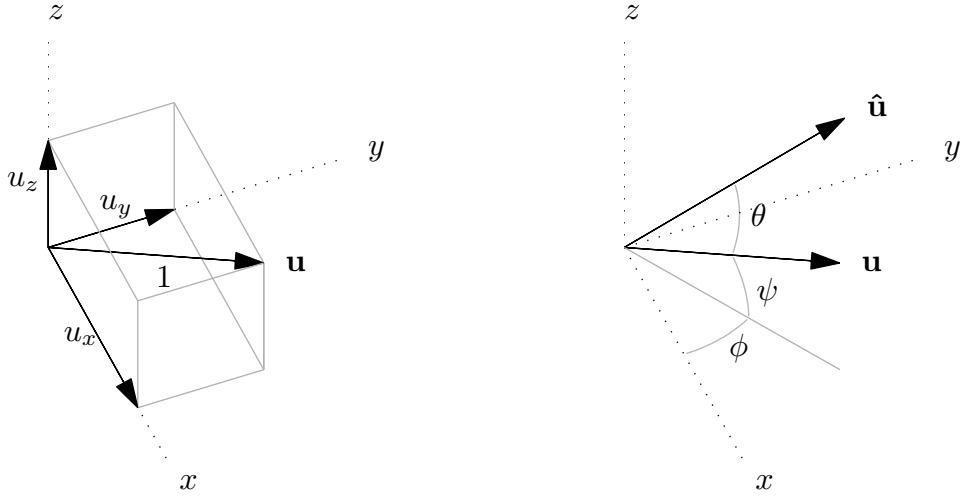


Figure 2.1.1: Unit vectors \mathbf{u} and $\hat{\mathbf{u}}$

by $\hat{\Theta}$ and Θ . From here I write the signal response and beampattern equations

$$y = y_p \cdot \frac{\eta^2 + \cos(\theta)}{\eta^2 + 1} \quad (2.1.9)$$

$$B = B_p \cdot \left| \frac{\eta^2 + \cos(\theta)}{\eta^2 + 1} \right|^2 \quad (2.1.10)$$

where B_p is the beampattern of the associated pressure-sensor array.

Before exploring these expressions in detail, I give a picture of the three-dimensional unit vectors \mathbf{u} and $\hat{\mathbf{u}}$ in Figure 2.1.1. The plot on the left illustrates \mathbf{u} and its components in the x , y , and z directions. Note that this unit vector \mathbf{u} has an arbitrary direction and need not lie in the x - y plane. The plot on the right shows the same unit vector and the hypothesized direction vector $\hat{\mathbf{u}}$. I have illustrated the azimuth angle ϕ and elevation angle ψ of \mathbf{u} . I have also illustrated θ , the angle between the two vectors. It is important to understand that θ is the angle between these vectors in \mathbb{R}^3 whereas ϕ is the angle one vector forms when projected onto the x - y plane. From this figure, it is clear that the $\cos \theta$ term comes from the scalar projection of \mathbf{u} onto $\hat{\mathbf{u}}$.

I now analyze the structure of Equation 2.1.10, specifically the vector-sensor mod-

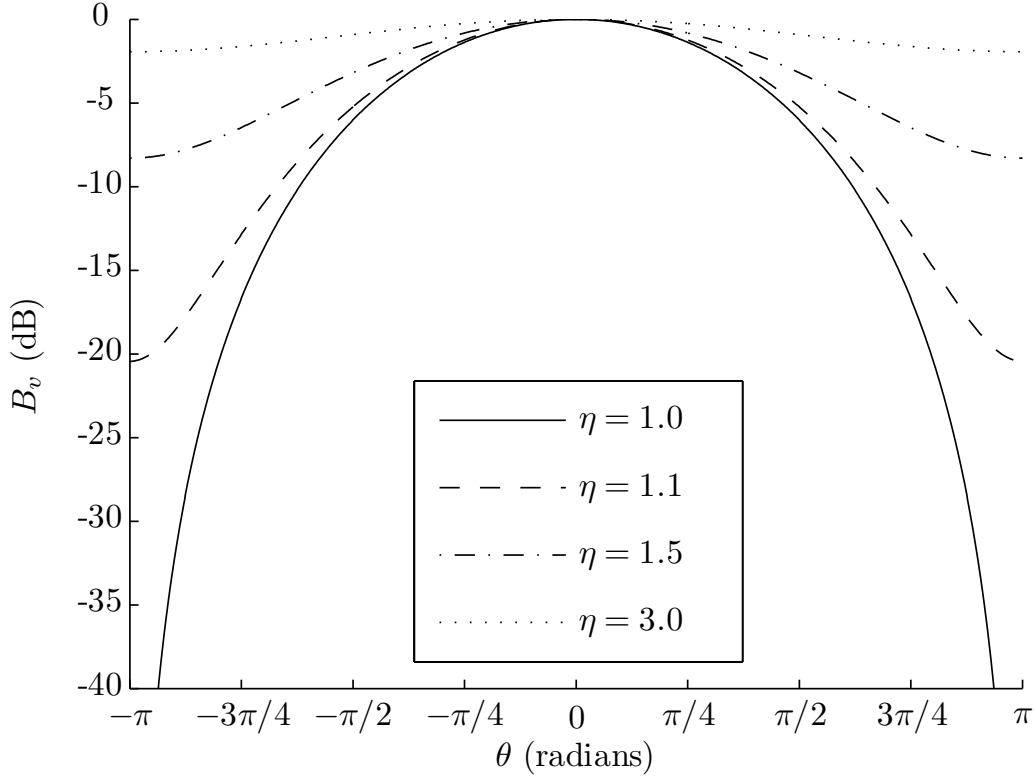


Figure 2.1.2: Vector-sensor modulation term B_v

ulation term

$$B_v \triangleq \left| \frac{\eta^2 + \cos(\theta)}{\eta^2 + 1} \right|^2. \quad (2.1.11)$$

In other contexts, this term is often referred to as the polarization term. This name originates in electromagnetic fields where the additional sensor gain comes from measuring wave polarization. Similarly, the angle θ is often called the polarization angle. Figure 2.1.2 plots this term versus θ as the normalization constant η varies. With proper normalization of the data, $\eta = 1$, giving the ideal null at $\theta = \pm\pi$. As the vector sensor gain decreases, $\eta \rightarrow \infty$ and $B_v \rightarrow 1$, *i.e.*, the vector-sensor array effectively becomes a pressure-sensor array. Also observe that the vector-sensor modulation term varies slowly with θ and forms an envelope for the pressure-sensor beam pattern. The “width” of the envelope B_v is generally much larger than the beamwidth of the pressure-sensor array response B_p . Thus, the mainlobe where $\cos(\theta) \approx 1$ is generally dominated by the pressure-sensor response; the vector-sensor terms affects the side-

lobe and ambiguity regions. Put another way, when $\theta \approx 0$ there is no polarization gain and the vector-sensor array effectively behaves like a pressure-sensor array. As a final note, it is easy to show that any effects from a spatial taper only enter through B_p and do not alter the vector-sensor term B_v .

2.2 Uniform Linear Array (ULA)

In this section, I apply the results above to the uniform linear array. This simple example illustrates the usefulness of the beampattern factorization and allows me to explore its use in more detail. The uniform linear array considered here is a uniformly weighted array of M linearly spaced sensors separated by d wavelengths. I further assume zero elevation or $\psi = 0$. Often, I use classical results for the pressure-sensor array which can be found in the thorough source [7].

2.2.1 Beampattern

I now examine in detail the factorization given by Equation 2.1.10. For the uniform linear array defined above, the pressure-sensor beampattern is given by

$$B_p = \frac{1}{M} \frac{\sin \left[\frac{M}{2} 2\pi d (\cos \phi - \cos \hat{\phi}) \right]}{\sin \left[\frac{1}{2} 2\pi d (\cos \phi - \cos \hat{\phi}) \right]} \quad (2.2.1)$$

and depends on Θ only through the azimuth angle ϕ . Again, choosing distance d in units of wavelengths greatly simplifies the expressions. Equation 2.2.1 is the familiar discrete sinc or Dirichlet beampattern appearing in classical array literature.

Before illustrating the vector-sensor ULA beampattern, I discuss the coordinate system used in Figures 2.2.1-2.2.3. Although the vector-sensor beampattern is a function of both azimuth and elevation through B_v , I only display a single scan at zero elevation. This is a reasonable restriction when considering sources whose distance is much greater than their depth. As mentioned in Section 1.5, $\phi = 0$ is forward endfire and $\phi = \pi/2$ is port broadside. The horizontal axis in Figures 2.2.1-2.2.3 scans from

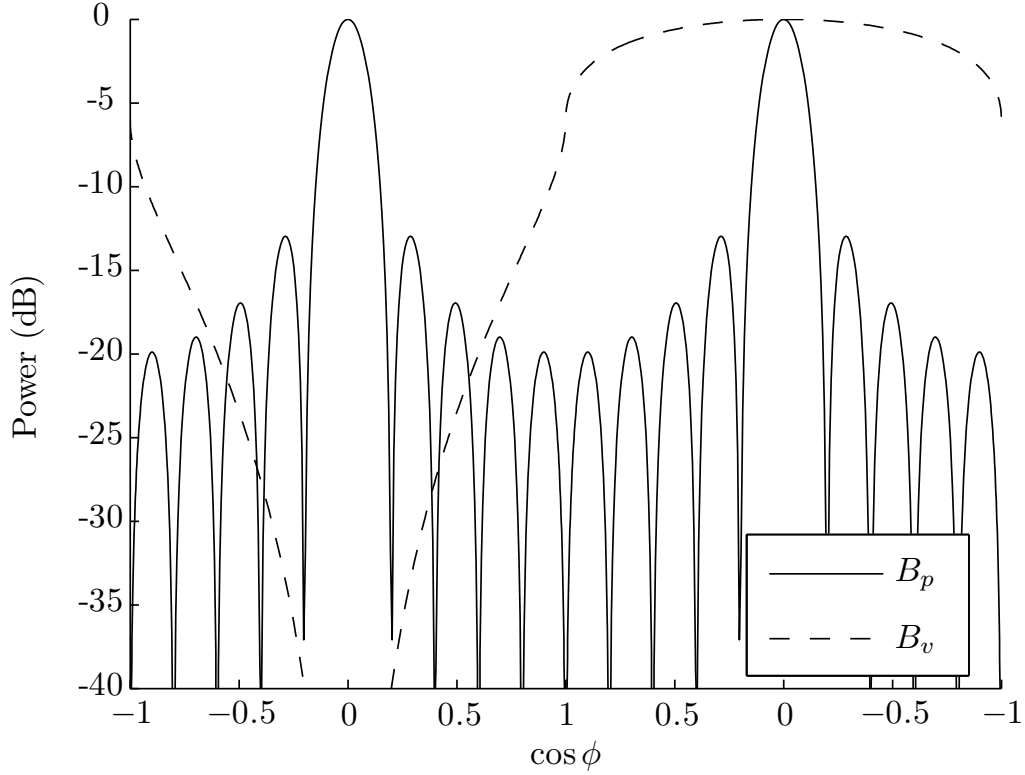


Figure 2.2.1: ULA Beampattern Components: $\hat{\phi} = \pi/2$

$\phi = -\pi$ to $\phi = \pi$ with constant cosine spacing. Thus, aft endfire is at the left and right edges and forward endfire is in the center. Also, the starboard side of the array appears on the left half of each plot and the port side appears on the right half. This counterintuitive starboard-to-port scan results from using a right-handed coordinate system with z directed upward.

Now examine the beampattern terms illustrated in Figure 2.2.1 for $M = 10$, $d = 1/2$, and $\hat{\phi} = \pi/2$ (port broadside). Both the pressure-sensor term, B_p , and the vector-sensor modulation term, B_v , for this example are illustrated in Figure 2.2.1. Because it is plotted versus $\cos \phi$, the shape of B_v is altered from that in Figure 2.1.2, raising an important distinction: a “natural” parameter space for the pressure-sensor component is $\cos \phi$; a “natural” parameter space for the vector-sensor component is θ . In general, $\psi \neq 0$ and these are different spaces (ϕ is conical angle but θ is polarization angle in three dimensions).

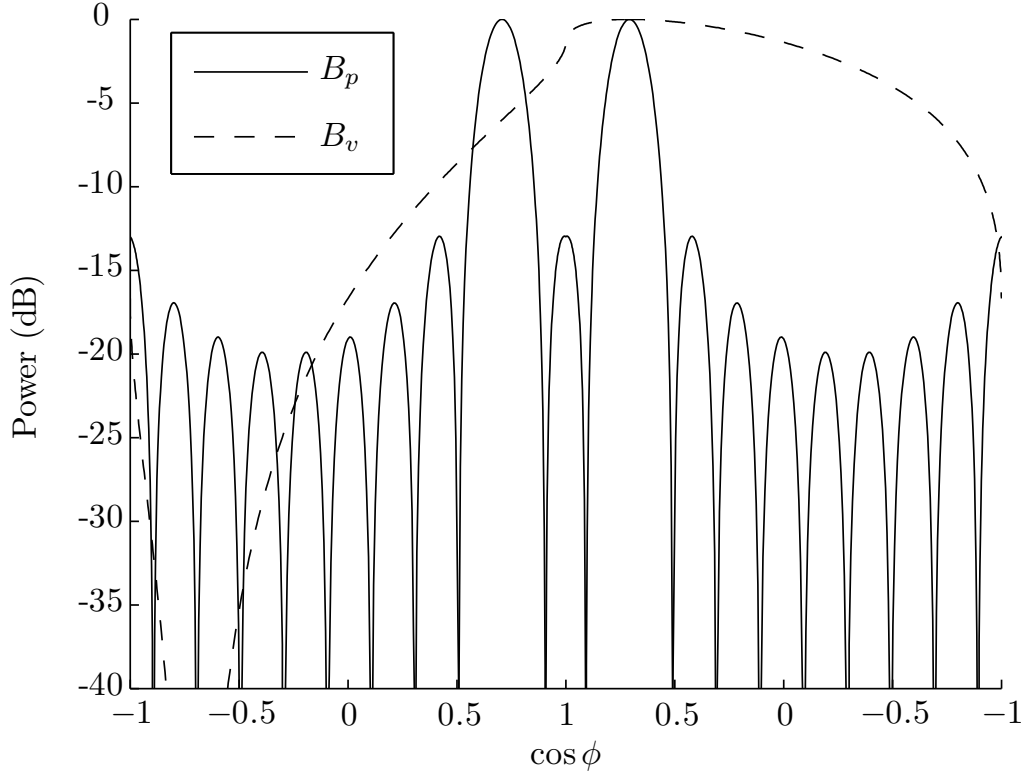


Figure 2.2.2: ULA Beampattern Components: $\hat{\phi} = \pi/4$

Although changing $\hat{\phi}$ or “steering” with a pressure-sensor array simply shifts the beampattern in wavenumber space, the same is not true for a vector sensor array. This effect is illustrated in Figure 2.2.2 with $\hat{\phi} = \pi/4$. This figure clearly shows the left/right, or port/starboard, ambiguity in B_p resulting from the conical angle ϕ . Comparing with Figure 2.2.1 reveals a different mapping of the vector-sensor modulation term. With $\hat{\phi} = \pi/2$ the vector-sensor null lies exactly on an ambiguity of B_p , but $\hat{\phi} = \pi/4$ produces a null in the sidelobe region.

The beampattern for a vector-sensor array is given by the product of B_p and B_v . For the examples above with $\hat{\phi} = \pi/2$ and $\pi/4$, the beampatterns are shown in Figure 2.2.3. This figure illustrates the effect of the vector-sensors on the left/right ambiguity inherent with the pressure-sensor array. The pressure-sensor ambiguity is nulled when the array is steered to broadside, but it becomes higher as it is steered toward endfire. As before, no spatial taper can further reduce the level of this pressure-

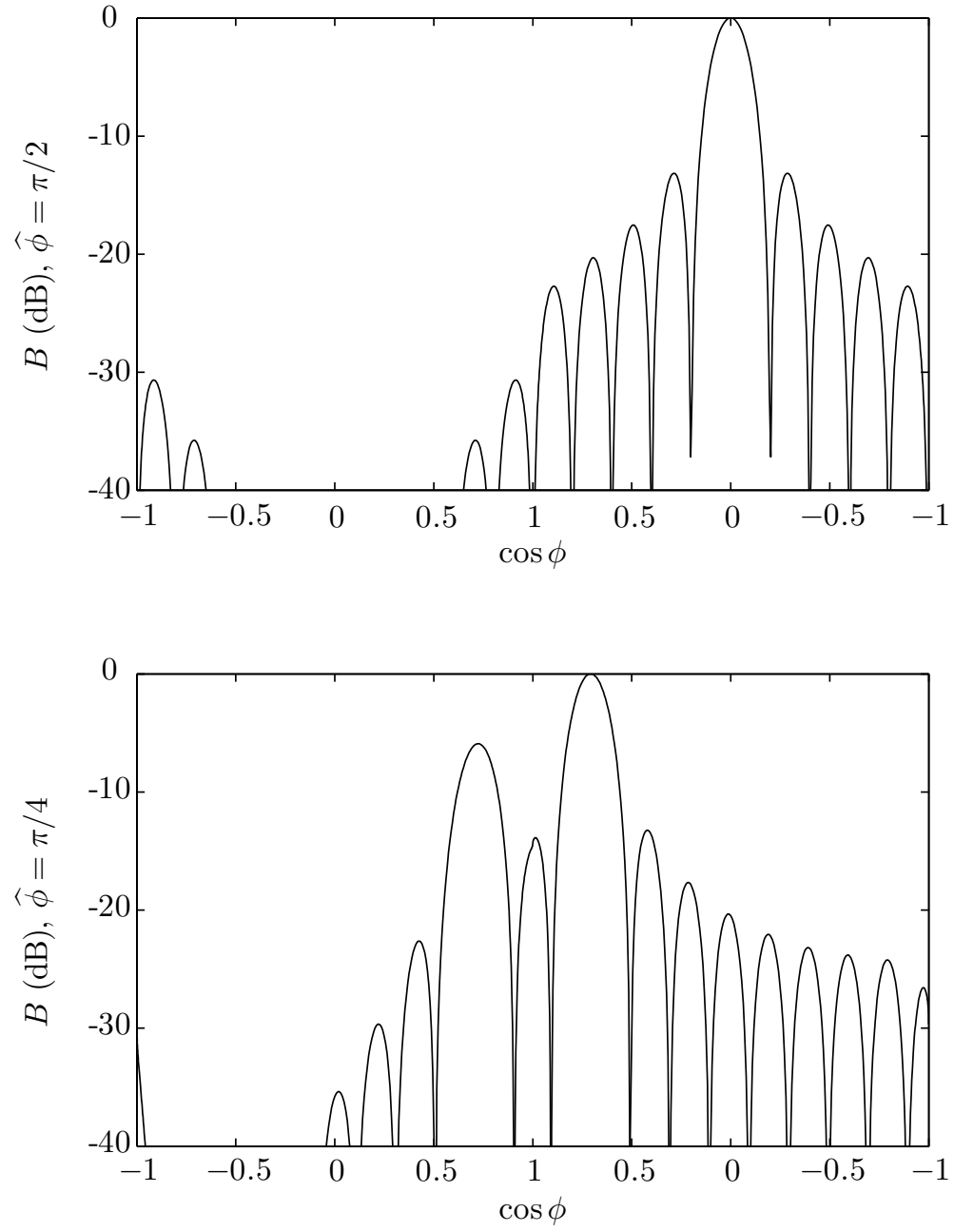


Figure 2.2.3: ULA Beampatterns: $\hat{\phi} = \pi/2$ and $\pi/4$

sensor ambiguity.

One of the more important theoretical benefits of a vector-sensor ULA is its ability to discriminate acoustic arrivals that would be ambiguous with a pressure-sensor ULA. In the next two subsections, I apply the beampattern factorization in Equation 2.1.10 to study the reduced level of pressure-sensor ambiguities.

2.2.2 Conical Angle (Left/Right) Discrimination

Because a linear pressure-sensor array is symmetric about rotation about its axis, its directional response is a function only of conical angle. This results in ambiguous arrivals: the array cannot determine its left from its right, or port from starboard. In practice, this means the array must maneuver to determine the true location of a source, a strict limitation. As demonstrated in the previous subsection, the velocity components of a vector-sensor array are not symmetric about any rotation. As a result, a linear vector-sensor array may “resolve” ambiguities that would be present with a pressure-sensor array.

With a linear pressure-sensor array, a source arriving at the hypothesized angle $\hat{\phi}$ is indistinguishable from a source at angle $\phi' = 2\pi - \hat{\phi}$, the same conical angle on the opposite side of the array. This ambiguity produces a peak or “backlobe” in the beampattern. Because the angle of these sources in three dimensions - the polarization angle θ - is clearly different, they are unambiguous with a vector-sensor array. In terms of the beampattern equations, this means

$$B_p(\hat{\phi}) = B_p(\phi') = 1 \quad (2.2.2)$$

$$B_v(\hat{\phi}) \neq B_v(\phi'). \quad (2.2.3)$$

The directional response at the backlobe ϕ' is given by

$$\begin{aligned}
B(\phi') &= B_v(\phi') \\
&= \left| \frac{\eta^2 + \cos(\hat{\phi} - \phi')}{\eta^2 + 1} \right|^2 \\
&= \left| \frac{\eta^2 - 1 + 2\cos^2(\hat{\phi})}{\eta^2 + 1} \right|^2
\end{aligned} \tag{2.2.4}$$

where the last step substitutes for ϕ' and applies a double angle identity. With proper normalization of the data, $\eta = 1$, resulting in the simple but useful relation

$$B(\phi') = \cos^4(\hat{\phi}) \tag{2.2.5}$$

giving the left/right suppression for any steering angle $\hat{\phi}$. Note that although ϕ' is an ambiguity for the pressure sensor array, it is not necessarily the highest point in the beampattern of a vector sensor array. One such example is Figure 2.2.3 for $\hat{\phi} = \pi/4$. The vector-sensor modulation shifts the peak of the backlobe very slightly toward endfire. In practice, Equation 2.2.5 is a good approximation to the peak value when the mainlobe is not excessively large. It is also important that the level and location of the backlobe are not affected by any spatial taper and do not vary with frequency or the number of sensors.

One use of Equation 2.2.5 is to give approximate regions over which a given left/right resolution is obtained: for at least 6 dB of resolution, $\pi/4 \leq \hat{\phi} \leq 3\pi/4$ or within $\pi/4$ radians of broadside.

2.2.3 Spatial Aliasing (Grating Lobe) Discrimination

In the same way that under-sampling a time signal produces frequency aliasing, spatially under-sampling a plane wave produces aliasing in the beampattern. The ambiguities or “grating lobes” resulting from this spatial aliasing limit the use of pressure-sensor arrays above a given design frequency. To further illustrate the usefulness of the beampattern results shown above, I quickly derive expressions for the location

and level of pressure-sensor grating lobes on the uniform linear vector-sensor array.

One theoretical benefit to acoustic vector-sensor arrays is their improved performance above the design frequency of corresponding pressure-sensor arrays. The design frequency of a uniform linear array is analogous to the Nyquist frequency for spatial sampling. Using previous notation, the perfect reconstruction criterion is $d < 1/2$. Denoting the inter-element spacing by δ units of length (whereas d is in wavelengths), this gives the design frequency

$$f_d = \frac{c}{2\delta}. \quad (2.2.6)$$

The spatial aliasing occurring above this design frequency enters only through B_p (Equation 2.2.1) because this function is periodic with respect to $\cos \phi$. In the following analysis I restrict my attention to pressure-sensor grating lobes existing within acoustic space. By symmetry, it is sufficient to consider only arrival angles $\hat{\phi} < \pi/2$. The location of the pressure-sensor grating lobes is then easily found as

$$\begin{aligned} \cos \phi' &= \cos \hat{\phi} - \frac{1}{d} \\ &= \cos \hat{\phi} - 2\frac{f_d}{f}. \end{aligned} \quad (2.2.7)$$

Examining this expression quickly reveals that the location of a pressure-sensor grating lobe depends on both arrival angle and analysis frequency. Even above the design frequency, pressure-sensor grating lobes for some angles $\hat{\phi}$ may not exist in physical space. Like the backlobes in the previous subsection, $B_p(\phi') = 1$, leaving

$$\begin{aligned} B(\phi') &= B_v(\phi') \\ &= \frac{1}{(\eta^2 + 1)^2} \left| \eta^2 + \cos \left[\hat{\phi} - \cos^{-1} \left(\cos \hat{\phi} - 2\frac{f_d}{f} \right) \right] \right|^2. \end{aligned} \quad (2.2.8)$$

Unlike the backlobe expression in Equation 2.2.5, Equation 2.2.8 is clearly dependent on frequency. Both results, however, are independent of the number of sensors and any spatial tapering. This leads to the important conclusion that neither backlobe nor grating lobe reduction is enhanced by increasing the number of sensors.

The behavior of Equation 2.2.8 is not immediately apparent, but a few examples

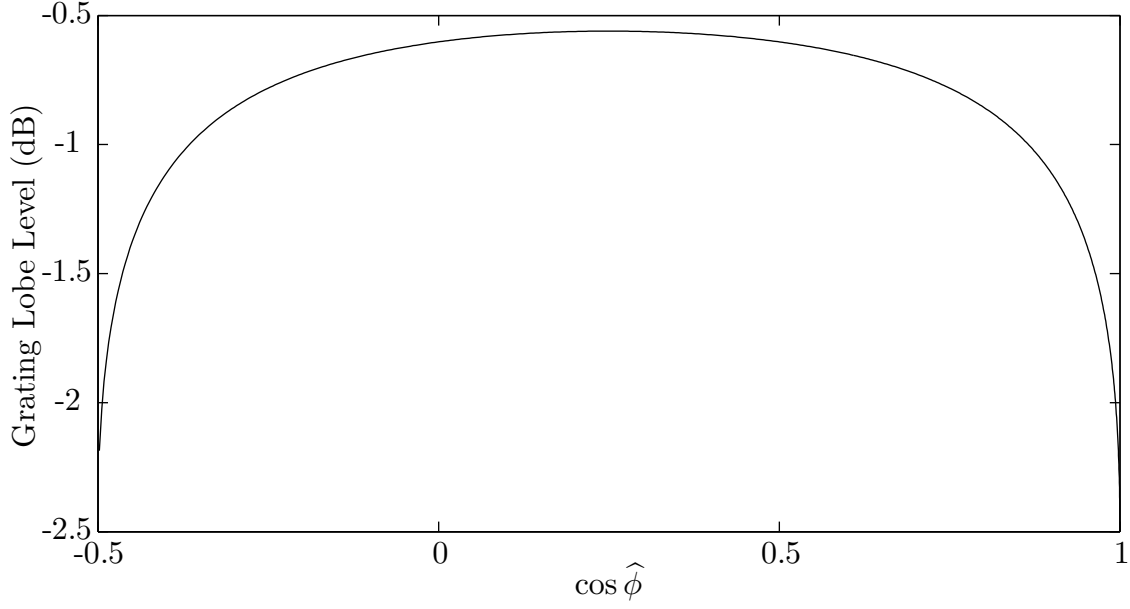


Figure 2.2.4: Vector-sensor “Grating Lobe” Height: $f/f_d = 4$

provide some insight. In the examples to follow, I again assume that $\eta = 1$. First, I fix the frequency at $f/f_d = 4$ and examine the “grating lobe” levels across all $\hat{\phi}$. I again put the term in quotations because spatially aliased sources are no longer ambiguous on a vector-sensor array and thus not true grating lobes. The result is shown in Figure 2.2.4. For this figure, the input range is restricted to $\cos \hat{\phi} \in [-1/2, 1]$ because no “grating lobes” occur in acoustic space when $\cos \hat{\phi} < -1/2$. As a second example, I fix the angle at broadside, $\hat{\phi} = \pi/2$, and consider the “grating lobe” level as a function of frequency. At this angle, “grating lobes” begin to appear when $f/f_d \geq 2$. As is shown in Figure 2.2.5, these lobes are at a -6 dB level when they first appear at twice the design frequency. As a final example, I plot contours of Equation 2.2.8 versus $\hat{\phi}$ and f/f_d . These contours are shown in Figure 2.2.6. The unlabeled contour corresponds to the angle beyond which no “grating lobes” appear in acoustic space. Figure 2.2.6 is readily used for design and analysis; for example, if an adaptive beamforming algorithm effectively nulls signals with 3 dB of mismatch, it may null these “grating lobes” up to approximately twice the design frequency.

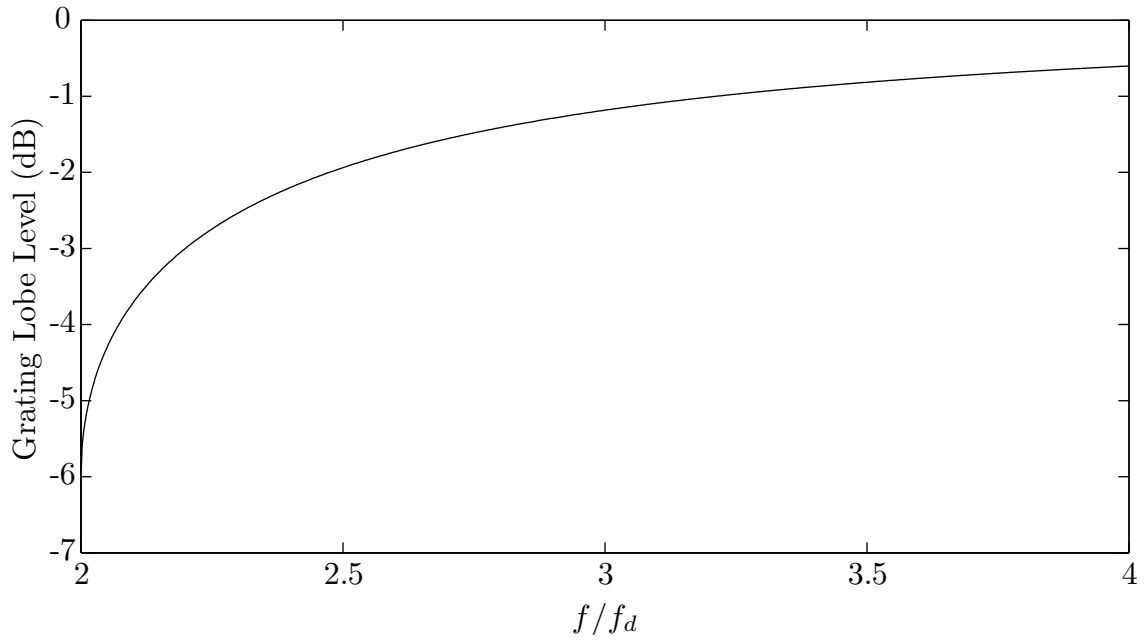


Figure 2.2.5: Vector-sensor “Grating Lobe” Height: $\hat{\phi} = \pi/2$

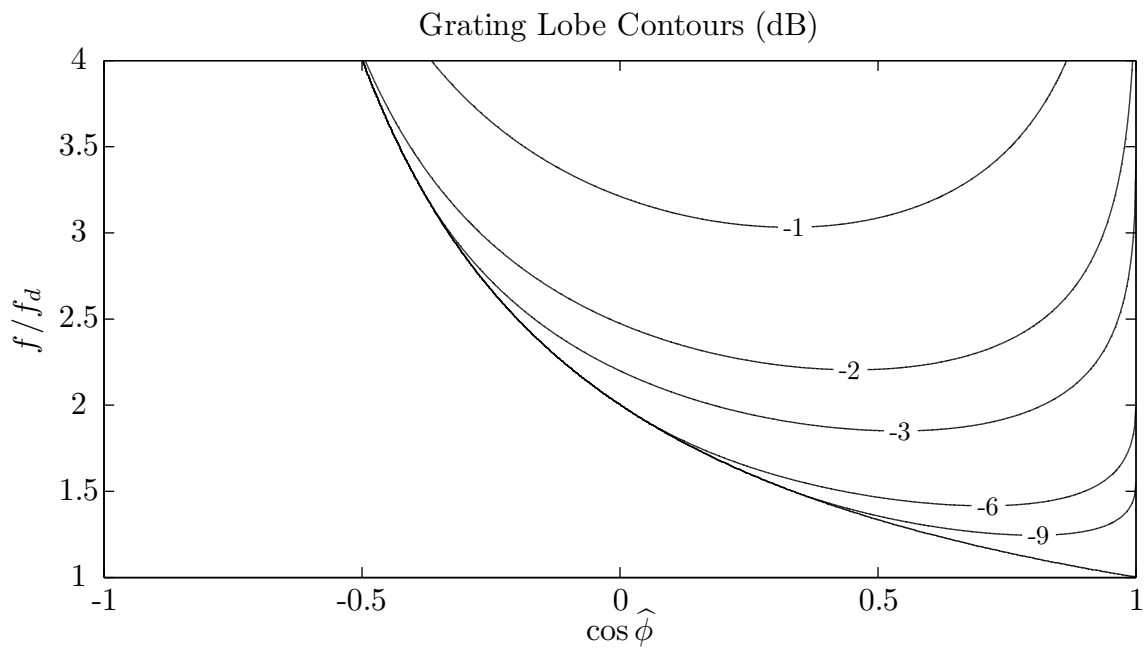


Figure 2.2.6: Vector-sensor “Grating Lobe” Height Contours

2.2.4 Near-Field Processing

Although the factorization given in Equation 2.1.10 is helpful for plane wave signals, it also holds well as an approximation when wavefront curvature increases. The “far-field” or plane-wave assumption that allows factoring the vector sensor term out of the array response is more strict than the traditional Fresnel rule-of-thumb. The Fresnel far-field distance for an array of aperture size L , usually given as $d_{ff} = L^2/2\lambda$, is based on a maximum phase error of $\pi/8$ [8]. With geophones, however, the angle-of-arrival error at the aperture edges produces differences in the beampattern at ranges greater than d_{ff} . At first glance, this seems to limit the practical use of Equation 2.1.10. However, this approximation is still useful because it may begin failing in the low sidelobe regions. These regions, although of theoretical interest, may already be unattainable in practice and may not contribute much integrated error. Because this thesis deals only with plane wave signals, a more detailed discussion of near-field effects is outside its scope.

Chapter 3

Mismatched Vector-Sensor Arrays

In the previous chapter, I considered “ideal” vector-sensor arrays. I term these arrays “ideal” because of two strong assumptions. First, I assume there are no errors in the model, *i.e.* the sensor positions and orientations are known. Second, I assume the received signal is deterministic and without sensor noise. In practice, both of these assumptions are typically violated. Sometimes, the sensor parameters - position and orientation - are fixed and known with some error tolerance. With other arrays, the sensor parameters vary slowly with time and can only be approximated. Signals, in practice, are usually embedded in noise and modeled stochastically.

This chapter expands my analysis to include these practical considerations. Of primary concern is the performance of acoustic vector-sensor arrays under “mismatch” or modeling errors. Now that I know how well “ideal” vector-sensor arrays can do, how does their performance degrade when the sensor parameters have random errors? The first section below considers the uniform linear array under Gaussian perturbations. Using some results from nonuniform sampling theory, it computes the average beampattern and studies the effects introduced by mismatch. The second section examines the problem of direction-of-arrival (DOA) estimation. With a Gaussian signal and error model, it derives and analyzes a Cramér-Rao lower bound for any unbiased DOA estimator.

3.1 Perturbed Linear Array

The previous chapter derived a simple form (see Equations 2.1.10 and 2.2.1) for the beampattern of a uniform linear array of vector-sensors. Before delving into direction-of-arrival estimation, it would be useful to understand how this beampattern changes when the array parameters have random errors. To keep the analysis simple, I consider only independent identically distributed (IID) Gaussian position errors. For a more detailed analysis of the beampattern under Gaussian modeling errors - including rotation, gain, and phase errors - see the work in [3] and [9]. Also, for an alternate analysis of random pressure-sensor arrays, see [10].

In the following subsections, I only consider the effect of position errors on the pressure-sensor array beampattern B_p . To motivate this discussion and justify ignoring the vector-sensor term B_v , examine the mean vector-sensor array beampattern with position errors only. I denote expectation with $E\{\cdot\}$ and use the random vector $\boldsymbol{\rho}^R$ to represent the position errors. Suppressing the dependence on $\hat{\boldsymbol{\Theta}}$, the expected beampattern is then

$$\begin{aligned} E\{B\} &= E_{\boldsymbol{\rho}^R}\{B_p \cdot B_v\} \\ &= E_{\boldsymbol{\rho}^R}\{B_p\} \cdot B_v \end{aligned} \tag{3.1.1}$$

where the factorization is possible because B_v is a function of known rotation parameters only and B_p is a function of position parameters only. When rotation errors are also considered as in [3], this factorization is not possible. As I show in this section, focusing on such a simplified model allows me to connect the field of “nonuniform sampling” to the analysis of a perturbed linear array. Although the resulting proof is less general than the one given in [9], it is more insightful thanks to the connections with nonuniform sampling.

3.1.1 Nonuniform Sampling

Before analyzing the perturbed linear array, I briefly state a useful result from the nonuniform sampling of a signal. The signal I define is the wide-sense stationary continuous-time random process $f(t)$. This signal is sampled at nonuniform times to give $\tilde{f}[n] = f(nT + \xi_n)$ where T is the nominal sampling period and ξ_n is a sequence of IID random variables. Recent work presented in [11] relates the discrete-time power spectrum density (PSD) $S_{\tilde{f}\tilde{f}}$ to the continuous-time PSD S_{ff} :

$$S_{\tilde{f}\tilde{f}}(e^{j\omega}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} S_{ff}\left(\frac{\omega - 2\pi k}{T}\right) \left| \varphi_{\xi}\left(\frac{\omega - 2\pi k}{T}\right) \right|^2 + \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{ff}(\Omega) [1 - |\varphi_{\xi}(\Omega)|^2] d\Omega \quad (3.1.2)$$

where $\varphi_{\xi}(s) = E\{e^{js\xi}\}$ is the characteristic function of ξ . Stating this result in words, the nonuniform sampling causes two effects: 1) the continuous-time PSD is windowed by φ_{ξ} and aliased, and 2) white noise is introduced. Also note that with uniform sampling $\varphi_{\xi} = 1$ and Equation 3.1.2 reverts to the standard uniform sampling expression for aliasing.

3.1.2 Relation to Perturbed ULA

To connect Equation 3.1.2 with the perturbed linear array, I must introduce the concept of an infinite linear aperture. The linear array obtains measurements at a finite number of points on a line, but an infinite linear aperture obtains measurements at every point on the line. Without loss of generality, assume the infinite linear aperture is oriented along the x axis and the single source is in the x - y plane. I now define

$$k_0 \triangleq \frac{2\pi}{\lambda} \quad (3.1.3)$$

$$k_x \triangleq -k_0 \cos \phi. \quad (3.1.4)$$

In words, the variable k_0 is the magnitude of the wavenumber and k_x is the wavenumber component along the array axis. The infinite linear aperture measurements are of acoustic pressure in phasor form, written as

$$p(x) = \exp \left(j \hat{k}_x x \right). \quad (3.1.5)$$

Because this array lies in the same x - y plane as the propagating wave, I need only consider position errors in the x and y directions. To formalize the modeling errors, assume the position perturbations are zero-mean Gaussian with variances given by σ_x^2 and σ_y^2 .

Having now defined an infinite linear aperture, I examine its relationship to nonuniform sampling. Using the same notation as Section 3.1.1, the discrete sensors with position perturbations are equivalent to a finite nonuniform spatial sampling of $p(x)$. Specifically, I write the discrete measurements as $a_{p_n} = p(n\delta + \xi_n)$, where ξ_n are zero-mean Gaussian IID random variables with variance $\sigma^2 = \sigma_x^2 + \sigma_y^2 \tan^2 \hat{\phi}$. Collapsing the position errors into a single variance is possible because errors along the y axis are equivalent to scaled errors along the x axis. Now, I relate time in Equation 3.1.2 to space along the infinite linear aperture. Formally, this means variables are remapped like

$$\begin{aligned} t &\leftrightarrow x \\ \Omega &\leftrightarrow k_x \end{aligned} \quad (3.1.6)$$

Now the importance of wavenumber k_x is clear: it is the spatial equivalent to angular frequency Ω .

To show the implications of this mapping from time to space, I apply the nonuniform sampling result to the perturbed array in four steps. The first step simply maps variables from space to time. This mapping gives the equivalent time series $f(t) = p(t)$ which is now sampled at intervals equal to the inter-element spacing $T = \delta$. The second step computes the terms in Equation 3.1.2. The time series is

simply a complex exponential and the ξ_n are Gaussian, so I have

$$S_{ff}(\Omega) = 2\pi \mathcal{D}(\Omega - \hat{k}_x) \quad (3.1.7)$$

$$|\varphi_\xi(\Omega)|^2 = \exp(-\sigma^2 \Omega^2) \quad (3.1.8)$$

where $\mathcal{D}(\cdot)$ is the familiar Dirac delta function. The sifting property of the delta function makes the evaluation of Equation 3.1.2 simple, giving

$$\gamma \triangleq \exp(-\sigma^2 \hat{k}_x^2) \quad (3.1.9)$$

$$= \exp\left[-k_0^2 \left(\sigma_x^2 \cos^2 \hat{\phi} + \sigma_y^2 \sin^2 \hat{\phi}\right)\right] \quad (3.1.10)$$

$$S_{\tilde{f}\tilde{f}}(e^{j\omega}) = \frac{2\pi\gamma}{\delta} \sum_{k=-\infty}^{\infty} \mathcal{D}\left(\frac{\omega - 2\pi k}{\delta} - \hat{k}_x\right) + (1 - \gamma). \quad (3.1.11)$$

Note that $S_{\tilde{f}\tilde{f}}$ is the PSD for *an infinite-length sequence of samples*. To reflect the finite number of samples in a linear array, the third step windows this sequence. I now define

$$w[n] \triangleq \begin{cases} 1/M & 0 \leq n < M \\ 0 & \text{otherwise} \end{cases} \quad (3.1.12)$$

$$g[n] \triangleq w[n]f(n\delta) \quad (3.1.13)$$

$$\tilde{g}[n] \triangleq w[n]\tilde{f}[n] \quad (3.1.14)$$

where the rectangular window $w[n]$ is normalized to agree with Equation 2.1.2. Note that $g[n]$ is the sequence obtained by uniform sampling and $\tilde{g}[n]$ is the sequence obtained by nonuniform sampling. Through linearity, windowing the first term in Equation 3.1.11 gives γ times the uniform sampling PSD. The second term gives white noise that is easily evaluated in the time domain. Combining the two produces

$$S_{\tilde{g}\tilde{g}}(e^{j\omega}) = \gamma \cdot S_{gg}(e^{j\omega}) + (1 - \gamma) \cdot \frac{1}{M}. \quad (3.1.15)$$

The fourth and final step is very brief and converts back to the space domain, giving

a result in terms of ϕ . The PSD of $g[n]$ becomes the nominal beampattern and the PSD of $\tilde{g}[n]$ becomes the expected beampattern. After the transformation, Equation 3.1.15 becomes

$$E\{B_p(\phi)\} = \gamma \cdot B_p(\phi) + (1 - \gamma) \cdot \frac{1}{M}. \quad (3.1.16)$$

Note that the derivation above is easily extended to non-rectangular windows and is valid as stated for any window normalized for unity gain. Note also that it is trivial to include Gaussian phase errors in the proof through Equation 3.1.10.

3.1.3 Analysis of the Expected Beampattern

Although the end result in Equation 3.1.16 is the same as that derived in [9] for a nominally linear array, the derivation shown here has two benefits. First, it connects the field of array processing to that of nonuniform sampling. Linear arrays have long been associated with discrete-time signal processing, but they have not been fully explored in the context of nonuniform sampling. Though the end result in [9] has a nice interpretation, the proof is strictly mathematical. A second benefit of my derivation is that it comes with an intuitive picture of how position mismatch affects the beampattern.

To gain this intuition, examine Figure 3.1.1. This figure results from the parameters $M = 16$, $f = 2f_d$, $\hat{\phi} = 4\pi/9$, $\sigma_x = \lambda_d/3$, and $\sigma_y = 0$. Each illustration is shown on the same horizontal axis of Ω , analogous to the horizontal wavenumber k_x . Part (a) of the illustration shows the characteristic function φ_ξ . As shown by the dotted lines, the mismatch factor γ comes from this curve evaluated at the source wavenumber \hat{k}_x . Because the position errors are Gaussian, φ_ξ takes a normalized Gaussian shape. As position errors decrease, the characteristic function spreads and γ increases. Part (b) shows the PSD $S_{\tilde{f}\tilde{f}}$ of the nonuniformly sampled infinite sequence $\tilde{f}[n]$. Compared to an uniformly sampled PSD, the impulse train is weighted by a factor of γ and white noise of power $1 - \gamma$ is added. In the limit of no mismatch the PSD converges to the familiar impulse train. Part (c) shows the effect of windowing the sequence $\tilde{f}[n]$ to obtain $\tilde{g}[n]$. The PSD of the window, in this case a sinc function, is convolved with

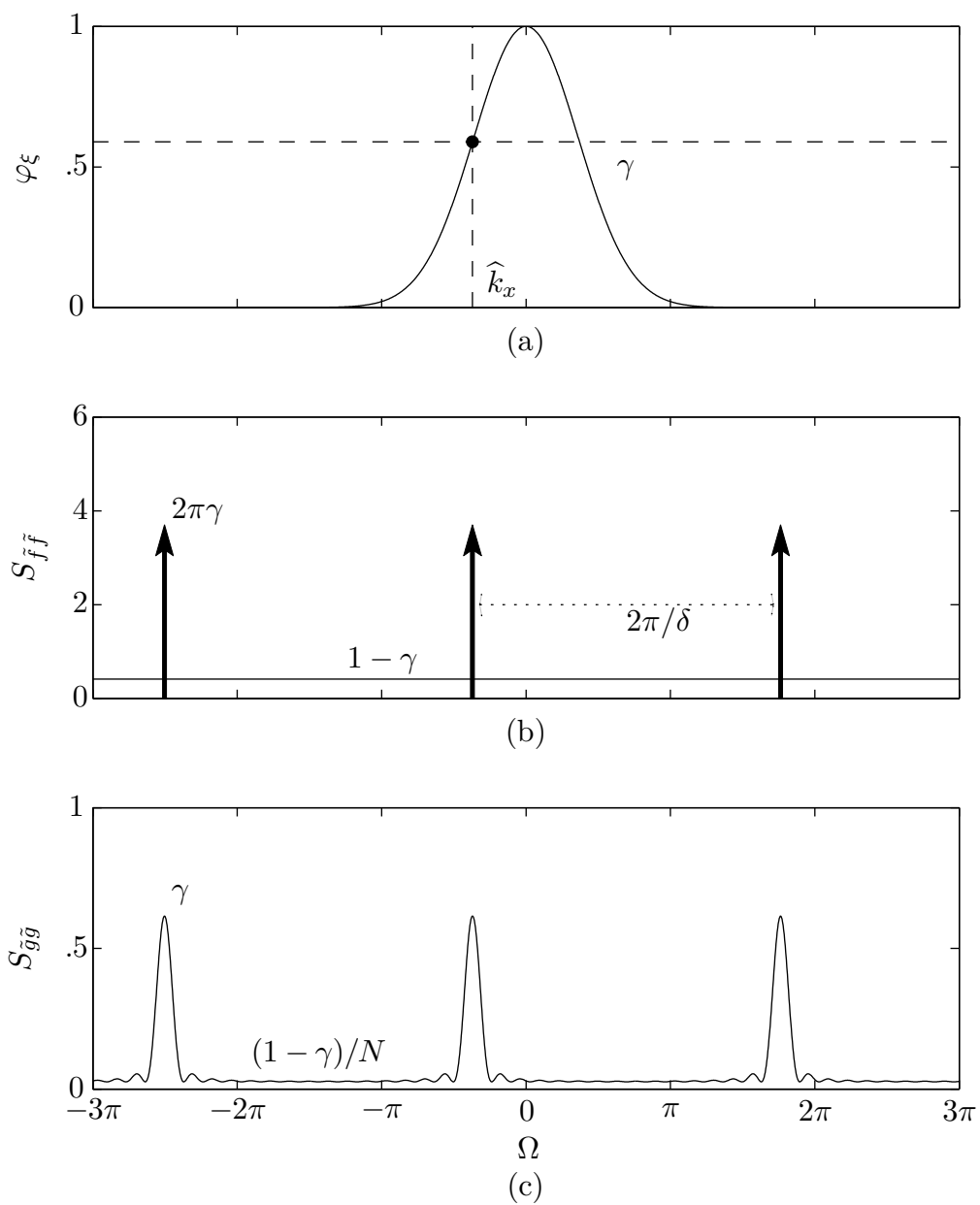


Figure 3.1.1: Illustration of Nonuniform Spatial Sampling

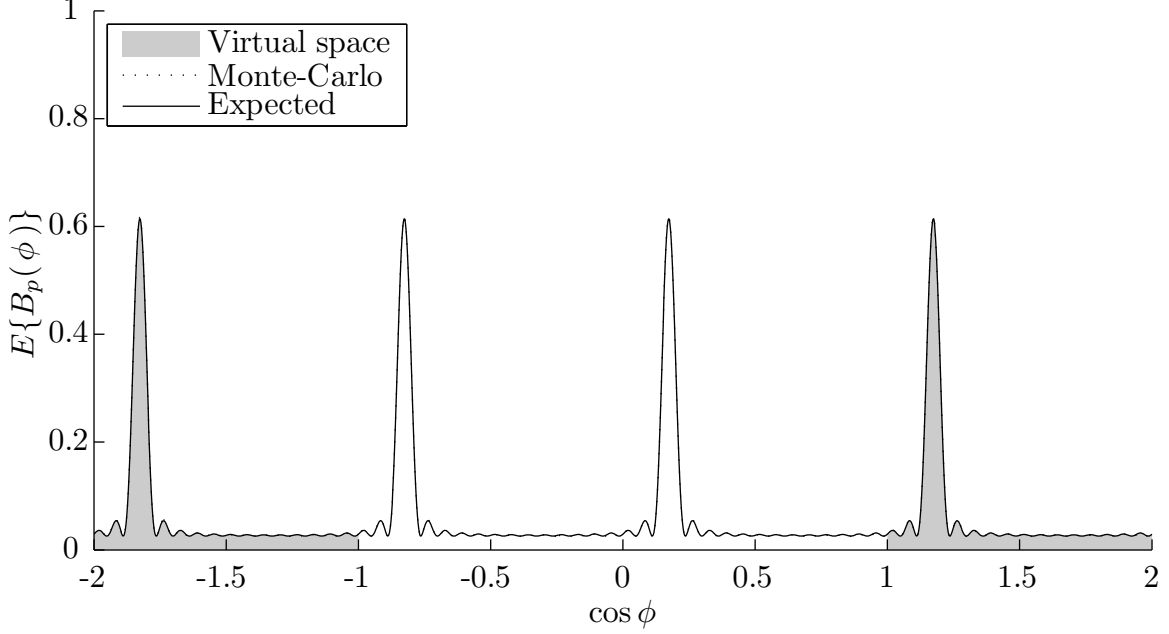


Figure 3.1.2: Mismatched Beampattern: $f = 2f_d$

$S_{\bar{f}\bar{f}}$ yielding a Dirichlet function.

To complete the picture of position mismatch, the PSD in Figure 3.1.1 must be mapped from Ω to ϕ . By definition, $\Omega \leftrightarrow k_x$ and $k_x = -k_0 \cos \phi$ so a region of Ω maps linearly to a region in $\cos \phi$. Figure 3.1.2 shows this mapping for the illustrated example in the previous paragraph. I have also included the average beampattern from a 4000-trial Monte-Carlo simulation although it lies almost exactly on the predicted curve. The shaded region corresponding to $\cos \phi \notin [-1, 1]$ is dubbed “virtual space” because no real ϕ maps to this region. The term “physical space” likewise refers to the complementary region of real ϕ . Because the analysis frequency is above the design frequency of the array more than one peak maps into physical space. This produces spatial aliasing, or grating lobes, in the beampattern. If I lower the analysis frequency to $f = f_d/2$ but keep the same source wavenumber \hat{k}_x and mismatch σ , the illustration in Figure 3.1.1 remains exactly the same. The mapping to ϕ , however, changes as shown in Figure 3.1.3. Because the new analysis frequency is below the array design frequency only one peak maps into physical space and no grating lobes are present.

Before moving on I summarize the effect of position mismatch on the beampattern.

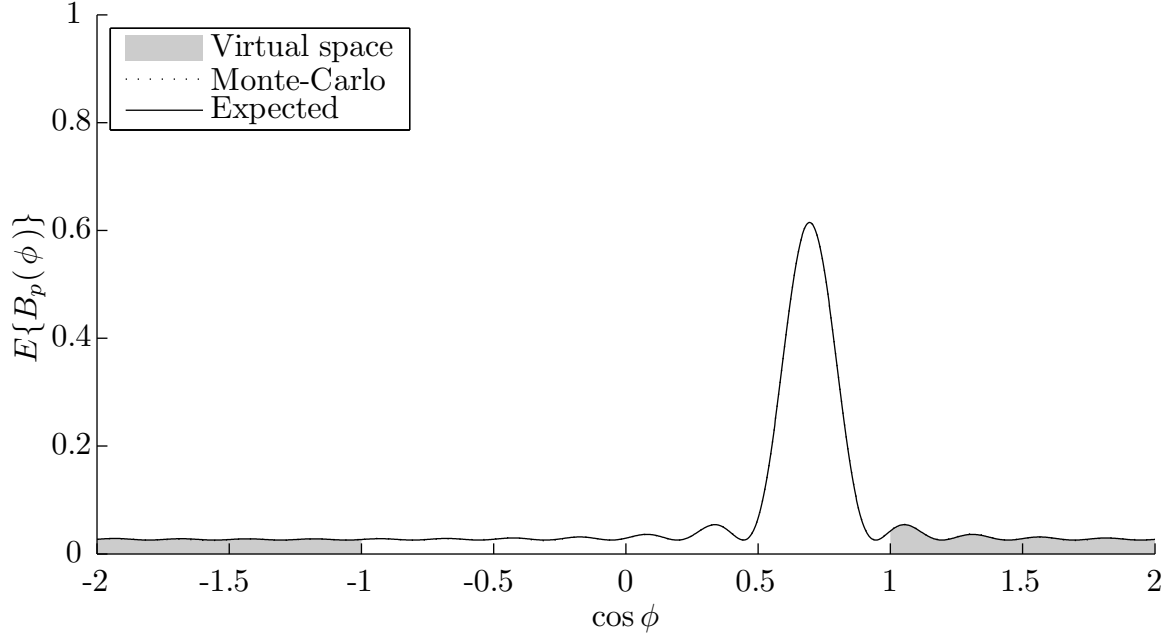


Figure 3.1.3: Mismatched Beampattern: $f = f_d/2$

The decrease in peak value caused by mismatch is simply a factor γ which in decibels grows linearly with $\sigma^2 \hat{k}_x^2$. The expected beampattern is a linear interpolation between the ideal beampattern and $1/M$ white noise - with γ as the interpolation factor. Having derived some intuition for the expected beampattern under position errors only, I now examine the vector-sensor array performance under a more general class of modeling errors.

3.2 Cramér-Rao Bound

This section derives a Cramér-Rao bound for direction-of-arrival estimation error with a single-source. The initial model is an arbitrary array of vector sensors whose position and rotation are perturbed by zero-mean Gaussian errors. Later subsections extend this model to include Gaussian gain and phase errors. The modeling errors here are the same as in [3] and are a vector-sensor extension of the Gilbert-Morgan model in [9]. Despite sharing a mismatch model, the DOA bound explored in this thesis is distinct from the beampattern expressions derived in [3]. All measurements under this model are of a zero-mean Gaussian random process of unknown power corrupted by additive white noise of unknown power. This is the more common scenario of “unknown signal in unknown noise.” To keep things shorter I use without proof the “hybrid” Cramér-Rao bound given in [7], Chapter 8.11. The bound used below is an approximation that is valid when the variance of the perturbations is small (again, see [7]). Note also that without modeling errors this bound reduces to the result given in [2].

3.2.1 Statement of the Hybrid Bound

The stochastic model outlined above may seem simple, but the bound it produces is quite complex to derive and express. This subsection elaborates on the model, defines several useful quantities, and states the hybrid Cramér-Rao bound explored in the rest of the chapter. It also clarifies some notation in an attempt to keep the derivation as simple as possible.

I begin by defining some helpful quantities and notation. Recalling the plane wave replica vector \mathbf{v} parameterized by $\Theta = [\phi \ \psi]^T$, suppose there are K independent measurements or “snapshots” of the form

$$\mathbf{x}_k(\Theta) = \mathbf{v}(\Theta) \cdot \mathcal{CN}(0, \sigma_s^2) + \mathcal{CN}(0, \sigma_n^2), \quad k = 1, 2, \dots, K \quad (3.2.1)$$

where $\mathcal{CN}(\mu, \sigma^2)$ denotes a complex Gaussian random variable with mean μ and

variance σ^2 . If the noise variance of the pressure sensors differs from that of the velocity sensors, it can be absorbed without loss of generality into the normalization constant η . This normalization introduced in Equation 1.5.7 allows me to treat the sensors as having equal noise powers. As stated above, my model assumes the more common scenario where both variances σ_s^2 and σ_n^2 are unknown. Also, each vector-sensor is perturbed by Gaussian position and rotation errors. It is further assumed that the components of each vector-sensor are rigidly connected and thus displaced and rotated together. To help keep a simple notation, I use a semicolon to denote vertical concatenation. I then define the perturbation parameters in column vectors

$$\boldsymbol{\rho}^R \triangleq [x ; y ; z] \quad (3.2.2)$$

$$\boldsymbol{\rho}^\Theta \triangleq [\alpha ; \beta ; \gamma] \quad (3.2.3)$$

$$\boldsymbol{\rho} \triangleq [\boldsymbol{\rho}^R ; \boldsymbol{\rho}^\Theta] \quad (3.2.4)$$

where $\boldsymbol{\rho}^R$ are the Euclidean coordinates of a sensor and $\boldsymbol{\rho}^\Theta$ are the Euler rotation angles about the corresponding coordinate axes. I also introduce notation to index into the source and rotation parameters, using Θ_l and ρ_l to indicate the l th parameter in $\boldsymbol{\Theta}$ and $\boldsymbol{\rho}$, respectively. This is a slight abuse of notation because each vector-sensor technically has its own perturbation parameters. For the m th vector-sensor, I denote these perturbation parameters by $\{x_m, y_m, z_m, \alpha_m, \beta_m, \gamma_m\}$. In the results that follow, I often find block matrices that cannot be expressed simply with Kronecker and Hadamard products. With such matrices, notation like $\mathbf{A} = [A_{ij}]$ indicates that the terms A_{ij} are concatenated over row index i and column index j to form the matrix \mathbf{A} . Another example of this notation is $[\partial \mathbf{A} / \partial \rho_i]$, indicating the vertical concatenation of the derivatives of the matrix \mathbf{A} with respect to the perturbation parameters $\boldsymbol{\rho}$.

Having setup some notation and defined the model parameters, I begin listing terms used in the following sections. First, I define the derivative of the replica

vector with respect to the source parameters

$$\mathbf{D}_\phi \triangleq \frac{\partial \mathbf{v}}{\partial \phi}, \text{ etc.} \quad (3.2.5)$$

$$\mathbf{D}_\Theta \triangleq [\mathbf{D}_\phi \quad \mathbf{D}_\psi] \quad (3.2.6)$$

where “etc.” indicates that \mathbf{D}_ψ is defined similarly. Because \mathbf{v} is a length $4M$ column vector, \mathbf{D}_ϕ is $4M \times 1$ and \mathbf{D}_Θ is $4M \times 2$. I also create derivatives of a slightly different form with respect to the perturbation parameters

$$\mathbf{D}_\mathbf{x} \triangleq \left[\frac{\partial v_1}{\partial x_1} ; \frac{\partial v_2}{\partial x_2} ; \dots ; \frac{\partial v_{4M}}{\partial x_{4M}} \right], \text{ etc.} \quad (3.2.7)$$

where the derivatives in Equation 3.2.7 go through all $4M$ sensor elements (M pressure sensors and $3M$ velocity sensors as defined in Equation 2.1.1). Just as \mathbf{D}_\star defined derivatives of the replica vector \mathbf{v} with respect to a given variable \star , the notation $\boldsymbol{\delta}_\star$ defines derivatives of the unit vector \mathbf{u} :

$$\boldsymbol{\delta}_\phi \triangleq \frac{\partial \mathbf{u}}{\partial \phi}, \text{ etc.} \quad (3.2.8)$$

Another helpful notation is to define derivatives of the \mathbf{h} vector similarly

$$\boldsymbol{\Delta}_\phi \triangleq \frac{\partial \mathbf{h}}{\partial \phi}, \text{ etc.}^1 \quad (3.2.9)$$

From the results from Section 2.1 recall that $\mathbf{v}^H \mathbf{v} = M(\eta^2 + 1)$. In keeping with the

¹For the derivatives considered, $\boldsymbol{\Delta}_\star = [0 ; \boldsymbol{\delta}_\star]$ for a given parameter \star .

notation from [7], I define the terms

$$S_f \triangleq \sigma_s^2 \quad (3.2.10)$$

$$\mathbf{S}_x \triangleq \mathbf{v} S_f \mathbf{v}^H + \sigma_n^2 \mathbf{I} \quad (3.2.11)$$

$$\begin{aligned} \Sigma &\triangleq S_f \mathbf{v}^H \mathbf{S}_x^{-1} \mathbf{v} S_f / \sigma_n^2 \\ &= \sigma_s^2 \mathbf{v}^H [\mathbf{v} \mathbf{v}^H \sigma_s^2 + \sigma_n^2 \mathbf{I}]^{-1} \mathbf{v} \sigma_s^2 / \sigma_n^2 \\ &= \gamma^2 \mathbf{v}^H [\gamma \mathbf{v} \mathbf{v}^H + \mathbf{I}]^{-1} \mathbf{v} \\ &= \gamma^2 \mathbf{v}^H \left[\mathbf{I} - \gamma \mathbf{v} (\mathbf{I} + \gamma \mathbf{v} \mathbf{v}^H)^{-1} \mathbf{v}^H \right] \mathbf{v} \\ &= \gamma^2 \mathbf{v}^H \mathbf{v} \left(1 - \frac{\gamma \mathbf{v}^H \mathbf{v}}{1 + \gamma \mathbf{v}^H \mathbf{v}} \right) \\ &= \gamma \cdot \frac{\gamma M (\eta^2 + 1)}{\gamma M (\eta^2 + 1) + 1} \end{aligned} \quad (3.2.12)$$

where $\gamma = \sigma_s^2 / \sigma_n^2$ is the element signal-to-noise ratio (SNR).² Note that S_f and Σ are scalar values because of the single source. Lastly, I define

$$\begin{aligned} \mathbf{P}_\mathbf{v}^\perp &\triangleq \mathbf{I} - \mathbf{v} (\mathbf{v}^H \mathbf{v})^{-1} \mathbf{v}^H \\ &= \mathbf{I} - [(\mathbf{a}_p \mathbf{a}_p^H) \otimes (\mathbf{h} \mathbf{h}^H)] / [M(\eta^2 + 1)], \end{aligned} \quad (3.2.13)$$

a projection matrix orthogonal to the replica subspace. In other words, the source replica vector \mathbf{v} spans the nullspace of the projection matrix $\mathbf{P}_\mathbf{v}^\perp$. This matrix can be viewed as a covariance matrix driving a perfect null in the direction of the source.

With the above definitions, I now state the hybrid Cramér-Rao bound. To simplify the resulting equation, I write it in block form and in terms of the matrices

$$\mathcal{A} \triangleq 2K \Sigma \cdot \text{Re} \{ \mathbf{D}_\Theta^H \mathbf{P}_\mathbf{v}^\perp \mathbf{D}_\Theta \} \quad (3.2.14)$$

$$\mathcal{B} \triangleq 2K \Sigma \cdot \text{Re} \left\{ \left[(\mathbf{P}_\mathbf{v}^\perp)^T \mathbf{D}_{\Theta_j}^* \odot \mathbf{D}_{\rho_i} \right] \right\} \quad (3.2.15)$$

$$\mathcal{C} \triangleq 2K \Sigma \cdot \text{Re} \left\{ \left[\mathbf{D}_{\rho_i} \mathbf{D}_{\rho_j}^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T \right] \right\} + \mathbf{\Lambda}_\rho^{-1}. \quad (3.2.16)$$

²This Σ is not exactly the same as that used in [7]. Here, it has a nice interpretation as the product of the element SNR and the array SNR.

In the above equations, $*$ denotes matrix conjugation. The covariance matrix $\mathbf{\Lambda}_\rho$ specifies the second-order statistics for the perturbation parameters. Again using notation similar to [7], the following equation for \mathbf{C}_{HCR} lower bounds the mean-square error of unbiased estimates for $\mathbf{\Theta}$ and ρ :

$$\mathbf{C}_{HCR}(\mathbf{\Theta}, \rho) = \left[\begin{array}{c|c} \mathbf{\mathcal{A}} & \mathbf{\mathcal{B}}^T \\ \hline \mathbf{\mathcal{B}} & \mathbf{\mathcal{C}} \end{array} \right]^{-1}. \quad (3.2.17)$$

For the remainder of this section, I only consider the mean-square error bounds on the source parameters $\mathbf{\Theta}$. That is, I treat the perturbations ρ as nuisance parameters. This allows me to rewrite Equation 3.2.17 for only the upper left partition as

$$\mathbf{C}_{HCR}(\mathbf{\Theta}) = [\mathbf{\mathcal{A}} - \mathbf{\mathcal{B}}^T \mathbf{\mathcal{C}}^{-1} \mathbf{\mathcal{B}}]^{-1}. \quad (3.2.18)$$

This form of the hybrid Cramér-Rao bound is presented in more detail in [7]. Although Equation 3.2.18 bounds estimation of $\mathbf{\Theta}$ in radians, it is often better to know the bound in cosine-space. Thankfully, applying a simple change of coordinates to the CRB is easily done as described in [7]. For the bound on ϕ this only requires multiplying by $\sin^2 \phi$. Keeping Equation 3.2.18 in mind, I now begin evaluating the terms of the bound in detail.

3.2.2 Evaluation of Terms

When deriving each term in the hybrid Cramér-Rao bound, a few observations become very helpful. First, the real components in \mathbf{h} are unaffected by position errors. Put another way, only the phase vector \mathbf{a}_p is affected when perturbing the sensor positions. Second, the phase components in \mathbf{a}_p are unaffected by orientation errors. The two statements above are another instance where the factorization in Equation 1.5.7 comes in very handy: position errors enter through \mathbf{a}_p and rotation errors enter through \mathbf{h} .

Evaluation of Equation 3.2.18 begins in the logical place with the matrix $\mathbf{\mathcal{A}}$. For

this, I need the derivatives that compose \mathbf{D}_{Θ} :

$$\begin{aligned}
\mathbf{D}_{\phi} &= \frac{\partial}{\partial \phi} \mathbf{v} \\
&= \frac{\partial}{\partial \phi} (\mathbf{a}_p \otimes \mathbf{h}) \\
&= \frac{\partial \mathbf{a}_p}{\partial \phi} \otimes \mathbf{h} + \mathbf{a}_p \otimes \frac{\partial \mathbf{h}}{\partial \phi} \\
&= (j2\pi \mathbf{R}^T \boldsymbol{\delta}_{\phi} \odot \mathbf{a}_p) \otimes \mathbf{h} + \mathbf{a}_p \otimes \boldsymbol{\Delta}_{\phi}
\end{aligned} \tag{3.2.19}$$

where $\mathbf{R} \triangleq [\mathbf{r}_1 \ \mathbf{r}_2 \ \dots \ \mathbf{r}_{4M}]$ is a matrix containing the element positions in units of wavelengths. The first term in Equation 3.2.19 is the derivative of the phase component; the part in parentheses is the equivalent derivative for a pressure-sensor array. The second term is the corresponding derivative for the directional gain. Similarly, the elevation derivative is

$$\mathbf{D}_{\psi} = (j2\pi \mathbf{R}^T \boldsymbol{\delta}_{\psi} \odot \mathbf{a}_p) \otimes \mathbf{h} + \mathbf{a}_p \otimes \boldsymbol{\Delta}_{\psi}. \tag{3.2.20}$$

It is now easy to calculate the derivatives of the unit vector \mathbf{u} as defined in Equation 3.2.8:

$$\boldsymbol{\delta}_{\phi} = [-\sin \phi \cos \psi ; \cos \phi \cos \psi ; 0] \tag{3.2.21}$$

$$\boldsymbol{\delta}_{\psi} = [-\cos \phi \sin \psi ; -\sin \phi \sin \psi ; \cos \psi]. \tag{3.2.22}$$

Without loss of generality, I assume the origin of the coordinate system is the array centroid. As was mentioned in [2], the three vectors $\{\mathbf{u}, \boldsymbol{\delta}_{\phi}, \boldsymbol{\delta}_{\psi}\}$ are orthogonal as illustrated in Figure 3.2.1. From this, it is easy to see that the vectors $\{\mathbf{h}, \boldsymbol{\Delta}_{\phi}, \boldsymbol{\Delta}_{\psi}\}$ are also orthogonal. Their orthogonality along with the choice of origin implies $\mathbf{D}_{\Theta}^H \mathbf{v} = \mathbf{0}$ and thus $\mathbf{D}_{\Theta}^H \mathbf{P}_{\mathbf{v}}^{\perp} = \mathbf{D}_{\Theta}^H$. Using these identities in

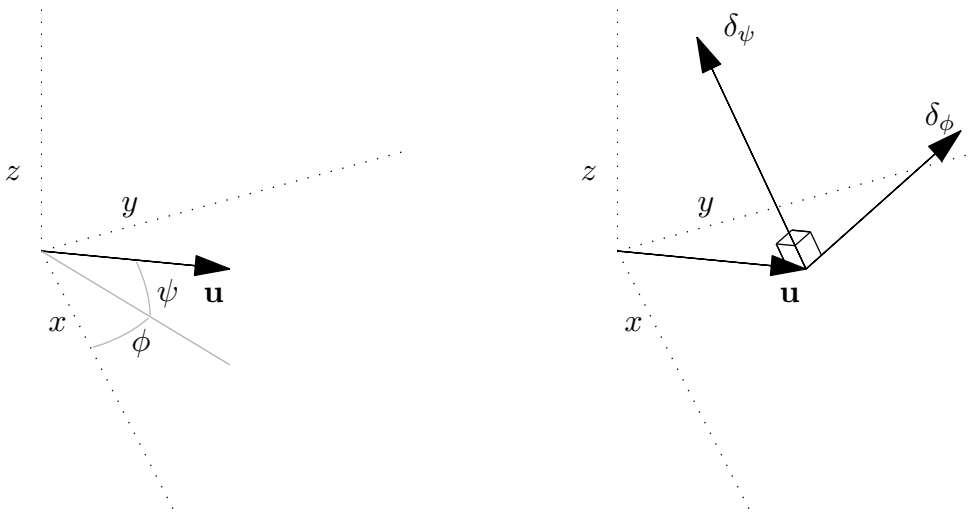


Figure 3.2.1: Orthogonal Vectors $\{\mathbf{u} , \boldsymbol{\delta}_\phi , \boldsymbol{\delta}_\psi\}$

Equation 3.2.14 above gives

$$\begin{aligned} \mathcal{A} &= 2K\Sigma \cdot \text{Re} \{ \mathbf{D}_\Theta^H \mathbf{D}_\Theta \} \\ &= 2K\Sigma \cdot \text{Re} \left\{ \begin{bmatrix} \mathbf{D}_\phi^H \mathbf{D}_\phi & \mathbf{D}_\phi^H \mathbf{D}_\psi \\ \mathbf{D}_\psi^H \mathbf{D}_\phi & \mathbf{D}_\psi^H \mathbf{D}_\psi \end{bmatrix} \right\}. \end{aligned} \quad (3.2.23)$$

I now evaluate the terms in this matrix, starting with the off-diagonal term

$$\begin{aligned} \mathbf{D}_\phi^H \mathbf{D}_\psi &= \{ (j2\pi \mathbf{R}^T \boldsymbol{\delta}_\phi \odot \mathbf{a}_p) \otimes \mathbf{h} + \mathbf{a}_p \otimes \boldsymbol{\Delta}_\phi \}^H \\ &\quad \{ (j2\pi \mathbf{R}^T \boldsymbol{\delta}_\psi \odot \mathbf{a}_p) \otimes \mathbf{h} + \mathbf{a}_p \otimes \boldsymbol{\Delta}_\psi \} \\ &= \{ (j2\pi \mathbf{R}^T \boldsymbol{\delta}_\phi \odot \mathbf{a}_p) \otimes \mathbf{h} \}^H \{ (j2\pi \mathbf{R}^T \boldsymbol{\delta}_\psi \odot \mathbf{a}_p) \otimes \mathbf{h} \} \\ &\quad + \{ \mathbf{a}_p \otimes \boldsymbol{\Delta}_\phi \}^H \{ \mathbf{a}_p \otimes \boldsymbol{\Delta}_\psi \} \\ &= 4\pi^2(\eta^2 + 1) \cdot \boldsymbol{\delta}_\phi^T (\mathbf{R} \mathbf{R}^T) \boldsymbol{\delta}_\psi + M \cdot \boldsymbol{\delta}_\phi^T \boldsymbol{\delta}_\psi. \end{aligned} \quad (3.2.24)$$

The first step above eliminates the cross-terms because the vectors $\{\mathbf{h} , \boldsymbol{\Delta}_\phi , \boldsymbol{\Delta}_\psi\}$ are orthogonal; the second step substitutes the norms of \mathbf{a}_p and \mathbf{h} . Modifying

Equation 3.2.24, I easily get the remaining terms to yield the simplified form

$$\mathbf{A} = 2K\Sigma \left\{ 4\pi^2(\eta^2 + 1) \begin{bmatrix} \boldsymbol{\delta}_\phi^T(\mathbf{R}\mathbf{R}^T)\boldsymbol{\delta}_\phi & \boldsymbol{\delta}_\phi^T(\mathbf{R}\mathbf{R}^T)\boldsymbol{\delta}_\psi \\ \boldsymbol{\delta}_\phi^T(\mathbf{R}\mathbf{R}^T)\boldsymbol{\delta}_\psi & \boldsymbol{\delta}_\psi^T(\mathbf{R}\mathbf{R}^T)\boldsymbol{\delta}_\psi \end{bmatrix} + M \begin{bmatrix} \cos^2 \psi & 0 \\ 0 & 1 \end{bmatrix} \right\} \quad (3.2.25)$$

which is the same expression given in [2] when there is no position or orientation uncertainty. The second term in this equation contains the inner products of the orthogonal vectors $\boldsymbol{\delta}_\phi$ and $\boldsymbol{\delta}_\psi$ and is thus diagonal. It is also easy enough to see that the second term in Equation 3.2.24 combines with the first to give

$$\mathbf{A} = 2K\Sigma \cdot [\boldsymbol{\delta}_\phi \ \boldsymbol{\delta}_\psi]^T \{ 4\pi^2(\eta^2 + 1) \cdot \mathbf{R}\mathbf{R}^T + M \cdot \mathbf{I} \} [\boldsymbol{\delta}_\phi \ \boldsymbol{\delta}_\psi]. \quad (3.2.26)$$

This representation is appealing because the array geometry only enters through the term in curly brackets, specifically through $\mathbf{R}\mathbf{R}^T$. Likewise, the source position only enters through the matrix term $[\boldsymbol{\delta}_\phi \ \boldsymbol{\delta}_\psi]$.

Having derived an expression for \mathbf{A} , I move to the next term \mathbf{B} . Recall that the paragraph above showed $\mathbf{D}_\Theta^H \mathbf{P}_\mathbf{v}^\perp = \mathbf{D}_\Theta^H$. Using this, I now seek the simpler but equivalent expression

$$\mathbf{B} = 2K\Sigma \cdot \text{Re} \left\{ \left[\mathbf{D}_{\Theta_j}^* \odot \mathbf{D}_{\rho_i} \right] \right\}. \quad (3.2.27)$$

Because the \mathbf{B} matrix is in block form, I begin by looking at a single block term $\mathbf{D}_\phi^* \odot \mathbf{D}_\mathbf{x}$. Having already computed \mathbf{D}_ϕ above, I start with $\mathbf{D}_\mathbf{x}$. Although $\mathbf{D}_\mathbf{x}$ is defined in Equation 3.2.7 using derivatives of each of the $4M$ sensor elements, it is easier to consider the derivative taken over each of the M vector-sensors. For the k th vector-sensor,

$$\begin{aligned} \frac{\partial \mathbf{v}_k}{\partial x_k} &= \frac{\partial}{\partial x_k} (a_{p_k} \otimes \mathbf{h}) \\ &= \mathbf{h} \cdot \frac{\partial}{\partial x_k} \exp \{ j2\pi(\mathbf{r}_k^T \mathbf{u}) \} \\ &= j2\pi u_x \cdot \mathbf{v}_k \end{aligned} \quad (3.2.28)$$

where u_x is the x-component of the unit vector \mathbf{u} . To make the first step shown, I use the property described above: the vector \mathbf{h} is invariant with respect to changes in position. Applying this result to every vector-sensor makes the complete derivative

$$\begin{aligned}\mathbf{D}_{\mathbf{x}} &= \left[\frac{\partial \mathbf{v}_1}{\partial x_1} ; \frac{\partial \mathbf{v}_2}{\partial x_2} ; \dots ; \frac{\partial \mathbf{v}_M}{\partial x_M} \right] \\ &= j2\pi u_x \cdot \mathbf{v}\end{aligned}\tag{3.2.29}$$

with similar results for $\mathbf{D}_{\mathbf{y}}$ and $\mathbf{D}_{\mathbf{z}}$. Denoting an M -length vector of ones with $\mathbf{1}_M$, I now compute the element-wise product

$$\begin{aligned}\mathbf{D}_{\phi}^* \odot \mathbf{D}_{\mathbf{x}} &= \left\{ (j2\pi \mathbf{R}^T \boldsymbol{\delta}_{\phi} \odot \mathbf{a}_p) \otimes \mathbf{h} + \mathbf{a}_p \otimes \boldsymbol{\Delta}_{\phi} \right\}^* \odot \{ j2\pi u_x \mathbf{a}_p \otimes \mathbf{h} \} \\ &= (4\pi^2 u_x \mathbf{R}^T \boldsymbol{\delta}_{\phi} \odot \mathbf{a}_p^* \odot \mathbf{a}_p) \otimes (\mathbf{h} \odot \mathbf{h}) \\ &\quad + (j2\pi u_x \mathbf{a}_p^* \odot \mathbf{a}_p) \otimes (\mathbf{h} \odot \boldsymbol{\Delta}_{\phi}) \\ &= (4\pi^2 u_x \mathbf{R}^T \boldsymbol{\delta}_{\phi}) \otimes (\mathbf{h} \odot \mathbf{h}) + (j2\pi u_x \mathbf{1}_M) \otimes (\mathbf{h} \odot \boldsymbol{\Delta}_{\phi})\end{aligned}\tag{3.2.30}$$

where the first step applies the Kronecker mixed-product property and the second uses the identity $\mathbf{a}_p^* \odot \mathbf{a}_p = \mathbf{1}_M$. Since I am only interested in the real part of this result, I need only the first term in Equation 3.2.30,

$$\text{Re} \{ \mathbf{D}_{\phi}^* \odot \mathbf{D}_{\mathbf{x}} \} = 4\pi^2 u_x \mathbf{R}^T \boldsymbol{\delta}_{\phi} \otimes (\mathbf{h} \odot \mathbf{h}).\tag{3.2.31}$$

From this result, it is easy to extrapolate every analogous term with $\{\mathbf{D}_{\mathbf{x}}, \mathbf{D}_{\mathbf{y}}, \mathbf{D}_{\mathbf{z}}\}$ and with \mathbf{D}_{ψ} . The only remaining terms are those involving the rotation parameters such as $\mathbf{D}_{\phi}^* \odot \mathbf{D}_{\boldsymbol{\alpha}}$. When I computed the position perturbation terms $\{\mathbf{D}_{\mathbf{x}}, \mathbf{D}_{\mathbf{y}}, \mathbf{D}_{\mathbf{z}}\}$, I simply differentiated with respect to existing position parameters \mathbf{r}_k . To follow the same procedure for the rotation parameters, I incorporate a rotation matrix about the three axes, $\mathbf{Q}(\alpha, \beta, \gamma)^T$. Suppressing the arguments of \mathbf{Q} , substitute the rotated

vector $\mathbf{h} = [\eta ; \mathbf{Q}^T \mathbf{u}]$ and write³

$$\Delta_\alpha = \left[0 ; \left(\frac{\partial \mathbf{Q}}{\partial \alpha} \bigg|_{\alpha=0} \right)^T \mathbf{u} \right]. \quad (3.2.32)$$

Using the same rotation matrix, analogous expressions result for Δ_β and Δ_γ . An expression for the rotation matrix may be easily found elsewhere, but I list the Δ_\star terms for convenience

$$\Delta_\alpha = [0 ; 0 ; \sin \psi ; -\sin \phi \cos \psi] \quad (3.2.33)$$

$$\Delta_\beta = [0 ; -\sin \psi ; 0 ; \cos \phi \cos \psi] \quad (3.2.34)$$

$$\Delta_\gamma = [0 ; \sin \phi \cos \psi ; -\cos \phi \cos \psi ; 0]. \quad (3.2.35)$$

Using these derivatives, computing one rotation perturbation term gives

$$\begin{aligned} \frac{\partial \mathbf{v}_k}{\partial \alpha_k} &= \frac{\partial}{\partial \alpha_k} (a_{p_k} \otimes \mathbf{h}) \\ &= a_{p_k} \otimes \Delta_\alpha \end{aligned} \quad (3.2.36)$$

$$\therefore \mathbf{D}_\alpha = \mathbf{a}_p \otimes \Delta_\alpha. \quad (3.2.37)$$

I can now express the element-wise product

$$\begin{aligned} \mathbf{D}_\phi^* \odot \mathbf{D}_\alpha &= \{ (j2\pi \mathbf{R}^T \boldsymbol{\delta}_\phi \odot \mathbf{a}_p) \otimes \mathbf{h} + \mathbf{a}_p \otimes \Delta_\phi \}^* \odot \{ \mathbf{a}_p \otimes \Delta_\alpha \} \\ &= (-j2\pi \mathbf{R}^T \boldsymbol{\delta}_\phi \odot \mathbf{a}_p^* \odot \mathbf{a}_p) \otimes (\mathbf{h} \odot \Delta_\alpha) \\ &\quad + (\mathbf{a}_p^* \odot \mathbf{a}_p) \otimes (\Delta_\alpha \odot \Delta_\phi) \\ &= (-j2\pi \mathbf{R}^T \boldsymbol{\delta}_\phi) \otimes (\mathbf{h} \odot \Delta_\alpha) + (\mathbf{1}_M) \otimes (\Delta_\alpha \odot \Delta_\phi). \end{aligned} \quad (3.2.38)$$

As before, I am only interested in the real part

$$\text{Re} \{ \mathbf{D}_\phi^* \odot \mathbf{D}_\alpha \} = (\mathbf{1}_M) \otimes (\Delta_\alpha \odot \Delta_\phi). \quad (3.2.39)$$

³Because the nominal rotations are zero, this is simply a more verbose definition of \mathbf{h} and does not change any previous results.

Enough representative terms have been derived now to write the matrix \mathcal{B} :

$$\begin{aligned}
\mathcal{B} &= 2K\Sigma \cdot \left[\begin{array}{c|c} 4\pi^2 u_x \mathbf{R}^T \boldsymbol{\delta}_\phi \otimes (\mathbf{h} \odot \mathbf{h}) & 4\pi^2 u_x \mathbf{R}^T \boldsymbol{\delta}_\psi \otimes (\mathbf{h} \odot \mathbf{h}) \\ 4\pi^2 u_y \mathbf{R}^T \boldsymbol{\delta}_\phi \otimes (\mathbf{h} \odot \mathbf{h}) & 4\pi^2 u_y \mathbf{R}^T \boldsymbol{\delta}_\psi \otimes (\mathbf{h} \odot \mathbf{h}) \\ 4\pi^2 u_z \mathbf{R}^T \boldsymbol{\delta}_\phi \otimes (\mathbf{h} \odot \mathbf{h}) & 4\pi^2 u_z \mathbf{R}^T \boldsymbol{\delta}_\psi \otimes (\mathbf{h} \odot \mathbf{h}) \\ \hline \mathbf{1}_M \otimes (\boldsymbol{\Delta}_\alpha \odot \boldsymbol{\Delta}_\phi) & \mathbf{1}_M \otimes (\boldsymbol{\Delta}_\alpha \odot \boldsymbol{\Delta}_\psi) \\ \mathbf{1}_M \otimes (\boldsymbol{\Delta}_\beta \odot \boldsymbol{\Delta}_\phi) & \mathbf{1}_M \otimes (\boldsymbol{\Delta}_\beta \odot \boldsymbol{\Delta}_\psi) \\ \mathbf{1}_M \otimes (\boldsymbol{\Delta}_\gamma \odot \boldsymbol{\Delta}_\phi) & \mathbf{1}_M \otimes (\boldsymbol{\Delta}_\gamma \odot \boldsymbol{\Delta}_\psi) \end{array} \right] \\
&= 2K\Sigma \cdot \left[\begin{array}{c} 4\pi^2 \mathbf{u} \otimes (\mathbf{R}^T [\boldsymbol{\delta}_\phi \quad \boldsymbol{\delta}_\psi]) \otimes (\mathbf{h} \odot \mathbf{h}) \\ [\mathbf{1}_M \otimes (\boldsymbol{\Delta}_{\rho_i^\Theta} \odot \boldsymbol{\Delta}_{\Theta_j})] \end{array} \right] \quad (3.2.40)
\end{aligned}$$

where the lower term in the last equation is itself a block matrix.

Having derived expressions for the matrices \mathcal{A} and \mathcal{B} , I naturally turn to the final block \mathcal{C} in the hybrid Cramér-Rao bound. For convenience, I restate the definition in Equation 3.2.16:

$$\mathcal{C} \triangleq 2K\Sigma \cdot Re \left\{ \left[\mathbf{D}_{\rho_i} \mathbf{D}_{\rho_j}^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T \right] \right\} + \boldsymbol{\Lambda}_\rho^{-1}.$$

Having already computed the derivatives \mathbf{D}_\star needed, I immediately begin computing a single term

$$\begin{aligned}
\mathbf{D}_x \mathbf{D}_y^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T &= (j2\pi u_x \mathbf{v})(j2\pi u_y \mathbf{v})^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T \\
&= 4\pi^2 u_x u_y \mathbf{v} \mathbf{v}^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T \\
&= 4\pi^2 u_x u_y (\mathbf{a}_p \mathbf{a}_p^H \otimes \mathbf{h} \mathbf{h}^T) \odot (\mathbf{P}_\mathbf{v}^\perp)^T. \quad (3.2.41)
\end{aligned}$$

Although it may not be immediately obvious, this term itself is real, so taking the real part is not necessary. Analogous results follow for the other position perturbation blocks. Looking at a rotation perturbation block gives

$$\begin{aligned}
\mathbf{D}_\alpha \mathbf{D}_\beta^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T &= (\mathbf{a}_p \otimes \boldsymbol{\Delta}_\alpha)(\mathbf{a}_p \otimes \boldsymbol{\Delta}_\beta)^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T \\
&= (\mathbf{a}_p \mathbf{a}_p^H \otimes \boldsymbol{\Delta}_\alpha \boldsymbol{\Delta}_\beta^T) \odot (\mathbf{P}_\mathbf{v}^\perp)^T \quad (3.2.42)
\end{aligned}$$

which again is a real matrix. Finally, the off-diagonal blocks in the \mathbf{C} matrix are terms like

$$\begin{aligned}\mathbf{D}_x \mathbf{D}_\alpha^H \odot (\mathbf{P}_v^\perp)^T &= (j2\pi u_x \mathbf{a}_p \otimes \mathbf{h})(\mathbf{a}_p \otimes \Delta_\alpha)^H \odot (\mathbf{P}_v^\perp)^T \\ &= j2\pi u_x (\mathbf{a}_p \mathbf{a}_p^H \otimes \mathbf{h} \Delta_\alpha^T) \odot (\mathbf{P}_v^\perp)^T.\end{aligned}\quad (3.2.43)$$

Although the diagonal blocks were real, these terms are purely imaginary. Because their real part is zero, these blocks form matrices of zeros in \mathbf{C} . Writing all blocks together and simplifying gives

$$\mathbf{C}_{1,1} \triangleq 4\pi^2 \mathbf{u} \mathbf{u}^T \otimes [\mathbf{v} \mathbf{v}^H \odot (\mathbf{P}_v^\perp)^T] \quad (3.2.44)$$

$$\mathbf{C}_{2,2} \triangleq [(\mathbf{a}_p \mathbf{a}_p^H \otimes \Delta_{\rho_i^\ominus} \Delta_{\rho_j^\ominus}^T) \odot (\mathbf{P}_v^\perp)^T] \quad (3.2.45)$$

$$\mathbf{C} = 2K\Sigma \cdot \begin{bmatrix} \mathbf{C}_{1,1} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{2,2} \end{bmatrix} + \Lambda_\rho^{-1}. \quad (3.2.46)$$

Although this expression can be expanded, I keep it in this form for brevity.

Having now computed enough terms to evaluate the hybrid Cramér-Rao bound, a few notes are worth mentioning. First, if any perturbations are deterministic or zero, they introduce singularities into the bound. Put another way, deterministic errors should not be included in the CRB. If any perturbation is nonrandom, the corresponding row and column in \mathbf{C}_{HCR}^{-1} should be removed. See [7] for more details and an example. Second, the rotation perturbations for each pressure sensor may be modeled as zero. Based on discussions above, the pressure sensor measurements do not depend on orientation. Combined with the first point, this means I could remove the rows and columns in \mathbf{B} and \mathbf{C} corresponding to pressure sensor rotation errors. Third, the vector-sensor array deteriorates into a pressure-sensor array as $\eta \rightarrow \infty$. This means that much of the work I have done, including the CRB, is valid for a pressure-sensor array if I let $\eta \rightarrow \infty$. Equivalently, to bound a pressure-sensor array I can let $\mathbf{h} = [1; 0; 0; 0]$ or the scalar $\mathbf{h} = 1$, although these require changing many of the derivatives I have computed. Thus, the CRB bound on DOA estimation

can be applied to pressure-sensor arrays with only simple modifications.

As a final note, I describe how these expressions might simplify under weak conditions. Specifically, when the rotation errors are uncorrelated with the position errors, the covariance matrix $\mathbf{\Lambda}_\rho$ is block diagonal. This implies that \mathbf{C} is also block diagonal, making its inverse simpler to compute. With a bit of algebra then, the second term in Equation 3.2.18 takes a simpler form. This term, $\mathbf{B}^T \mathbf{C}^{-1} \mathbf{B}$, summarizes the effect of the modeling errors on the bound. A simpler form might reveal analytically which errors dominate the bound under different circumstances and geometries. Under stronger conditions like IID perturbations and a uniform linear array, the bound may even simplify further.

3.2.3 Including Gain and Phase Errors in the CRB

Thus far, I have only considered errors in vector-sensor position and orientation. In practice, however, each sensor also has gain and phase errors. In this section, I consider how the hybrid Cramér-Rao bound changes when Gaussian gain and phase errors are introduced. The first matrix examined in the previous section, \mathbf{A} , is only a function of the source parameters and does not change with the addition of new perturbation parameters. In the matrices \mathbf{B} and \mathbf{C} , however, new gain and phase blocks appear. Denoting the new gain and phase parameters by g and p , I examine the expanded matrices \mathbf{B}' and \mathbf{C}' below.

For reasons that become clear soon, I begin by examining the new gain blocks in the matrix \mathbf{C}' . The element-wise derivative of the replica vector with respect to gain errors is easily computed as

$$\begin{aligned} \mathbf{D}_g &= \left[\frac{\partial}{\partial g_i} \mathbf{v}_i (1 + g_i) \Big|_{g_i=0} \right] \\ &= \mathbf{v} \end{aligned} \tag{3.2.47}$$

where the gain perturbations enter through g_i . The $(1 + g_i)$ term here mirrors the work done on the rotation derivatives in Equation 3.2.32. As before, the nominal

value of the perturbations $g_i = 0$ leads to the expressions I have already derived. I now examine \mathbf{C}' , computing the α - g rotation-gain cross term

$$\begin{aligned}\mathbf{D}_\alpha \mathbf{D}_g^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T &= (\mathbf{a}_p \otimes \Delta_\alpha)(\mathbf{a}_p \otimes \mathbf{h})^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T \\ &= (\mathbf{a}_p \mathbf{a}_p^H \otimes \Delta_\alpha \mathbf{h}^T) \odot (\mathbf{P}_\mathbf{v}^\perp)^T\end{aligned}\tag{3.2.48}$$

which is strictly real. By symmetry, I know the β - g and γ - g blocks are real as well. These real blocks bring up an important point: the addition of gain errors keeps \mathbf{C}' from being block diagonal. The previous section spent time enumerating each block in an effort to show how the resulting term $\mathbf{B}^T \mathbf{C}^{-1} \mathbf{B}$ might simplify when \mathbf{C} is block diagonal. With the addition of gain terms, however, this simplification does not generally happen.

Unfortunately, the phase errors have the same effect. The element-wise derivative of the replica vector with respect to phase errors is also easy to compute:

$$\begin{aligned}\mathbf{D}_p &= \left[\frac{\partial}{\partial p_i} (\mathbf{v}_i \cdot e^{jp_i}) \Big|_{p_i=0} \right] \\ &= j\mathbf{v}.\end{aligned}\tag{3.2.49}$$

Interestingly, this derivative is j times the equivalent gain derivative. Using this relationship, work done in the previous paragraph reveals the α - p , β - p , and γ - p blocks in \mathbf{C}' are zero. I must then look at the blocks including position errors, starting with

$$\begin{aligned}\mathbf{D}_x \mathbf{D}_p^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T &= (j2\pi u_x \mathbf{v})(j\mathbf{v})^H \odot (\mathbf{P}_\mathbf{v}^\perp)^T \\ &= (2\pi u_x \mathbf{v} \mathbf{v}^H) \odot (\mathbf{P}_\mathbf{v}^\perp)^T.\end{aligned}\tag{3.2.50}$$

Recalling Equation 3.2.41, recognize that this term is different only by a real constant, implying it is also real. Just like the rotation-gain blocks in the previous paragraph, the position-phase blocks keep \mathbf{C}' from being block diagonal. That these blocks are nonzero provides important insight into vector-sensors and is discussed in the CRB analysis section.

Although I could continue expanding the new terms in \mathbf{B}' and \mathbf{C}' , the resulting expression would be complicated and provide little insight. So instead, I simply define the new perturbation parameter vector

$$\boldsymbol{\rho}' = [\boldsymbol{\rho} ; g ; p] \quad (3.2.51)$$

and give the updated matrices in terms of $\boldsymbol{\rho}'$. From Equations 3.2.27 and 3.2.16, I easily get

$$\mathbf{B}' = 2K\Sigma \cdot \text{Re} \left\{ \left[\mathbf{D}_{\Theta_j}^* \odot \mathbf{D}_{\rho'_i} \right] \right\} \quad (3.2.52)$$

$$\mathbf{C}' = 2K\Sigma \cdot \text{Re} \left\{ \left[\mathbf{D}_{\rho'_i} \mathbf{D}_{\rho'_j}^H \odot (\mathbf{P}_{\mathbf{v}}^\perp)^T \right] \right\} + \boldsymbol{\Lambda}_{\boldsymbol{\rho}'}^{-1}. \quad (3.2.53)$$

Because the previous sections list every variable and derivative used here, these equations form a good basis for numerically evaluating the hybrid Cramér-Rao bound.

Before moving on, observe a few notes about the hybrid Cramér-Rao bound under gain and phase errors. First, Gaussian phase errors between vector-sensors may be modeled as position error and incorporated into the existing bound. Second, phase errors between elements within a single vector-sensor may be moderated in practice by the fact that phase information is measured redundantly. Third, if each velocity sensor rotates independently, the resulting modeling errors may be approximated as gain errors.

3.2.4 The Maximum Likelihood (ML) Estimator

This chapter has spent a significant amount of effort discussing the Cramér-Rao lower bound for DOA estimation without mentioning any explicit estimation algorithms. Partly, this is because the CRB is a lower bound for the variance of *any* unbiased estimator. Now, however, it is appropriate to look at the maximum likelihood or ML estimator.

I begin by defining the ML estimator and stating its relevance to the CRB. The “likelihood” of some measured data \mathbf{X} given a set of parameters \mathbf{P} is simply the

conditional distribution $p_{\mathbf{X}}(\mathbf{X}; \mathbf{P})$. The maximum likelihood estimator is then

$$\hat{\mathbf{P}}_{ML} \triangleq \arg \max_{\mathbf{P}} p_{\mathbf{X}}(\mathbf{X}; \mathbf{P}). \quad (3.2.54)$$

The ML estimator is generally important for several reasons. First, it is practical to obtain as is shown below. Second, when an efficient estimator exists, it is the ML estimator.⁴ This does not mean, however, that the ML estimator must be efficient or even unbiased. Third, the ML estimator is often asymptotically efficient and consistent.⁵

I now examine the ML estimator under my mismatched data model. Beginning with the signal model in Equation 3.2.1, I need only define the vector \mathbf{y} as the complex magnitude and phase of each received snapshot. The parameter vector to estimate is then

$$\mathbf{P} \triangleq [\boldsymbol{\rho} ; \boldsymbol{\Theta} ; \mathbf{y}]. \quad (3.2.55)$$

The estimation of additive noise is implied as it can be determined from the other parameters. With this definition, the data is a nonlinear deterministic matrix function \mathbf{F} of the parameters added to complex Gaussian noise. The likelihood function is then only the probability of the additive noise given the parameters. Because the noise is IID, this probability is written

$$p_{\mathbf{X}}(\mathbf{X}; \mathbf{P}) = \prod_{i=1}^K \cdot \prod_{j=1}^{4M} \mathcal{CN}(\mathbf{X}_{ij} - \mathbf{F}(\mathbf{P})_{ij} ; 0, \sigma_n^2). \quad (3.2.56)$$

Maximizing the likelihood is equivalent to maximizing the log-likelihood

$$\ln p_{\mathbf{X}}(\mathbf{X}; \mathbf{P}) = -\frac{1}{2\sigma_n^2} \sum_{i=1}^K \sum_{j=1}^{4M} |\mathbf{X}_{ij} - \mathbf{F}(\mathbf{P})_{ij}|^2 - 2KM \ln 2\pi\sigma_n^2. \quad (3.2.57)$$

⁴“Efficient” simply means the estimator achieves the CRB with equality.

⁵A “consistent” estimator asymptotically converges to the correct value.

Ignoring the constant term and the coefficient, I now write the ML estimator as

$$\begin{aligned}\hat{\mathbf{P}}_{ML} &= \arg \min_{\mathbf{P}} \sum_{i=1}^K \sum_{j=1}^{4M} |\mathbf{X}_{ij} - \mathbf{F}(\mathbf{P})_{ij}|^2 \\ &= \arg \min_{\mathbf{P}} \text{tr} \left[(\mathbf{X} - \mathbf{F}(\mathbf{P}))^H (\mathbf{X} - \mathbf{F}(\mathbf{P})) \right]\end{aligned}\quad (3.2.58)$$

where $\text{tr}[\cdot]$ indicates the trace of a matrix, or the sum of its diagonal elements. I now go a little farther by writing

$$\mathbf{F}(\mathbf{P}) = \mathbf{v}(\mathbf{P})\mathbf{y}^T \quad (3.2.59)$$

where $\mathbf{v}(\mathbf{P})$ is the replica vector produced by the parameters \mathbf{P} . At this point, I pause to say something about the nonlinear vector function $\mathbf{v}(\mathbf{P})$: any possible output is produced by more than one input. When both gain and phase errors are considered, I can say something even stronger: any arbitrary output is possible. That is, there are many ways to produce *any* output vector by judiciously choosing \mathbf{P} . This statement has the important implication that the maximum likelihood estimate is not unique. Thus, any ML algorithm only estimates what output \mathbf{v} is produced when the likelihood is maximized. To find this ML output vector, rewrite the optimization problem as

$$\begin{aligned}\hat{\mathbf{P}}_{ML} &= \arg \min_{\mathbf{v}, \mathbf{y}} \text{tr} \left[(\mathbf{X} - \mathbf{v}\mathbf{y}^T)^H (\mathbf{X} - \mathbf{v}\mathbf{y}^T) \right] \\ &= \arg \min_{\mathbf{v}, \mathbf{y}} -2\text{tr} [\mathbf{X}^H \mathbf{v}\mathbf{y}^T] + \text{tr} [\mathbf{y}^* \mathbf{v}^H \mathbf{v}\mathbf{y}^T] \\ &= \arg \max_{\mathbf{v}, \mathbf{y}} 2\text{tr} [\mathbf{X}^H \mathbf{v}\mathbf{y}^T] - \|\mathbf{v}\|^2 \cdot \|\mathbf{y}\|^2 \\ &= \arg \max_{\mathbf{v}, \mathbf{y}} 2\mathbf{y}^T \mathbf{X}^H \mathbf{v} - \|\mathbf{v}\|^2 \cdot \|\mathbf{y}\|^2.\end{aligned}\quad (3.2.60)$$

Because \mathbf{v} is a replica vector, its norm is fixed. The solution to the optimization problem is now evident: given $\|\mathbf{y}\|$ the maximum likelihood is achieved when $\mathbf{y}^T \mathbf{X}^H \mathbf{v}$ is maximized. Using the Cauchy-Schwarz inequality, this occurs when \mathbf{y} is parallel to

$\mathbf{X}^H \mathbf{v}_{ML}$ and

$$\begin{aligned}
\mathbf{v}_{ML} &= \arg \max_{\mathbf{v}} \|\mathbf{X}^H \mathbf{v}\|^2 \\
&= \arg \max_{\mathbf{v}} \mathbf{v}^H \mathbf{R} \mathbf{v} \\
&\propto \boldsymbol{\Phi}_1
\end{aligned} \tag{3.2.61}$$

where $\mathbf{R} = \mathbf{X}\mathbf{X}^H$ is the sample covariance matrix and $\boldsymbol{\Phi}_1$ is the principal eigenvector - the eigenvector with the largest eigenvalue - of \mathbf{R} . Thus, any choice of parameters \mathbf{P} satisfying $\mathbf{v}(\mathbf{P}) \propto \boldsymbol{\Phi}_1$ maximizes the likelihood function. As long as the ML estimator is not unique, an estimator that maximizes the likelihood function does not necessarily satisfy any of the above properties. It may not be asymptotically efficient or consistent; it may never meet the CRB with equality; it may not even be unbiased. Again, a non-unique ML estimator occurs when gain and phase errors are included. Without gain or phase errors, the solution may require a high-dimensional search and both the model and algorithm may be impractical.

3.2.5 Analysis and Examples

Having established a Cramér-Rao bound for direction-of-arrival estimation, I now analyze the bound with examples. After introducing a standard mismatched ULA model, I relate the different sources of modeling errors. Then, I provide some results contrasting pressure and vector-sensor arrays. Finally, I compare simple direction-of-arrival algorithms to the CRB.

Although the Cramér-Rao bound applies for arbitrary arrays, the examples in this section are only uniform linear arrays. To focus the discussion, I use the following standard ULA mirroring the Gilbert-Morgan model in [3]:

- $M = 13$ equally spaced elements on the x -axis
- $\gamma = 1/4$ element level SNR
- $K = 10$ independent snapshots

- IID perturbations
- Analysis at frequency $\frac{7}{8} \cdot f_d$
- 10,000 independent Monte-Carlo trials when simulated
- Single source at $\hat{\phi} = -\pi/3$, $\hat{\psi} = 0$ in x - y plane
- Position error $\sigma_x = \sigma_y = \sigma_z = \lambda_d/10$
- Rotation error $\sigma_\alpha = \sigma_\beta = \sigma_\gamma = \pi/18$
- Sensor gain error $\sigma_g = 0.1$
- Sensor phase error $\sigma_p = \pi/18$

where λ_d is one wavelength at the design frequency f_d . In the following paragraphs, I explicitly note any deviations from this model. In each example, the CRB is computed in cosine-space and plotted in decibels of a beamwidth. Although I only present examples of uniform linear arrays, many of my conclusions apply equally to any array.

Deriving the Cramér-Rao bound matrix \mathbf{C}' with sensor gain and phase errors revealed that only a select few off-diagonal blocks were nonzero. The implied connections between rotation and gain - or similarly between position and phase - are not by chance. Although I introduced vector-sensors as measuring the physical quantities of pressure and particle velocity, one can equivalently think of each vector-sensor as measuring amplitude and phase information. Now the reason for the nonzero off-diagonal terms is clear: sensor gain and rotation errors appear as amplitude measurements but sensor phase and position errors affect phase measurements. Taking this a step farther, consider the asymptotic CRB as SNR or observation time increases. In this limit if either amplitude or phase measurements are error-free, one can exactly determine direction-of-arrival and the CRB decays to zero. These relationships between modeling errors are illustrated in Figures 3.2.2 and 3.2.3. In these figures, only the given sources of mismatch are present. Notice the striking asymptotic similarities between sensor gain and rotation errors and between sensor phase and position errors.

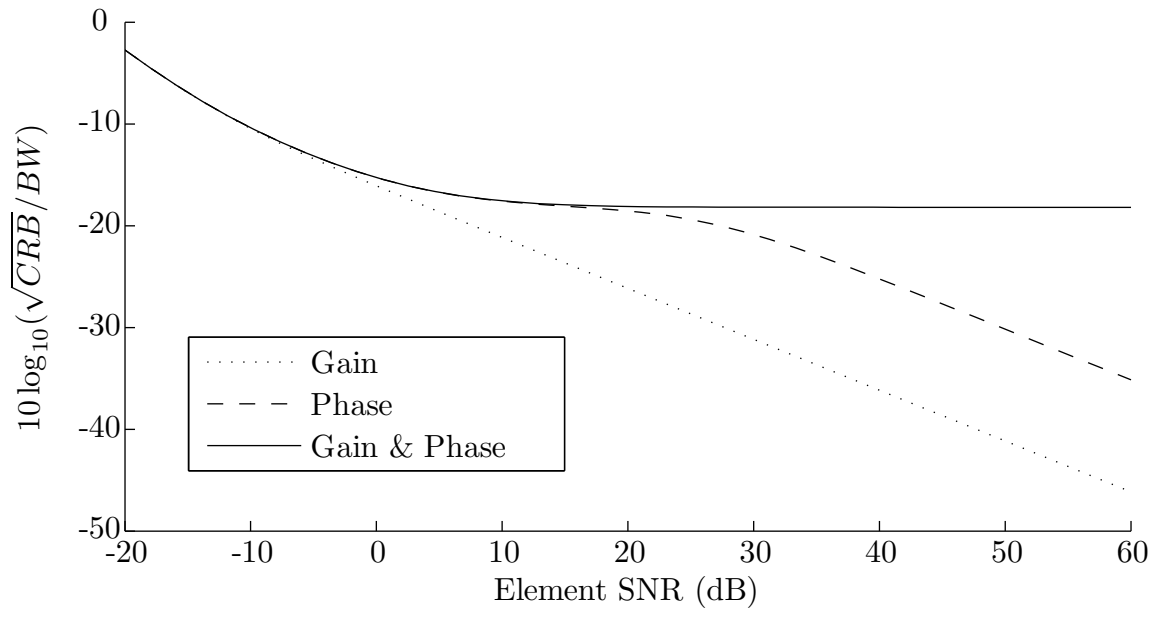


Figure 3.2.2: CRB with Gain and/or Phase Errors

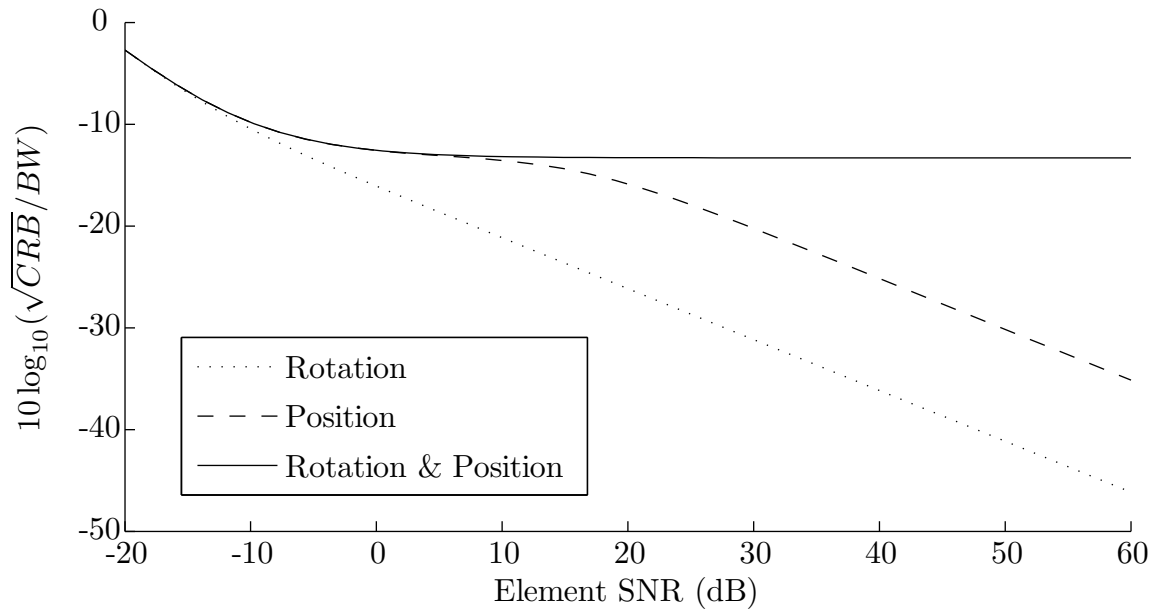


Figure 3.2.3: CRB with Rotation and/or Position Errors

Although the asymptotic region begins at a higher SNR for sensor phase errors, the figures are almost identical. Also note that when amplitude and phase measurement errors are *both* present the bound does not decay to zero. Under these conditions any estimator is asymptotically limited by the mismatch and not by observation time or SNR.

The theoretical advantages of vector-sensor over pressure-sensor ULAs attract much research attention. In this section, I contrast vector and pressure-sensor array Cramér-Rao bounds. First examine the CRB as a function of SNR in Figure 3.2.4.

Because both magnitude and phase measurement errors are modeled, the bounds asymptotically approach a constant. For the acoustic vector-sensor array, however, the bound is several decibels lower indicating better direction-of-arrival performance may be possible. Also note that rotation errors, which do not affect pressure-sensors, increase the vector-sensor CRB only slightly. Figure 3.2.5 reveals similar results when the CRB is plotted versus the number of snapshots, K . In this case, however, rotation errors hardly change the vector-sensor array bound. These plots, combined with the initial beampattern study, suggest that vector-sensor performance may not be overly sensitive to rotation errors. An intuitive explanation is that the geophone response - and similarly, the vector-sensor modulation term B_v - has a very wide mainlobe. For rotation errors to significantly affect the performance, very large errors would need to be present. Both Figures 3.2.4 and 3.2.5 show vector-sensor arrays bounded lower than pressure-sensor arrays. Intuitively, the increased number of measurements should make vector-sensors robust to some modeling errors.

Although the Cramér-Rao bound shows vector-sensors might improve performance, it would be useful to find simple DOA algorithms that operate much better with vector-sensors. This section analyzes the performance of a single direction-of-arrival algorithm compared with the Cramér-Rao bound. The algorithm chosen is a simple matched filter searching over azimuth to maximize CBF power. Although this algorithm is a simple one-dimensional search, it seemed to perform as well asymptot-

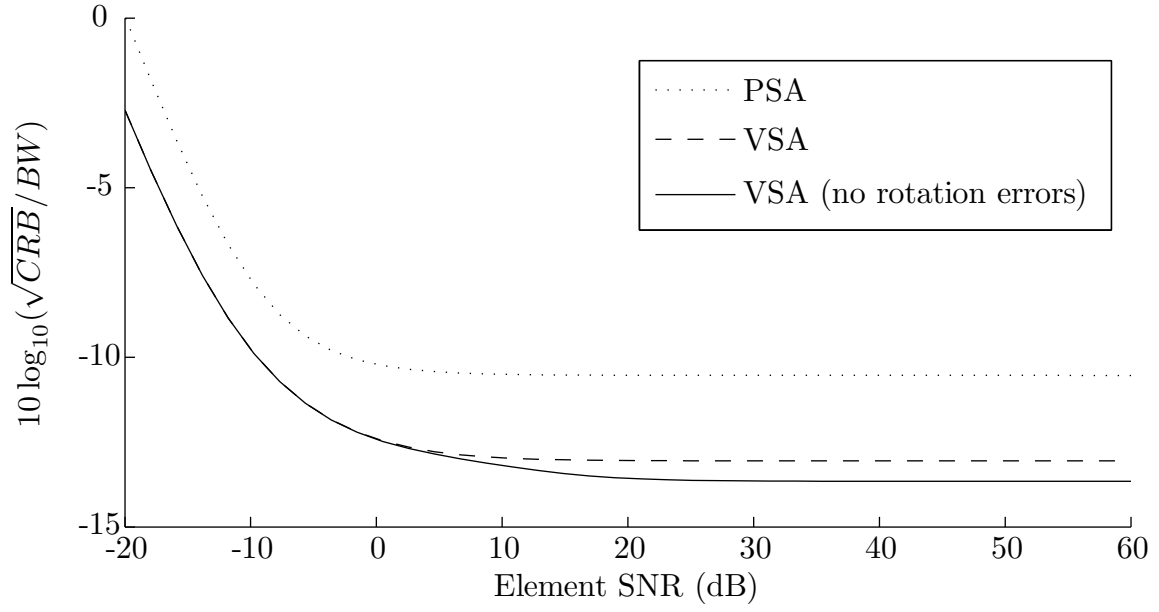


Figure 3.2.4: Pressure and Vector-Sensor Arrays: CRB vs. SNR

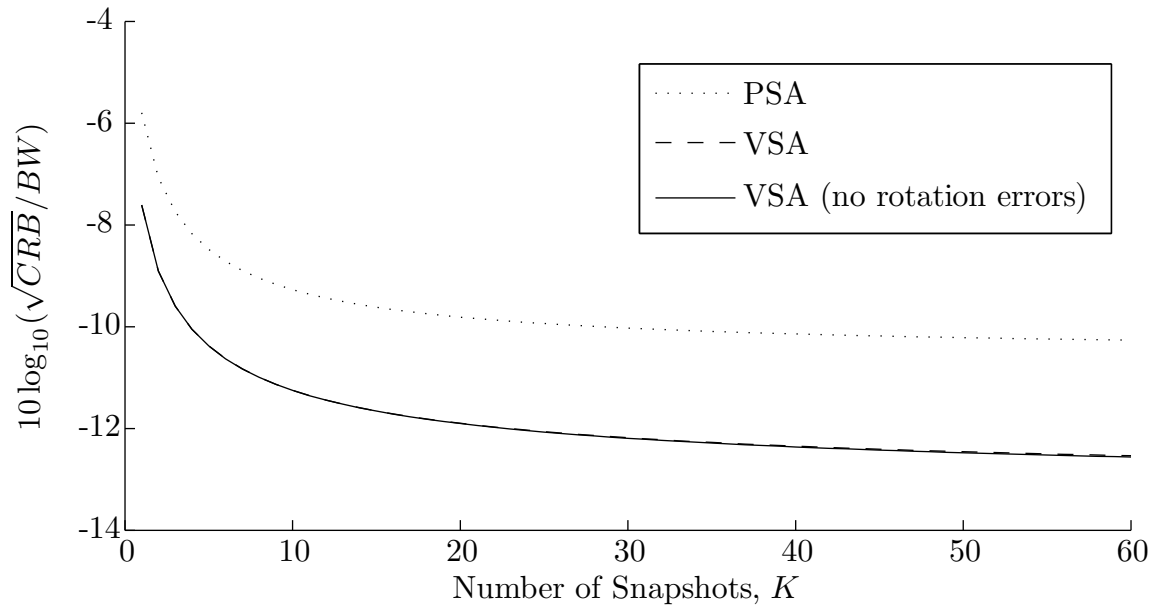


Figure 3.2.5: Pressure and Vector-Sensor Arrays: CRB vs. Snapshots

ically as more complex approaches. In function form, the algorithm maximizes

$$F(\phi) = \|\mathbf{v}(\phi)^H \mathbf{X}\|^2 \quad (3.2.62)$$

where \mathbf{v} is evaluated with $\psi = 0$. Figure 3.2.6 reveals that, with a pressure-sensor ULA, this algorithm is asymptotically very close to the bound. In Figure 3.2.7, however, the algorithm does not meet the vector-sensor CRB. It performs equally well in both cases, despite the additional velocity measurements. These figures suggest that, although DOA performance may possibly improve with vector-sensor arrays, efficient algorithms could be difficult to develop. In particular, MAP self-calibration algorithms might approach the CRB but require much more computation given the high-dimensional search space. In this discussion, recall that a principal motivation behind acoustic vector-sensors is not their *direction-of-arrival* performance but their ability to *resolve pressure-sensor ambiguities*.

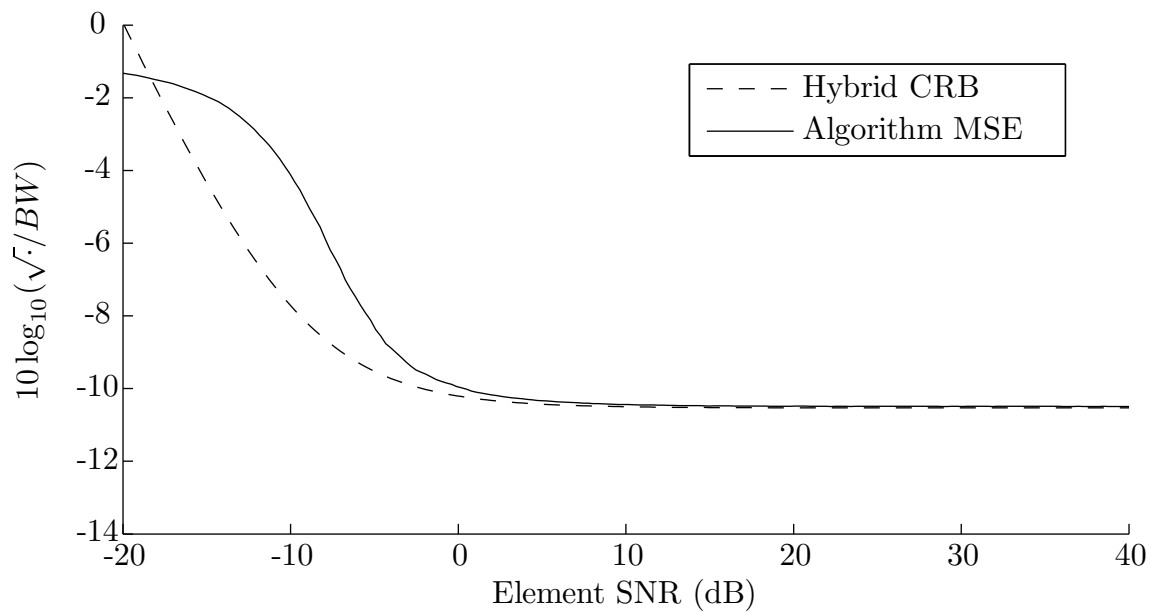


Figure 3.2.6: Pressure-Sensor Array Algorithm Performance

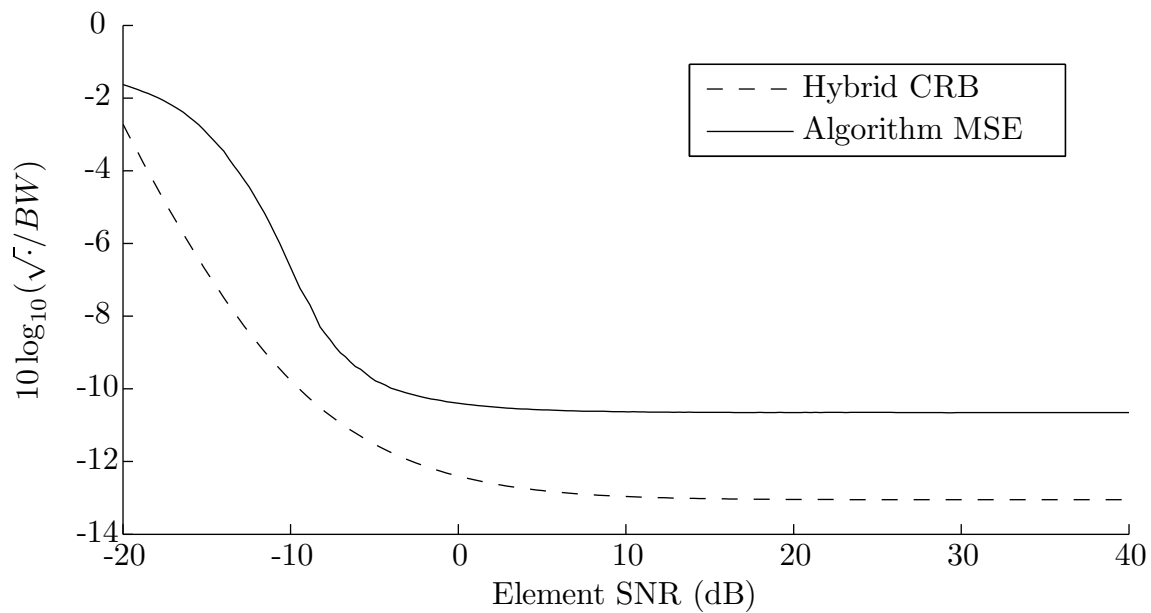


Figure 3.2.7: Vector-Sensor Array Algorithm Performance

Chapter 4

Conclusion

The performance tools developed in this thesis should be useful both in theory and practice. As acoustic vector-sensors increase in capability and number, a more complete picture of their performance will aid in the design of new arrays and the analysis of data.

4.1 Summary

Just as this thesis organizes into two parts, the conclusions also fall into two categories. First, there are initial observations about ideal arrays, or arrays without mismatch. Second, these results are leveraged to analyze vector-sensor arrays under Gaussian modeling errors.

The first section of this thesis ignores the effects of modeling errors to provide tools like those used with pressure-sensor arrays. It reveals the importance of a beampattern factorization in developing simple intuition for the vector-sensor array response. It also quantifies the ability of vector-sensor CBF to null pressure-sensor ambiguities.

The second section of this thesis shows that like pressure-sensor arrays, vector-sensor arrays are reasonably robust to mismatch. It develops a useful beampattern expression under position mismatch and a hybrid Cramér-Rao bound for DOA estimation under Gaussian modeling errors. Analysis of the CRB implies that although

vector-sensors require additional orientation parameters, DOA performance is not overly sensitive to rotation mismatch. Furthermore, the CRB hints that performance may be improved with vector-sensors because of the increased number of measurements.

4.2 Future Work

This thesis is by no means comprehensive, and could be extended in several directions. First, one could research alternative performance bounds. Some bounds, such as the Ziv-Zakai bound in [12], are more complex but can be tighter than the CRB. Analysis of these bounds could better quantify rotation errors or approximate their effect on vector-sensor performance. Second, one could design practical algorithms to exploit vector-sensors. Algorithm design goals might be to approach the theoretical performance limit, stay computationally efficient, and remain robust to mismatch.

Appendix A

Nomenclature

A.1 Acronyms

Acronym	Description
CBF	Conventional Beamforming
CRB	Cramér-Rao Bound
DFT	Discrete Fourier Transform
DOA	Direction of Arrival
HCR	Hybrid Cramér-Rao Bound
IID	Independent and Identically Distributed
MAP	Maximum a Posteriori
ML	Maximum Likelihood
PSA	Pressure-sensor Array
PSD	Power Spectrum Density
SNR	Signal to Noise Ratio
ULA	Uniform Linear Array
VSA	Vector-sensor Array

A.2 Notation

Notation	Description	Example
a	Scalar variable	Eqn. 3.1.4
\mathbf{a}	Vector variable	Eqn. 1.5.1
a_m	m^{th} element of vector \mathbf{a} unless stated otherwise	Eqn. 1.5.6
\mathbf{a}^H	Conjugate (or Hermitian) transpose	Eqn. 2.1.2
\mathbf{a}^*	Conjugation	Eqn. 3.2.15
\mathbf{a}^T	Transpose	Eqn. 1.5.1
$\mathbf{a} \odot \mathbf{b}$	Element-wise (or Hadamard) product	Eqn. 3.2.16
$\mathbf{a} \otimes \mathbf{b}$	Tensor (or Kronecker) product	Eqn. 2.1.1
$[\mathbf{a} \ , \ \mathbf{b}]$ or $[\mathbf{a} \ \mathbf{b}]$	Horizontal concatenation	Eqns. 1.5.1 or 3.2.26
$[\mathbf{a} \ ; \ \mathbf{b}]$ or $\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}$	Vertical concatenation	Eqns. 3.2.4 or 1.5.7
\star	An arbitrary variable	Sec. 3.2.1
$[\mathbf{A}_{\alpha_i \alpha_j}]$	Block matrix notation	Eqn. 3.2.15 or Sec. 3.2.1
$\mathcal{CN}(\mu, \sigma^2)$	Complex Gaussian random variable with mean μ and variance σ^2	Eqn. 3.2.1

Bibliography

- [1] A. Nehorai and E. Paldi. Acoustic vector-sensor array processing. *IEEE Trans. Signal Processing*, 42(9):2481–2491, September 1994.
- [2] M. Hawkes and A. Nehorai. Effects of sensor placement on acoustic vector-sensor array performance. *IEEE J. Oceanic Eng.*, 24(1):33–40, January 1999.
- [3] A. J. Poulsen and A. B. Baggeroer. Vector sensor array sensitivity and mismatch: generalization of the Gilbert-Morgan formula. In *Fourth Joint Meeting of the Acoustical Society of America and the Acoustical Society of Japan*, December 2006.
- [4] H. Cox and H. Lai. Performance of line arrays of vector and higher order sensors. In *IEEE Asilomar Conference*, November 2007.
- [5] M. Hawkes and A. Nehorai. Acoustic vector-sensor beamforming and capon direction estimation. *IEEE Trans. Signal Processing*, 46(9):2291–2304, September 1998.
- [6] F. B. Jensen, W. A. Kuperman, M. B. Porter, and H. Schmidt. *Computational Ocean Acoustics*. Springer-Verlag New York, Inc., Fifth Avenue, New York, NY 10010, 2000.
- [7] H. L. Van Trees. *Optimum Array Processing. Part IV of Detection, Estimation, and Modulation Theory*. John Wiley & Sons, Inc., New York, NY, 2002.
- [8] C. A. Balanis. *Antenna Theory: Analysis and Design*. Wiley-Interscience, Hoboken, NJ, 3rd edition, 2005.

- [9] E. N. Gilbert and S. P. Morgan. Optimum design of directive antenna arrays subject to random variations. *The Bell System Technical Journal*, 34:637–663, May 1955.
- [10] B. D. Steinberg. *Principles of Aperture and Array System Design*. Wiley-Interscience, New York, NY, 1976.
- [11] S. Maymon. Nonuniform sampling. Presented to MIT DSP Group, December 2007.
- [12] W. Xu, A. B. Baggeroer, and K. L. Bell. A bound on mean-square estimation error with background parameter mismatch. *IEEE Trans. Information Theory*, 50(4):621–632, April 2004.