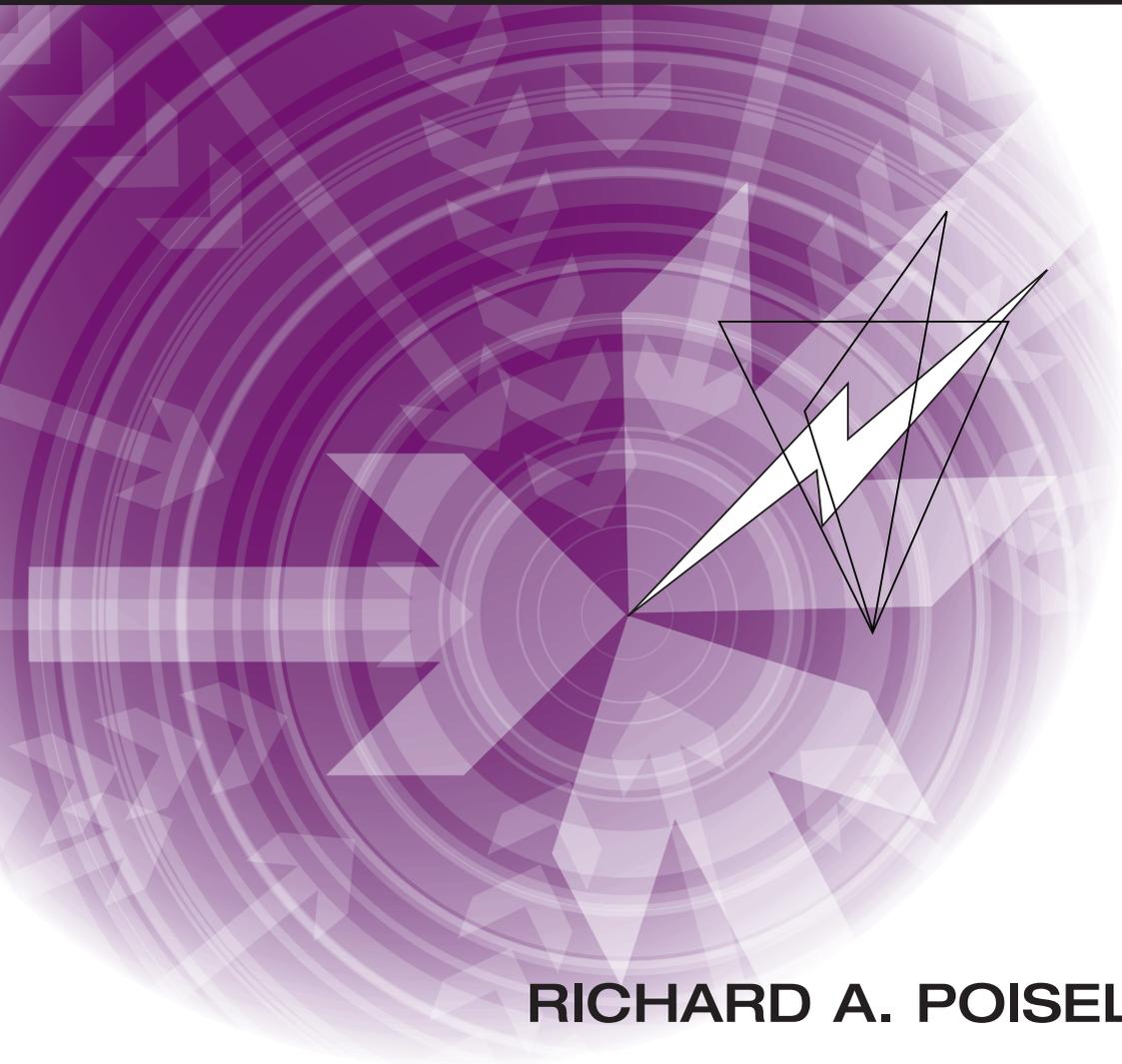


# ELECTRONIC WARFARE TARGET LOCATION METHODS

SECOND EDITION



**RICHARD A. POISEL**

# **Electronic Warfare Target Location Methods**

**Second Edition**

For a complete listing of titles in the  
*Artech House Radar Library*,  
turn to the back of this book.

# **Electronic Warfare Target Location Methods**

**Second Edition**

Richard A. Poisel



**ARTECH  
HOUSE**

BOSTON | LONDON  
[artechhouse.com](http://artechhouse.com)

**Library of Congress Cataloging-in-Publication Data**

A catalog record for this book is available from the U.S. Library of Congress.

**British Library Cataloguing in Publication Data**

A catalogue record for this book is available from the British Library.

**Cover design by Vicki Kane**

ISBN 13: 978-1-60807-523-2

© 2012 ARTECH HOUSE

685 Canton Street

Norwood, MA 02062

All rights reserved. Printed and bound in the United States of America. No part of this book may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Artech House cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

10 9 8 7 6 5 4 3 2 1

*To Herb Hovey (posthumously),  
my friend, mentor, and leader.  
He gave me all the rope  
I needed to sink or swim.*



# Contents

Preface		xv
Chapter 1	Introduction to Emitter Geolocation	1
1.1	Introduction	1
1.2	Gradient Descent Algorithm	3
1.3	Concluding Remarks	12
	References	12
Chapter 2	Triangulation	15
2.1	Introduction	15
2.2	Basic Concepts	16
2.3	Least-Squares Error Estimation	20
2.4	Total Least-Squares Estimation	27
2.5	Least-Squares Distance Error PF Algorithm	30
2.5.1	Brown's Least-Squares Triangulation Algorithm	30
2.5.2	Hemispheric Least-Squares Error Estimation Algorithm	34
2.5.3	Pages-Zamora Least-Squares	41
2.5.4	Total Least-Squares Error	45
2.6	Minimum Mean-Squares Error Estimation	47
2.6.1	Dynamical Systems	47
2.6.2	Linear Minimum Mean-Squares Estimation	52
2.6.3	Target Location Estimation with the Linear Model	59
2.6.4	Kalman Filter Methods	62
2.7	The Discrete Probability Density Method	75
2.8	Generalized Bearings	80
2.9	Maximum Likelihood PF Algorithm	84
2.9.1	Maximum Likelihood Estimation Triangulation Algorithm	86
2.9.2	Maximum Likelihood Estimation Algorithm Comparison	90
2.10	Multiple Sample Correlation	99

2.11	Bearing-Only Target Motion Analysis	104
2.12	Sources of Error in Triangulation	110
	2.12.1 Geometric Dilution of Precision in Triangulation	110
	2.12.2 LOB Error	112
	2.12.3 Effects of Bias on Bearing-Only PF	112
	2.12.4 Combining Noisy LOB Measurements	116
	2.12.5 Effects of Navigation Error	122
2.13	Concluding Remarks	126
	References	129
	Appendix 2A Least-Squares Error Estimation Program Listing	131
	Appendix 2B Generalized Bearing Program Listing	135
Chapter 3	DF Techniques	139
3.1	Introduction	139
3.2	Array Processing Direction of Arrival Measurement Methods	140
	3.2.1 Introduction	140
	3.2.2 The Model	141
	3.2.3 Array Covariance Modeling	144
	3.2.4 Direction of Arrival	146
	3.2.5 Subspace-Based Methods	147
	3.2.6 Beamforming AOA Estimation	149
	3.2.7 Maximum Likelihood AOA Estimation	151
	3.2.8 Least-Square Error AOA Estimation	151
	3.2.9 Decoupling Sample Source Signals from AOA Parameters	152
	3.2.10 Gram-Schmidt Orthogonalization	153
	3.2.11 Nonlinear Programming	154
3.3	Other Methods of Estimating the AOA	156
	3.3.1 Phase Interferometry	156
	3.3.2 Amplitude Systems	157
	3.3.3 Doppler Direction Finder	159
3.4	MSE Phase Interferometer	159
	3.4.1 Introduction	159
	3.4.2 The Algorithm	161
	3.4.3 Simulation	166
3.5	DF with a Butler Matrix	168
	3.5.1 Introduction	168
	3.5.2 Beamforming Network	169
	3.5.3 Summary	180
3.6	Phase Difference Estimation Using SAW Devices	180
	3.6.1 Introduction	180

3.6.2	SAW Characteristics	182
3.7	Concluding Remarks	183
	References	186
Chapter 4	MUSIC	189
4.1	Introduction	189
4.2	MUSIC Overview	190
4.3	MUSIC	191
4.3.1	The MUSIC Algorithm	191
4.4	Performance of MUSIC in the Presence of Modeling Errors	193
4.4.1	Model Errors	193
4.4.2	Error Expressions	195
4.4.3	Results	199
4.5	Determining the Number of Wavefields	201
4.6	Effect of Phase Errors on the Accuracy of MUSIC	203
4.6.1	Introduction	203
4.6.2	Accuracy	204
4.6.3	Solutions for Errors	206
4.6.4	Statistics	207
4.6.5	Horizontal Planar Arrays	209
4.6.6	Simulations	211
4.6.7	Summary	213
4.7	Other Superresolution Algorithms	213
4.7.1	Maximum Likelihood Method	215
4.7.2	Adaptive Angular Response	215
4.7.3	Thermal Noise Algorithm	215
4.7.4	Maximum Entropy Method	215
4.7.5	Comparisons	217
4.8	Concluding Remarks	221
	References	223
Chapter 5	Quadratic Position-Fixing Methods	225
5.1	Introduction	225
5.2	TDOA Position-Fixing Techniques	226
5.2.1	Introduction	226
5.2.2	TDOA	227
5.2.3	Calculating the PF with TDOAs	229
5.2.4	Nonlinear Least-Squares	233
5.2.5	TDOA Measurement Accuracy	234
5.2.6	TDOA PFs with Noisy Measurements	237
5.2.7	TDOA Dilution of Precision	244

	5.2.8	Bias Effects of TDOA PF Estimation	248
	5.2.9	Effects of Movement on TDOA PF Estimation	252
5.3		Differential Doppler	255
	5.3.1	Introduction	255
	5.3.2	DD	256
	5.3.3	DD Measurement Accuracy	257
	5.3.4	Maximum Likelihood DD Algorithms	262
	5.3.5	Cross-Ambiguity Function	267
	5.3.6	Estimating the DD of a Sinusoid in Noise Using Phase Data	268
	5.3.7	Effects of Motion on DD PF Estimating	271
5.4		Range Difference Methods	273
	5.4.1	Introduction	273
	5.4.2	Least-Squared Range Difference Methods	274
	5.4.3	Range Difference Using Feasible Bivectors	281
5.5		Concluding Remarks	291
		References	291
Chapter 6		Time Delay Estimation	293
	6.1	Introduction	293
	6.2	System Overview	293
	6.3	Cross Correlation	296
	6.3.1	Error Analysis of the Cross-Correlation Method	296
	6.3.2	Flat Noise Spectra: Arbitrary Signal Spectrum	300
	6.3.3	Flat Noise Spectra: Flat Signal Spectrum	301
	6.4	Generalized Cross-Correlation	302
	6.5	Estimating the Time Delay with the Generalized Correlation Method	306
	6.5.1	Roth Process	307
	6.5.2	Smoothed Coherence Transform (SCOT)	308
	6.5.3	Phase Transform (PHAT)	309
	6.5.4	Eckart Filter	310
	6.5.5	Maximum Likelihood	310
	6.5.6	Variance of the Delay Estimators	311
	6.6	Time Delay Estimation Using the Phase of the Cross-Spectral Density	311
	6.6.1	Introduction	311
	6.6.2	Data Model	312
	6.6.3	Properties of the Sample CSD	313
	6.6.4	TDOA Estimation	315
	6.6.5	Cramer-Rao Bound	316
	6.6.6	Other Considerations	319

6.6.7	Summary	321
6.7	Effects of Frequency and Phase Errors in EW TDOA Direction-Finding Systems	321
6.7.1	Introduction	321
6.7.2	Perfect Synchronization	321
6.7.3	Errors in Synchronization	322
6.7.4	Effects of Finite Sample Time	323
6.7.5	White Noise Signal	325
6.7.6	Simulation Results	326
6.7.7	Estimator Performance	327
6.7.8	Ramifications	327
6.7.9	Summary	332
6.8	Concluding Remarks	332
	References	332
Chapter 7	Single-Site Location Techniques	335
7.1	Introduction	335
7.2	HF Signal Propagation	335
	7.2.1 Ionograms	338
	7.2.2 Magnetic Field Effects	340
7.3	Single-Site Location	341
7.4	Passive SSL	341
7.5	Determining the Reflection Delay with the Cepstrum	345
7.6	MUSIC Cepstrum SSL	349
7.7	Earth Curvature	353
7.8	Skywave DF Errors	354
	7.8.1 Introduction	354
	7.8.2 Magnetic Field Effects	355
	7.8.3 Ross Curve	355
	7.8.4 Bailey Curve	355
7.9	Ray Tracing	355
	7.9.1 Parabolic Modeling	361
7.10	Accuracy Comparison of SSL and Triangulation for Ionospherically Propagated Signals	362
	7.10.1 Introduction	362
	7.10.2 Spherical Model	365
	7.10.3 Plane Model	368
	7.10.4 Comparison Between SSL and Triangulation	370
	7.10.5 Summary	373
7.11	Concluding Remarks	373
	References	373

Appendix A	Grassmann Algebra	375
A.1	Background	375
A.2	Introduction	375
A.3	Exterior Product	376
A.3.1	Properties of the Exterior Product	377
A.3.2	$m$ -Vectors	379
A.4	Regressive Product	380
A.4.1	Unions and Intersections of Spaces	381
A.4.2	Properties of the Regressive Product	381
A.4.3	The Common Factor Axiom	383
A.4.4	The Common Factor Theorem	384
A.4.5	The Intersection of Two Bivectors in a 3-D Space	384
A.5	Geometric Interpretations	385
A.5.1	Points and Vectors	385
A.5.2	Sums and Differences of Points	386
A.5.3	Lines and Planes	387
A.5.4	Intersection of Two Lines	389
A.6	The Complement	392
A.6.1	The Complement as a Correspondence Between Spaces	392
A.6.2	The Euclidean Complement	392
A.6.3	The Complement of a Complement	393
A.6.4	The Complement Axiom	393
A.7	The Interior Product	394
A.7.1	Inner Products and Scalar Products	394
A.7.2	Calculating Interior Products	394
A.7.3	Expanding Interior Products	395
A.7.4	The Interior Product of a Bivector and a Vector	396
A.8	Concluding Remarks	396
	References	397
Appendix B	Nonlinear Programming Algorithms	399
B.1	Introduction	399
B.2	Steepest Descent	399
B.2.1	Introduction	399
B.2.2	Method of Steepest Descent	400
B.2.3	Convergence	401
B.2.4	Scaling	403
B.2.5	Extensions	403
B.3	Gauss-Newton Method	404
B.4	Levenberg-Marquardt Algorithm	407

B.4.1	Introduction	407
B.4.2	Nonlinear Least-Squares Minimization	407
B.4.3	LM as a Blend of Gradient Descent and Gauss-Newton Iteration	409
B.5	Concluding Remarks	412
References		412
Acronyms		413
Symbols		417
About the Author		419
Index		421



# Preface

This book is about methods and techniques for geolocating noncooperative targets of interest that are emitting radio frequency signals. This is a critical function for most military electronic warfare systems; tactical ones at least.

Added to this second edition, which were not included in the first edition, are chapters on methods of estimating the fundamental parameters that allow the position fixes to be calculated. These are either the measurement of angles of arrival of the signal of interest (as in interferometry), or some other time, frequency, amplitude, or phase parameter such as time of arrival or differential frequency. In any real-world situation these parameters are always statistical as random components introduced by noise processes are always present.

This material is intended to be useful in applications as it is presented, allowing implementation of hardware and algorithms with the information provided. It can also serve as a launching point for further developments in the methods for geoposition estimation.

The material is also suitable for a short course on electronic warfare emitter geolocation. End of chapter problems are not included, so its utility as a text on the subject for part of a full course is limited. When augmented with problems, however, it could fill that need as well. In such a role, it would best be taught as part of a fourth year or first level graduate engineering curriculum.

The intended audience for this book is the same as the first edition. It is aimed primarily at technical personnel with at least a four-year degree in an engineering or scientific discipline, new to the field of electronic warfare. It also serves as a source for experienced engineers who would like to have a reference on the topic of emitter geolocation. Most of the more common techniques are included so the coverage is reasonably complete.

One area that is not explored in detail here is array-beamforming. It is a method for determining the angle of arrival of signals of interest, while rejecting interfering signals from other directions. The reason this topic is not covered in depth here is because a thorough coverage of it is contained in [1]. Another area not described at length is the theory of phase interferometry. The reason is similar: considerable detail is included in [2].

After the introduction in Chapter 1, we delve into the concepts behind triangulation in Chapter 2. This chapter did not change much from the first edition. Triangulation is the notion of estimating the position fix of a target emitter based on measurements of the angle of arrival of the signal at 2 or more intercept sites. The estimate of the target location is the intersection of the lines defined by these angles, since the locations of the receiving systems are assumed to be known.

Chapter 3 is all new in this edition. We did not include any descriptions in the first edition of how to obtain the angles of arrival used in the triangulation methods discussed in Chapter 2. We now have included discussions of several techniques. One of the more important such technique is called MUSIC, and it is an example (perhaps *the* example) of techniques that carry the appellation subspace methods. Chapter 4 is devoted to the MUSIC algorithm and its characteristics.

Chapter 5, like Chapter 2, did not change much from the first edition. It covers quadratic position fixing methods including time difference of arrival, time of arrival, differential Doppler, and range difference methods. Chapter 6 explores techniques for estimating the time delay of signals at several receiving systems. This topic was not included in the first edition.

We included discussions about single site location technology in the first edition, and we expand on that in this edition. That is the subject of Chapter 7.

Two appendices are included that explore in more detail two of the more mathematical topics mentioned in the main text.

Errors tend to creep into technical works such as this monograph. The author accepts all the responsibility for any such errors or omissions. As always, constructive feedback is welcomed.

## References

- [1] Poisel, R. A., *Antenna Systems and Electronic Warfare Applications*, Norwood, MA: Artech House, 2012.
- [2] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, 2nd Ed., Norwood, MA: Artech House, 2008, Ch. 8.

# Chapter 1

## Introduction to Emitter Geolocation

### 1.1 Introduction

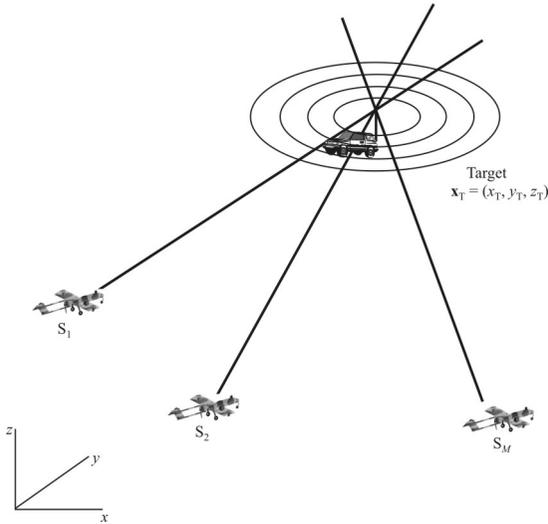
Determining the location of an emitting target is one of the fundamental functions of communication *electronic warfare* (EW) systems. Knowing the location of targets is useful for several purposes. First, knowing the location of targets indicates the disposition of forces. Second, precision location of targets allows for use of *global positioning system* (GPS)-enabled fire-and-forget munitions for negation of the target. Third, clustering different types of emitters in the same region can give an indication of the type of entity at a particular location.

This monograph presents several modern (and some not-so-modern) techniques for computing the *position fix* (PF) of a target based on differing information available from measurements performed on intercepted signals.

The azimuth *angle of arrival* (AOA) of a signal, or its *line of bearing* (LOB), is a frequently used parameter for PF computation. Two or more LOBs, assumed to be measured on the same target at more or less the same time, may intersect as illustrated in Figure 1.1 [1–6]. Such a technique for PF determination is referred to as *triangulation*. Triangulation is discussed at length in Chapter 2, where several different algorithms are presented.

There are several methods to estimate the AOA of signals impinging on the antenna array at an intercept site. They are all based on measuring the *time difference of arrival* (TDOA) or phase difference of the signals at two antennas that are spaced half a wavelength or less apart (at least two of the set of antennas used must satisfy this; otherwise, ambiguities arise) [7]. We present some of the more common methods in Chapter 3. One of the most important and widely used methods carries the appellation *multiple signal classification* (MUSIC) [8]. This technique is discussed at length in Chapter 4.

Another possibility is to measure the *time of arrival* (TOA) of the signal at several dispersed sensors that are located substantially more than a wavelength



**Figure 1.1** Intersection of measured LOBs.

apart. The TOA itself can be used to compute the PF, where the isocontours are circles, but more typically the TOAs are transferred to a central site where the TDOAs are computed between the sensors, two at a time. Closely related to the method of using the TDOAs is to compute the range differences between the sensors and the target. Such *range differences* (RD) are related to the TDOAs by the speed of propagation in the medium through which the signal propagates. In the air this is normally assumed to be the speed of light. If the signal is an audio signal, then in the air it is the speed of propagation of sound through the air, which can be influenced by humidity and other parameters in the air. If the signal is an audio signal underwater, then the speed of propagation is that of sound through water.

Another form of quadratic processing is to measure the *differential Doppler* (DD), otherwise known as *differential frequency*. Such measurements generate target location isochrones upon which the target is estimated to lie. Its greatest challenge is the target cannot be moving; otherwise, large errors can occur when trying to measure Doppler differences. Examination of DD is included as well.

All of the techniques that use the TOA, TDOA, RD, or DD generate quadratic curves, called *lines of position* (LOP), upon which the emitter lies, subject to measurement errors and noise perturbations. The intersection of these LOPs is used to estimate the PF. Quadratic PF techniques are discussed in Chapter 5. As

discussed in Chapter 5, if the target is moving, errors can occur in the PF calculations. These errors can be mitigated if the motion is detected. It is required to estimate the time delay between the sensors to determine the LOPs. Some techniques to do so are presented in Chapter 6.

Signals in the *high frequency* (HF) range can propagate for considerable distances using reflections off the Earth's ionosphere. Such bending of the signal is actually being caused by refraction of the signals within the ionosphere due to variations in the electron and ion density in the ionosphere.

Estimating the geoposition of such targets is possible using the AOA estimation techniques presented in Chapter 2. However, knowing the azimuth angle of arrival of the signal, as well as its elevation angle, combined with an estimate of the equivalent height of the ionosphere where the signals are reflected allows estimation of the PF of a target with a single sensor. Such PF techniques are known as *single-site location* (SSL). SSL methods are discussed in Chapter 7. Comparison of the virtues of AOA methods to the SSL technique are also presented in Chapter 7.

Perhaps the most used form of AOA estimation is amplitude comparison since that is the popular technique used in *aircraft survivability equipment* (ASE), in particular, in *radar warning receivers* (RWRs). Every combat aircraft in the Air Force and Navy has this equipment employed on it. The downfall of the method is that it is fairly inaccurate. Nevertheless, we include a discussion on how it is used.

## 1.2 Gradient Descent Algorithm

Foy presented a PF algorithm based on the simple process of using a Taylor series expansion of the defining equations (also known as Gauss or Gauss-Newton interpolation) [9]. This development provides a convenient introduction to the algorithmic approach to computing PFs. It contains many of the characteristics typical of solving the PF estimation problem, so it is presented in this introduction. It is also one of the most general and most accurate methods available. Other types of descent algorithms exist and can be used as well [6]. More in-depth information on the decent algorithms is provided in Appendix B.

The equations express the geometry involved in mathematical terms and are frequently nonlinear. Expanding the nonlinearities in a Taylor series and retaining only the linear terms allows for employing Newton-Raphson methods of gradient descent to iteratively find a solution. An initial guess of the solution is made, and the estimation process proceeds iteratively until an estimated solution is found. At each stage, a correction factor is calculated based on the local linear least-sum square error.

**Table 1.1** Advantages and Disadvantages of the Gradient Descent Approach to PF

Advantages	Disadvantages
Multiple independent measurements to a single station are averaged naturally Multiple measurements and mixed-mode measurements are combined properly, that is, with the correct geometric factors, and can be weighted according to their a priori accuracies The statistical spread of the solution can be found easily and naturally Experience indicates that the initial position guess can be quite far off without preventing good convergence Failure to converge is easy to detect Simulation is easy, so convergence can be readily tested Computational complexity is less than that of a Kalman-Bucy filter	The method is iterative, requiring an initial guess It is computationally complex compared to simple plotting of lines of position Being a local correction, its convergence is not assured

Source: [9].

The principal disadvantages and advantages according to Foy are as shown in Table 1.1.

Suppose  $\vec{x}_T = (x_T, y_T)$  represents the true position of the target emitter and  $\vec{x}_k = (x_k, y_k), k = 1, 2, \dots, K$  represents the true positions of the  $K$  PF sensors.<sup>1, 2</sup> Let  $n_{ki}$  represent the  $n$ th measurement at sensor  $k$ . These measurements, at this point, are not restricted to LOBs, ranges, and so forth, but could represent any measurement useful for calculating the PF. Thus,

$$f_i(x_T, y_T, x_k, y_k) = u_i = n_{ki} + \epsilon_i, \quad k = 1, 2, \dots, K \quad (1.1)$$

where

$n_{ki}$  = true value of the measured quantity

and

$\epsilon_i$  = error in the  $n_{ki}$  measurement

For example, if the measurement is that of an LOB from the PF sensor to the emitter, then

<sup>1</sup> Bold capital letters (**X**) refer to matrices, small italicized letters with an arrow ( $\vec{x}$ ) refer to vectors, and italicized letters refer to variables.

<sup>2</sup> This analysis is restricted to two dimensions. Extension to higher dimensionality is straightforward.

$$f_i(x_T, y_T, x_k, y_k) = \tan^{-1} \left( \frac{y_T - y_k}{x_T - x_k} \right) + \epsilon_i \quad (1.2)$$

The *covariance matrix* for two random vectors  $\bar{y} = [y_1 \ y_2 \ \cdots \ y_{C(\bar{y})}]$  and  $\bar{z} = [z_1 \ z_2 \ \cdots \ z_{C(\bar{z})}]$ , where  $C(\cdot)$  stands for the cardinality of the argument, is a matrix whose element in the  $i, j$  position is the covariance between the  $i$ th and  $j$ th elements of the vectors. It is given by

$$\mathbf{C}_{yz} = \mathbf{Cov}\{\bar{y}, \bar{z}\} = \mathcal{E}\{[\bar{y} - \mathcal{E}\{\bar{y}\}][\bar{z} - \mathcal{E}\{\bar{z}\}]^H\} \quad (1.3)$$

where  $^H$  stands for the Hermitian operation (conjugate transpose) and  $\mathcal{E}\{\}$  denotes statistical expectation.<sup>3</sup> If the two vectors have zero means, then obviously

$$\mathbf{C}_{yz} = \mathcal{E}\{\bar{y}\bar{z}^H\} \quad (1.4)$$

If the two vectors are the same then

$$\mathbf{C}_{yy} = \mathbf{Var}\{y\} \quad (1.5)$$

The covariance matrix is a measure of the quality of the relationship between the two constituent vectors.

The goal is to find  $(x_T, y_T)$  given the measurements and known locations of the sensors. The errors in the measurements  $\epsilon_i$  are assumed to be *independent* (of each other) and *identically distributed* (i.i.d.) with zero means,  $\mathcal{E}\{\epsilon_i\} = 0$ .

The *error covariance matrix* is given by

$$\mathbf{C}_{\epsilon\epsilon} = [c_{ij}] \quad (1.6)$$

with entries

$$c_{ij} = \mathcal{E}\{\epsilon_i \epsilon_j\} \quad (1.7)$$

Let  $(\hat{x}_T, \hat{y}_T)$  be guesses of the true position  $(x_T, y_T)$ . Then

---

<sup>3</sup> Strictly speaking, specification of expectations requires a domain (ensemble) over which the expectation applies. When that domain is obvious, it will not be indicated.

$$x_T = \hat{x}_T + \delta_x \quad (1.8)$$

and

$$y_T = \hat{y}_T + \delta_y \quad (1.9)$$

The Taylor series is an expansion of a (usually) nonlinear function. Its purpose is to typically retain only the linear terms (the first couple), or perhaps up to the second order, to simplify analysis. When retaining only the linear terms, the process linearizes an otherwise nonlinear, and thus generally intractable, function. The result is normally a linear approximation to the original function, and can be quite accurate in a small region around a given point.

The Taylor series expansion of  $f_i()$  around point  $a$  is given by [10]

$$f(x) = f(a) + (x-a)f'(a) + \frac{(x-a)^2}{2!} f''(a) + \dots + \frac{(x-a)^n}{n!} f^{(n)}(a) + \dots \quad (1.10)$$

where the notation  $f^{(n)}(a)$  refers to the  $n$ th derivative of  $f(x)$  evaluated at point  $a$ . In two dimensions, this becomes

$$\begin{aligned} f(a+h, b+k) &= f(a, b) + \left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right) f(x, y) \Big|_{\substack{x=a \\ y=b}} + \dots \\ &+ \frac{1}{n!} \left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right)^n f(x, y) \Big|_{\substack{x=a \\ y=b}} + \dots \end{aligned} \quad (1.11)$$

where the bar and subscripts mean that, after differentiation,  $x$  is replaced with  $a$  and  $y$  is replaced with  $b$ . Also, in this notation,

$$\begin{aligned} \left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right) f(x, y) &= \left( h \frac{\partial f(x, y)}{\partial x} + k \frac{\partial f(x, y)}{\partial y} \right) \\ \left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right)^2 f(x, y) &= h^2 \frac{\partial^2 f(x, y)}{\partial x^2} + 2hk \frac{\partial^2 f(x, y)}{\partial x \partial y} + k^2 \frac{\partial^2 f(x, y)}{\partial y^2} \\ &\dots \end{aligned}$$

For the case at hand, the two-dimensional Taylor expansion, after deleting all terms higher than the linear ones, is

$$\hat{f}_i + \left. \frac{\partial f_i(O)}{\partial x} \right|_{\substack{x=\hat{x}_T \\ y=\hat{y}_T}} \delta_x + \left. \frac{\partial f_i(O)}{\partial y} \right|_{\substack{x=\hat{x}_T \\ y=\hat{y}_T}} \delta_y \approx n_{ki} + \epsilon_i \quad (1.12)$$

where

$$\hat{f}_i = f_i(\hat{x}_T, \hat{y}_T, x_k, y_k) \quad (1.13)$$

To put this development into matrix form for easier manipulation, define

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ \vdots & \vdots \\ h_{N_s 1} & h_{N_s 2} \end{bmatrix} = \begin{bmatrix} \left. \frac{\partial f_1(O)}{\partial x} \right|_{\substack{x=\hat{x}_T \\ y=\hat{y}_T}} & \left. \frac{\partial f_1(O)}{\partial y} \right|_{\substack{x=\hat{x}_T \\ y=\hat{y}_T}} \\ \left. \frac{\partial f_2(O)}{\partial x} \right|_{\substack{x=\hat{x}_T \\ y=\hat{y}_T}} & \left. \frac{\partial f_2(O)}{\partial y} \right|_{\substack{x=\hat{x}_T \\ y=\hat{y}_T}} \\ \vdots & \vdots \\ \left. \frac{\partial f_{N_s}(O)}{\partial x} \right|_{\substack{x=\hat{x}_T \\ y=\hat{y}_T}} & \left. \frac{\partial f_{N_s}(O)}{\partial y} \right|_{\substack{x=\hat{x}_T \\ y=\hat{y}_T}} \end{bmatrix} \quad (1.14)$$

$$\vec{\delta} = \begin{bmatrix} \delta_x \\ \delta_y \end{bmatrix} \quad (1.15)$$

$$\vec{z} = \begin{bmatrix} n_{k1} - \hat{f}_1 \\ n_{k2} - \hat{f}_2 \\ \vdots \\ n_{kN_s} - \hat{f}_{N_s} \end{bmatrix} \quad (1.16)$$

and

$$\vec{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_{N_s} \end{bmatrix} \quad (1.17)$$

then (1.12) can be rewritten as

$$\mathbf{H}\vec{\delta} \approx \vec{z} + \vec{\epsilon} \quad (1.18)$$

The form of  $\vec{\delta}$  that yields the least sum squared error with the terms weighted according to the covariance matrix is [11]

$$\vec{\delta} = [\mathbf{H}^T \mathbf{Y}^{-1} \mathbf{H}]^{-1} \mathbf{Y}^{-1} \vec{z} \quad (1.19)$$

where  $\mathbf{Y}$  is a weighting matrix used to factor in one or more parameters.  $\mathbf{Y}$  is arbitrary, but must be positive definite and full rank so that  $\mathbf{Y}^{-1}$  exists. Therefore, during one step of the iteration,  $\vec{\delta}$  is computed according to (1.19), and new estimates (guesses) are obtained via

$$\begin{aligned} \hat{x}_{\text{new}} &\leftarrow \hat{x}_{\text{old}} + \delta_x \\ \hat{y}_{\text{new}} &\leftarrow \hat{y}_{\text{old}} + \delta_y \end{aligned} \quad (1.20)$$

in (1.18), and the iteration is repeated until there is a satisfactory minimization of change in  $(x_g, y_g)$  from one iteration to the next ( $\vec{\delta} \approx \vec{0}$ ).

The covariance matrix of the PF estimate is given by

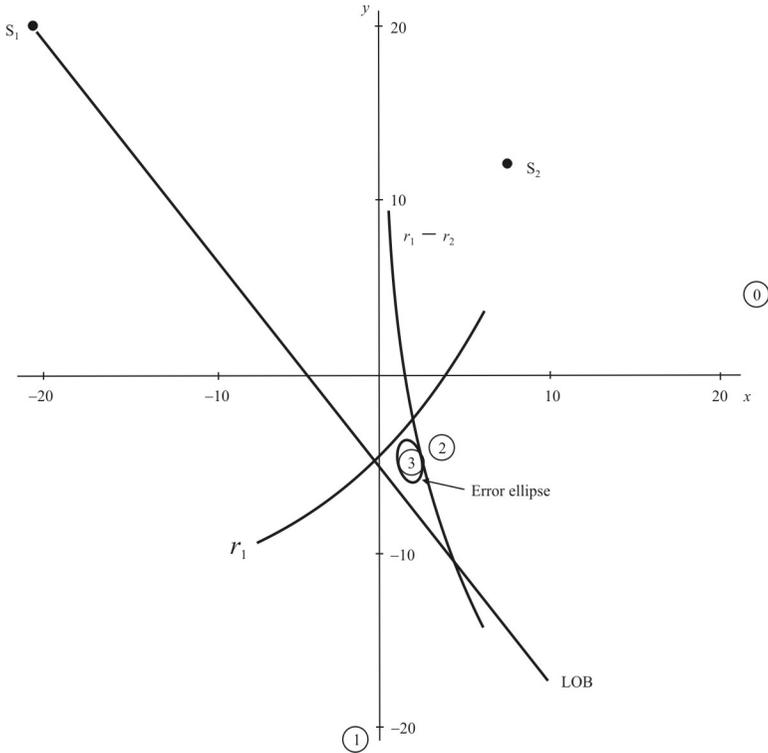
$$\mathbf{C}_0 = [\mathbf{H}^T \mathbf{Y}^{-1} \mathbf{H}]^{-1} = \begin{bmatrix} \sigma_x^2 & \rho_{xy} \\ \rho_{xy} & \sigma_y^2 \end{bmatrix} \quad (1.21)$$

If the error statistics are normal, then the error region is an ellipse, called the *elliptical error probable* (EEP), with semimajor axis  $a$  and semiminor axis  $b$  given by [12]

$$a^2 = 2 \frac{\sigma_x^2 \sigma_y^2 - \rho_{xy}^2}{\sigma_x^2 + \sigma_y^2 - [\sigma_x^2 - \sigma_y^2 + 4\rho_{xy}^2]^{1/2}} c_e^2 \quad (1.22)$$

$$b^2 = 2 \frac{\sigma_x^2 \sigma_y^2 - \rho_{xy}^2}{\sigma_x^2 + \sigma_y^2 + [\sigma_x^2 - \sigma_y^2 + 4\rho_{xy}^2]^{1/2}} c_e^2 \quad (1.23)$$

where  $c_e = -2 \ln(1 - P_e)$ , with  $P_e$  being the confidence that the target lies within



**Figure 1.2** Mixed observation example geometry.

the EEP (e.g., 0.5 for 50%, 0.9 for 90%, and so forth). The tilt angle  $\theta$  of the semimajor axis relative to the  $x$ -axis is given by

$$\theta = \frac{1}{2} \tan^{-1} \frac{2\rho_{xy}}{\sigma_y^2 - \sigma_x^2} \quad (1.24)$$

The *circular error probable* (CEP) is similar in concept to the EEP. It is a circle centered on the computed PF with an area such that the target lies within the circle with a prescribed probability. Based on the fact that the trace of a matrix is equal to the sum of its eigenvalues, the CEP can be estimated to within 10% by [7, 13]

$$\text{CEP} \approx 0.75\sqrt{a^2 + b^2} \quad (1.25)$$

### Example

This example is from [9] and is a good illustration of the mixed observation capability of the gradient descent method. Consider two sensors as shown in Figure 1.2. The two sensors are located at  $(-20, 20)$  and  $(8, 12)$ . Three measurements are made: an LOB from sensor  $S_1$  at  $-38^\circ$  with  $\sigma_\phi = 3^\circ = 0.0524$  radian, a range measurement also from  $S_1$  at  $r_1 = 32$  units with  $\sigma_r = 2$  units, and a range difference between the two sensors  $r_1 - r_2 = \Delta r_{12} = 16$  units with  $\sigma_{\Delta r} = 1$  unit. The bearing can be expressed as

$$\phi_1 = \sin^{-1} \left[ \frac{y_1 - y_k}{\sqrt{(x_1 - x_k)^2 + (y_1 - y_k)^2}} \right]$$

for which the first-order expansion is

$$\left[ \frac{y_1 - y_k}{r_1^2} \right] \delta_x - \left[ \frac{x_1 - x_k}{r_1^2} \right] \delta_y \approx \phi_1 - \tan^{-1} \left( \frac{y_1 - y_k}{x_1 - x_k} \right) - \epsilon_{\phi_1}$$

This result must be weighted by multiplying by the distance between the current guess and  $S_1$ , yielding

$$\left[ \frac{y_1 - y_k}{r_1} \right] \delta_x - \left[ \frac{x_1 - x_k}{r_1} \right] \delta_y \approx r_1 \left[ \phi_1 - \tan^{-1} \left( \frac{y_1 - y_k}{x_1 - x_k} \right) - \epsilon_{\phi_1} \right]$$

This is used for  $a_{11}$ ,  $a_{12}$ , and  $z_1$ .

The true range from  $S_1$  to the current guess is given by

$$r_1 = \sqrt{(x_k - x_1)^2 + (y_k - y_1)^2}$$

The first-order Taylor series expansion is given by

$$\frac{x_k - x_1}{r_{k1}} \delta_x + \frac{y_k - y_1}{r_{k1}} \delta_y \approx r_{\text{measured}} - r_{k1} - \epsilon_{r_1}$$

while the first-order Taylor series expansion of the range difference is given by

$$\begin{aligned} & \left[ \frac{x_k - x_1}{r_{k1}} - \frac{x_k - x_2}{r_{k2}} \right] \delta_x + \left[ \frac{y_k - y_1}{r_{k1}} - \frac{y_k - y_2}{r_{k2}} \right] \delta_y \\ & \approx \Delta r_{12} - (r_{k1} - r_{k2}) - \epsilon_{\Delta r_{12}} \end{aligned}$$

When the measurement errors are independent, the error covariance matrix of the measurements is

$$\mathbf{C}_{\epsilon\epsilon} = \begin{bmatrix} \sigma_\phi^2 & 0 & 0 \\ 0 & \sigma_r^2 & 0 \\ 0 & 0 & \sigma_{\Delta r}^2 \end{bmatrix} = \begin{bmatrix} (0.0524)^2 r_{1k}^2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The algorithm was started with the initial point of  $x_0 = 22$  and  $y_0 = 4$ , as illustrated in Figure 1.2. This results in  $\delta_x = -23.5$  and  $\delta_y = -24.4$  from (1.19). These values yield  $x_1 = -1.5$  and  $y_1 = -20.4$  from (1.20). Continuing the algorithm, the final solution at the end of iteration 3 is, as shown in Figure 1.2,  $x_3 = 2$ ,  $y_3 = -5.1$ , after which  $\delta_x = 0.05$  and  $\delta_y = -0.05$  and the algorithm halts. At this point,

$$\mathbf{H} = \begin{bmatrix} 0.752 & 0.659 \\ 0.659 & -0.752 \\ 0.990 & 0.192 \end{bmatrix}$$

and

$$\mathbf{C}_0 = [\mathbf{H}^T \mathbf{P}^{-1} \mathbf{H}]^{-1} = \begin{bmatrix} 0.899 & -0.640 \\ -0.640 & 3.578 \end{bmatrix}$$

From (1.22) to (1.24), the EEP parameters are

$$a = 1.929 \text{ units} \quad b = 0.868 \text{ units} \quad \theta = 102.76^\circ$$

yielding

$$\text{CEP} = 1.587 \text{ units}$$

The EEP is plotted in Figure 1.2.

In this example, convergence occurred after only three iterations and the resulting EEP was reasonable, considering the variances of the measurements.

### 1.3 Concluding Remarks

There are many PF estimation methods, some of which are presented in this monograph. There has been no attempt to include all possibilities. GPS alone has spurred considerable interest and research into different techniques of computing PFs (although all of those related to GPS are based on time differences). The ongoing implementation of the Federal Communications Commission (FCC) ruling in the United States mandating geoposition availability of all 911 calls has spurred considerable research into methods for computing PFs as well.

It is hoped that the reader garners a basic understanding of the problems and some useful approaches for estimating the locations of emitters.

### References

- [1] Jenkins, H. H., *Small-Aperture Radio Direction-Finding*, Norwood, MA: Artech House, 1991.
- [2] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, Norwood, MA: Artech House, 2002, Chapters 11 and 12.
- [3] Neri, F., *Introduction to Electronic Defense Systems*, 2nd ed., Norwood, MA: Artech House, 2001.
- [4] Van Trees, H. L., *Optimum Array Processing, Part IV of Detection, Estimation, and Modulation Theory*, New York: Wiley, 2002.
- [5] Cadzow, J. A., "Methods of Direction-of-Arrival," *Proceedings IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing*, May 9–10, 1991.
- [6] Cadzow, J. A., "Signal Processing via Least Squares Error Modeling," *IEEE ASSP Magazine*, October 1990, pp. 12–31.
- [7] Rusu, P., "A TOA-FOA Approach to Find the Position and Velocity of RF Emitters," Applied Research Laboratories, University of Texas at Austin.
- [8] Schmidt, R. O., "Multiple Emitter Location and Signal Parameter Estimation," *IEEE Transactions on Antennas and Propagation*, Vol. AP-34, No. 3, March 1986, pp. 276–280.
- [9] Foy, W. H., "Position-Location Solutions by Taylor-Series Estimation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-12, No. 2, March 1976, pp. 187–193.

- [10] Abramowitz, M., and I. A. Stegun, *Handbook of Mathematical Functions*, New York: Dover, 1972, p. 880.
- [11] Whalen, A. D., *Detection of Signals in Noise*, New York: Academic Press, 1971, pp. 374–376.
- [12] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, Norwood, MA: Artech House, 2002, pp. 384–388.
- [13] Torrieri, D. J., “Statistical Theory of Passive Location Systems,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-20, No. 2, March 1984, pp. 183–198.



# Chapter 2

## Triangulation

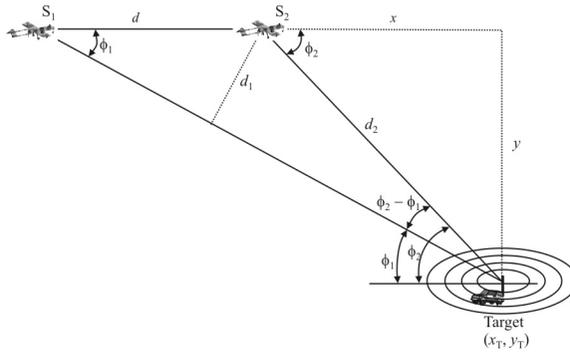
### 2.1 Introduction

*Triangulation* is the appellation applied to estimating a PF by calculating the most likely point for the target, given the intersection of two or more LOBs from sensors at known locations. It is widely used in this role, and there are several techniques available for measuring the azimuth angles of arrival of signals. Noise, measurement errors, and multipath reflections typically limit the PF accuracy.

Perhaps the first documented exploration of the triangulation concept was done by Stansfield [1]. Although not noted at the time, the Stansfield algorithm is a *maximum likelihood estimation* (MLE) technique. It is, however, biased (explained shortly). We examine Stansfield's method in this chapter.

Triangulation can be implemented on all varieties of platforms, including aircraft, ships, and ground vehicles. If signal phase is used as the parameter for computing the LOBs, then triangulation requires an array of antennas, with a baseline shorter than half a wavelength to avoid ambiguities in phase angle measurement. Parameters other than phase, such as relative amplitude, can also be used as the angle indicator. Most methods, however, require the antenna array.

This chapter starts with a simple illustration of triangulation to introduce the fundamental concepts. Several of the available algorithms for optimum PF calculation are based on the *least-square error* (LSE) estimation technique. Therefore, a generalized introduction to the LSE method is presented next. An extension to the LSE method is the *total least-square error* (TLSE) technique, which is somewhat more general and is presented next. PF algorithms based on the method of least-squared distance error are presented. This is followed by a discussion of optimum PF estimation based on the method of minimum mean-square error. A technique based on dividing up the *area of interest* (AOI) into discrete intervals, called the *discrete probability density* (DPD) method, is



**Figure 2.1** Geometric relationships for PF by triangulation.

discussed. The concept of generalized bearings, where the LOBs are not based on the Cartesian coordinates but are three-dimensional, yields a different algorithm for PF estimation; this is discussed next. Using concepts from the statistical process of maximum-likelihood estimation also yields PF estimation algorithms, and two of these are presented. A method of PF estimation based on the intersection of LOB *fans*, defined as  $\pm$ maximum LOB error, is discussed next. Then the discussion shifts to presentations of the theoretical observability requirements for estimation of the PF and velocity of a target. The last topics discussed in this chapter are the sources of error in PF estimation using triangulation. These include systematic errors (bias) and random errors (noise).

## 2.2 Basic Concepts

The geometry shown in Figure 2.1 can be used to describe the triangulation concept. Initially consider just two dimensions. There are two sensors,  $S_1$  and  $S_2$ , separated by distance  $d$ , and a single target. Of course, the two sensors could be the same sensor that has moved distance  $d$ . Each of the sensors computes an LOB relative to some reference, which is the same for both sensors. By simple trigonometry,

$$\sin \phi_1 = \frac{d_1}{d}$$

$$d_1 = d \sin \phi_1$$

and

$$\sin(\phi_2 - \phi_1) = \frac{d_1}{d_2}$$

$$d_2 = \frac{d_1}{\sin(\phi_2 - \phi_1)}$$

yielding

$$d_2 = \frac{d \sin \phi_1}{\sin(\phi_2 - \phi_1)} \quad (2.1)$$

With knowledge of  $d_2$  and  $\phi_2$ ,  $x$  and  $y$ , the distances to the target from  $S_2$ , can be computed as

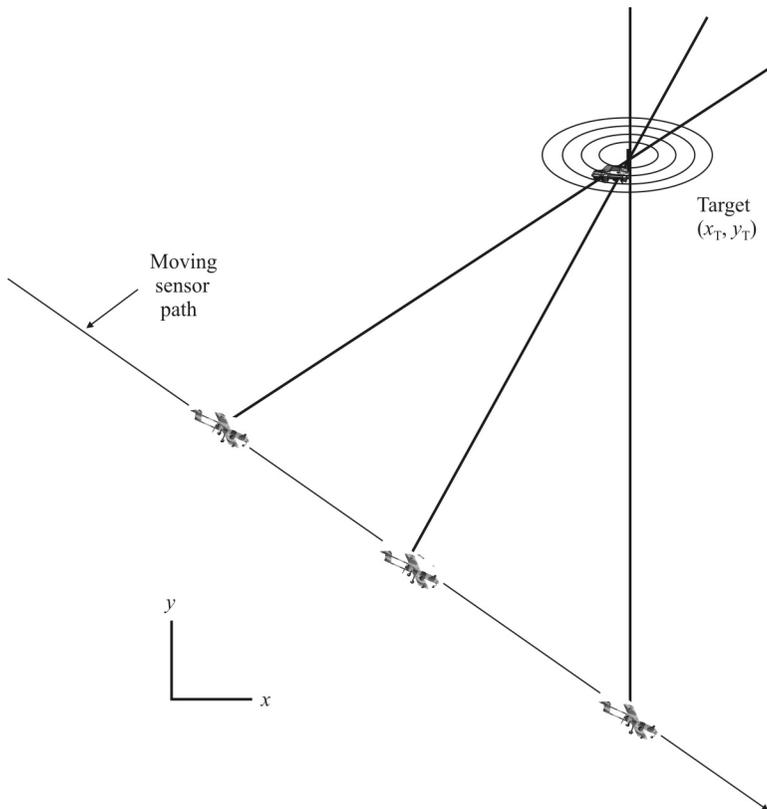
$$x = d_2 \cos \phi_2$$

$$y = d_2 \sin \phi_2 \quad (2.2)$$

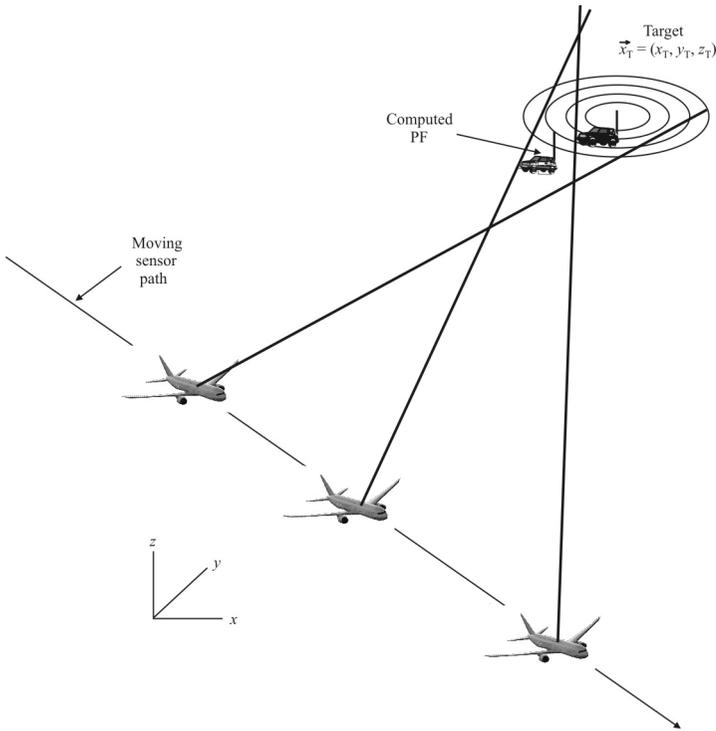
These notions extend to more than two sensors and to three dimensions in the obvious way, taking two sensors at a time and computing the coordinates of the target. The resulting coordinates can then be averaged. Other methods of using more than two LOBs are discussed in this chapter.

Another technique for PF with triangulation is to plot (figuratively or literally) the measured LOBs and see where they cross, as shown in Figure 2.2 (the sensors could be one moving as shown or three stationary, and the same results would apply). In the absence of errors in the measurement process, these LOBs will all cross at a single point. However, in general, the LOBs are corrupted with measurement error and noise. This noise is frequently assumed to be zero mean *additive white Gaussian noise* (AWGN). The result on the PF computation, however, is to cause the LOBs to move away from crossing at a single point. If the noise is random, the measured LOB could be larger than or smaller than the actual LOB, as illustrated in Figure 2.3. The result is an error ellipse. The sensors could also exhibit biases, which many parameter estimators do. Biases can be caused by being an inherent property of the algorithm to compute the PF or they can be due to systematic errors in the parameter measurement device.

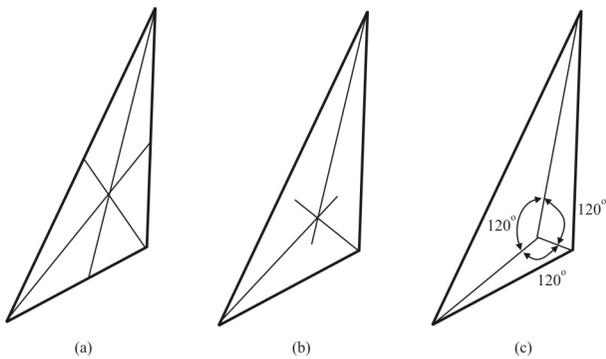
If there are only three LOBs available, they will form a triangle hopefully close to the actual location of the target as illustrated in Figure 2.2. There are three popular nonstatistical methods to estimate the location of the target given the triangle. These are shown in Figure 2.4 [2]; they are all techniques for estimating the centroid of the area of the triangle. In Figure 2.4(a), the medians of the sides are connected with the opposite angle. In Figure 2.4(b), the angles are bisected and the point where the bisectors intersect is chosen as the PF. In Figure 2.4(c), the



**Figure 2.2** Triangulation.



**Figure 2.3** Random error effects on computing the PF.



**Figure 2.4** Nonstatistical PF calculations with only three LOBs: (a) intersection of medians, (b) intersection of angle bisectors, and (c) Steiner point (defined by the point where the angles between the lines from the corners are all  $120^\circ$  as shown). They are all methods for estimating the centroid of the triangle.

intersection of lines drawn from each angle to the point in the triangle where the lines form  $120^\circ$  angles is chosen as the PF.

If there are more than three LOBs in the calculation, such as those illustrated in Figure 2.5(a), they can be taken three at a time, as shown in Figure 2.5(b–e). The coordinates of the resultant centroids can then be averaged to determine the final fix, as shown by the white circle in Figure 2.6. Note that averaging the intersections of bearings may lead to biased results.

## 2.3 Least-Squares Error Estimation

Many of the optimization methods to be presented rely on finding an estimate based on minimizing the error between the estimate and the actual value of the PF. Therefore, this section presents the general development of the optimal LSE estimator. Here, the recursive form of LSE estimation is assumed such that measurements are made sequentially at time instants  $0, 1, \dots, k, \dots, N - 1$ . That is, measurements are available at time instants  $k$  and before. LSE estimation can be applied to any appropriate set of data points [3], however, to include the case when the estimate is not obtained until all the data points are available.

Let the linear estimation model for the  $k$ th time instant be given by<sup>1</sup>

$$\vec{z}_k = \mathbf{H}_k \vec{\theta}_k + \vec{n}_k \quad (2.3)$$

where  $\vec{z}_k$  is the vector of measurements up to time  $k$ , and  $\mathbf{H}_k$  is the observation matrix such that measurement  $z_k$  at time instant  $k$  is related to the vector  $\vec{\theta}_k$  by

$$z_k = \vec{h}_k \vec{\theta}_k + n_k \quad (2.4)$$

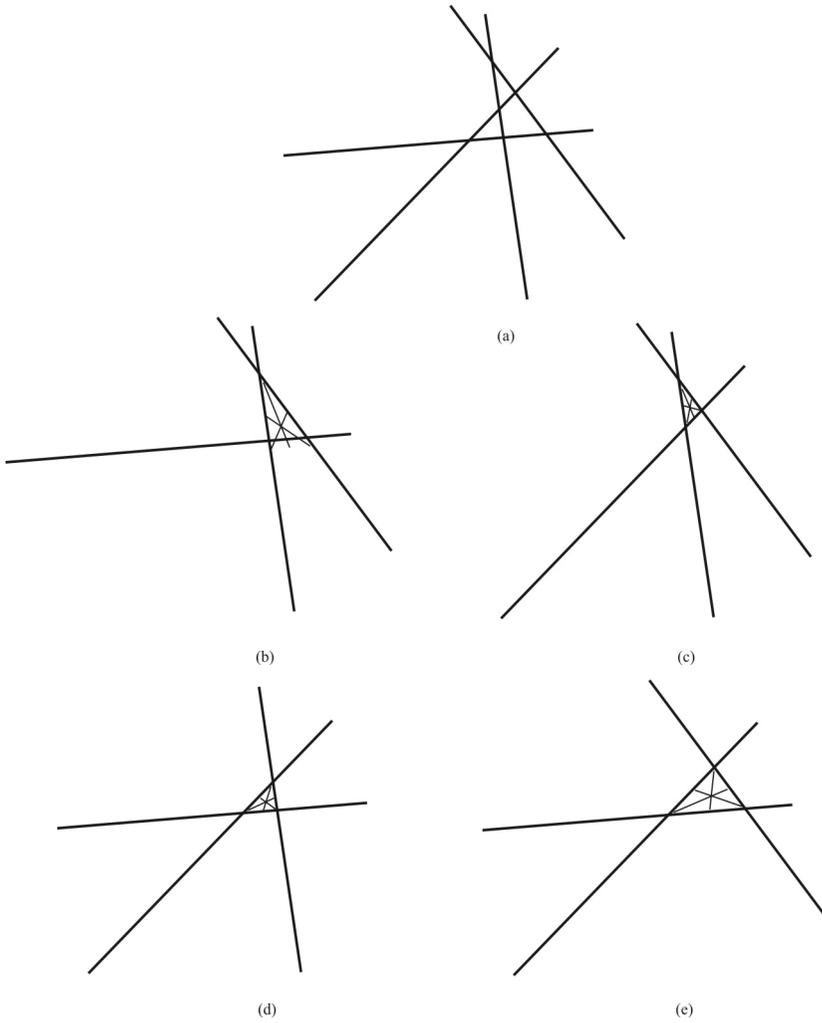
where  $\vec{h}_k$  is the  $k$ th row vector in  $\mathbf{H}_k$  and  $\vec{\theta}_k$  is the unknown parameter vector up to  $k$ . Noise term  $n_k$  is the measurement noise vector for sample time  $k$ .

The estimation model for  $\vec{z}_k$  is given by

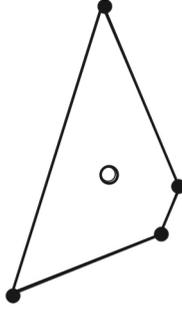
$$\hat{\vec{z}}_k = \mathbf{H}_k \hat{\vec{\theta}}_k \quad (2.5)$$

---

<sup>1</sup> Vectors are denoted with an overhead arrow ( $\vec{x}$ ), usually, but not necessarily, small case, while matrices are denoted in bold ( $\mathbf{X}$ ), usually, but not necessarily, capitalized.



**Figure 2.5(a-e)** Calculating the PF when there are more than three LOBs. PFs are computed using three LOBs at a time, then the results are combined.



**Figure 2.6** The centroid of the polygon formed by connecting the centroids of the triangles formed with three LOBs yields the final fix.

The goal is to find the optimum estimate for  $\vec{\theta}$ , denoted by  $\hat{\vec{\theta}}$ , such that some cost function is minimized. For LSE estimation, that cost function is the squared error of the difference between the actual value for  $\vec{\theta}$  and the estimate. Thus, the cost function to be minimized is given by

$$c(\hat{\vec{\theta}}_k) = \vec{\epsilon}_k^T \mathbf{W}_k \vec{\epsilon}_k \quad (2.6)$$

where  $\mathbf{W}_k$  is a weighting matrix, which is arbitrary but must be symmetric and positive definite, and is usually used to enhance some aspect of the estimation process. When  $\mathbf{W}_k = \mathbf{I}$ , then the process is called LSE estimation. If  $\mathbf{W}_k \neq \mathbf{I}$ , then it is called *weighted LSE* (WLSE) estimation.  $\mathbf{W}_k$  may be a constant matrix or it may be updated for each  $k$ . The difference between the actual value of  $\vec{z}_k$  and its estimate  $\hat{\vec{z}}_k$  at instant  $k$  is called the *error term*, denoted by  $\vec{\epsilon}_{z_k}$ , and is given by

$$\vec{\epsilon}_{z_k} = \vec{z}_k - \hat{\vec{z}}_k \quad (2.7)$$

and

$$\vec{\epsilon}_{z_k} = [\vec{\epsilon}_{z_k} \quad \vec{\epsilon}_{z_{k-1}} \quad \cdots \quad \vec{\epsilon}_{z_{k-N+1}}]^T \quad (2.8)$$

Substituting (2.7) into (2.6) yields

$$c(\hat{\vec{\theta}}_k) = (\vec{z}_k - \hat{\vec{z}}_k)^T \mathbf{W}_k (\vec{z}_k - \hat{\vec{z}}_k) \quad (2.9)$$

Using (2.5) in (2.9),

$$c(\hat{\theta}_k) = (\vec{z}_k - \mathbf{H}_k \hat{\theta}_k)^\top \mathbf{W}_k (\vec{z}_k - \mathbf{H}_k \hat{\theta}_k) \quad (2.10)$$

$$\begin{aligned} &= (\vec{z}_k^\top - \hat{\theta}_k^\top \mathbf{H}_k^\top) \mathbf{W}_k (\vec{z}_k - \mathbf{H}_k \hat{\theta}_k) \\ &= (\vec{z}_k^\top \mathbf{W}_k - \hat{\theta}_k^\top \mathbf{H}_k^\top \mathbf{W}_k) (\vec{z}_k - \mathbf{H}_k \hat{\theta}_k) \\ &= \vec{z}_k^\top \mathbf{W}_k \vec{z}_k - 2\vec{z}_k^\top \mathbf{W}_k \mathbf{H}_k \hat{\theta}_k + \hat{\theta}_k^\top \mathbf{H}_k^\top \mathbf{W}_k \mathbf{H}_k \hat{\theta}_k \end{aligned} \quad (2.11)$$

This last step is possible because all three terms in (2.11) are scalars and therefore symmetric.

The minimum of the cost function is found by finding the derivative of (2.11) and setting it to zero. The first term in (2.11) is independent of  $\vec{\theta}_k$ , so its derivative is zero. The derivative of the second term is based on a result from vector calculus that says for two  $n \times 1$  vectors  $\vec{a}$  and  $\vec{b}$ ,

$$\frac{d\vec{a}^\top \vec{b}}{d\vec{b}} = \vec{a} \quad (2.12)$$

Thus,

$$\frac{d(-2\vec{z}_k^\top \mathbf{W}_k \mathbf{H}_k \hat{\theta}_k)}{d\hat{\theta}_k} = -2(\vec{z}_k^\top \mathbf{W}_k \mathbf{H}_k)^\top = -2\mathbf{H}_k^\top \mathbf{W}_k \vec{z}_k \quad (2.13)$$

The derivative of the third term in (2.11) is found by using the fact that for  $n \times n$  matrix  $\mathbf{A}$

$$\frac{d\vec{b}^\top \mathbf{A} \vec{b}}{d\vec{b}} = 2\mathbf{A} \vec{b} \quad (2.14)$$

Therefore,

$$\frac{d\hat{\theta}_k^\top \mathbf{H}_k^\top \mathbf{W}_k \mathbf{H}_k \hat{\theta}_k}{d\hat{\theta}_k} = 2\mathbf{H}_k^\top \mathbf{W}_k \mathbf{H}_k \hat{\theta}_k \quad (2.15)$$

Using (2.13) and (2.15),

$$\frac{dc(\hat{\theta}_k)}{d\hat{\theta}_k} = \vec{0} = -2\mathbf{H}_k^T \mathbf{W}_k \vec{z}_k + 2\mathbf{H}_k^T \mathbf{W}_k \mathbf{H}_k \hat{\theta}_k \quad (2.16)$$

so

$$\vec{\theta}_k = [\mathbf{H}_k^T \mathbf{W}_k \mathbf{H}_k]^{-1} \mathbf{H}_k^T \mathbf{W}_k \vec{z}_k \quad (2.17)$$

which is the optimal LSE estimate desired. Expression (2.17) is sometimes referred to as the *normal equation*. That this is a minimum can be established by taking the second derivative of (2.11), which is

$$\frac{d^2 c(\hat{\theta}_k)}{d\hat{\theta}_k^2} = 2\mathbf{H}_k^T \mathbf{W}_k \mathbf{H}_k \quad (2.18)$$

which is in quadratic form and, since  $\mathbf{W}_k$  is positive definite,  $\mathbf{H}_k^T \mathbf{W}_k \mathbf{H}_k > 0$ , and  $\hat{\theta}_k$  is a minimum.

The accuracy or quality of the estimate provided by (2.17) is given by the covariance matrix, which is

$$\mathbf{C}_k = [\mathbf{H}_k^T \mathbf{W}_k^{-1} \mathbf{H}_k]^{-1} \quad (2.19)$$

In three dimensions, when the covariances are fixed, this matrix is ( $k$  assumed)

$$\mathbf{C}_{xyz} = \begin{bmatrix} \sigma_x^2 & \rho_{xy} \sigma_x \sigma_y & \rho_{xz} \sigma_x \sigma_z \\ \rho_{xy} \sigma_x \sigma_y & \sigma_y^2 & \rho_{yz} \sigma_y \sigma_z \\ \rho_{xz} \sigma_x \sigma_z & \rho_{yz} \sigma_y \sigma_z & \sigma_z^2 \end{bmatrix} \quad (2.20)$$

where  $\sigma_x^2$  is the variance of  $x$ ,  $\rho_{xy}$  is the correlation coefficient between  $x$  and  $y$ , and so forth. It is often assumed that  $x$ ,  $y$ , and  $z$  are uncorrelated so that  $\rho_{xy} = \rho_{xz} = \rho_{yz} = 0$ , in which case

$$\mathbf{C}_{xyz} = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix} \quad (2.21)$$

The nonrecursive form of the LSE estimation procedure is identical to the above except that  $k$  is not the sample time but is interpreted as the size of the final measurement set.

### Example

Suppose it is desired to fit a set of  $N$  paired points  $(x_i, y_i)$  to a straight line. The equation for the line is

$$y = \beta_1 x + \beta_0$$

The problem can be cast into matrix form as

$$\begin{aligned} \vec{y} &= [y_1 \quad y_2 \quad \cdots \quad y_N]^T \\ \vec{\beta} &= [\beta_0 \quad \beta_1]^T \\ \mathbf{X} &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_N \end{bmatrix}^T \end{aligned}$$

The LSE estimate based on (2.17) is given by

$$\hat{\boldsymbol{\theta}} = [\mathbf{X}^T \mathbf{W} \mathbf{X}]^{-1} \mathbf{X}^T \mathbf{W} \vec{y}$$

where  $\mathbf{W} = \mathbf{I}$ , yielding

$$\hat{\boldsymbol{\theta}} = [\mathbf{X}^T \mathbf{X}]^{-1} \mathbf{X}^T \vec{y} \quad (2.22)$$

Now,

$$\mathbf{X}^T \mathbf{X} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_N \\ \vdots & \vdots & & \vdots \\ 1 & x_N & & \end{bmatrix}$$

$$= \begin{bmatrix} N & \sum_{i=1}^N x_i \\ \sum_{i=1}^N x_i & \sum_{i=1}^N x_i^2 \end{bmatrix}$$

so

$$[\mathbf{X}^T \mathbf{X}]^{-1} = \frac{1}{N \sum_{i=1}^N x_i^2 - \left( \sum_{i=1}^N x_i \right)^2} \begin{bmatrix} \sum_{i=1}^N x_i^2 & -\sum_{i=1}^N x_i \\ -\sum_{i=1}^N x_i & N \end{bmatrix}$$

and

$$\mathbf{X}^T \vec{y} = \begin{bmatrix} \sum_{i=1}^N y_i \\ \sum_{i=1}^N x_i y_i \end{bmatrix}$$

From (2.22),

$$\hat{\vec{\beta}} = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{bmatrix} = \frac{1}{N \sum_{i=1}^N x_i^2 - \left( \sum_{i=1}^N x_i \right)^2} \begin{bmatrix} \sum_{i=1}^N x_i^2 & -\sum_{i=1}^N x_i \\ -\sum_{i=1}^N x_i & N \end{bmatrix} \begin{bmatrix} \sum_{i=1}^N y_i \\ \sum_{i=1}^N x_i y_i \end{bmatrix}$$

$$= \frac{1}{N \sum_{i=1}^N x_i^2 - \left( \sum_{i=1}^N x_i \right)^2} \begin{bmatrix} \sum_{i=1}^N y_i \sum_{i=1}^N x_i^2 - \sum_{i=1}^N x_i \sum_{i=1}^N x_i y_i \\ N \sum_{i=1}^N x_i y_i - \sum_{i=1}^N x_i \sum_{i=1}^N y_i \end{bmatrix}$$

which are the optimal estimates of the coefficients in an LSE sense.

Substituting (2.3) into (2.17) with  $\mathbf{W} = \mathbf{I}$  (and dropping the  $k$ ) yields

$$\hat{\vec{\theta}} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\mathbf{H} \vec{\theta} + \vec{n}) \quad (2.23)$$

$$= \underbrace{(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{H}}_{=\mathbf{I}} \vec{\theta} + (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \vec{n} \quad (2.24)$$

so

$$\hat{\vec{\theta}} = \vec{\theta} + (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \vec{n} \quad (2.25)$$

The *bias error* is given by

$$\vec{\delta} = \mathcal{E}\{\hat{\vec{\theta}}\} - \vec{\theta} \quad (2.26)$$

The resulting LSE estimation bias is approximated by [4]

$$\vec{\delta} \approx \mathcal{E}\left\{\frac{1}{N} \mathbf{H}^T \mathbf{H}\right\} \mathcal{E}\{\mathbf{H}^T \vec{n}\} \quad (2.27)$$

Therefore, in general, the LSE estimation process produces biased estimates. However, specific cases of LSE estimation, as will be shown, can produce unbiased results.

## 2.4 Total Least-Squares Estimation

A generalization of the LSE estimation technique discussed in Section 2.2 allows for the existence of noise in not only the measurements  $\vec{z}_k$  in (2.3), but also for errors in the observation matrix  $\mathbf{H}_k$ . When there is no noise in  $\mathbf{H}_k$ , and the noise represented by  $\vec{n}_k \sim \mathcal{N}(0, \sigma^2)$  (this notation means that the noise is normal, or

Gaussian, with zero mean and variance  $\sigma^2$ ), then the LSE solution  $\hat{\vec{\theta}}_{k,\text{LS}}$  is the same as the maximum likelihood estimate. However, when noise is present in  $\mathbf{H}_k$ , (2.17) with  $\mathbf{W}_k = \mathbf{I}$  is no longer optimal. It exhibits bias and increased covariance. To determine the optimal LSE estimate in this case, the method of TLSE estimation was developed [5]. The subscript  $k$  will be dropped here for notational convenience.

As described in Section 2.2, the LSE estimate is obtained by finding  $\hat{\vec{\theta}}$  by calculating

$$\min_{\vec{\theta}} \left\| \mathbf{H}\vec{\theta} - \vec{z} \right\|_2 \quad (2.28)$$

where  $\|\cdot\|_2$  is the  $\mathcal{L}_2$  norm of the argument. The solution to (2.28) is given by

$$\vec{\theta}_{\text{LS}} = \mathbf{H}^\dagger \vec{z} \quad (2.29)$$

where  $\mathbf{H}^\dagger$  is the pseudoinverse of  $\mathbf{H}_{N \times P}$ .<sup>2</sup> It is typically assumed that  $N > P$  and that  $\mathbf{H}$  has full rank, so that  $\mathbf{H}^\dagger = (\mathbf{H}^* \mathbf{H})^{-1} \mathbf{H}^*$  and

$$\hat{\vec{\theta}}_{\text{LS}} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^* \vec{z} \quad (2.30)$$

The measurement vector and observation matrix can be expressed as

$$\vec{z} = \vec{z}_0 + \Delta \vec{z} \quad \mathbf{H} = \mathbf{H}_0 + \Delta \mathbf{H} \quad (2.31)$$

where  $\Delta \vec{z}$  and  $\Delta \mathbf{H}$  are the noise perturbations. With no noise,  $\mathbf{H}_0 \vec{\theta}_0 = \vec{z}_0$ . The TLSE estimate is obtained from

$$\min_{\Delta \mathbf{D}, \vec{\theta}} \left\| \Delta \mathbf{D} \right\|_F^2 \quad \text{subject to} \quad (\mathbf{D} + \Delta \mathbf{D}) \begin{pmatrix} \vec{\theta} \\ -1 \end{pmatrix} = \vec{0} \quad (2.32)$$

$\mathbf{D}$  is given by

$$\mathbf{D} = [\mathbf{H} : \vec{z}] \quad (2.33)$$

---

<sup>2</sup> For any full rank matrix  $\mathbf{A}_{(m \times n)}$  ( $m > n$ ) ( $\text{rank} = n$ ) the *pseudoinverse* of  $\mathbf{A}$  is  $\mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$ . One way to compute  $\mathbf{A}^\dagger$  is to use *singular value decomposition* (SVD). With  $\mathbf{A} = \mathbf{U} \mathbf{S} \mathbf{V}^T$ , where  $\mathbf{U}_{(m \times m)}$  and  $\mathbf{V}_{(n \times n)}$  are orthogonal and  $\mathbf{S}_{(m \times n)} = \begin{bmatrix} \mathbf{D}_{(n \times n)} & \mathbf{0}_{(m-n \times n)} \end{bmatrix}^T$ , with  $\mathbf{D}_{(n \times n)}$  diagonal with real, nonnegative singular values  $d_i$ , then  $\mathbf{A}^\dagger = \mathbf{V} (\mathbf{S}^T \mathbf{S})^{-1} \mathbf{S}^T \mathbf{U}^T$ . If  $r = \text{rank}(\mathbf{A}) < n$ , the inverse of  $\mathbf{S}^T \mathbf{S}$  does not exist, and the first  $r$  singular values only are used in  $\mathbf{S}$ , and  $\mathbf{U}$  and  $\mathbf{V}$  are correspondingly smaller.

and  $\Delta \mathbf{D}$  is given by

$$\Delta \mathbf{D} = [\Delta \mathbf{H} : \Delta \vec{z}] \quad (2.34)$$

The value  $\|\bullet\|_F$  stands for the *Frobenius norm* of the argument, which is given by  $\|\vec{z}\|_F = \sqrt{\text{tr}(\vec{z}^* \vec{z})}$ . Let the SVD of  $\mathbf{D}$  be  $\mathbf{D} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^*$  ( $N \times P$ ), where  $\mathbf{\Sigma}$  has real diagonal elements  $\sigma_i$  such that

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(N,P)} \geq 0, \quad i = 1, 2, \dots, N+1 \quad (2.35)$$

The  $\sigma_i$  are the singular values of  $\mathbf{D}$  and the first  $\min(N, P)$  columns of  $\mathbf{U}$  and  $\mathbf{V}$  are the left and right singular vectors of  $\mathbf{D}$ . These singular values and singular vectors satisfy

$$\mathbf{D} \vec{v}_i = \sigma_i \vec{u}_i \quad \text{and} \quad \mathbf{D}^* \vec{u}_i = \sigma_i \vec{v}_i \quad (2.36)$$

where  $\vec{u}_i$  and  $\vec{v}_i$  are the  $i$ th columns of  $\mathbf{U}$  and  $\mathbf{V}$ , respectively.

Based on these derivations, the TLS estimate is given by

$$\hat{\theta} = -\frac{1}{v_i} \vec{v}_i' \quad (2.37)$$

or

$$\hat{\theta}_{\text{TLS}} = (\mathbf{D}^* \mathbf{D} - \sigma_i^2 \mathbf{I})^{-1} \mathbf{D}^* \vec{z} \quad (2.38)$$

The factor  $v_i$  is the last component of  $\mathbf{V}$  in the  $i$ th row and  $\vec{v}_i'$  is the first element of  $\mathbf{V}$ . In (2.38),  $\sigma_i$  is the smallest singular value and  $\sigma_i^2 \mathbf{I}$  helps to reduce the bias that the noise in the observation matrix introduces in  $\mathbf{D}^* \mathbf{D}$ .

When  $\Delta \mathbf{D} \sim \mathcal{N}(0, \sigma_b^2)$ , TLSE estimation is very effective and more accurate than LSE.

## 2.5 Least-Squares Distance Error PF Algorithm

### 2.5.1 Brown's Least-Squares Triangulation Algorithm

An algorithm developed by Brown [6] will be presented in this section for calculating the PF, which is based on minimizing the square of the miss distance of the PF from the measured LOBs. This algorithm was presented in [7] and is included here for completeness.

Referring to Figure 2.7 [6], to minimize the sum of the squares of the total miss distance, formulate

$$D = \sum_{i=1}^N d_i^2 \quad (2.39)$$

$$\begin{aligned} &= \sum_{i=1}^N a_i^2 x_T^2 + \sum_{i=1}^N 2a_i b_i x_T y_T - \sum_{i=1}^N 2a_i c_i x_T \\ &\quad + \sum_{i=1}^N b_i^2 y_T^2 - \sum_{i=1}^N 2b_i c_i y_T + \sum_{i=1}^N c_i^2 \end{aligned} \quad (2.40)$$

where

$$a_i = \sin \phi_i$$

$$b_i = -\cos \phi_i$$

$$c_i = x_i \sin \phi_i - y_i \cos \phi_i$$

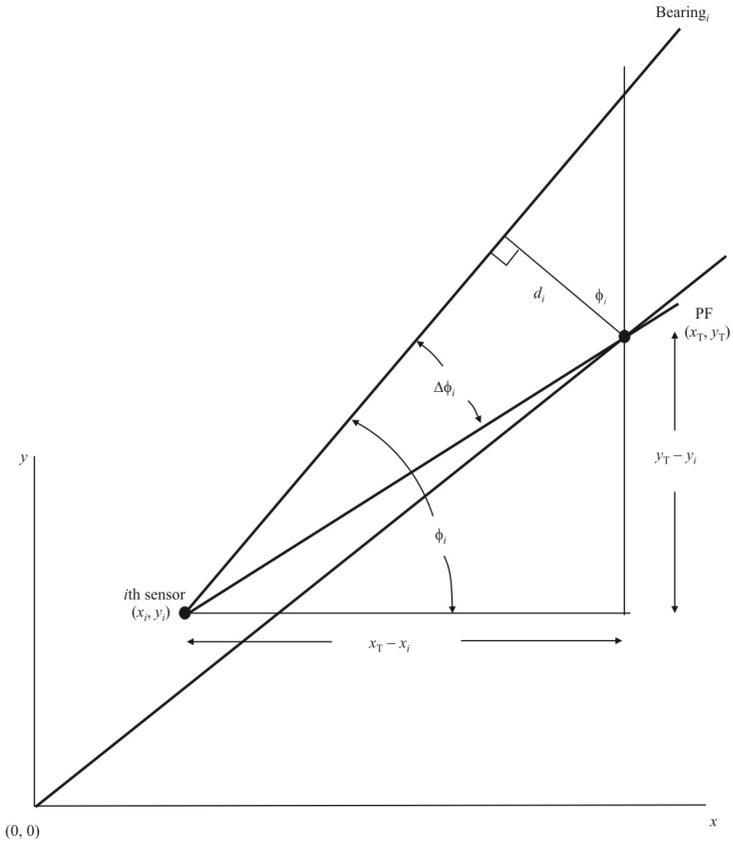
$N$  is the number of LOBs

Setting the first partial derivative of  $D$  with respect to  $x_T$  and then  $y_T$  equal to zero will find the values of  $x_T$  and  $y_T$  for which the total squares distance is minimized.

$$\frac{\partial D}{\partial x_T} = 0 = 2x_T \sum_{i=1}^N a_i^2 + 2y_T \sum_{i=1}^N a_i b_i - 2 \sum_{i=1}^N a_i c_i \quad (2.41)$$

$$\frac{\partial D}{\partial y_T} = 0 = 2x_T \sum_{i=1}^N a_i b_i + 2y_T \sum_{i=1}^N b_i^2 - 2 \sum_{i=1}^N b_i c_i \quad (2.42)$$

which yield



**Figure 2.7** Definitions of the terms for derivation of Brown's mean-squares distance algorithm.

$$x_T = \frac{\sum_{i=1}^N b_i^2 \sum_{i=1}^N a_i c_i - \sum_{i=1}^N a_i b_i \sum_{i=1}^N b_i c_i}{\sum_{i=1}^N a_i^2 \sum_{i=1}^N b_i^2 - \left( \sum_{i=1}^N a_i b_i \right)^2} \quad (2.43)$$

$$y_T = \frac{\sum_{i=1}^N a_i^2 \sum_{i=1}^N b_i c_i - \sum_{i=1}^N a_i b_i \sum_{i=1}^N a_i c_i}{\sum_{i=1}^N a_i^2 \sum_{i=1}^N b_i^2 - \left( \sum_{i=1}^N a_i b_i \right)^2} \quad (2.44)$$

The above miss distance for sensor  $i$  is expressed as

$$d_i = a_i x_T + b_i y_T - c_i \quad (2.45)$$

where  $i$  is the  $i$ th measurement of a line of bearing, and  $a_i$ ,  $b_i$ , and  $c_i$  are as given above. In matrix form this is

$$\vec{d} = \mathbf{H} \vec{x}_T - \vec{c} \quad (2.46)$$

In this expression,

$$\vec{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{bmatrix} \quad \vec{x}_T = \begin{bmatrix} x_T \\ y_T \end{bmatrix} \quad \mathbf{H} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ \vdots & \vdots \\ a_N & b_N \end{bmatrix} \quad \vec{d} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix} \quad (2.47)$$

The LSE estimator for the target location vector  $\vec{x}_T$  is given by (2.17) as

$$\vec{x}_T = \left[ \mathbf{H}^T \mathbf{W}^{-1} \mathbf{H} \right]^{-1} \mathbf{H}^T \mathbf{W}^{-1} \vec{c} \quad (2.48)$$

where, as usual, the superscript  $^{-1}$  denotes inverse and  $^T$  denotes transpose. We assume that the noise is zero mean AWGN. The variance of this estimator is given by the two-dimensional covariance matrix (2.19)

$$\mathbf{C}_{xy} = [\mathbf{H}^T \mathbf{W}^{-1} \mathbf{H}]^{-1} = \begin{bmatrix} \sigma_x^2 & \rho_{xy} \sigma_x \sigma_y \\ \rho_{xy} \sigma_x \sigma_y & \sigma_y^2 \end{bmatrix} \quad (2.49)$$

The EEP parameters are related to the elements of this covariance matrix as follows:

$$\begin{aligned} L_A &= \text{semimajor axis} \\ &= \frac{2(\sigma_x^2 \sigma_y^2 - \rho_{xy}^2 \sigma_x^2 \sigma_y^2) b^2}{\sigma_x^2 + \sigma_y^2 - [(\sigma_y^2 - \sigma_x^2)^2 + 4\rho_{xy}^2 \sigma_x^2 \sigma_y^2]^{1/2}} \end{aligned} \quad (2.50)$$

$$\begin{aligned} L_I &= \text{semiminor axis} \\ &= \frac{2(\sigma_x^2 \sigma_y^2 - \rho_{xy}^2 \sigma_x^2 \sigma_y^2) b^2}{\sigma_x^2 + \sigma_y^2 + [(\sigma_y^2 - \sigma_x^2)^2 + 4\rho_{xy}^2 \sigma_x^2 \sigma_y^2]^{1/2}} \end{aligned} \quad (2.51)$$

$$\tan 2\zeta = \frac{2\rho_{xy}^2 \sigma_x \sigma_y}{\sigma_y^2 - \sigma_x^2} \quad (2.52)$$

$$b = -2 \ln(1 - P_e) \quad (2.53)$$

$$P_e \text{ is probability of being inside} \quad (2.54)$$

Here,  $\zeta$  is the tilt angle of the semimajor axis of the ellipse relative to the  $x$ -axis.

The weighting matrix  $\mathbf{W}^{-1}$  is used to optimize the performance. In one application of this algorithm,  $\mathbf{W}^{-1}$  is given by

$$\mathbf{W}^{-1} = \frac{1}{\sum_i QF_i} \begin{bmatrix} QF_1 & 0 & 0 & \dots & 0 \\ 0 & QF_2 & 0 & \dots & 0 \\ 0 & 0 & QF_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & QF_N \end{bmatrix}$$

$$\times \begin{bmatrix} \frac{1}{\sigma_{d_1}^2} & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{\sigma_{d_2}^2} & 0 & \cdots & 0 \\ 0 & 0 & \frac{1}{\sigma_{d_3}^2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{\sigma_{d_N}^2} \end{bmatrix} \quad (2.55)$$

where  $QF_i$  is some quality factor associated with measurement  $i$ . It might be the variance of the measurement, it could be some higher order statistic as it makes sense, or it could be a measure of the *signal-to-noise ratio* (SNR), as examples.

### 2.5.1.1 Mean and Covariance of the Bias in Brown's Algorithm

Matrix  $\mathbf{H}$  is given by (2.47) which contains no random components. Therefore,  $\mathbf{H}^T \mathbf{H}$  contains no random components. Thus, from (2.27),

$$\vec{\delta} \approx -\frac{1}{N^2} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathcal{E}\{\vec{n}\} \quad (2.56)$$

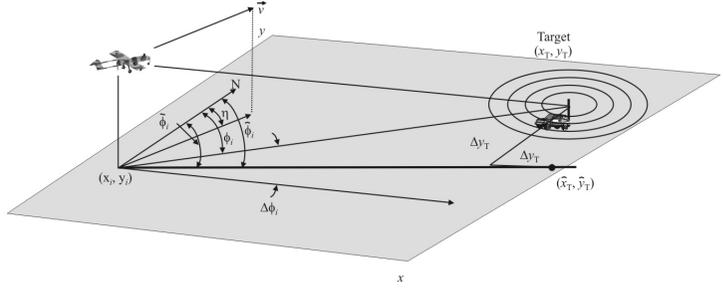
Since it is assumed that  $\vec{n} \sim \mathcal{N}(\vec{0}, \sigma^2)$ ,  $\mathcal{E}\{\vec{n}\} = 0$  and therefore  $\vec{\delta} = 0$ .

Denoting the covariance matrix of  $\vec{d}$  by  $\mathbf{C}_{dd}$  and assuming for simplicity that  $\mathbf{W} = \mathbf{I}$ , the variance of the estimator described by (2.48) is given by [8]

$$\text{Var}(\vec{c}) = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{C}_{dd} \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \quad (2.57)$$

## 2.5.2 Hemispheric Least-Squares Error Estimation Algorithm

A different least-squared algorithm for PF estimation is presented in this section. The measured bearing from a sensor,  $\phi_i$ , as illustrated in Figure 2.8, projected onto the surface of the Earth in the northern hemisphere (the other hemisphere will work as well with appropriate change of variables) can be characterized by [9, 10]



**Figure 2.8** Geometry for hemispheric least-squared.

$$\cos(\phi_i) = \frac{y_T - y_i}{\sqrt{(y_T - y_i)^2 + (x_T - x_i)^2}} \quad (2.58)$$

where the variables are defined in Figure 2.8 as (all in degrees)

$x_i$  is the longitude of the sensor during the  $i$ th observation interval

$y_i$  is the latitude of the sensor during the  $i$ th observation interval

$x_T$  is the actual longitude of the target emitter

$y_T$  is the actual latitude of the target emitter

$\eta$  is the heading of the sensor relative to north

$\phi_i$  is the  $i$ th measured bearing

$\Delta\phi_i$  is the error in the  $i$ th measured bearing

$\Delta x_i$  is the error in the calculation of the longitude in the  $i$ th observation interval

$\Delta y_i$  is the error in the calculation of the latitude in the  $i$ th observation interval

It is assumed that all LOBs are referenced to north; therefore, adjustments for vehicle/aircraft navigation/orientation/attitude have been previously accounted for. It is further assumed that  $N$  LOB measurements are made at regular intervals as indicated by the index  $i$ .

If the bearing errors are normally distributed with zero mean, the best least-squared estimates of the PF,  $(\hat{x}_T, \hat{y}_T)$ , assuming that  $(x_0, y_0)$  is available or estimated, are given by

$$x_i = \hat{x}_i - \Delta x_i \quad (2.59)$$

and

$$y_i = \hat{y}_i - \Delta y_i \quad (2.60)$$

Using (1.10), expanding  $(x_i - \Delta x_i)$  about  $\Delta x_T = \Delta y_T = 0$  yields

$$\Phi_i = x_{i1} \Delta x_T + x_{i2} \Delta y_T + \epsilon_i \quad (2.61)$$

where

$$\begin{aligned} \Phi_i &= \tilde{\phi}_i - \hat{\phi}_i \\ x_{i1} &= \frac{\hat{n}_i}{\hat{m}_i^2 + \hat{n}_i^2} \\ x_{i2} &= \frac{\hat{m}_i}{\hat{m}_i^2 + \hat{n}_i^2} \\ \hat{\phi}_i &= \cos^{-1} \left( \frac{\hat{m}_i}{\sqrt{\hat{m}_i^2 + \hat{n}_i^2}} \right) \\ \hat{m}_i &= \hat{x}_T - x_i \\ \hat{n}_i &= \hat{y}_T - y_i \\ \epsilon_i &= \Delta \phi_i + \text{higher order terms} \end{aligned}$$

If there is systematic error manifest in a bias in the measurements given by  $\phi_s$ , then

$$\Delta \phi_i = \phi_s + \Delta \phi_i^* \quad (2.62)$$

where  $\Delta \phi_i^*$  is the random part of  $\Delta \phi_i$ , and (2.61) is not necessarily the best estimator.

This procedure yields an estimate of the location of the target according to the expression

$$\vec{x} = \mathbf{X} \vec{\beta} + \vec{\epsilon} \quad (2.63)$$

where

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{N1} \\ x_{12} & x_{22} & \cdots & x_{N2} \end{bmatrix}^T \quad (2.64)$$

$$\vec{\beta} = [\Delta y_T \quad \Delta x_T]^T \quad (2.65)$$

$$\vec{y} = \vec{\phi} - \vec{\phi} \quad (2.66)$$

and

$$\vec{\epsilon} = [\epsilon_1 \quad \epsilon_2 \quad \cdots \quad \epsilon_N]^T \quad (2.67)$$

The estimate of  $\vec{\beta}$  is obtained by solving (2.63) for  $\vec{\beta}$  as in (2.17) with  $\mathbf{W} = \mathbf{I}$ :

$$\vec{\beta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \vec{x} = [\hat{\beta}_1 \quad \hat{\beta}_2]^T \quad (2.68)$$

where  $\hat{\beta}_1$  and  $\hat{\beta}_2$  are the best estimate of the errors in calculating the PF of the target. The target location is thus obtained by subtracting these values from the initial estimates given by  $\lambda_0$  and  $\psi_0$  as

$$\hat{y}_T = y_0 - \hat{\beta}_1 \quad (2.69)$$

and

$$\hat{x}_T = x_0 - \hat{\beta}_2 \quad (2.70)$$

This is one iteration through the procedure. It can be repeated to generate better estimates as many times as desired by using the computed values of  $\hat{\psi}_T$  and  $\hat{\lambda}_T$  from (2.69) and (2.70), respectively, as the initial values of the next iteration, so that

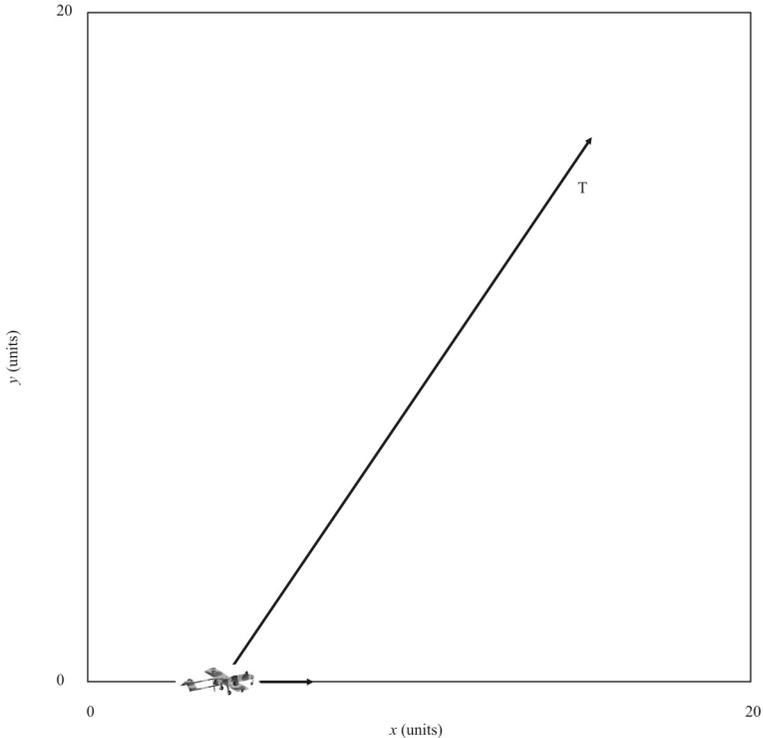
$$y_0 \leftarrow \hat{y}_T \quad (2.71)$$

and

$$x_0 \leftarrow \hat{x}_T \quad (2.72)$$

For the same reasons discussed above for Brown's algorithm, since the mean of the noise is zero, the resulting bias due to the algorithm is zero as well.

A MathCAD<sup>®</sup> program is given in Appendix 2A to compute one pass of this technique.

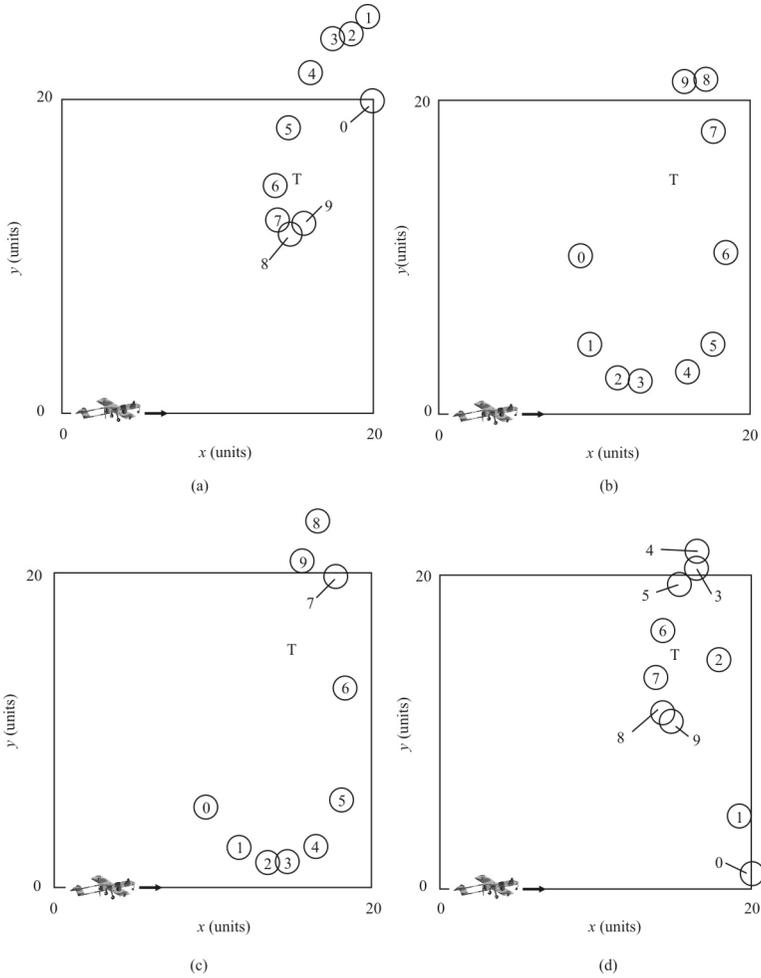


**Figure 2.9** Scenario for the least-squared example. The sensor is flying across the  $x$ -axis while the target is located at (15, 15).

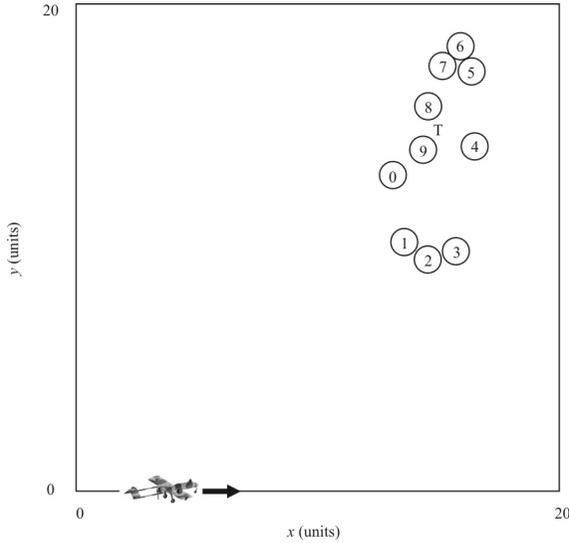
### Example

Consider the case when the target is located at (15, 15) units and the sensor is an aircraft flying along the  $x$ -axis as indicated in Figure 2.9. There are 21 measurements taken at unit intervals. The bearings are corrupted by Gaussian noise with a variance of 4 degrees squared.

A few results of estimating the PF of the target with the algorithm described in this section are shown in Figure 2.10. Although the cases illustrated in Figure 2.10 all appear to be converging to the target location, they do so slowly. Convergence was not obtained in all cases. When the initial points were (5, 10), (1, 1), and (10, 20), the algorithm diverged, for example.



**Figure 2.10** Illustrative results of estimating the PF of the target at coordinates (15, 15) with the least-squared algorithm. In (a) the initial point was (20, 20), in (b) it was (10, 10), in (c) it was at (10, 5), and in (d) it was at (20, 1).



**Figure 2.11** Results of LSE when the initial point was closer to the target location at (13, 13).

For those cases illustrated in Figure 2.10, more than 10 iterations are required to have the results converge. If the initial point is closer to the actual target location, then convergence is faster, although it is still not assured. The results when the initial point was (13, 13) are illustrated in Figure 2.11. While convergence appears to occur, it is still fairly slow.

A well-known problem with the least-squared approach to PF is that an initial estimate of the emitter location is required and it is only valid for small bearing errors. Convergence is not guaranteed, as illustrated in the previous example. Also, slow convergence may occur even when convergence occurs. A technique using total least-squares is described in Section 2.4.4 that does not have these shortcomings.

Since the bearing errors were assumed to have zero mean, the covariance matrix of the estimate, assuming no weighting is applied, is given by

$$\mathbf{C}_{xy} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{C}_{\epsilon\epsilon} \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \quad (2.73)$$

where  $\mathbf{C}_{\epsilon\epsilon}$  is the covariance matrix of  $\vec{\epsilon}$ .

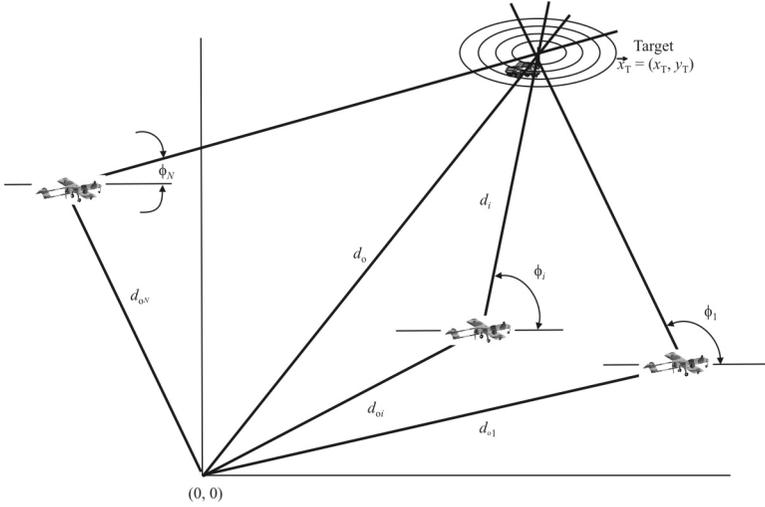


Figure 2.12 Geometry describing the algorithm developed by Pages-Zamora, Vidal, and Brooks.

### 2.5.3 Pages-Zamora Least-Squares

Pages-Zamora, Vidal, and Brooks [11] presented an LSE algorithm based on the cellular phone requirements imposed by the FCC in the United States. The fundamental algorithm is based on the geometry shown in Figure 2.12 [11] and recognition of the fact that

$$\vec{d}_o = \vec{d}_{oi} + d_i \vec{v}_i \quad \forall i \tag{2.74}$$

where  $d_i$  is the distance between the target and sensor  $i$  and  $\vec{v}_i$  is the unit vector given by

$$\vec{v}_i = [\cos \phi_i \quad \sin \phi_i]^T \tag{2.75}$$

The distance  $d_i$  can be removed from (2.74) as follows:

$$\vec{d}_o = \begin{bmatrix} x_T \\ y_T \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix} + d_{oi} \begin{bmatrix} \cos \phi_i \\ \sin \phi_i \end{bmatrix}$$

and changing these equations from matrix form to conventional form gives:

$$\begin{aligned}x_T &= x_i + d_{oi} \cos \phi_i \\y_T &= y_i + d_{oi} \sin \phi_i\end{aligned}\tag{2.76}$$

From (2.76),

$$x_T \sin \phi_i = x_i \sin \phi_i + d_{oi} \sin \phi_i \cos \phi_i\tag{2.77}$$

$$y_T \cos \phi_i = y_i \cos \phi_i + d_{oi} \sin \phi_i \cos \phi_i\tag{2.78}$$

Subtract (2.77) from (2.78) and we get

$$-x_i \sin \phi_i + y_i \cos \phi_i = -x_T \sin \phi_i + y_T \cos \phi_i\tag{2.79}$$

Doing this for all the sensors yields, in matrix form,

$$\begin{bmatrix} -x_1 \sin \phi_1 + y_1 \cos \phi_1 \\ -x_2 \sin \phi_2 + y_2 \cos \phi_2 \\ \vdots \\ -x_N \sin \phi_N + y_N \cos \phi_N \end{bmatrix} = \begin{bmatrix} -\sin \phi_1 & \cos \phi_1 \\ -\sin \phi_2 & \cos \phi_2 \\ \vdots & \vdots \\ \sin \phi_N & \cos \phi_N \end{bmatrix} \begin{bmatrix} x_T \\ y_T \end{bmatrix}\tag{2.80}$$

or

$$\vec{a}(\vec{\phi}) = \mathbf{H}(\vec{\phi})\vec{x}_T\tag{2.81}$$

Generally, this matrix expression is over determined—that is,  $N > 2$ , so there are more equations than unknowns. Therefore,  $\mathbf{H}$  cannot be directly inverted and the pseudoinverse must be used which results from the least-squares solution. The least-squared solution of (2.81) is given by

$$\hat{\vec{x}}_T = \left[ \mathbf{H}^T(\vec{\phi})\mathbf{H}(\vec{\phi}) \right]^{-1} \mathbf{H}^T(\vec{\phi})\vec{a}(\vec{\phi})\tag{2.82}$$

Now assume that the bearing measurements contain a small amount of error denoted by  $\delta\vec{\phi}$  so that  $\vec{\phi}' = \vec{\phi} + \delta\vec{\phi}$ . It is assumed that  $\delta\vec{\phi} \sim \mathcal{N}(\vec{0}, \sigma^2)$ . It is further assumed that the  $\sigma_i^2$  are uncorrelated with each other. Now,

$$\vec{a}(\vec{\phi}') = \vec{a} + \delta\vec{a}(\vec{\phi})\tag{2.83}$$

and

$$\mathbf{H}(\vec{\phi}') = \mathbf{H}(\vec{\phi}) + \delta\mathbf{H}(\vec{\phi}) \quad (2.84)$$

$$\vec{a}(\vec{\phi}) = \begin{bmatrix} -x_1 \sin \phi_1 + y_1 \cos \phi_1 \\ -x_2 \sin \phi_2 + y_2 \cos \phi_2 \\ \vdots \\ -x_N \sin \phi_N + y_N \cos \phi_N \end{bmatrix} \quad (2.85)$$

so that

$$\frac{\partial \vec{a}(\vec{\phi})}{\partial \vec{\phi}} = \begin{bmatrix} -x_1 \cos \phi_1 - y_1 \sin \phi_1 \\ -x_2 \cos \phi_2 + y_2 \sin \phi_2 \\ \vdots \\ -x_N \cos \phi_N + y_N \sin \phi_N \end{bmatrix} \quad (2.86)$$

Using the approximations that  $\delta\vec{\phi} \approx \partial\vec{\phi}$  and  $\delta\vec{a} \approx \partial\vec{a}$ , then

$$\delta\vec{a} = \begin{bmatrix} -x_1 \delta\phi_1 \cos \phi_1 - y_1 \delta\phi_1 \sin \phi_1 \\ -x_2 \delta\phi_2 \cos \phi_2 + y_2 \delta\phi_2 \sin \phi_2 \\ \vdots \\ -x_N \delta\phi_N \cos \phi_N + y_N \delta\phi_N \sin \phi_N \end{bmatrix} \quad (2.87)$$

Likewise,

$$\mathbf{H}(\vec{\phi}) = \begin{bmatrix} -\sin \phi_1 & \cos \phi_1 \\ -\sin \phi_2 & \cos \phi_2 \\ \vdots & \vdots \\ \sin \phi_N & \cos \phi_N \end{bmatrix} \quad (2.88)$$

so that

$$\frac{\partial \mathbf{H}(\vec{\phi})}{\partial \vec{\phi}} = \begin{bmatrix} -\cos \phi_1 & -\sin \phi_1 \\ -\cos \phi_2 & -\sin \phi_2 \\ \vdots & \vdots \\ -\cos \phi_N & -\sin \phi_N \end{bmatrix} \quad (2.89)$$

Using the same approximations that lead to (2.87) yields

$$\delta \mathbf{H} = \begin{bmatrix} -\delta \phi_1 \cos \phi_1 & -\delta \phi_1 \sin \phi_1 \\ -\delta \phi_2 \cos \phi_2 & -\delta \phi_2 \sin \phi_2 \\ \vdots & \vdots \\ -\delta \phi_N \cos \phi_N & -\delta \phi_N \sin \phi_N \end{bmatrix} \quad (2.90)$$

Substituting (2.83) and (2.84) into (2.81) yields

$$(\mathbf{H} + \delta \mathbf{H})^T (\vec{a} + \delta \vec{a}) = (\mathbf{H} + \delta \mathbf{H})^T (\mathbf{H} + \delta \mathbf{H})(\vec{x}_T + \delta \vec{x}_T) \quad (2.91)$$

In this expression,  $\delta \vec{x}_T$  is identified as the bias in the PF algorithm. Carrying out the multiplications and discarding all terms where two or more incremental terms are products yields

$$\delta \vec{x}_T = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\delta \vec{a} - \delta \mathbf{H} \vec{x}_T) \quad (2.92)$$

Substituting (2.87) and (2.90) into (2.92) shows that  $\delta \vec{x} = 0$  and the estimator is therefore unbiased.

The covariance matrix of  $\delta \vec{x}_T$  is given by

$$\mathbf{C}_{xx} = \mathcal{E} \{ \delta \vec{x}_T \delta \vec{x}_T^T \} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{\Lambda} [(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T]^T \quad (2.93)$$

where

$$\mathbf{\Lambda} = \begin{bmatrix} \sigma_1^2 d_1^2 & & & \\ & \sigma_2^2 d_2^2 & & \\ & & \ddots & \\ & & & \sigma_N^2 d_N^2 \end{bmatrix} \quad (2.94)$$

### 2.5.4 Total Least-Squares Error

The method of *total least-squares estimation* (TLSE) starts with squaring (2.58):

$$\cos^2(\phi_i) = \frac{(x_T - x_i)^2}{(x_T - x_i)^2 + (y_T - y_i)^2} \quad (2.95)$$

Manipulating this equation eventually yields

$$a_i x_T + b_i y_T + c_i x_T^2 + d_i y_T^2 = -e_i \quad (2.96)$$

where

$$a_i = -2x_i \cos^2(\phi_i) + 2x_i$$

$$b_i = -2 \cos^2(\phi_i) y_i \cos^2 x_i$$

$$c_i = \cos^2(\phi_i) - 1$$

$$d_i = \cos^2(\phi_i) \cos^2 x_i$$

$$e_i = x_i^2 \cos^2(\phi_i) + \cos^2(\phi_i) y_i^2 \cos^2 x_i - x_i^2$$

For  $N$  observations, (2.96) generates  $N$  overdetermined equations as

$$\mathbf{A} \vec{\theta}_T = -\vec{e} \quad (2.97)$$

where

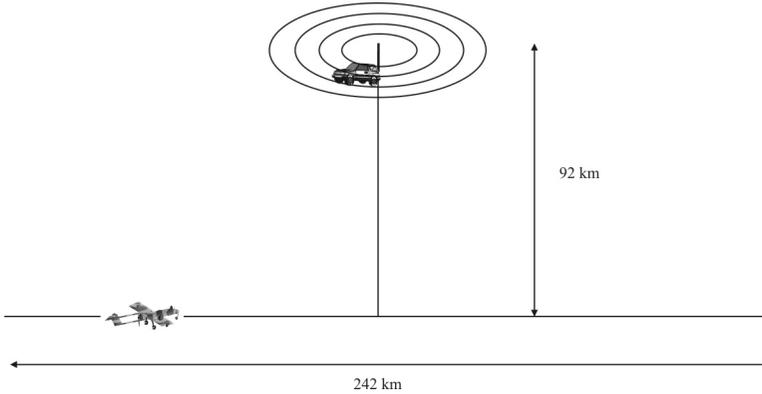
$$\mathbf{A} = \begin{bmatrix} a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ \vdots & \vdots & \vdots & \vdots \\ a_N & b_N & c_N & d_N \end{bmatrix} \quad (2.98)$$

$$\vec{\theta}_T = [x_T \quad y_T \quad x_T^2 \quad y_T^2]^T \quad (2.99)$$

$$\vec{e} = [e_1 \quad e_2 \quad \cdots \quad e_N]^T \quad (2.100)$$

Rewriting (2.97) as

$$\mathbf{A} \vec{\theta}_T + \vec{e} = \vec{0} \quad (2.101)$$



**Figure 2.13** Geometry for the TLSE example.

which can be expressed as

$$[\mathbf{A} : -\vec{e}][\vec{\theta}_T^T : -1]^T = \vec{0} \quad (2.102)$$

The *singular value decomposition* (SVD) of the first term in (2.102) can be expressed as

$$[\mathbf{A} : -\vec{e}] = \mathbf{U}\mathbf{S}\mathbf{V}^H \triangleq [\mathbf{U}_s \quad \mathbf{U}_o] \begin{bmatrix} \mathbf{S}_s & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}_s^H \\ \mathbf{V}_o^H \end{bmatrix} \quad (2.103)$$

The  $\sigma_i$  in  $\mathbf{S}$  are singular values of  $[\mathbf{A} : -\vec{e}]$ . The solution of (2.102) is given by the last column of  $\mathbf{V}$  if the smallest singular value of  $\mathbf{A}$  is greater than the smallest singular value of  $[\mathbf{A} : -\vec{e}]$ . The TLSE solution is obtained by scaling  $\vec{v}_{M+1}$  until the last component is  $-1$ :

$$\begin{bmatrix} \vec{\theta}_T^T & -1 \end{bmatrix} = -\frac{\vec{v}_{N+1}}{v_{N+1,N+1}} \quad (2.104)$$

It is tacitly assumed here that the errors in the bearings are i.i.d. with zero mean.

**Table 2.1** Results of the TLSE Algorithm for the Example

Case	Random Error	Systematic Error	Error (km)
1	$\sim\mathcal{N}(0, 1)$	$0^\circ$	0.2637
2	$\sim\mathcal{N}(0, 1)$	$-3^\circ$	2.917
3	$\sim\mathcal{N}(0, 1)$	$6^\circ$	7.339

Source: [12].

### Example

Rao and Reddy conducted a simulation experiment to evaluate the performance of the TLSE approach to geolocation [12]. The geometry studied is illustrated in Figure 2.13 [12], where 122 bearings were taken along the sensor path as shown. The PF was computed using the TLSE algorithm as presented. The results for three cases are given in Table 2.1. The estimates produced by the TLS method increased substantially when there was systematic bias in the bearing measurements.

The effects of the  $\gamma = \text{SNR}$  on the TLSE method were also examined during the same experiment. The results are shown in Figure 2.14 [12]. As expected, the error decreased as the SNR increased, with the error dropping to less than 1% of range (from the perpendicular bisector of the baseline) for values of SNR above about 25 dB.

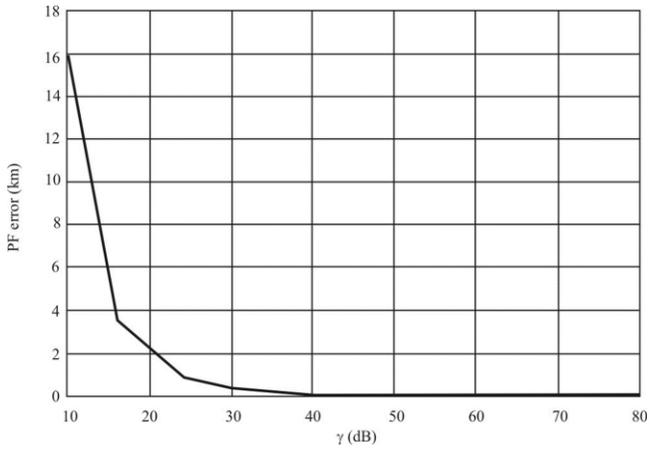
The effects of decreasing the number of observations were also examined for the TLS algorithm, with the results as indicated in Figure 2.15 [12]. One percent of range was achieved for 80 observations and above, but even with only 10 observations the error was only 1.5 km.

## 2.6 Minimum Mean-Squares Error Estimation

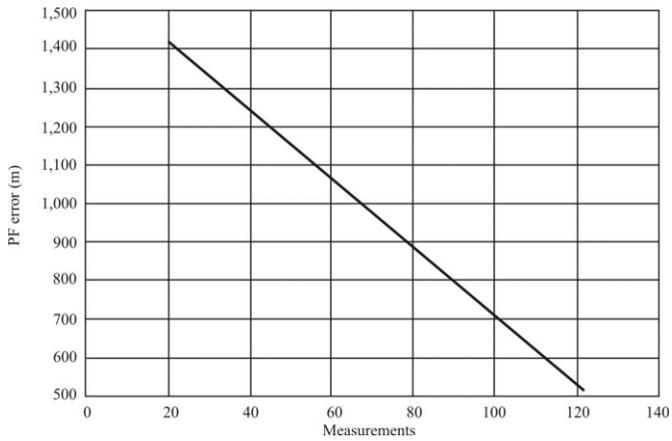
The PF algorithms presented in this section are based on dynamic system models. These algorithms are typically applicable when the measurements are obtained recursively with the PF updated during each iteration. The section begins with a background discussion on dynamical systems before the algorithms are presented.

### 2.6.1 Dynamical Systems

Dynamical systems can be described by the state transition equation:



**Figure 2.14** PF performance of TLSE versus SNR.



**Figure 2.15** TLSE performance versus the number of observations.

$$\vec{s}_i = \Phi_i \vec{s}_{i-1} + \mathbf{B} \vec{u}_i + \vec{n}_i, \quad i = 0, 1, \dots \quad (2.105)$$

with  $\vec{s} \in \mathfrak{R}^N$  and the measurement vector given by

$$\vec{z}_i = \mathbf{H} \vec{s}_i + \vec{\eta}_i, \quad i = 0, 1, \dots \quad (2.106)$$

with  $\vec{z}_i \in \mathfrak{R}^p$ , where  $\vec{s}$  is the state of the system at time  $i$  and  $\Phi_i$  is the state transition matrix, which can be linear, nonlinear, time variant, or time invariant. Matrix  $\mathbf{B}$  relates the effects of the input  $\vec{u}_i$  to the state  $\vec{s}_i$  at sample time  $i$ . The vector of random noise  $\vec{n}_i$  is usually modeled as white noise:

$$\mathcal{E}\{\vec{n}_i\} = \vec{0} \quad (2.107)$$

$$\mathcal{E}\{\vec{n}_i \vec{n}_j^T\} = \begin{cases} \sigma_n^2, & i = j \\ 0, & \text{otherwise} \end{cases} \quad (2.108)$$

so

$$\mathbf{C}_{nn} = \text{diag}[\sigma_n^2 \quad \sigma_n^2 \quad \dots \quad \sigma_n^2] = \sigma_n^2 \mathbf{I} \quad (2.109)$$

where  $\mathbf{C}_{nn}$  is the covariance matrix of the process noise.

The measurement noise is characterized by

$$\mathcal{E}\{\vec{\eta}_i\} = \vec{0} \quad (2.110)$$

$$\mathcal{E}\{\vec{\eta}_i \vec{\eta}_j^T\} = \begin{cases} \sigma_{\eta}^2, & i = j \\ 0, & \text{otherwise} \end{cases} \quad (2.111)$$

so

$$\mathbf{C}_{\eta\eta} = \text{diag}[\sigma_{\eta_1}^2 \quad \sigma_{\eta_2}^2 \quad \dots \quad \sigma_{\eta_p}^2] \quad (2.112)$$

is the  $\mathbf{C}_{\eta\eta}$  measurement error covariance matrix. The state variable noise and measurement noise processes are assumed to be independent:

$$\mathcal{E}\{\vec{\eta}_i \vec{n}_j^T\} = 0, \quad \forall i, j \quad (2.113)$$

The  $N \times N$  matrix  $\Phi_i$  relates the state at the previous time step  $i - 1$  to the state at the current step  $i$ , in the absence of any driving function  $\vec{u}_i$  or process noise  $\vec{n}_i$ . The  $N \times L$  matrix  $\mathbf{B}$  relates the control input  $\vec{u}$  (if present) to the state  $\vec{s}_i$ . The  $P \times N$  matrix  $\mathbf{H}$  relates the state to the measurement  $\vec{z}_i$ .

Thus, we wish to determine the values of  $\vec{s}_i$  at instant  $i$  but are not able to measure the states directly. Instead, we measure  $\vec{z}_i = \tilde{\vec{z}}_i$ , which are functions of  $\vec{s}_i$ . Knowing these  $\vec{z}_i$ , we wish to estimate the values of  $\vec{s}_i$ . For *minimum mean-square error* (MMSE) estimation, the optimality criterion used is to minimize the mean of the squared error given by

$$\mathcal{E}\{(\vec{s}_i - \hat{\vec{s}}_i)^2\} \quad (2.114)$$

We assume that the *probability density functions* (PDF)  $p(\vec{s}_i)$  and  $p(\vec{s}_i | \vec{z}_i)$  are known. Therefore, we want to find the best guess of  $\vec{s}_i$  given that  $\vec{z}_i = \tilde{\vec{z}}_i$  in the MMSE sense. In other words, we want to find a function  $\hat{\vec{s}}_i = g(\hat{\vec{z}}_i)$  such that we can minimize

$$J = \text{MSE} = \mathcal{E}\{[\vec{s}_i - g(\tilde{\vec{z}}_i)]^2 | \vec{z}_i = \tilde{\vec{z}}_i\} \quad (2.115)$$

Now,

$$\frac{dJ}{dg} = \frac{d}{dg} \mathcal{E}\{[\vec{s}_i - g(\tilde{\vec{z}}_i)]^2 | \vec{z}_i = \tilde{\vec{z}}_i\} = 0 \quad (2.116)$$

yields

$$\begin{aligned} \mathcal{E}\left\{\frac{d}{dg}[\vec{s}_i - g(\tilde{\vec{z}}_i)]^2\right\} &= 0 \\ \mathcal{E}\left\{\frac{d}{dg}[\vec{s}_i^2 - 2\vec{s}_i g(\tilde{\vec{z}}_i) + g^2(\tilde{\vec{z}}_i)]\right\} &= 0 \\ -2\mathcal{E}\{\vec{s}_i\} + 2g(\tilde{\vec{z}}_i) &= 0 \\ g(\tilde{\vec{z}}_i) &= \mathcal{E}\{\vec{s}_i | \vec{z}_i = \tilde{\vec{z}}_i\} \end{aligned} \quad (2.117)$$

So the optimum estimate given  $\vec{z}_i = \tilde{z}_i$  is the expected value, or mean value, of  $\vec{s}_i$ .

The stochastic function  $g(\vec{z}_i) = \mathcal{E}\{\vec{s}_i | \vec{z}_i\}$  minimizes the unconditional *mean-square error* (MSE)  $\mathcal{E}\{[\vec{s}_i - g(\vec{z}_i)]^2\}$  in addition to minimizing the conditional MSE  $\mathcal{E}\{[\vec{s}_i - g(\vec{z}_i)]^2 | \vec{z}_i\}$ . Using iterated expectations,

$$\begin{aligned} \mathcal{E}\{[\vec{s}_i - g(\vec{z}_i)]^2\} &= \mathcal{E}\{\mathcal{E}\{[\vec{s}_i - g(\vec{z}_i)]^2 | \vec{z}_i\}\} \\ &= \int_{-\infty}^{\infty} \left\{ \mathcal{E}\{[\vec{s}_i - g(\vec{z}_i)]^2 | \vec{z}_i = \tilde{z}_i\} \right\} p_{\vec{z}_i}(\tilde{z}_i) d\tilde{z}_i \end{aligned}$$

Since  $g(\vec{z}_i) = \mathcal{E}\{\vec{s}_i | \vec{z}_i = \tilde{z}_i\}$  minimizes  $\mathcal{E}\{[\vec{s}_i - g(\vec{z}_i)]^2 | \vec{z}_i = \tilde{z}_i\}$  for each  $\tilde{z}_i$ , the integral is also minimized. Let  $\hat{s}_i = \mathcal{E}\{\vec{s}_i | \vec{z}_i\}$  and  $\hat{s} = \vec{s}_i - \hat{s}_i$ . Then

$$\begin{aligned} \mathcal{E}\{\hat{s}_i | \vec{z}_i = \tilde{z}_i\} &= \mathcal{E}\{\vec{s}_i - \hat{s}_i | \vec{z}_i = \tilde{z}_i\} \\ &= \mathcal{E}\{\vec{s}_i | \vec{z}_i = \tilde{z}_i\} - \mathcal{E}\{\hat{s}_i | \vec{z}_i = \tilde{z}_i\} \\ &= \mathcal{E}\{\vec{s}_i | \vec{z}_i = \tilde{z}_i\} - \mathcal{E}\{\hat{s}_i | \vec{z}_i = \tilde{z}_i\} \\ &= 0 \\ \mathcal{E}\{\hat{s}_i\} &= \mathcal{E}\{\mathcal{E}\{\vec{s}_i | \vec{z}_i\}\} = 0 \end{aligned}$$

so the MMSE estimator is unbiased.

In addition,

$$\begin{aligned} \mathbf{C}_{\hat{s}_i, \hat{s}_i} &= \mathcal{E}\left\{(\hat{s}_i - \mathcal{E}\{\hat{s}_i\})(\hat{s}_i - \mathcal{E}\{\hat{s}_i\})\right\} \\ &= \mathcal{E}\left\{\hat{s}_i(\hat{s}_i - \mathcal{E}\{\hat{s}_i\})\right\} \\ &= \mathcal{E}\left\{\mathcal{E}\{\hat{s}_i(\hat{s}_i - \mathcal{E}\{\hat{s}_i\}) | \vec{z}_i\}\right\} \\ &= \mathcal{E}\left\{(\hat{s}_i - \mathcal{E}\{\hat{s}_i\}) \underbrace{\mathcal{E}\{\hat{s}_i | \vec{z}_i\}}_{=0}\right\} = 0 \end{aligned}$$

so  $\hat{s}_i$  and  $\hat{s}_i$  are uncorrelated, and as a result,

$$\mathbf{V}(\vec{s}_i) = \mathbf{V}(\hat{s}_i) + \mathbf{V}(\bar{s}_i) \quad (2.118)$$

### 2.6.2 Linear Minimum Mean-Square Estimation

The MMSE estimator requires the a posteriori density  $p(\vec{\theta}|\vec{z})$  [ $p(\vec{s}_i|\vec{z}_i)$  above] to be known, which is frequently difficult to calculate. If the analysis is restricted to the class of estimators that are affine (linear with a possible translation) functions of the observation vector  $\vec{z}$ , however, the *linear minimum mean-square error* (LMMSE) estimator ensues which is a function of the first and second moments of  $\vec{\theta}$  and  $\vec{z}$  only. Furthermore, if  $\vec{z}$  and  $\vec{\theta}$  are jointly Gaussian, the LMMSE estimator is the same as the MMSE estimator.

The LMMSE estimator for an  $K$ -dimensional unknown vector  $\vec{\theta}$  and an  $N$ -dimensional observation vector  $\vec{z}$  will be considered, which are of the form

$$\hat{\vec{\theta}} = \mathbf{H}\vec{z} + \vec{b} \quad (2.119)$$

where  $\mathbf{H}$  is an  $K \times N$  matrix and  $\vec{b}$  is an  $K$  element vector.  $\mathbf{H}$  and  $\vec{b}$  are chosen to minimize

$$\mathcal{E}_{z_0} \left\{ \left( \vec{\theta} - \hat{\vec{\theta}} \right)^T \left( \vec{\theta} - \hat{\vec{\theta}} \right) \right\} \quad (2.120)$$

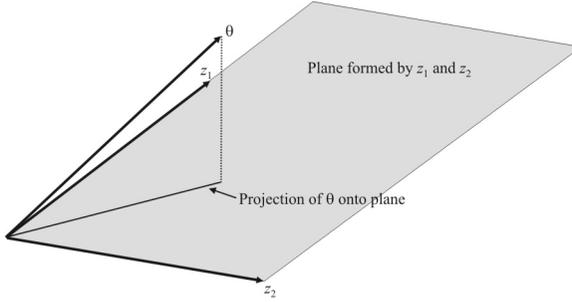
The LMMSE estimate is

$$\vec{\theta} = \mathcal{E}\{\vec{\theta}\} + \mathbf{C}_{\theta z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E}\{\vec{z}\}) \quad (2.121)$$

The value  $\mathcal{E}\{\vec{\theta}\}$  is the mean of  $\vec{\theta}$ ,  $\mathcal{E}\{\vec{z}\}$  is the mean of  $\vec{z}$ ,  $\mathbf{C}_{\theta z}$  is the cross covariance matrix of  $\vec{\theta}$  and  $\vec{z}$  given by

$$\mathbf{C}_{\theta z} = \mathcal{E} \left\{ \left( \vec{\theta} - \mathcal{E}\{\vec{\theta}\} \right) \left( \vec{z} - \mathcal{E}\{\vec{z}\} \right)^T \right\} \quad (2.122)$$

and  $\mathbf{C}_{zz}$  is the autocovariance matrix of  $\vec{z}$  given by



**Figure 2.16**  $N = 2$  example of projection of  $\vec{\theta}$  onto the plane formed by  $z_1$  and  $z_2$ .

$$\mathbf{C}_{zz} = \mathcal{E} \left\{ (\vec{z} - \mathcal{E}\{\vec{z}\})(\vec{z} - \mathcal{E}\{\vec{z}\})^T \right\} \quad (2.123)$$

Assume that  $\vec{z}$  and  $\vec{\theta}$  have zero means. Then the orthogonality principle applies.

### Property: Orthogonality Principle

For the LMMSE estimate  $\hat{\vec{\theta}}$ , the estimate error is orthogonal to the data. That is,

$$\mathcal{E} \left\{ \left( \vec{\theta} - \hat{\vec{\theta}} \right) \vec{z}^T \right\} = \mathbf{0} \quad (2.124)$$

Note that in this case,  $\mathbf{0}$  is a  $K \times N$  matrix.

An example when  $N = 2$  is illustrated in Figure 2.16. The projection of  $\hat{\vec{\theta}}$  onto the plane formed by the measurement vectors will be the smallest (zero in this case) when  $\hat{\vec{\theta}}$  is orthogonal to that plane.

*Proof:* Let  $\mathbf{A}$  be a  $K \times N$  matrix for which the orthogonality principle holds:

$$\mathcal{E} \left\{ \left( \vec{\theta} - \mathbf{A}\vec{z} \right) \vec{z}^T \right\} = \mathbf{0} \quad (2.125)$$

Let  $\mathbf{B}$  be any other  $K \times N$  matrix and  $\mathbf{B}\bar{\mathbf{z}}$  be the estimate that it gives for  $\bar{\boldsymbol{\theta}}$ . The expected squared error of this estimate is

$$\begin{aligned}
 \mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{B}\bar{\mathbf{z}}\right)^{\top}\left(\bar{\boldsymbol{\theta}}-\mathbf{B}\bar{\mathbf{z}}\right)\right\} &= \mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}+\mathbf{A}\bar{\mathbf{z}}-\mathbf{B}\bar{\mathbf{z}}\right)^{\top}\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}+\mathbf{A}\bar{\mathbf{z}}-\mathbf{B}\bar{\mathbf{z}}\right)\right\} \\
 &= \mathcal{E}\left\{\left[\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}+(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]^{\top}\left[\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}+(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]\right\} \\
 &= \mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)^{\top}\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)\right\} \\
 &\quad + \mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)^{\top}(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right\} \\
 &\quad + \mathcal{E}\left\{\left[(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]^{\top}\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)\right\} \\
 &\quad + \mathcal{E}\left\{\left[(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]^{\top}\left[(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]\right\}
 \end{aligned} \tag{2.126}$$

$$\begin{aligned}
 \mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)^{\top}(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right\} &= \text{tr}\mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)\left[(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]^{\top}\right\} \\
 &= \text{tr}\mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)\bar{\mathbf{z}}^{\top}(\mathbf{A}-\mathbf{B})^{\top}\right\} \\
 &= \text{tr}\mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)\bar{\mathbf{z}}^{\top}\right\}(\mathbf{A}-\mathbf{B})^{\top} \\
 &= \text{tr}\mathbf{0}(\mathbf{A}-\mathbf{B})^{\top} \\
 &= 0
 \end{aligned} \tag{2.127}$$

In a similar fashion, it can be shown that  $\mathcal{E}\left\{\left[(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]^{\top}\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)\right\} = 0$ . Thus, the squared error becomes

$$\begin{aligned}
 \mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{B}\bar{\mathbf{z}}\right)^{\top}\left(\bar{\boldsymbol{\theta}}-\mathbf{B}\bar{\mathbf{z}}\right)\right\} &= \mathcal{E}\left\{\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)^{\top}\left(\bar{\boldsymbol{\theta}}-\mathbf{A}\bar{\mathbf{z}}\right)\right\} \\
 &\quad + \mathcal{E}\left\{\left[(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]^{\top}(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right\}
 \end{aligned} \tag{2.128}$$

but

$$\mathcal{E}\left\{\left[(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right]^{\top}(\mathbf{A}-\mathbf{B})\bar{\mathbf{z}}\right\} \geq 0 \tag{2.129}$$

so

$$\mathcal{E}\left\{\left(\vec{\theta} - \mathbf{B}\vec{z}\right)^T \left(\vec{\theta} - \mathbf{B}\vec{z}\right)\right\} \geq \mathcal{E}\left\{\left(\vec{\theta} - \mathbf{A}\vec{z}\right)^T \left(\vec{\theta} - \mathbf{A}\vec{z}\right)\right\} \quad (2.130)$$

Thus, the expected squared error of any linear estimator cannot be smaller than the expected squared error of a linear estimator that satisfies the orthogonality principle. Therefore, the LMMSE estimator must satisfy the orthogonality principle.

There is a geometric interpretation of the orthogonality principle. A collection of random variables can be considered as a Hilbert space (a vector space). The random variables  $z_0$  through  $z_{N-1}$  and  $\vec{\theta}$  are elements of the space. The inner product between elements is

$$(z_i, z_j) = \mathcal{E}\{z_i z_j\} \quad (2.131)$$

The magnitude of an element of the space is

$$\|z_i\| = \sqrt{(z_i, z_i)} = \sqrt{\mathcal{E}\{z_i z_i\}} \quad (2.132)$$

The vector  $\hat{\vec{\theta}}$  is a linear combination of  $z_0$  through  $z_{N-1}$ . To make the magnitude of the estimate error as small as possible, the estimate is a projection of  $\hat{\vec{\theta}}$  onto  $z_0$  through  $z_{N-1}$ ; in other words, the error should be perpendicular (in  $N$ -space) to  $z_0$  through  $z_{N-1}$ .

Given that the estimator has the form  $\hat{\vec{\theta}} = \mathbf{H}\vec{z}$ , the orthogonality principle is used to solve for the value of  $\mathbf{H}$  as follows.

$$\mathcal{E}\left\{\left(\vec{\theta} - \mathbf{H}\vec{z}\right)\vec{z}^T\right\} = \mathbf{0} \quad (2.133)$$

Expanding the inner product,

$$\mathcal{E}\left\{\vec{\theta}\vec{z}^T\right\} - \mathcal{E}\left\{\mathbf{H}\vec{z}\vec{z}^T\right\} = \mathcal{E}\left\{\vec{\theta}\vec{z}^T\right\} - \mathbf{H}\mathcal{E}\left\{\vec{z}\vec{z}^T\right\} = \mathbf{0} \quad (2.134)$$

These expected value terms are identified as the cross-covariance of  $\vec{\theta}$  and  $\vec{z}$  and the autocovariance of  $\vec{z}$ , respectively. Therefore,

$$\mathbf{C}_{\theta z} - \mathbf{H}\mathbf{C}_{zz} = \mathbf{0} \quad (2.135)$$

and

$$\mathbf{H} = \mathbf{C}_{\theta z} \mathbf{C}_{zz}^{-1} \quad (2.136)$$

so

$$\hat{\vec{\theta}} = \mathbf{C}_{\theta z} \mathbf{C}_{zz}^{-1} \vec{z} \quad (2.137)$$

If  $\vec{z}$  and  $\vec{\theta}$  do not have zero means, new random variables are created that do have zero means as

$$\tilde{\vec{z}} = \vec{z} - \mathcal{E}\{\vec{z}\} \quad (2.138)$$

$$\tilde{\vec{\theta}} = \vec{\theta} - \mathcal{E}\{\vec{\theta}\} \quad (2.139)$$

The LMMSE estimate of  $\tilde{\vec{\theta}}$  is

$$\begin{aligned} \hat{\tilde{\vec{\theta}}} &= \mathbf{C}_{\tilde{\theta} \tilde{z}} \mathbf{C}_{\tilde{z} \tilde{z}}^{-1} \tilde{\vec{z}} \\ &= \mathbf{C}_{\theta z} \mathbf{C}_{zz}^{-1} \tilde{\vec{z}} \end{aligned} \quad (2.140)$$

From (2.139),

$$\hat{\tilde{\vec{\theta}}} = \hat{\vec{\theta}} + \mathcal{E}\{\vec{\theta}\} = \mathcal{E}\{\vec{\theta}\} + \mathbf{C}_{\theta z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E}\{\vec{z}\}) \quad (2.141)$$

This estimator is unbiased:

$$\mathcal{E}\{\hat{\tilde{\vec{\theta}}}\} = \mathcal{E}\{\vec{\theta}\} + \mathbf{C}_{\theta z} \mathbf{C}_{zz}^{-1} \mathcal{E}\{(\vec{z} - \mathcal{E}\{\vec{z}\})\} = \mathbf{0} \quad (2.142)$$

with the expected squared error matrix given by

$$\begin{aligned}
\mathcal{E} \left\{ \left( \vec{\theta} - \hat{\vec{\theta}} \right) \left( \vec{\theta} - \hat{\vec{\theta}} \right) \vec{z}^T \right\} &= \mathcal{E} \left\{ \begin{aligned} &\left[ \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} - \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{x} - \mathcal{E} \{ \vec{x} \}) \right] \\ &\times \left[ \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} - \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \}) \right]^T \end{aligned} \right\} \\
&= \mathcal{E} \left\{ \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right) \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right)^T \right\} \\
&\quad - \mathcal{E} \left\{ \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right) \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \})^T \right] \right\} \\
&\quad - \mathcal{E} \left\{ \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \}) \right] \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right)^T \right\} \\
&\quad + \mathcal{E} \left\{ \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \}) \right] \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \}) \right]^T \right\} \\
&= \mathcal{E} \left\{ \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right) \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right)^T \right\} \\
&\quad - \mathcal{E} \left\{ \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right) \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \})^T \right] \right\} \\
&\quad - \mathcal{E} \left\{ \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \}) \right] \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right)^T \right\} \\
&\quad + \mathcal{E} \left\{ \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \}) \right] \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \}) \right]^T \right\} \\
&= \mathcal{E} \left\{ \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right) \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right)^T \right\} \\
&\quad - \mathcal{E} \left\{ \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right) (\vec{z} - \mathcal{E} \{ \vec{z} \})^T (\mathbf{C}_{zz}^{-1})^T \mathbf{C}_{0z}^T \right\} \\
&\quad - \mathcal{E} \left\{ \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{x} \}) \right] \left( \vec{\theta} - \mathcal{E} \{ \vec{\theta} \} \right)^T \right\} \\
&\quad + \mathcal{E} \left\{ \left[ \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} (\vec{z} - \mathcal{E} \{ \vec{z} \}) \right] (\vec{z} - \mathcal{E} \{ \vec{z} \})^T (\mathbf{C}_{zz}^{-1})^T \mathbf{C}_{0z}^T \right\} \\
&= \mathbf{C}_{00} - \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} \mathbf{C}_{z0} \\
&\quad - \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} \mathbf{C}_{z0} + \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} \mathbf{C}_{zz} \mathbf{C}_{zz}^{-1} \mathbf{C}_{z0} \\
&= \mathbf{C}_{00} - \mathbf{C}_{0z} \mathbf{C}_{zz}^{-1} \mathbf{C}_{z0} \tag{2.143}
\end{aligned}$$

**Example**

$N = 1$  and  $\theta$  has a mean  $\mu_\theta$  and a variance  $\sigma_\theta^2$ :

$$z_0 = \theta + n_0$$

This example might apply to a single LOB from a single sensor. The noise  $n_0 \sim \mathcal{N}(0, \sigma_n^2)$  and is independent of  $\theta$ .

$$\begin{aligned} \mu_x &= \mathcal{E}\{x_0\} = \mu_\theta \\ c_{zz} &= \mathcal{E}\{(z_0 - \mu_z)^2\} \\ &= \mathcal{E}\{(\theta + n_0 - \mu_\theta)^2\} \\ &= \mathcal{E}\{(\theta - \mu_\theta)^2\} + \underbrace{\mathcal{E}\{2(\theta - \mu_\theta)n_0\}}_{=0} + \mathcal{E}\{n_0^2\} \\ &= \sigma_\theta^2 + \sigma_n^2 \\ c_{\theta x} &= \mathcal{E}\{(\theta - \mu_\theta)(z_0 - \mu_z)\} \\ &= \mathcal{E}\{(\theta - \mu_\theta)((\theta - \mu_\theta) + n_0)\} \\ &= \mathcal{E}\{(\theta - \mu_\theta)(\theta - \mu_\theta)\} + \underbrace{\mathcal{E}\{(\theta - \mu_\theta)n_0\}}_{=0} \\ &= \sigma_\theta^2 \\ \hat{\theta} &= \mu_\theta + c_{\theta z} c_{zz}^{-1} (z_0 - \mu_z) \\ &= \mu_\theta + \frac{\sigma_\theta^2}{\sigma_\theta^2 + \sigma_n^2} (z_0 - \mu_\theta) \\ &= \frac{\sigma_\theta^2}{\sigma_\theta^2 + \sigma_n^2} z_0 + \frac{\sigma_n^2}{\sigma_\theta^2 + \sigma_n^2} \mu_\theta \end{aligned}$$

The expected squares error for this estimator is

$$\mathcal{E}\left\{\left(\hat{\theta} - \theta\right)^2\right\} = c_{\theta\theta} - c_{\theta z} c_{zz}^{-1} c_{z\theta}$$

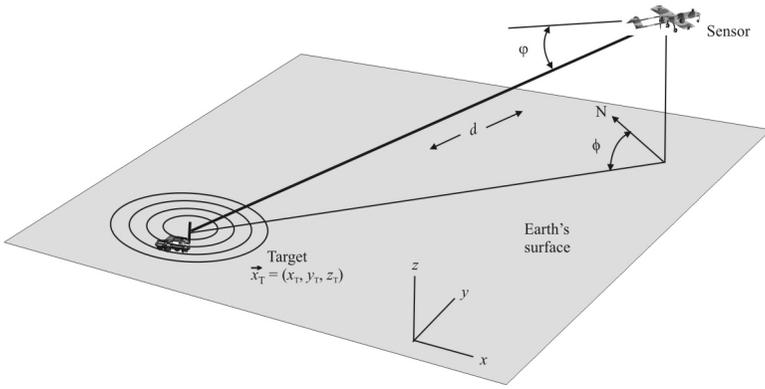


Figure 2.17 Target location geometry.

$$\begin{aligned}
 &= \sigma_{\theta}^2 - \frac{(\sigma_{\theta}^2)^2}{\sigma_{\theta}^2 + \sigma_n^2} \\
 &= \frac{1}{\frac{1}{\sigma_{\theta}^2} + \frac{1}{\sigma_n^2}}
 \end{aligned}$$

■

### 2.6.3 Target Location Estimation with the Linear Model

For the linear model, the observation vector is given by  $\vec{z}$ , which consists of  $N$  random observations. The parameters to be estimated are given by the  $K$  element vector  $\vec{\theta}$ , with means given by the mean vector  $\vec{\mu}_{\theta}$  and covariance matrix  $\mathbf{C}_{zz}$ . The observation matrix,  $\mathbf{H}$ , is known, deterministic, and is  $N \times K$ . The noise,  $\vec{n}$ , is an  $N$ -element vector with mean zero and covariance matrix  $\mathbf{C}_{nn}$ . It is assumed that  $\vec{\theta}$  and  $\vec{n}$  are uncorrelated.

The model is given by

$$\vec{z} = \mathbf{H}\vec{\theta} + \vec{n} \tag{2.144}$$

As seen in Figure 2.17, the observations consist of the azimuth of the target ( $\phi$ ), elevation of the target ( $\phi$ ), and range to the target ( $d$ ) corrupted by noise:

$$\vec{z} = [\phi \quad \varphi \quad d]^T + \vec{n} \quad (2.145)$$

The unknown is the target position in Cartesian coordinates:

$$\vec{\theta} = \vec{x}_T = [x_T \quad y_T \quad z_T]^T \quad (2.146)$$

The relationship between the observations and the target position is

$$\phi = \tan^{-1} \frac{y_T - y_i}{x_T - x_i} \quad (2.147)$$

$$\varphi = \tan^{-1} \frac{z_T - z_i}{\sqrt{(x_T - x_i)^2 + (y_T - y_i)^2}} \quad (2.148)$$

and

$$d = \sqrt{(x_T - x_i)^2 + (y_T - y_i)^2 + (z_T - z_i)^2} \quad (2.149)$$

where the position of the sensor at instant  $i$  is given by  $\vec{x}_S = (x_i, y_i, z_i)$ . These are put into matrix form as

$$\vec{z} = \mathbf{H}\vec{\theta} + \vec{n} \quad (2.150)$$

Note the squares, square roots, and  $\tan^{-1}$  functions in (2.147) to (2.149) are nonlinear functions and therefore do not fit the basic assumptions given above. However, (2.150) can be expanded in a Taylor series about some nominal value  $\vec{\theta}_0$  to get

$$\vec{z} \approx \mathbf{H}\vec{\theta}_0 + \vec{h}(\vec{\theta} - \vec{\theta}_0) + \vec{n} \quad (2.151)$$

where

$$\vec{h} = \left. \frac{\partial \mathbf{H}\vec{\theta}}{\partial \vec{\theta}} \right|_{\vec{\theta} = \vec{\theta}_0} \quad (2.152)$$

One possibility is to set  $\vec{\theta}_0 = \vec{\mu}_0$ . Make the following definitions:

$$\vec{z} = \vec{z} - \mathbf{H}(\vec{\mu}_0) \quad (2.153)$$

$$\vec{\theta} = \vec{\theta} - \vec{\mu}_0 \quad (2.154)$$

and

$$\vec{z} = \mathbf{H}\vec{\theta} + \vec{n} \quad (2.155)$$

This is now in the form of the linear model.

The LMMSE estimator for this linear model is found as follows.

$$\vec{\mu}_z = \mathcal{E}\{\vec{z}\} = \mathcal{E}\{\mathbf{H}\vec{\theta} + \vec{n}\} = \mathbf{H}\vec{\mu}_0 \quad (2.156)$$

$$\begin{aligned} \mathbf{C}_{zz} &= \mathcal{E}\{(\vec{z} - \vec{\mu}_z)(\vec{z} - \vec{\mu}_z)^T\} \\ &= \mathcal{E}\{(\mathbf{H}\vec{\theta} + \vec{n} - \mathbf{H}\vec{\mu}_0)(\mathbf{H}\vec{\theta} + \vec{n} - \mathbf{H}\vec{\mu}_0)^T\} \\ &= \mathbf{H}\mathbf{C}_{\theta\theta}\mathbf{H}^T + \mathbf{C}_{nn} \end{aligned} \quad (2.157)$$

$$\begin{aligned} &= \mathcal{E}\{\mathbf{H}(\vec{\theta} - \vec{\mu}_0)(\vec{\theta} - \vec{\mu}_0)^T\} + \mathcal{E}\{\mathbf{H}(\vec{\theta} - \vec{\mu}_0)\vec{n}^T\} \\ &\quad + \mathcal{E}\{\vec{n}(\vec{\theta} - \vec{\mu}_0)^T\mathbf{H}^T\} + \mathcal{E}\{\vec{n}\vec{n}^T\} \end{aligned} \quad (2.158)$$

$$\begin{aligned} \mathbf{C}_{\theta z} &= \mathcal{E}\{(\vec{\theta} - \vec{\mu}_0)(\vec{z} - \vec{\mu}_z)\} \\ &= \mathcal{E}\{\vec{\theta}\vec{\mu}_0^T(\mathbf{H}\vec{\theta} + \vec{n} - \mathbf{H}\vec{\mu}_0)^T\} \\ &= \mathcal{E}\{(\vec{\theta} - \vec{\mu}_0)(\vec{\theta} - \vec{\mu}_0)^T\mathbf{H}^T\} + \mathcal{E}\{(\vec{\theta} - \vec{\mu}_0)\vec{n}^T\} \\ &= \mathbf{C}_{\theta\theta}\mathbf{H}^T \end{aligned} \quad (2.159)$$

and

$$\begin{aligned} \hat{\vec{\theta}} &= \vec{\mu}_0 + \mathbf{C}_{\theta z}\mathbf{C}_{zz}^{-1}(\vec{z} - \vec{\mu}_z) \\ &= \vec{\mu}_0 + \mathbf{C}_{\theta\theta}\mathbf{H}^T(\mathbf{H}\mathbf{C}_{\theta\theta}\mathbf{H}^T + \mathbf{C}_{nn})^{-1}(\vec{z} - \mathbf{H}\vec{\mu}_0) \end{aligned} \quad (2.160)$$

The squared error matrix from (2.143) is

$$\begin{aligned}
 \mathcal{E} \left\{ \left( \hat{\vec{\theta}} - \vec{\theta} \right) \left( \hat{\vec{\theta}} - \vec{\theta} \right)^T \right\} &= \mathbf{C}_{\theta\theta} - \mathbf{C}_{\theta z} \mathbf{C}_{zz}^{-1} \mathbf{C}_{z\theta} \\
 &= \mathbf{C}_{\theta\theta} - \mathbf{C}_{\theta\theta} \mathbf{H}^T \left( \mathbf{H} \mathbf{C}_{\theta\theta} \mathbf{H}^T + \mathbf{C}_{nn} \right)^{-1} \mathbf{H} \mathbf{C}_{\theta\theta} \\
 &= \left( \mathbf{C}_{\theta\theta}^{-1} + \mathbf{H}^T \mathbf{C}_{nn}^{-1} \mathbf{H} \right)^{-1}
 \end{aligned} \tag{2.161}$$

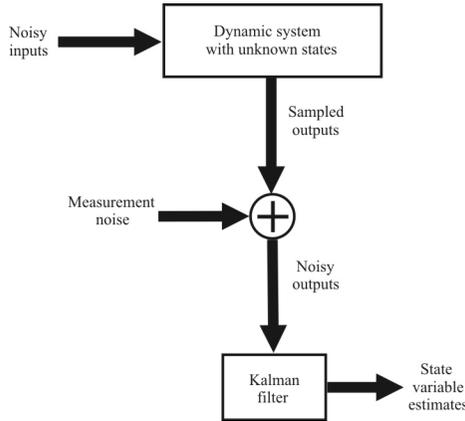
### 2.6.4 Kalman Filter Methods

In some cases, knowing only the PF of a target is inadequate. Examples of this include subsurface oceanographic targets that are tracked with active sonar and surface and airborne targets that are tracked by active radar. Thus, methods were devised to accommodate such tracking. Kalman filtering is one of those methods, although Kalman filters have much broader applications than tracking targets. Kalman filtering is based on the principles of MMSE.

In 1960, R. E. Kalman published a seminal paper describing a recursive solution to the discrete-data linear filtering problem [13]. That solution was based on modeling dynamic linear systems as discrete-data systems, and the resulting state variables are used to solve the Wiener filtering problem. Kalman filtering is a technique for optimally estimating the state of a dynamical system at time  $t_1$  given the value of the state variables at time  $t$ . If  $t_1 < t$ , then the estimate is an *interpolation* of the state variables. If  $t_1 = t$ , then the system is a *filter*. If  $t_1 > t$ , the problem is one of *prediction* of a future state given the current state. Of interest here is the filtering problem.

Kalman filter estimation uses a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of noisy measurements. The behavior of a dynamic system can be described by the time evolution of the state variables. As indicated previously, the state variables of a dynamic system cannot be determined exactly by direct measurements; instead, the measurements available are functions of the state variables corrupted by random noise. It is then necessary to estimate the state variables from the noisy observations. The measurement and estimation process is illustrated in Figure 2.18. The purpose of the Kalman filter is to optimally estimate the state variables in the dynamical system. *Optimum* in this sense means that the mean squared estimation error is minimized.

The Kalman filter can be applied to the PF estimation problem in a variety of ways, two of which are presented here. In the case of the standard Kalman filter, the results are optimum in the least mean-squares error sense. In the case of the



**Figure 2.18** Kalman filter application.

*extended Kalman filter* (EKF) described later, the results are not optimum. The EKF is an ad hoc estimator because of the approximations employed.

### 2.6.4.1 Standard Kalman Filter

Denote the state vector with  $\vec{s}_i \in \mathfrak{R}^N$  and the measurement vector with  $\vec{z}_i \in \mathfrak{R}^P$ . The Kalman filter assumes that the state vector follows the equation given by (2.105), and the measurements are given by (2.106).

The equations for the Kalman filter can be derived by a number of methods. Let  $\hat{\vec{s}}_{i-} \in \mathfrak{R}^N$  denote the a priori state estimate at instant  $i$  given information about the process prior to instant  $i$  (that is, the measurements at instant  $i$  have not been factored in to update the state at instant  $i$ ) and  $\hat{\vec{s}}_i \in \mathfrak{R}^N$  denote the a posteriori state estimate at instant  $i$ , given measurement  $\vec{z}_i$ . Two estimate errors can then be defined as

$$\vec{\epsilon}_{i-} \triangleq \vec{s}_i - \hat{\vec{s}}_{i-} \tag{2.162}$$

and

$$\vec{\epsilon}_i \triangleq \vec{s}_i - \hat{\vec{s}}_i \tag{2.163}$$

which are the a priori and a posteriori estimate errors, respectively. The associated (weighted) error covariance matrices are

$$\mathbf{C}_{i|-} = \mathcal{E}\{\vec{\epsilon}_{i|-} \mathbf{W} \vec{\epsilon}_{i|-}^T\} \quad (2.164)$$

and

$$\mathbf{C}_i = \mathcal{E}\{\vec{\epsilon}_i \mathbf{W} \vec{\epsilon}_i^T\} \quad (2.165)$$

where  $\mathbf{W}$  is any positive semi-definite weighting matrix, chosen to optimize some criterion.

The Kalman filter implements the derivation of an a posteriori estimate of the state as a linear combination of the a priori estimate and a weighted difference between the actual measurement  $\vec{z}_k$  and a measurement prediction of the state  $\mathbf{H}\hat{\mathbf{s}}_{i|-}$

$$\hat{\mathbf{s}}_i = \hat{\mathbf{s}}_{i|-} + \mathbf{K}(\vec{z}_i - \mathbf{H}\hat{\mathbf{s}}_{i|-}) \quad (2.166)$$

The last term in (2.166),  $(\vec{z}_i - \mathbf{H}\hat{\mathbf{s}}_{i|-})$ , is called the *innovation* of the process. Matrix  $\mathbf{K}$ , which is  $N \times P$ , is selected to minimize the a posteriori error covariance (2.165).  $\mathbf{K}$  is given by

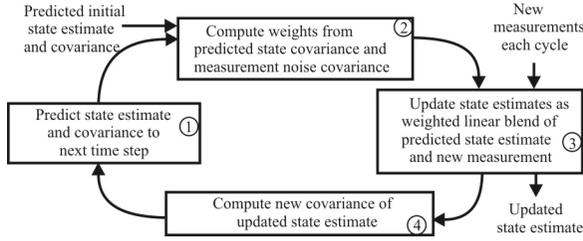
$$\mathbf{K}_i = \mathbf{C}_{i|-} \mathbf{H}^T (\mathbf{H} \mathbf{C}_{i|-} \mathbf{H}^T + \mathbf{C}_{\eta\eta})^{-1} \quad (2.167)$$

The Kalman filter is a variable gain filter, as indicated by (2.167). As the measurement noise covariance is reduced ( $\mathbf{C}_{\eta\eta} \rightarrow \mathbf{0}$ ),

$$\lim_{\mathbf{C}_{\eta\eta} \rightarrow \mathbf{0}} \mathbf{K}_i = \mathbf{C}_{i|-} \mathbf{H}^T (\mathbf{H}^T)^{-1} \mathbf{C}_{i|-}^{-1} \mathbf{H}^{-1} = \mathbf{H}^{-1} \quad (2.168)$$

and the gain is adjusted with weighting favoring the residual—that is, more credibility is given to the measurements. On the other hand, as the a priori estimate error covariance  $\mathbf{C}_{i|-}$  decreases,

$$\lim_{\mathbf{C}_{i|-} \rightarrow \mathbf{0}} \mathbf{K}_i = \mathbf{C}_{i|-} \mathbf{H}^T (\mathbf{H} \mathbf{C}_{i|-} \mathbf{H}^T + \mathbf{C}_{\eta\eta})^{-1} = \mathbf{0} \quad (2.169)$$



**Figure 2.19** The standard Kalman filter algorithm.

and the residual is weighted less heavily.

A flowchart for the *standard Kalman filter* is illustrated in Figure 2.19. It is defined by the evolution equations as follows (the numbers correspond to the numbers in the blocks in Figure 2.19):

- State prediction

$$(1) \quad \hat{s}_{i| -} = \Phi \hat{s}_{i-1} + \mathbf{B} \vec{u}_{k-1} \quad (2.170)$$

- Prediction of covariance matrix of states

$$(1) \quad \mathbf{C}_{i| -} = \Phi \mathbf{C}_{i-1} \Phi^T + \mathbf{C}_{\eta\eta} \quad (2.171)$$

- Kalman gain matrix computation

$$(2) \quad \mathbf{K}_i = \mathbf{C}_{i| -} \mathbf{H}^T (\mathbf{H} \mathbf{C}_{i| -} \mathbf{H}^T + \mathbf{C}_{\eta\eta})^{-1} \quad (2.172)$$

- Update state estimation

$$(3) \quad \hat{s}_i = \hat{s}_{i| -} + \mathbf{K}_i (\vec{z}_i - \mathbf{H}_i \hat{s}_{i| -}) \quad (2.173)$$

- Update covariance matrix of states

$$(4) \quad \mathbf{C}_i = (\mathbf{I} - \mathbf{K}_i \mathbf{H}) \mathbf{C}_{i| -} \quad (2.174)$$

- Initialization

$$\begin{aligned}\vec{s}_{0|-} &= \mathcal{E}\{\vec{s}_0\} \\ \mathbf{C}_{0|-} &= \mathbf{Cov}\{\vec{s}_0\}\end{aligned}\quad (2.175)$$

Thus, the equations for the Kalman filter fall into two groups: time update equations and measurement update equations. The time update equations, given by (2.170) and (2.171), are responsible for projecting forward in time the current state and error covariance estimates to obtain the estimates for the next a priori time step. The measurement update equations, given by (2.173) and (2.174), are responsible for the feedback, that is, for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate.

Noise processes characterized by  $\mathbf{C}_i$  and  $\mathbf{C}_{\eta_i}$  are measured or assumed to be of some value. They are often specified via an assumed value of SNR, and the SNR is varied to determine system performance trade-offs. As mentioned, frequently the noise processes are reasonably assumed to be uncorrelated so that

$$\mathbf{C}_i = \text{diag}[\sigma_{n_1}^2 \quad \sigma_{n_2}^2 \quad \cdots \quad \sigma_{n_N}^2] \quad (2.176)$$

and

$$\mathbf{C}_{\eta_i} = \text{diag}[\sigma_{\eta_1}^2 \quad \sigma_{\eta_2}^2 \quad \cdots \quad \sigma_{\eta_p}^2] \quad (2.177)$$

### Example

Suppose it is necessary to ascertain the diameter of a steel rod that is nominally 1 mm. Measurements of the diameter are sequentially made and processed in a Kalman filter. In this case, there is only one state variable: the diameter. Therefore  $\Phi = 1$ ,  $\mathbf{B} = 0$ , and

$$s_{i+1} = s_i \quad (2.178)$$

There is no process noise here, but suppose the manufacturer of the rod specifies that the diameter is 1 mm with a variance of 1% (0.01 mm<sup>2</sup>). Therefore, initially the best estimate of the diameter is  $s_{0|-} = 1$  mm with an uncertainty variance of  $\mathbf{C}_{0|-} = 0.01$ .

Repeated micrometer measurements

$$z_i = s_i + \eta_i$$

indicate the diameter to within the measurement noise  $\eta_i$ . Thus,  $\mathbf{H} = 1$ . It is assumed (reasonably) that the noise is uncorrelated from one measurement to the next. The micrometer manufacturer specifies that the micrometer is accurate to within an uncertainty variance of 0.1% (0.001 mm<sup>2</sup> in this case), and therefore  $C_0 = 0.001$ .

Starting at  $i = 0$ , with the initial estimate of 1 mm and an uncertainty of 0.01, the weight for updating with the first measurement is

$$K_0 = \frac{\Gamma_{0|-}}{\Gamma_{0|-} + C_0} = \frac{0.01}{0.01 + 0.001} = 0.91$$

with an updated state estimate as

$$s_{0|0} = (1 - K_0)s_{0|-} + K_0 z_0$$

where  $s_{0|0}$  denotes the best estimate at time 0, based on the measurement at time 0. The associated variance of the updated estimate is

$$C_0 = (1 - K_0)C_{0|-} = (0.09)(0.01) = 9 \times 10^{-4}$$

According to (2.178), the state projects identically to time 1, so the state at time 1 and variance projection is

$$s_{1|0} = s_{0|0} \quad C_{1|0} = C_{0|0} = 9 \times 10^{-4}$$

Repeating the cycle yields

$$K_1 = \frac{C_{1|0}}{C_{1|0} + C_1} = \frac{9 \times 10^{-4}}{9 \times 10^{-4} + 10^{-3}} = 0.47$$

and

$$C_{1|1} = (1 - K_1)C_{1|0} = (0.53)(9 \times 10^{-4}) = 4.8 \times 10^{-4}$$

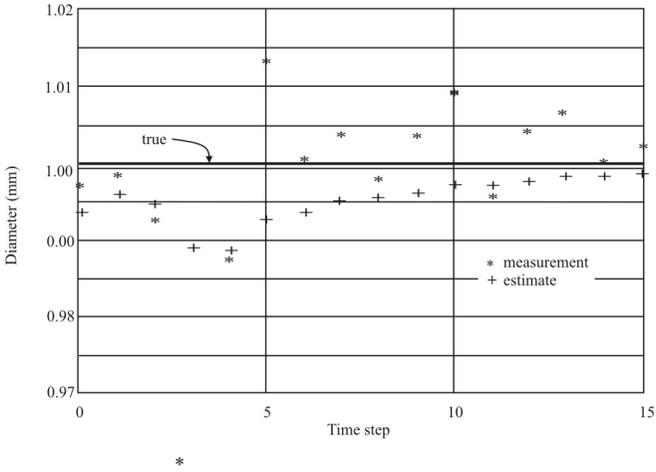


Figure 2.20 Diameter of the steel rod for the example.

Figure 2.20 illustrates a simulation of this process. The path of the covariance for this simulation is illustrated in Figure 2.21. The covariance gets to be quite small as the estimate approaches the true value of the diameter.

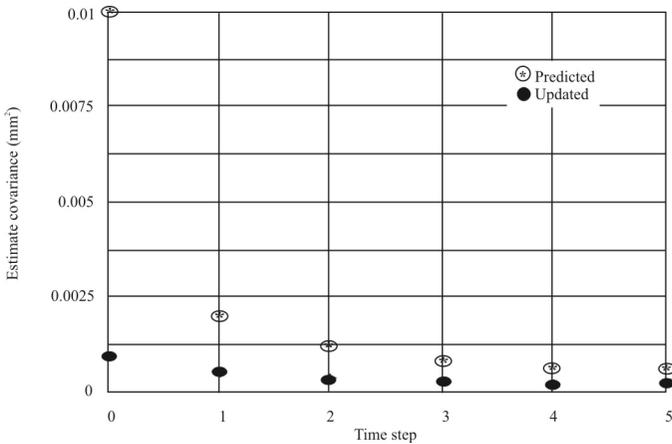
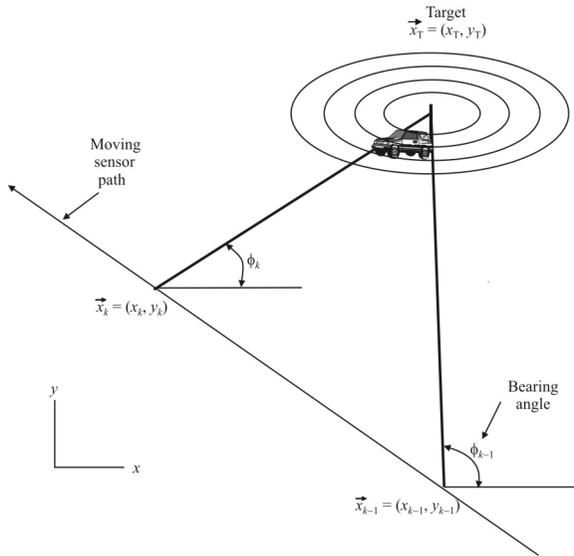


Figure 2.21 Variance of the rod diameter for the example.



**Figure 2.22** Geometry for EKF PF analysis.

This discussion assumed that the Kalman filter is *linear, time invariant (LTI)*. In general, nonlinear and time-variant versions of the Kalman filter are possible. In fact, it is these characteristics that make the Kalman filter as ubiquitous as it is. In particular, if  $\Phi_i \vec{s}_i$  is nonlinear or a linear relationship between  $\vec{x}_i$  and  $\vec{s}_i$  is not available, the *extended Kalman filter (EKF)* can be applied.

#### 2.6.4.2 Extended Kalman Filter

The EKF approach is to apply the standard Kalman filter for the linear systems just discussed to nonlinear systems with AWGN by linearizing the nonlinearity via the Taylor series expansion and ignoring the terms of second order and higher. In other words, a linear Taylor approximation of the system function at the previous state estimate and that of the observation function at the corresponding predicted position are considered. Convergence to a reasonable estimate may not be obtained if the initial guess is poor or if the noise is so large that the linearization is inadequate to describe the system. Also, as mentioned, the EKF is an ad hoc estimator due to the approximations incorporated (in particular, the linearization). It should be noted, however, that the EKF can have instability problems caused by the linearization approximation [14].

### 2.6.4.3 Position Fix Estimation with the Extended Kalman Filter

An analysis of using the EKF for the location of stationary emitters from a moving platform was conducted by Spingarn [15]. The two-dimensional geometry is illustrated in Figure 2.22. In this case,

$$\hat{\vec{z}}_k = \vec{f}(\hat{\vec{z}}_{k-1}, \vec{0}, \vec{0}) = \begin{bmatrix} \hat{x}_{T,k} \\ \hat{y}_{T,k} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{T,k-1} \\ \hat{y}_{T,k-1} \end{bmatrix} \quad (2.179)$$

The state transition matrix in this case is the identity matrix because the target is stationary. If the target were moving, then there would be dynamic terms involved. The state transition in this case is linear. The observables are the bearing angles  $\phi_i$  and the measurement model is given by

$$\phi_k = h(\hat{\vec{z}}_k) + \eta_k \quad (2.180)$$

where  $\eta_k$  represents the noise process, assumed here to be zero mean AWGN such that

$$\mathcal{E}\{\eta_j \eta_k\} = \sigma_\eta^2 \delta_{jk} \quad (2.181)$$

and

$$\mathbf{C}_{\eta\eta} = \sigma_\eta^2 \quad (2.182)$$

The state estimation equation is

$$\hat{\vec{z}}_k = \hat{\vec{z}}_{k|k-} + \mathbf{K} \left[ \phi_k - h(\hat{\vec{z}}_{k|k-}) \right] \quad (2.183)$$

where

$$h(\hat{\vec{z}}_{k|k-}) = \hat{\phi}_k = \tan^{-1} \left( \frac{\hat{y}_{T,k|k-} - y_k}{\hat{x}_{T,k|k-} - x_k} \right) \quad (2.184)$$

The update for the covariance matrix of states is

$$\mathbf{C}_k = \left[ \mathbf{I} - \mathbf{KJ} \left( \hat{\mathbf{z}}_{k|k-} \right) \right] \mathbf{C}_{k|k-} \quad (2.185)$$

where  $\mathbf{J}$  is the Jacobian matrix corresponding to  $h$ :

$$\begin{aligned} \mathbf{J} \left( \hat{\mathbf{z}}_{k|k-} \right) &= \left. \frac{\partial h(\bar{\mathbf{z}})}{\partial \bar{\mathbf{z}}} \right|_{\bar{\mathbf{z}} = \hat{\mathbf{z}}_{k|k-}} \\ &= [J_{11} \quad J_{12}] \end{aligned} \quad (2.186)$$

where

$$J_{11} = \left. \frac{\partial h}{\partial x_T} \right|_{x_T = \hat{x}_{T,k|k-}} = \frac{-u}{1+u^2} \frac{1}{\hat{x}_{T,k|k-} - x_k} \quad (2.187)$$

and

$$J_{12} = \left. \frac{\partial h}{\partial y_T} \right|_{y_T = \hat{y}_{T,k|k-}} = \frac{1}{1+u^2} \frac{1}{\hat{x}_{T,k|k-} - x_k} \quad (2.188)$$

with

$$u = \frac{\hat{y}_{T,k|k-} - y_k}{\hat{x}_{T,k|k-} - x_k} \quad (2.189)$$

The Kalman gain matrix is

$$\mathbf{K} = \mathbf{C}_{k|k-} \mathbf{J}^T \left( \hat{\mathbf{z}}_{k|k-} \right) \left[ \mathbf{J} \left( \hat{\mathbf{z}}_{k|k-} \right) \mathbf{C}_{k|k-} \mathbf{J}^T \left( \hat{\mathbf{z}}_{k|k-} \right) + \mathbf{C}_{\eta\eta,k} \right]^{-1} \quad (2.190)$$

where

$$\mathbf{C}_{\eta\eta,k} = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \quad (2.191)$$

and the error covariance is

$$\mathbf{C}_{k+1} = \mathbf{J}\mathbf{C}_k\mathbf{J}^T + \mathbf{Q} \quad (2.192)$$

where

$$\mathbf{Q} = \begin{bmatrix} q_{11} & 0 \\ 0 & q_{22} \end{bmatrix} \quad (2.193)$$

$\mathbf{Q}$  is a function of the noise. In this case, there is no noise in the model, so  $q_{11} = 0$  and  $q_{22}$  is set to a small positive value.

In this development, the state estimate equation is linear but the measurement equation is a  $\tan^{-1}$  function—highly nonlinear. Initial values of the state variables (location) are required, denoted by  $\hat{\mathbf{z}}_{e0}$ . Initial values for the error covariance are also required:

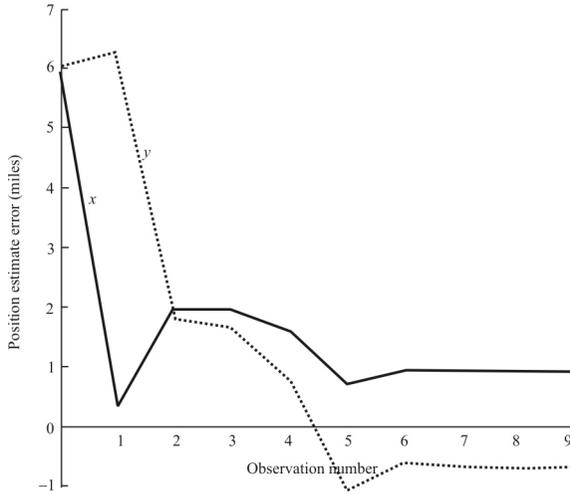
$$\mathbf{C}_{0|0} = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \quad (2.194)$$

### Example

Example results of a Monte Carlo simulation of the EKF model are shown in Figure 2.23 [15]. For this figure,  $\sigma_n^2 = (0.0175)^2 \text{ rad}^2 = (1^\circ)^2$ . The target was located at (141, 141) miles and the initial position of the aircraft was  $(x, y) = (130, 11)$  miles. The velocity of the sensor was  $v = 0.1$  miles/sec. Measurements were made every 300 seconds and the closest the sensor got to the target was 100 miles. The bearing error was caused by AWGN with  $\sigma = 1^\circ$ . The initial estimate of target location to initialize the iterations was  $\hat{x}_{T,0} = \hat{y}_{T,0} = 135$  miles,  $q_{11} = 0$  (because there is no process noise included),  $q_{22} = 0.0252 \text{ rad}^2$ , and  $\Gamma_{0|0} = \text{diag}(5^2) \text{ miles}^2$ . The  $x$ -error converges to about 1 mile, while  $y$ -error converges to about  $-0.7$  mile after nine observations. ■

#### 2.6.4.4 Total Least-Square Error with a Kalman Filter

If the bearing errors contain a bias  $\Phi$ , as expressed in (2.62), the TLSE solution is not necessarily the best statistical estimate. An alternate technique using TLSE and



**Figure 2.23** Results of Monte Carlo simulation of a passive position fix Kalman filter.

a Kalman filter is discussed in this section [12]. This Kalman filter approach deviates somewhat from the earlier development in Section 2.5.2.1, which serves to illustrate the considerable flexibility available with the Kalman filter.

The bearing errors have a constant part given by  $\Phi$  and a random part given by  $\Delta\phi_i^*$  :

$$\phi_i = \Phi + \Delta\phi_i^* \tag{2.195}$$

First the TLSE estimate is found, and that estimate is used in (2.58). The state equations of the filter are given by

$$\delta\phi_{i+1} = \delta\phi_i \tag{2.196}$$

$$\Phi_{i+1} = \Phi_i \tag{2.197}$$

while the observation equation is

$$\tilde{\phi}_i - \hat{\phi}_i = \Delta\hat{\phi}_i = \delta\phi_i + \Phi_i + \Delta\phi_i^* \tag{2.198}$$

where  $\delta\phi_i$  is the bearing error at interval  $i$  due to the error in the estimation of  $(\psi_T, \lambda_T)$ .

The Kalman filter algorithm is found as follows. Suppose that  $N$  observations are available.

$$\begin{aligned} &\text{For } i = 1, 2, \dots, N \\ &\begin{bmatrix} \hat{\phi}_i \\ \delta\hat{\phi}_i \end{bmatrix} = \begin{bmatrix} \hat{\phi}_{i-1} \\ \delta\hat{\phi}_{i-1} \end{bmatrix} + \mathbf{K}_i \left( \Delta\hat{\phi}_i - \vec{\mathbf{1}}_2^T \begin{bmatrix} \hat{\phi}_{i-1} \\ \delta\hat{\phi}_{i-1} \end{bmatrix} \right) \\ &\mathbf{K}_i = \mathbf{C}_{i-1} \frac{\vec{\mathbf{1}}_2}{\mathbf{C}_{\phi,i} + \vec{\mathbf{1}}_2^T \mathbf{C}_{\psi\lambda,i-1} \vec{\mathbf{1}}} \\ &\mathbf{C}_i = \mathbf{C}_{i-1} - \mathbf{C}_{i-1} \vec{\mathbf{1}}_2 \vec{\mathbf{1}}_2^T \frac{\mathbf{C}_{\psi\lambda,i-1}}{\mathbf{C}_{\phi,i} + \vec{\mathbf{1}}_2^T \mathbf{C}_{\psi\lambda,i-1} \vec{\mathbf{1}}} \end{aligned}$$

$\mathbf{K}_i$  is the Kalman gain vector

$\mathbf{C}_\phi$  is the covariance of  $\Delta\hat{\phi}_i^*$

$\mathbf{C}_{\psi\lambda,i}$  is the covariance matrix

$$\vec{\mathbf{1}}_2 = [1 \quad 1]^T$$

The set of observations  $\tilde{\phi}_i = \tilde{\phi}_i - \hat{\phi}_i$  for all  $i$  are then used with the TLS algorithm above to obtain a new estimate  $(\tilde{\psi}_T, \tilde{\lambda}_T)$ . These values are then used in (2.95) to generate a new estimate of  $\theta_i$ . An updated estimate  $\tilde{\Phi}_n$  is then obtained with the Kalman filter. This procedure is repeated until the bias disappears.

### Example

In addition to conducting a simulation experiment for TLSE PF computations, Rao and Reddy [12] also included a simulation experiment using TLSE with the Kalman filter. The geometry was as indicated in Figure 2.13, and 122 bearings were taken on the baseline length of 242 km. The results are indicated in Table

**Table 2.2** Results of TLS Estimation with a Kalman Filter

Case	Random Error	Systematic Error	Error (km)
1	$\sim\mathcal{N}(0, 1)$	$0^\circ$	0.524
2	$\sim\mathcal{N}(0, 1)$	$-3^\circ$	0.524
3	$\sim\mathcal{N}(0, 1)$	$6^\circ$	0.524

Source: [1].

2.2. This indicates that TLSE alone is not necessarily optimal when there is a systematic bias in the bearings. Using TLS with the Kalman filter produced significantly better results in that case.

#### 2.6.4.5 Unscented Kalman Filter

The EKF linearizes the system nonlinearity by expanding it in a Taylor series and retaining only the linear terms. When the nonlinearity is highly nonlinear, retaining only the linear terms can lead to significant modeling errors and divergent behavior. An extension to the EKF was proposed by Julier and Uhlman [16] called the *unscented Kalman filter* (UKF). This method improves on the performance of the EKF by, inter alia, retaining up through the third-order nonlinear terms in the Taylor series. This, coupled with retaining only specially selected sample points to propagate from one state to the next, can lead to much better models for nonlinear systems.

## 2.7 The Discrete Probability Density Method

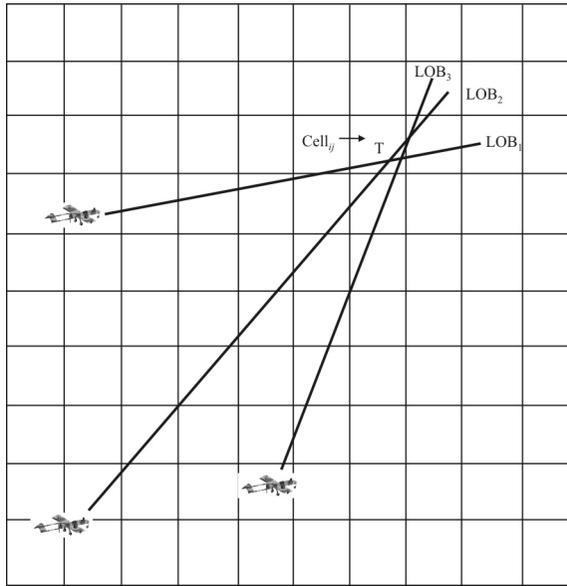
All of the position fixing algorithms discussed here are subject to erroneous results due to errors in the data used to compute the fixes. The effects of errors vary. Presented in this section is a technique that mitigates some of the erroneous results when the input data for the fix computations are LOBs [17].

The AOI is segmented into small squares or rectangles with a grid such as the one shown in Figure 2.24. This grid can be as granular as desired/required; however, the smaller the segmentation, the higher the computational burden and the higher the resolution (but not necessarily the accuracy). A  $100 \times 100$  grid requires 100 times the amount of computations of a  $10 \times 10$  grid, for example.

The position-fixing algorithm computes the probability that an emitter lies within each of the grid segments based on the LOBs provided. The LOBs are assumed to have a von Mises PDF (also called the Tikhonov PDF [18]) that is defined over the angular interval  $(-\pi, \pi)$ . The Tikhonov PDF is given by [17]

$$p(\phi) = \begin{cases} \frac{\exp[\nu \cos(\phi - \mu)]}{2\pi I_0(\nu)}, & -\pi < \phi \leq \pi \\ \text{undefined,} & \text{elsewhere} \end{cases} \quad (2.199)$$

where  $\nu$  is the *concentration* and indicates the dispersion of  $\phi$  around  $\mu$  and is similar to the inverse of the variance of the Gaussian density.  $\mu$  is the mean of the



**Figure 2.24** Discrete probability density grid with three LOBs.

distribution and, for the purposes here, is the measured LOB, which ranges over  $(-\pi, \pi)$  radians. This function is plotted in Figure 2.25 for some values of  $\nu$ . As  $\nu$  gets larger, the Tikhonov PDF gets larger at the mean value (zero in this case) and narrower, and the variance is smaller.

The distribution function corresponding to (2.199) is given by

$$P(\phi) = \frac{1}{2\pi I_0(\nu)} \left\{ \phi I_0(\nu) + 2 \sum_{i=1}^{\infty} \frac{I_i(\nu) \sin[i(\phi - \mu)]}{i} \right\} \quad (2.200)$$

which is plotted in Figure 2.26 for the same values. As expected and predicted from Figure 2.25, the larger  $\nu$  is, the more rapid is the transition from low to high probability around the mean value.

The PDF expands as the distance from the sensor increases. Close to the sensor, the PDF is narrow, with probability values close to the mean as illustrated in Figure 2.27. Farther from the sensor, the PDF expands, encompassing more grid squares.

The errors in the LOBs from different platforms are independent random variables, so the overall PDF obtained by combining  $N$  individual distributions is given by

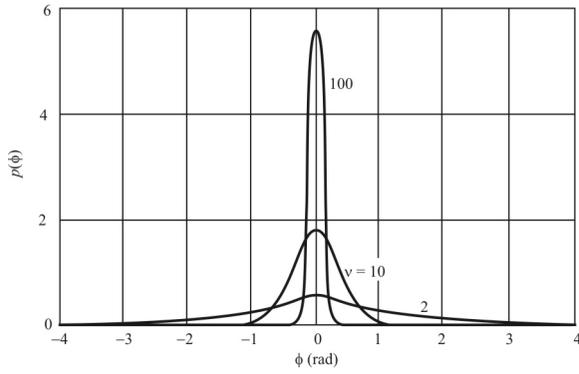


Figure 2.25 von Mises PDF.

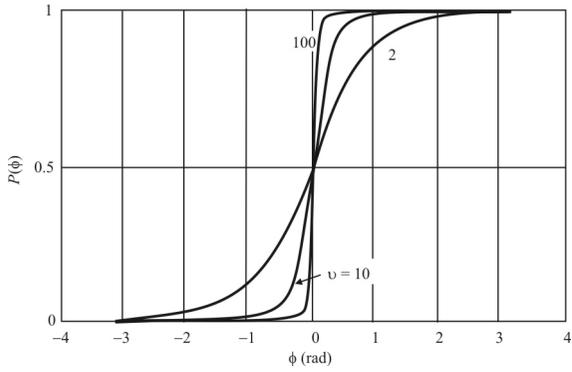
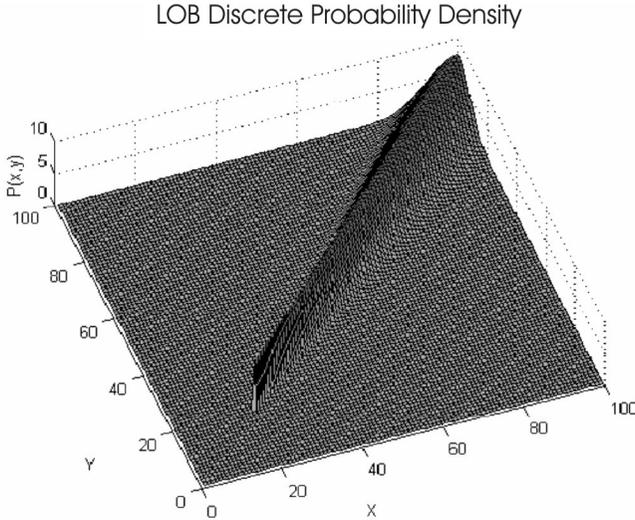


Figure 2.26 von Mises distribution function.



**Figure 2.27** DPD LOB distribution. (Source: [16]. © 2003 British Crown. Reprinted with permission.)

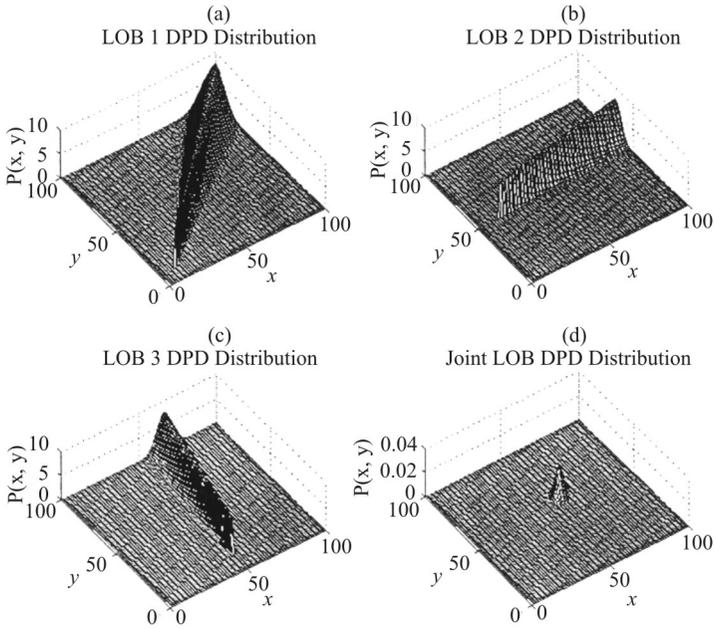
$$F(x, y) = k \prod_{i=1}^N \frac{\exp[v_i \cos(\phi_i - \mu_i)]}{2\pi I_0(v_i)} \quad (2.201)$$

The constant  $k$  is present to normalize the total volume under the total grid surface and is present to avoid numerical stability problems. It is calculated after all the probability terms have been determined. After normalization, the probability value for each cell is given by  $F(x, y)$ .

Based on this PDF, the probability that the target lies within a particular segment is computed by first computing the probabilities at each of the corners of the grid segment. The overall probability of being within the segment is then the average of these probabilities, to wit,

$$\text{Pr}(\text{within cell}_{i,j}) = \frac{1}{4} [F(x_i, y_j) + F(x_{i+1}, y_j) + F(x_i, y_{j+1}) + F(x_{i+1}, y_{j+1})] \quad (2.202)$$

A wild bearing tends to have less effect on this method of fix computation. This is because the overall fix PDF is the product of the  $N$  individual PDFs. Even



**Figure 2.28** (a–d) Example of the DPD PF algorithm with three PF systems. (Source: [16]. © 2003 British Crown. Reprinted with permission.)

if the wild bearing has a small associated variance, its contribution to the overall PDF will be small because the wild PDF is largest where the tails of the good PDFs are located. Assuming the remainder of the PDFs are of good quality, their product will be larger than the product of the wild PDF main lobe and the good PDF tails, even though the former PDF is multiplied by the wild PDF tail.

### Example

Figure 2.28 illustrates the technique for three PF sensors located according to the data in Table 2.3. The product of the PDFs is significant only over a small segment of the grid, as shown in Figure 2.28(d). In this case, the *root mean-square* (RMS) error is  $3^\circ$ , corresponding to

**Table 2.3** Bearings for the DPD Example

PF System	x-Coordinate	y-Coordinate	Bearing from y-Axis (degrees)
1	10	10	45
2	10	50	80
3	50	10	10

$$v = \frac{1}{\left(3 \frac{\pi}{180}\right)^2} = 364.8$$

The volume under the curve in Figure 2.28(d) between  $55 < x < 65$  and  $55 < y < 65$  is approximately 0.92 and therefore approximately equal to the 90% error ellipse. This ellipse has a major axis of 14 cells and a minor axis of 8 cells, with a major axis orientation of  $45^\circ$ .

## 2.8 Generalized Bearings

A different approach to iterating toward a PF solution was presented by Paradowski [19]. Given the  $i$ th azimuth AOA  $\phi_i$  and the  $i$ th elevation AOA  $\varphi_i$ , the *generalized bearing*  $\alpha_i$  is determined as indicated in Figure 2.29. The generalized bearing is in the plane defined by triangle  $TS_iA$  and is given by

$$f_i(\vec{x}_i) = \alpha_i = \tan^{-1} \frac{\sqrt{(x_T - x_i)^2 + (z_T - z_i)^2}}{y_T - y_i} \quad (2.203)$$

The measurements are expressed as

$$u_i = f_i(\vec{x}_i) + n_i, \quad i = 1, 2, \dots, N \quad (2.204)$$

or

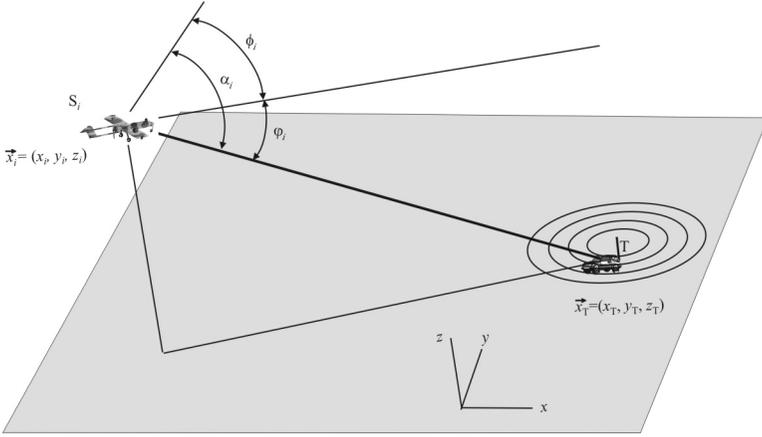


Figure 2.29 Generalized bearing, denoted by  $\alpha_i$ .

$$\vec{u} = \vec{f}(\vec{x}) + \vec{n} \quad (2.205)$$

where, in the absence of measurement errors  $u_i = f_i(\vec{x}_i) = \alpha_i$ , and  $f_i(\vec{x}_i)$  is the function in which the PF and LOB information is contained. The measurement noise,  $n_i$ , is characterized by

$$\mathcal{E}\{n_i\} = 0 \quad \mathcal{E}\{n_i n_j\} = \sigma_{ij} \quad (2.206)$$

$$\sigma_{ij} = \begin{cases} \sigma_{ij}^2 & i = j \\ \rho_{ij} \sigma_i \sigma_j & i \neq j \end{cases} \quad (2.207)$$

$\sigma_i$  and  $\sigma_j$  are standard deviations for the  $i$ th and  $j$ th measurements and  $\rho_{ij}$  is the correlation coefficient between these measurements.

The gradient vector at the  $m$ th step in the iteration toward the solution is given by

$$\vec{g}_{m_i} = \left[ \frac{\hat{x}_m - x_i}{d_{m_i}^2 s_{m_i}} \quad -\frac{\hat{y}_m - y_i}{d_{m_i}} s_{m_i} \quad \frac{\hat{z}_m - z_i}{d_{m_i}^2 s_{m_i}} \right]^T \quad (2.208)$$

where

$$d_{m_i}^2 = (\hat{x}_m - x_i)^2 + (\hat{y}_m - y_i)^2 + (\hat{z}_m - z_i)^2 \quad (2.209)$$

and

$$s_{m_i} = \sqrt{\left(\frac{d_{m_i}}{\hat{y}_m - y_i}\right)^2 - 1} \quad (2.210)$$

The gradient matrix consists of rows that are the gradient vectors

$$\mathbf{G}_m = [\vec{g}_{m1} \quad \vec{g}_{m2} \quad \cdots \quad \vec{g}_{mM}]^T \quad (2.211)$$

At the  $m$ th iteration, the covariance matrix of generalized bearing errors is a diagonal matrix expressed as

$$\mathbf{C}_{\alpha\alpha m} = \text{diag}[\sigma_{\alpha m_1}^2, \sigma_{\alpha m_2}^2, \dots, \sigma_{\alpha m_m}^2] \quad (2.212)$$

Where  $\sigma_{\alpha m_i}^2$  is the variance of the  $\alpha_i$  error at the  $m$ th iteration. These can be expressed in terms of the variances of the measurements as

$$\sigma_{\alpha m_i}^2 = a_{m_i} \sigma_{\phi_i}^2 + b_{m_i} \sigma_{\varphi_i}^2 + c_{m_i} \rho_i \sigma_{\phi_i} \sigma_{\varphi_i} \quad (2.213)$$

where

$$a_{m_i} = \left( \frac{1}{s_{m_i}} \frac{\hat{x}_m - x_i}{\hat{y}_m - y_i} \right)^2$$

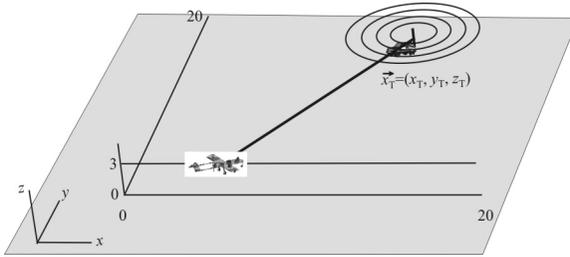
$$b_{m_i} = \left\{ s_{m_i} \left[ \left( \frac{d_{m_i}}{\hat{z}_m - z_i} \right) - 1 \right]^{1/2} \right\}^{-2}$$

$$c_{m_i} = \sqrt{a_{m_i} b_{m_i}}$$

Given that the noise is AWGN, then the iteration equation is given by

$$\hat{\mathbf{x}}_{m+1} = (\mathbf{G}_m^T \mathbf{C}_{\alpha\alpha m}^{-1} \mathbf{G}_m)^{-1} \mathbf{G}_m^T \mathbf{C}_{\alpha\alpha m}^{-1} \vec{w}_m \quad (2.214)$$

where



**Figure 2.30** Example scenario for the generalized bearing algorithm.

$$\vec{w}_m = \vec{u} + \mathbf{G}_m \hat{x}_m - \vec{f}(\hat{x}_m) \tag{2.215}$$

and where  $\vec{f}(\hat{x}_m)$  is  $\vec{f}(\vec{x})$  evaluated at  $\vec{x} = \hat{x}_m$ .

Of course, being an iterative procedure, and an approximate procedure, a good initial guess is required or the solution may settle to a local minimum. Also, convergence cannot be guaranteed.

### Example

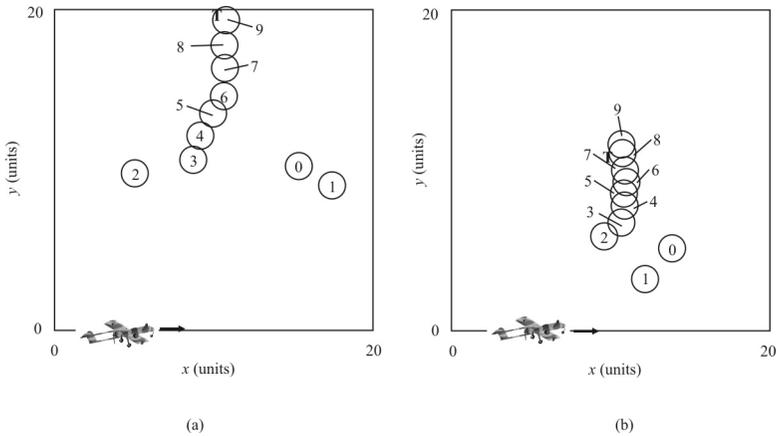
Consider the geometry shown in Figure 2.30. The unattended aerial system (UAS) sensor is flying along the  $x$ -axis at an altitude of 3,000 m and the targets are somewhere in the  $20 \times 20$  km region. The  $y$ -coordinate of the sensor is therefore unchanging and set to zero. The sensor makes an LOB measurement every 1 km along its straight route.

The results of this algorithm for two target locations are shown in Figure 2.31. The results appear to be converging to the correct location, although there were many cases that did not converge.

The parameters used in the example cause the convergence to be particularly sensitive to  $\hat{z}_0$ . This is likely due to the low altitude at which the sensor was flown compared to the distance to the targets; the slant angle was fairly low.

■

A MathCAD<sup>®</sup> program listing is contained in Appendix 2B that implements one pass through the iterative procedure for the generalized bearing algorithm described.

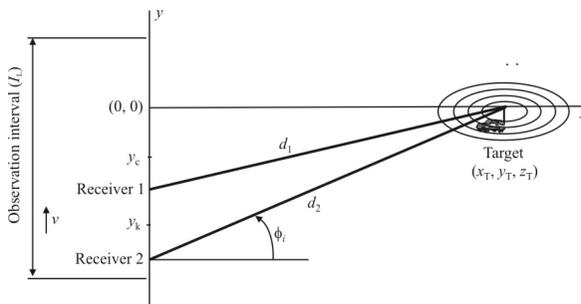


**Figure 2.31** Results of the generalized bearing algorithm example. The target in (a) is located at  $\vec{x}_T = (10, 20, 0.5)$ , while the target in (b) is at  $(10, 10, 0.5)$ . Variations in  $\hat{z}$  are not shown.

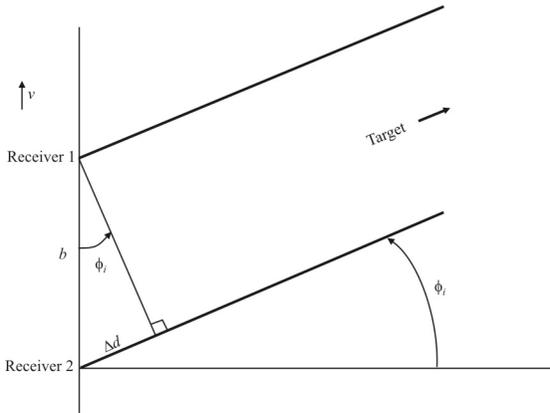
### 2.9 Maximum Likelihood PF Algorithm

The analysis of interferometric LOB algorithms using a maximum likelihood approach is presented in this section. The development follows that in [20].

Consider the geometry shown in Figure 2.32 [20], which contains an airborne sensor flying in the  $+y$  direction with velocity  $v$ . It contains two receivers as noted. Distance  $y_k$  is at the center of the two receivers at instant  $k$ . Measurements are taken over the observation interval, which has a center located at  $y_c$ . The target is located at coordinates  $(x_T, y_T)$  (the  $z$ -dimension is ignored for now). Initially it is



**Figure 2.32** Geometry for the maximum likelihood algorithm.



**Figure 2.33** Details of the geometry shown in Figure 2.17.

assumed that  $y_T = 0$ .  $N = 2P + 1$  measurements are collected over interval  $I_L$ .

The sensor measures the phase difference of the signals received by the two receivers. This difference is related to the ranges  $d_1$  and  $d_2$  by

$$\Delta\phi_i = 2\pi \frac{\Delta d}{\lambda} \tag{2.216}$$

When the baseline between the two receivers is small compared with the range to the target and the bearing angle is small, the two rays that delineate  $d_1$  and  $d_2$  in Figure 2.32 are approximately parallel, as shown in Figure 2.33 [20]. Then

$$\sin \phi = \frac{\Delta d}{b}$$

but for small  $\phi$ ,

$$\sin \phi \approx \phi$$

so

$$\phi \approx \frac{\Delta d}{b}$$

so the error in the range difference, as indicated by the standard deviation  $\sigma_{\Delta d}$ , is related to the error in the angle measurement as indicated by the standard deviation  $\sigma_\phi$  as

$$\sigma_\phi \approx \frac{\sigma_{\Delta d}}{b}, \quad \phi \ll 1 \text{ rad} \quad (2.217)$$

The noise is assumed to be Gaussian with zero mean and variance  $\sigma_n^2$ . The estimator used in this PF system is a least-squared estimator that implements an iterative approach.

### 2.9.1 Maximum Likelihood Estimation Triangulation Algorithm

The unknowns in the geometry shown in Figure 2.32 are  $y_c$ , the center of the observation interval, and  $x_T$ , the location of the target (since it is assumed for now that  $y_T = 0$ ). Therefore, the vector of unknowns is

$$\vec{\theta} = [x_T \quad y_c]^T \quad (2.218)$$

For the interferometer, from Figure 2.32, the noiseless measurements are

$$\tilde{h}_n(x_T, y_c) = \phi_n = \tan^{-1} \left( -\frac{y_n}{x_T} \right) \quad (2.219)$$

where

$$y_n = y_c + k\Delta L$$

The  $N = 2P + 1$  noisy measurements, denoted by  $z_n$ , where each receiver makes  $P$  measurements, corrupted by samples of the Gaussian noise process denoted by  $n_n$ , are given by

$$z_n(\vec{\theta}) = \tilde{z}_n(\vec{\theta}) + n_n, \quad n = 0, \pm 1, \pm 2, \dots, \pm N \quad (2.220)$$

Let

$$\partial_{x_n} = \frac{\partial \tilde{z}_n(\vec{\theta})}{\partial x_T} \quad (2.221)$$

$$\partial_{y_n} = \frac{\partial \tilde{z}_n(\vec{\theta})}{\partial y_c} \quad (2.222)$$

define the matrix of partial derivatives as

$$\mathbf{G} = \begin{bmatrix} \partial_{-xN} & \cdots & \partial_{x_n} & \cdots & \partial_{xN} \\ \partial_{-yN} & \cdots & \partial_{y_n} & \cdots & \partial_{yN} \end{bmatrix}^T \quad (2.223)$$

and

$$\vec{z} = [z_{-N} \quad \cdots \quad z_n \quad \cdots \quad z_N]^T \quad (2.224)$$

$$\tilde{\vec{z}} = [\tilde{z}_{-N} \quad \cdots \quad \tilde{z}_n \quad \cdots \quad \tilde{z}_N]^T \quad (2.225)$$

$$\vec{n} = [n_{-N} \quad \cdots \quad n_n \quad \cdots \quad n_N]^T \quad (2.226)$$

$$\vec{w} = [\Delta x_T \quad \Delta y_c]^T \quad (2.227)$$

If the noise is i.i.d. with covariance matrix

$$\mathbf{C}_{nn} = \sigma_n^2 \mathbf{I} \quad (2.228)$$

then the correction vector for the next iteration is given by

$$\vec{w} = (\mathbf{G}^T \mathbf{G})^{-1} (\vec{z} - \tilde{\vec{z}}) \quad (2.229)$$

and the estimate for the next iteration is

$$\vec{\theta}_{i+1} = \vec{\theta}_i + \vec{w} \quad (2.230)$$

The iterating ceases when  $\vec{w}$  vanishes.

The final covariance matrix for the variables gives the PF error. It is given by

$$\mathbf{C}_{zz} = \sigma_n^2 (\mathbf{G}^T \mathbf{G})^{-1} \quad (2.231)$$

From (2.219), (2.221), and (2.222),

$$\partial_{x_n} = \frac{y_c + n\Delta L}{x_T^2 + (y_c + n\Delta L)^2} \quad (2.232)$$

and

$$\partial_{yn} = \frac{-x_T}{x_T^2 + (y_c + n\Delta L)^2} \quad (2.233)$$

so that

$$\mathbf{G}^T \mathbf{G} = \begin{bmatrix} \sum_{n=-N}^N (\partial_{xn})^2 & \sum_{n=-N}^N \partial_{xk} \partial_{yk} \\ \sum_{n=-N}^N \partial_{xn} \partial_{yn} & \sum_{n=-N}^N (\partial_{yn})^2 \end{bmatrix} \quad (2.234)$$

When  $y_c = 0$ , the off-diagonal terms in (2.234) are zero, so

$$V^{-1} x_T = \frac{1}{\sigma_\phi^2} \sum_{n=-N}^N \frac{(n\Delta L)^2}{[x_T^2 + (n\Delta L)^2]^2} \quad (2.235)$$

and

$$V^{-1} y_c = \frac{1}{\sigma_\phi^2} \sum_{n=-N}^N \frac{x_T^2}{[x_T^2 + (n\Delta L)^2]^2} \quad (2.236)$$

Again, assuming that the observations are dense, so that  $\Delta L \approx dL$ , multiplying by  $dL$  inside the sum, dividing by  $\Delta L$  outside the sum, and replacing the sum with an integral with

$$k\Delta L = y, N\Delta L = \frac{L}{2} \quad (2.237)$$

then

$$V^{-1} x_T = \frac{1}{\sigma_\phi^2 \Delta L} \int_{-L/2}^{L/2} \frac{y^2}{[x_T^2 + y^2]^2} dy \quad (2.238)$$

and

$$V^{-1} y_c = \frac{1}{\sigma_\phi^2 \Delta L} \int_{-L/2}^{L/2} \frac{x_T^2}{[x_T^2 + y^2]^2} dy \quad (2.239)$$

Assuming  $x_T \gg L$  and the fact that

$$N = 2P + 1 \approx 2P = \frac{I_L}{\Delta L} \quad (2.240)$$

then (2.238) and (2.239) become

$$V^{-1} x_T \approx \frac{N I_L^2}{12 \sigma_\phi^2 x_T^4} \quad (2.241)$$

and

$$V^{-1} y_c \approx \frac{N}{\sigma_\phi^2 x_T^2} \quad (2.242)$$

so that

$$\sigma_x \approx \sqrt{\frac{12}{N}} \frac{\sigma_\phi x_T^2}{I_L} \quad (2.243)$$

and

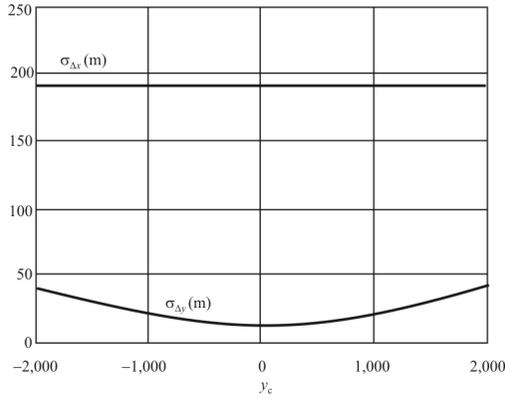
$$\sigma_y \approx \frac{\sigma_\phi x_T}{\sqrt{N}} \quad (2.244)$$

When  $y_c \neq 0$ , (2.243) remains the same while (2.244) changes to

$$\sigma_y \approx \sqrt{\frac{1}{N}} \sigma_\phi x_T \sqrt{1 + 12 \frac{y_c^2}{I_L^2}} \quad (2.245)$$

### Example

The performance of the LOB algorithm is illustrated in Figure 2.34 [20], where  $x_0 = 10$  km,  $I_L = 1.6$  km,  $N = 40$ ,



**Figure 2.34** Interferometer results.

$\sigma_{\Delta d} = 0.0141\text{m}$ , and the initial guess of  $b = 50$  m. The down-range standard deviation remained constant as dictated by (2.243), while the cross-range standard deviation varied as  $y_c$  was varied, as expected from (2.245).

## 2.9.2 Maximum Likelihood Estimation Algorithm Comparison

Gavish and Weiss compared the performance of two MLE PF algorithms [21]. Assuming the noise is Gaussian with zero mean, the MLE of the location of the target is given by

$$\hat{\vec{x}} = \underset{\vec{x}}{\operatorname{argmin}} F(\vec{x}, \vec{\phi}) \quad (2.246)$$

where

$$F(\vec{x}, \vec{\phi}) = \frac{1}{2} [\vec{g}(\vec{x}) - \vec{\phi}]^T \mathbf{C}_{xx}^{-1} [\vec{g}(\vec{x}) - \vec{\phi}] \quad (2.247)$$

is a cost function. The components of the cost function are

$$\vec{g}(\vec{x}) = [g_1(\vec{x}) \quad \cdots \quad g_N(\vec{x})]^T \quad (2.248)$$

In two dimensions,

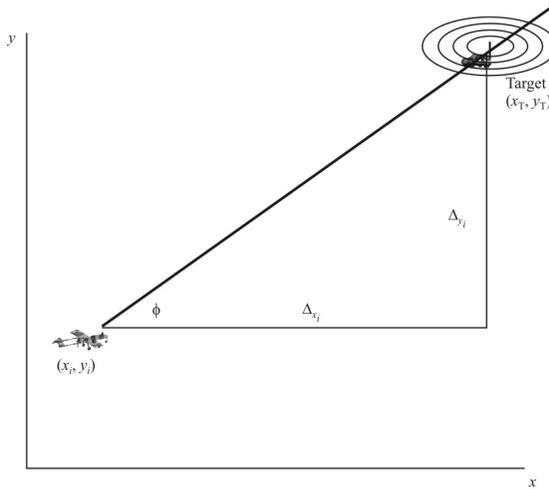


Figure 2.35 MLE LOBs.

$$\vec{x} = [x_T \quad y_T]^T \tag{2.249}$$

and the bearing measurements are given by (see Figure 2.35)

$$\vec{\phi} = [\phi_1 \quad \phi_2 \quad \dots \quad \phi_N]^T \tag{2.250}$$

The components of  $\vec{g}(\vec{x})$  are given by

$$g_n(\vec{x}) = \tan^{-1} \left( \frac{\Delta y_n}{\Delta x_n} \right) \tag{2.251}$$

where

$$\begin{aligned} \Delta x_n &= x_T - x_n \\ \Delta y_n &= y_T - y_n \end{aligned} \tag{2.252}$$

Finally,  $C_{xx} = \text{diag}(\sigma_1^2, \sigma_2^2, \dots, \sigma_N^2)$  is the  $N \times N$  covariance matrix of the LOB observations.

$N$  is the number of observations and there can be multiple observations per sensor. However, the number of observations from all sensors is assumed to be the same.

The LOB observations are corrupted with AWGN given by

$$\vec{\delta\phi} = [\delta\phi_1 \quad \delta\phi_2 \quad \dots \quad \delta\phi_N]^T \quad (2.253)$$

When the true LOBs are given by  $\vec{\phi}_0$ , the observed LOBs are given by

$$\vec{\phi} = \vec{\phi}_0 + \vec{\delta\phi} \quad (2.254)$$

Expression (2.247) can be put in the form

$$F(\vec{x}, \vec{\phi}) = \frac{1}{2} \vec{f}^T \mathbf{C}_{xx}^{-1} \vec{f} = \frac{1}{2} \sum_{n=1}^N \frac{f_n^2}{\sigma_n^2} \quad (2.255)$$

where

$$\vec{f} = [f_1 \quad f_2 \quad \dots \quad f_N]^T = \vec{g}(\vec{x}) - \vec{\phi} \quad (2.256)$$

The solution of (2.246), which is a nonlinear equation, can be found numerically by the Newton-Gauss method, which iterates to a solution with

$$\hat{\vec{x}}_{m+1} = \hat{\vec{x}}_m + [\vec{g}_x^T \mathbf{C}_{xx}^{-1} \vec{g}_x]^T \vec{g}_x^T \mathbf{C}_{xx}^{-1} [\vec{\phi} - \vec{g}(\hat{\vec{x}}_m)] \quad (2.257)$$

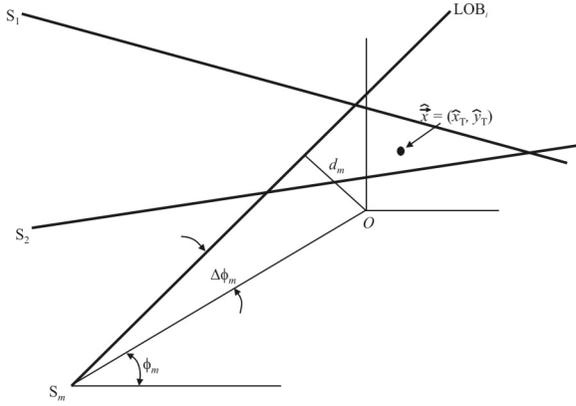
where  $\vec{g}_x = \partial \vec{g} / \partial \vec{x}$  evaluated at the true target position, yielding

$$\vec{g}_x = \begin{bmatrix} -\frac{\Delta_{y_1}}{d_1^2} & -\frac{\Delta_{y_2}}{d_2^2} & \dots & -\frac{\Delta_{y_N}}{d_N^2} \\ \frac{\Delta_{x_1}}{d_1^2} & \frac{\Delta_{x_2}}{d_2^2} & \dots & \frac{\Delta_{x_N}}{d_N^2} \end{bmatrix} \quad (2.258)$$

with

$$d_n^2 = \Delta_{x_n}^2 + \Delta_{y_n}^2 \quad (2.259)$$

■



**Figure 2.36** Stansfield's PF geometry.

The two MLE techniques analyzed by [21] are the original Stansfield algorithm [1] and the true MLE algorithm, as given by (2.255) to (2.257). The analysis presented in this section follows [22].

One of the first algorithms developed for the purpose of calculating the location of an emitter based on multiple lines of bearing was due to Stansfield [1]. In that algorithm, it is assumed that the bearing errors of the EW systems are normally distributed. The joint probability density function of multiple lines of bearings is then a multivariate Gaussian probability density function. A maximum likelihood estimator for the PF ensues by maximizing the exponent in the equation for the joint probability density function (which will minimize the total probability of error because the exponent is negative).

Stansfield's original technique for calculating the PF of an emitter is probably the first example of maximum likelihood-like estimation applied to the PF problem. The geometry involved is illustrated in Figure 2.36. As described in [1, 22, 23], the goal is to minimize the expression for the joint probability of miss given as a function of the miss distances given by

$$P_{\text{miss}}(d_1, d_2, \dots, d_M) = \frac{1}{(2\pi)^{M/2} \prod_{m=1}^M \sigma_{p_m}} \exp\left(-\frac{1}{2} \sum_{m=1}^M \frac{d_m^2}{\sigma_{p_m}^2}\right) \quad (2.260)$$

when there are  $M$  sensors. This expression is minimized when the argument of  $\exp$  is maximized. The result is the PF that is the most likely location of the target.

The MLE cost function from (2.255) is given by

$$F_{\text{ML}}(\vec{x}, \vec{\phi}) = \frac{1}{2} \sum_{m=1}^M \frac{f_m^2}{\sigma_m^2} \quad (2.261)$$

where  $\vec{x}_T = (x_T, y_T)$  is the target location to be estimated. The Stansfield fix algorithm is an example of an MLE-like algorithm, where it is assumed that the errors in bearing measurement are small. In (2.261), the  $f_i$  represent the difference between the measured bearings and the bearings corresponding to a target at the estimated position given by  $\hat{\vec{x}} = (\hat{x}_T, \hat{y}_T)$ , as shown in Figure 2.36. Thus,  $f_m = \Delta\phi_m$ . Stansfield's approach assumes that the bearing errors are small enough so that  $\sin(\Delta\phi) \approx \Delta\phi$ . The cost function is thus given by

$$F_{\text{ST}}(\vec{x}, \vec{\phi}) = \frac{1}{2} \sum_{m=1}^M \frac{\sin^2 \Delta\phi_m}{\sigma_m^2} \quad (2.262)$$

However [20],

$$\begin{aligned} \sin f_m &= \sin \left[ \tan^{-1} \left( \frac{\Delta y_m}{\Delta x_m} \right) - \phi_m \right] \\ &= \frac{\Delta y_m \cos \phi_m - \Delta x_m \sin \phi_m}{\Delta_m} \\ &= \frac{(y_T - y_m) \cos \phi_m - (x_T - x_m) \sin \phi_m}{\Delta_m} \end{aligned}$$

where

$$\Delta_m^2 = \Delta_{x_m}^2 + \Delta_{y_m}^2, \quad m = 1, 2, \dots, M$$

Then

$$\begin{aligned} F_{\text{ST}}(\vec{x}, \vec{\phi}) &= \frac{1}{2} \sum_{m=1}^M \frac{[(y_T - y_m) \cos \phi_m - (x_T - x_m) \sin \phi_m]^2}{\Delta_m^2 \sigma_m^2} \\ &= \frac{1}{2} (\mathbf{A}\vec{x} - \vec{b})^T \mathbf{D}^{-1} \mathbf{C}^{-1} (\mathbf{A}\vec{x} - \vec{b}) \end{aligned} \quad (2.263)$$

where

$$\mathbf{A} = \begin{bmatrix} \sin \phi_1 & -\cos \phi_1 \\ \vdots & \vdots \\ \sin \phi_M & -\cos \phi_M \end{bmatrix}$$

$$\vec{b} = \begin{bmatrix} x_1 \sin \phi_1 - y_1 \cos \phi_1 \\ \vdots \\ x_M \sin \phi_M - y_M \cos \phi_M \end{bmatrix}$$

$$\mathbf{D} = \text{diag}(\Delta_1^2, \Delta_2^2, \dots, \Delta_M^2)$$

The value of  $\vec{x}_{\text{ST}}$  that minimizes (2.263) is given by (2.17) with the weighting matrix as  $\mathbf{W} = \mathbf{D}^{-1}\mathbf{C}_{\text{xx}}^{-1}$ , which yields

$$\hat{\vec{x}}_{\text{ST}} = (\mathbf{A}^T \mathbf{D}^{-1} \mathbf{C}_{\text{xx}}^{-1} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{D}^{-1} \mathbf{C}_{\text{xx}}^{-1} \vec{b} \quad (2.264)$$

### 2.9.2.1 Bias and Variance of the Stansfield PF Estimator

The bias and variance of the Stansfield PF estimator are examined in this section. As in Section 2.7.1, it is assumed that there are  $M$  sensors, each making  $P$  observations so that  $N = MP$  is the total number of observations. It is shown in [20] that the bias of the Stansfield PF estimator is given by

$$\mathcal{E}\{\delta\vec{x}\} \approx -\frac{1}{P} \tilde{\mathbf{C}}_{\text{xx}} \vec{h} \quad (2.265)$$

where

$$\vec{h} = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} = \sum_{m=1}^M \frac{1}{\Delta_m^2} \left( \frac{1}{\sigma_m^2 \Delta_m^4} \begin{bmatrix} 2\Delta_{x_m} \Delta_{y_m} & \Delta_{y_m}^2 - \Delta_{x_m}^2 \\ \Delta_{y_m}^2 - \Delta_{x_m}^2 & -2\Delta_{x_m} \Delta_{y_m} \end{bmatrix} \times \tilde{\mathbf{C}}_{\text{xx}} \begin{bmatrix} -\Delta_{y_m} \\ \Delta_{x_m} \end{bmatrix} + P \begin{bmatrix} \Delta_{x_m} \\ \Delta_{y_m} \end{bmatrix} \right) \quad (2.266)$$

and

$$\tilde{\mathbf{C}}_{\text{xx}}^{-1} = \tilde{\mathbf{g}}_x^T \mathbf{C}_{\text{xx}}^{-1} \tilde{\mathbf{g}}_x = \sum_{m=1}^M \frac{1}{\sigma_m^2 \Delta_m^4} \begin{bmatrix} \Delta_{y_m}^2 & -\Delta_{x_m} \Delta_{y_m} \\ -\Delta_{x_m} \Delta_{y_m} & \Delta_{x_m}^2 \end{bmatrix} \quad (2.267)$$

The tilde indicates that the matrix is evaluated at  $P = 1$ .

Thus,

$$\lim_{P \rightarrow \infty} \mathcal{E}\{\delta \vec{x}\} \approx -\tilde{\mathbf{C}}_{\text{xx}} \sum_{m=1}^M \begin{bmatrix} \frac{\Delta_{x_m}}{\Delta_m^2} \\ \frac{\Delta_{y_m}}{\Delta_m^2} \end{bmatrix} \quad (2.268)$$

which indicates that the Stansfield algorithm produces biased results, since, even for very large numbers of samples,  $M$ , the bias does not disappear.

The covariance of this estimator is given by [20]

$$\mathbf{C}_{\delta \vec{x}} \approx \frac{1}{P} \tilde{\mathbf{C}} \tilde{\mathbf{H}} \tilde{\mathbf{S}}^{-1} \tilde{\mathbf{H}}^T \tilde{\mathbf{C}} \Big|_{(\vec{x}_b, \vec{\phi}_0)} \quad (2.269)$$

where

$$\tilde{\mathbf{H}} = \sum_{m=1}^M \frac{1}{\Delta_m^2} \begin{bmatrix} \Delta_{x_m} \sin 2\phi_m - \Delta_{y_m} \cos 2\phi_m \\ -\Delta_{x_m} \sin 2\phi_m - \Delta_{y_m} \cos 2\phi_m \end{bmatrix} \tilde{\mathbf{e}}_m^T$$

$$\vec{x}_b = \vec{x}_0 + \mathcal{E}\{\delta \vec{x}\}$$

$\tilde{\mathbf{e}}_m^T$  is the  $m$ th column of the  $M \times M$  identity matrix  $\mathbf{I}_M$

Because the Stansfield PF estimator is biased, the *Cramer-Rao bound* (CRB) cannot be used as an estimate of performance, since the CRB only applies to unbiased estimators. If an unbiased PF estimator is sought, the Stansfield algorithm is probably a poor choice.

### Example

Suppose there are two sensor locations at  $(-a, 0, 0)$  and  $(a, 0, 0)$  and a target located at  $(0, b, 0)$ , as shown in Figure 2.37. Each sensor collects  $P$  observations with variance  $\sigma^2$ , assumed to be the same for all sensors. Therefore,  $M = 2$ ,  $N = 2P$ . In this case, from (2.258),

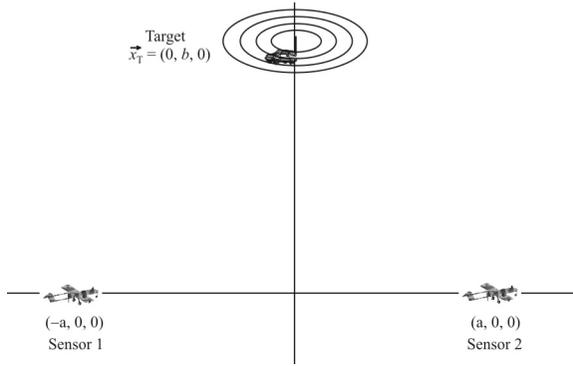


Figure 2.37 Geometry for MLE algorithm example.

$$\mathbf{g}_x^T = \left[ \begin{array}{c} \overbrace{\begin{array}{ccccc} -b & -a & \cdots & -b & -a \\ \frac{-b}{a^2+b^2} & \frac{-a}{a^2+b^2} & \cdots & \frac{-b}{a^2+b^2} & \frac{-a}{a^2+b^2} \end{array}}^{N=2P \text{ times}} \\ -b & a & \cdots & -b & a \\ \frac{-b}{a^2+b^2} & \frac{a}{a^2+b^2} & \cdots & \frac{-b}{a^2+b^2} & \frac{a}{a^2+b^2} \end{array} \right]$$

and

$$\mathbf{C}_{xx}^{-1} = \begin{bmatrix} \frac{1}{\sigma^2} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sigma^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\sigma^2} \end{bmatrix}$$

with matrix  $\mathbf{B}^{-1}$  given as

$$\mathbf{B}^{-1} = \vec{g}_x^T \mathbf{C}_{xx}^{-1} \vec{g}_x$$

$$= \begin{bmatrix} \frac{-b}{a^2+b^2} & \frac{-a}{a^2+b^2} & \cdots & \frac{-b}{a^2+b^2} & \frac{-a}{a^2+b^2} \\ \frac{-b}{a^2+b^2} & \frac{a}{a^2+b^2} & \cdots & \frac{-b}{a^2+b^2} & \frac{a}{a^2+b^2} \end{bmatrix}$$

which reduces to

$$\mathbf{B} = \frac{\sigma^2(b^2+a^2)^2}{N} \begin{bmatrix} \frac{1}{b^2} & 0 \\ 0 & \frac{1}{a^2} \end{bmatrix} \\ \times \begin{bmatrix} \frac{1}{\sigma^2} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sigma^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\sigma^2} \end{bmatrix} \begin{bmatrix} \frac{-b}{a^2+b^2} & \frac{-b}{a^2+b^2} \\ \frac{-a}{a^2+b^2} & \frac{a}{a^2+b^2} \\ \vdots & \vdots \\ \frac{-b}{a^2+b^2} & \frac{-b}{a^2+b^2} \\ \frac{-a}{a^2+b^2} & \frac{a}{a^2+b^2} \end{bmatrix}$$

From (2.265) and (2.266),

$$\mathcal{E}\{\delta\vec{x}\} \approx -\frac{\sigma^2 b(b^2+a^2)}{a^2} \begin{bmatrix} 0 \\ 1 - \frac{3}{P} + \frac{a^2}{Nb^2} \end{bmatrix}$$

so that

$$\lim_{P \rightarrow \infty} \mathcal{E}\{\delta\vec{x}\} = -\frac{\sigma^2 b(b^2+a^2)}{a^2} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

In the  $x$  direction, the limit converges to zero, forming an unbiased estimator. The same cannot be said for the  $y$  direction. There is a residual bias.

The results of this analysis for the two estimators are illustrated in Figure 2.38 [20], with the MLE as solid lines and

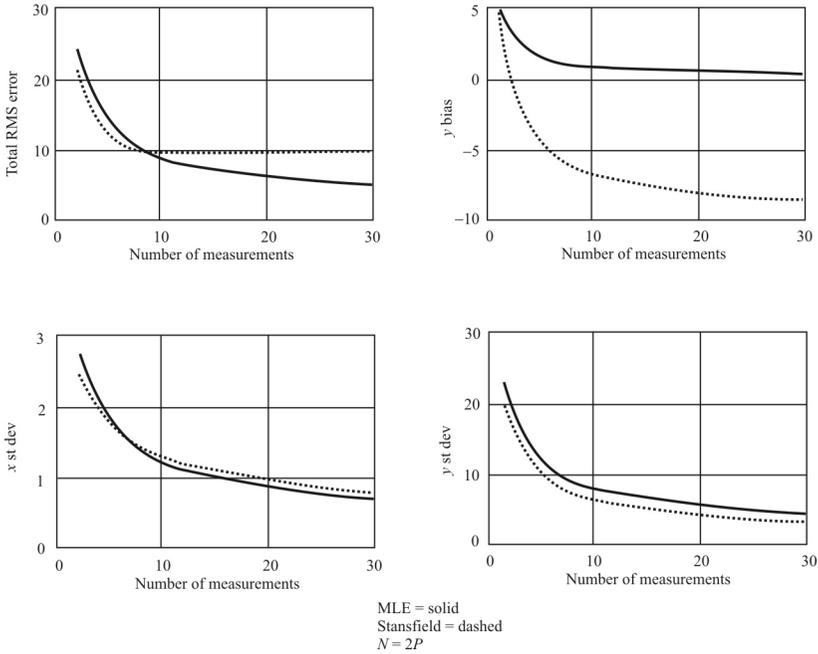
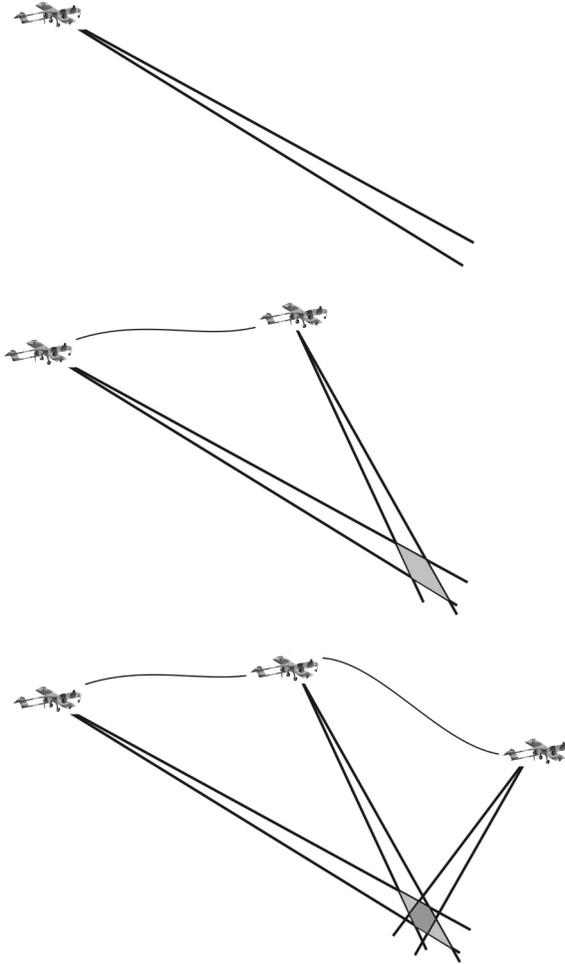


Figure 2.38 MLE example results.

the Stansfield estimator as dashed lines. The total RMS error continues to decrease for the MLE as the number of measurements increases, while that for the Stansfield estimator levels after about 10 measurements. The  $y$  bias for the MLE approaches zero, while the  $y$  bias for the Stansfield estimator settles to a relatively large negative value. The  $x$  and  $y$  standard deviations for both estimators are approximately the same.

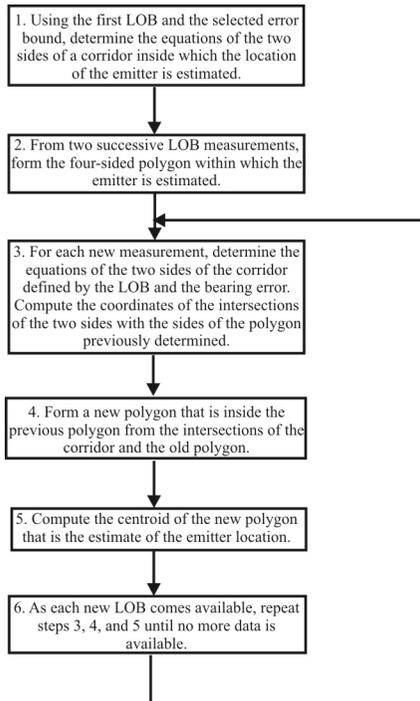
## 2.10 Multiple Sample Correlation

Fu, Vian, and Grose developed an algorithm for position fixing that utilizes the intersection of the LOB fans, defined as the measured LOB plus and minus the maximum error in the sensor [24]. Using the maximum error guarantees that the emitter is within the resulting area. If  $1\sigma$  or other statistical quantity is used instead of the maximum error, this guarantee cannot be assumed. However, the EEP can

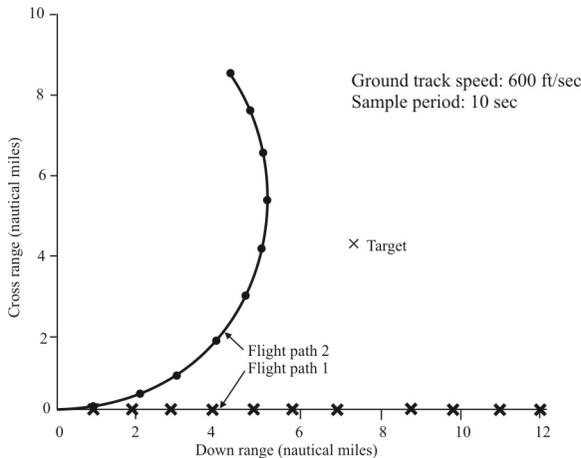


**Figure 2.39** Multiple sample correlation position fixing.

be calculated with a certain probability. The basic idea is illustrated in Figure 2.39. Successive LOBs are obtained from a single sensor or simultaneous LOBs from multiple sensors. The area on the surface of the Earth defined by the sides of these fans is the estimate of the region within which the target lies. The steps in the algorithm are given in Figure 2.40 [24].



**Figure 2.40** Multiple sample correlation algorithm.



**Figure 2.41** Flight paths for the multiple sample correlation.

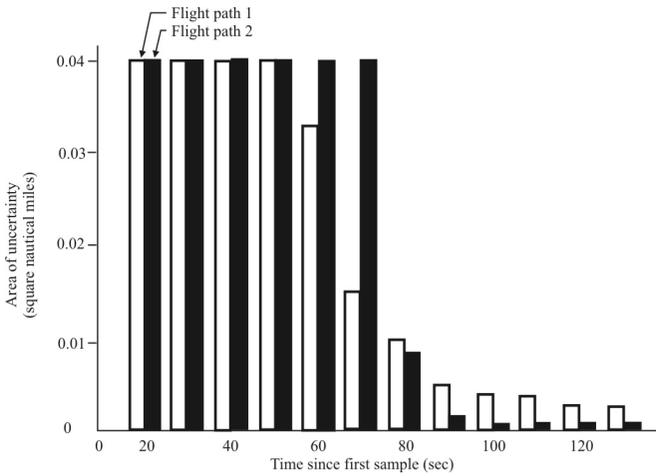
### Example

The reference describes a test case, the flight paths of which are shown in Figure 2.41 [24]. Two flight paths are described, the first a straight line along the  $x$ -axis and the second a curvilinear path with a constant turn rate of  $1^\circ$  per second. The bearing measurement error was assumed to be  $\pm 1^\circ$ .

Figure 2.42 [24] illustrates the results. Flight path 2 ultimately yielded more accurate PFs, primarily because of how close to the target the sensor traveled.

A comparison to the Kalman filter described by Spingarn [14] was also described based on Monte Carlo simulations of both algorithms. The results are shown in Table 2.4.

The idea of using maximum error is appealing because it guarantees the emitter is within the error bounds. Unfortunately, it is difficult to guarantee that there is some maximum error associated with an LOB measurement device. Operational factors are frequently the dominant source of LOB error, especially for sensors close to or on the ground, where reflections off buildings, the ground, passing vehicles, and so forth can cause LOBs that are off by a full half-circle. In those cases where multipath reflections are not likely to cause problems, such as satellite sensors or high-flying UAVs, the approach is more viable.



**Figure 2.42** Emitter area of uncertainty. For the first 40 seconds for flight path 1 and 60 seconds for flight path 2, the areas of uncertainty are off the scale (larger than 0.04 nautical mile).

**Table 2.4** Emitter Location Error

Standard Deviation (degrees)	Maximum Error (degrees)	Percent with Maximum Error	EKF Error (feet)	MSC* Error (feet)
1	1	68.3	5,477	1,633
1	2	95.6	5,477	3,267
1	3	99.7	5,477	4,900

Source: [24]. \*Multiple sample correlation.

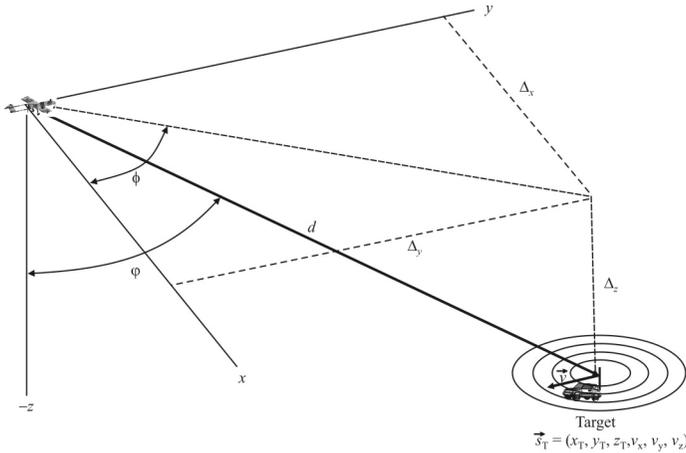


Figure 2.43 Geometry to establish observability requirements.

## 2.11 Bearing-Only Target Motion Analysis

It is possible to track moving targets with bearing-only data [called *bearing-only target motion analysis* (TMA)]. In TMA it is desirable to determine the location and velocity of a moving target, in general. For a set of instantaneous LOBs, the velocity cannot be determined, which is a special case of TMA—the target is not moving [25].

TMA had its origins in tracking ships at sea, both surface ships and subsurface with sonar. This, by far, is not the only application of TMA.

TMA is based on the scenario illustrated in Figure 2.43. The three-dimensional formulation shown in Figure 2.43 can be analyzed using only the azimuth and elevation bearings [26]. The target, located at  $(x_T, y_T, z_T)$ , is moving with a constant velocity  $(v_{Tx}, v_{Ty}, v_{Tz})$ . The state vector associated with the target at time  $i$  is thus

$$\vec{s}_T(i) = \begin{bmatrix} x_T(i) & y_T(i) & z_T(i) & v_{Tx}(i) & v_{Ty}(i) & v_{Tz}(i) \end{bmatrix}^T \quad (2.270)$$

The sensor state vector is defined similarly as

$$\vec{s}_R(i) = \begin{bmatrix} x_R(i) & y_R(i) & z_R(i) & v_{R_x}(i) & v_{R_y}(i) & v_{R_z}(i) \end{bmatrix}^T \quad (2.271)$$

The velocity of the sensor is not necessarily constant, so it is therefore a function of  $i$ . The relative target-sensor state vector is given by

$$\vec{s}(i) = \vec{s}_T(i) - \vec{s}_R(i) = \begin{bmatrix} \Delta_x(i) & \Delta_y(i) & \Delta_z(i) & v_x(i) & v_y(i) & v_z(i) \end{bmatrix}^T \quad (2.272)$$

The discrete time equation describing this dynamic system is given by

$$\vec{s}(i) = \mathbf{\Phi}(i, i-1)\vec{s}(i-1) + \vec{u}(i) \quad (2.273)$$

where the state transition matrix is given by

$$\mathbf{\Phi}(i, i-1) = \begin{bmatrix} \mathbf{I}_2 & [i - (i-1)]\mathbf{I}_2 \\ \mathbf{Z}_2 & \mathbf{I}_2 \end{bmatrix} \quad (2.274)$$

$$\mathbf{I}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2.275)$$

$$\mathbf{Z}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (2.276)$$

and  $i$  is the  $i$ th sample index while  $i-1$  is the  $i-1$ st sample index. The term

$$\vec{u}(i) = [0 \quad 0 \quad 0 \quad u_x(i) \quad u_y(i) \quad u_z(i)]^T$$

in (2.273) accounts for the effects of acceleration of the sensor.<sup>3</sup>

The measurements in this system consist of azimuth bearings collected at the sensor referenced to the positive  $x$ -axis and measured counterclockwise-positive, corrupted by noise as

$$\phi_i = \tilde{\phi}_i + n_{\phi_i} \quad (2.277)$$

where  $\tilde{\phi}_i$  is the true bearing to the target given by

---

<sup>3</sup> Acceleration could be as simple as turning the sensor.

$$\tilde{\phi}_i = \tan^{-1} \frac{\Delta_x(i)}{\Delta_y(i)} \quad (2.278)$$

and  $n_{\phi_i} \sim \mathcal{N}(0, \sigma_{n\phi}^2)$ ,  $n_{\phi_i} \in (-\pi, \pi)$ . In addition, the elevation angle of arrival,  $\phi_i$ , measured from the negative  $z$ -axis, also corrupted by noise, given by

$$\phi_i = \tilde{\phi}_i + n_{\phi_i} \quad (2.279)$$

where  $\tilde{\phi}_i$  is the true elevation bearing to the target given by

$$\tilde{\phi}_i = \tan^{-1} \frac{\sqrt{\Delta_x^2(i) + \Delta_y^2(i)}}{\Delta_z(i)} \quad (2.280)$$

and  $n_{\phi_i} \sim \mathcal{N}(0, \sigma_{n\phi}^2)$ ,  $n_{\phi_i} \in (-\pi, \pi)$ . We denote the measurement vector as

$$\bar{\theta}_i = [\phi_i \quad \phi_i]^\top \quad (2.281)$$

and the noise vector as

$$\bar{n}_i = [n_{\phi_i} \quad n_{\phi_i}]^\top \quad (2.282)$$

These equations describe a nonlinear process. They can be converted into a linear problem by defining a new measurement given by

$$\alpha_i = \vec{h}^\top(\bar{\theta}_i) \vec{s}_R(i) \quad (2.283)$$

and

$$\alpha_i = \vec{h}^\top(\bar{\theta}_i) \vec{s}_T + d_i \sin \bar{n}_i \quad (2.284)$$

where

$$d_i = \sqrt{\Delta_x^2(i) + \Delta_y^2(i) + \Delta_z^2(i)} \quad (2.285)$$

and

$$\vec{h}^T(\vec{\theta}_i) = [\cos \vec{\theta}_i \quad -\sin \vec{\theta}_i \quad -\sin \vec{\theta}_i \quad 0 \quad 0 \quad 0] \quad (2.286)$$

Since the noise is assumed to be Gaussian and additive, the series of bearing measurements given by the vector  $\vec{\theta} = [\vec{\theta}_1 \quad \vec{\theta}_2 \quad \dots \quad \vec{\theta}_i]$  have the likelihood function

$$P_{\vec{\theta}|\vec{s}} = \frac{1}{\sqrt{(2\pi)^i \det(\mathbf{W})}} \exp \left\{ -\frac{1}{2} [\vec{\theta} - \vec{\tilde{\theta}}(\vec{s})]^T \mathbf{W}^{-1} [\vec{\theta} - \vec{\tilde{\theta}}(\vec{s})] \right\} \quad (2.287)$$

where  $\vec{\tilde{\theta}}(\vec{s})$  is the vector of correct bearings and

$$\mathbf{W} = \text{diag}(\sigma_{\alpha_i}^2) \quad (2.288)$$

The MLE is given by the solution of

$$\frac{\partial \ln P_{\vec{\theta}|\vec{s}}}{\partial \vec{s}} = 0 \quad (2.289)$$

Using (2.287), this equates to

$$\left[ \frac{\partial \vec{\theta}(\vec{s})}{\partial \vec{s}} \right]^T \mathbf{W}^{-1} [\vec{\theta} - \vec{\tilde{\theta}}(\vec{s})] = \vec{0} \quad (2.290)$$

Carrying out the math finally yields the iteration equation

$$\vec{s}_{i+1} = \vec{s}_i - \lambda_i [\hat{\Gamma}_\theta^T \hat{\Psi}^{-1} \mathbf{W}^{-1} \hat{\Psi}^{-1} \hat{\Gamma}_\theta]^{-1} \hat{\Gamma}_\theta^T \hat{\Psi}^{-1} \mathbf{W}^{-1} [\vec{\theta} - \hat{\vec{\theta}}] \quad (2.291)$$

where

$$\hat{\vec{\theta}} = \vec{\theta}(\hat{\vec{s}})$$

$$\mathbf{\Gamma}_{\hat{\theta}} = \begin{bmatrix} \cos \hat{\theta}_1 & -\sin \hat{\theta}_1 & -\sin \hat{\theta}_1 & (i-m)\cos \hat{\theta}_1 & (i-m)\sin \hat{\theta}_1 & (i-m)\sin \hat{\theta}_1 \\ \cos \hat{\theta}_2 & -\sin \hat{\theta}_2 & -\sin \hat{\theta}_2 & (i-m)\cos \hat{\theta}_2 & (i-m)\sin \hat{\theta}_2 & (i-m)\sin \hat{\theta}_2 \\ \vdots & & & & & \\ \cos \hat{\theta}_i & -\sin \hat{\theta}_i & -\sin \hat{\theta}_i & (i-m)\cos \hat{\theta}_i & (i-m)\sin \hat{\theta}_i & (i-m)\sin \hat{\theta}_i \end{bmatrix}$$

$$\hat{\Psi} = \text{diag}(d_i)$$

Parameter  $\lambda_i$  is the step size at step  $i$ , and  $m$  is the reference index when the time scale started.

The estimated information matrix is

$$\hat{\Omega} = \hat{\Gamma}_{\hat{\theta}}^T \hat{\Psi}^{-1} \mathbf{W}^{-1} \hat{\Psi}^{-1} \hat{\Gamma}_{\hat{\theta}} \quad (2.292)$$

and the estimated error covariance matrix is

$$\hat{\mathbf{P}}_{\text{MLE}} = \hat{\Omega}_{\text{MLE}}^{-1} \quad (2.293)$$

The general requirement for being able to perform bearing-only TMA in terms of observability requirements was established by Hammel and Aidala [26] in an underwater sonar context. Assuming a single moving sensor system, then with the notation as indicated in Figure 2.43, the requirements to be able to determine the location and velocity of the target (to be able to *observe* the target<sup>4</sup>) are given by

$$\det(\mathbf{H}^T \mathbf{H}) \neq 0 \quad (2.294)$$

where  $\mathbf{H}$  is the matrix given by

<sup>4</sup> A dynamical system described by the normal form equations

$$\dot{\vec{x}} = \mathbf{A}\vec{x} + \mathbf{B}\vec{v} \quad \vec{y} = \mathbf{C}\vec{x} + \mathbf{D}\vec{v} + \mathbf{E}\dot{\vec{v}}$$

with  $n$  states in vector  $\vec{x}$ ,  $m$  inputs in vector  $\vec{v}$ , and  $p$  outputs in vector  $\vec{y}$ , with  $\mathbf{A}_{n \times n}$  the state transition matrix,  $\mathbf{B}_{n \times m}$ ,  $\mathbf{C}_{p \times n}$ ,  $\mathbf{D}_{p \times m}$ , and  $\mathbf{E}_{p \times m}$ , is *controllable* if, given any initial state  $\vec{x}_0$  at  $t_0$ , a set of inputs exists that can drive the system to the zero state at  $t_1$ . This system is *observable* if, observing the outputs over  $(t_0, t_1)$ , the initial state of the system at  $t_0$  can be determined.

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_2 & \vec{h}_0 & \mathbf{Z}_2 & \vec{0} \\ \mathbf{Z}_2 & \vec{h}_1 & \mathbf{I}_2 & \vec{h}_0 \\ \mathbf{Z}_2 & \vec{h}_2 & \mathbf{Z}_2 & 2\vec{h}_1 \\ \mathbf{Z}_2 & \vec{h}_3 & \mathbf{Z}_2 & 3\vec{h}_2 \\ \mathbf{Z}_2 & \vec{h}_4 & \mathbf{Z}_2 & 4\vec{h}_3 \\ \mathbf{Z}_2 & \vec{h}_5 & \mathbf{Z}_2 & 5\vec{h}_4 \end{bmatrix} \quad (2.295)$$

with  $\mathbf{I}_2$  and  $\mathbf{Z}_2$  given by (2.275) and (2.276) and

$$\vec{h}_k = \begin{bmatrix} f_k \\ g_k \end{bmatrix} = - \begin{bmatrix} \frac{d^i(\sin \phi \cot \phi)}{dt^i} \\ \frac{d^i(\cos \phi \cot \phi)}{dt^i} \end{bmatrix}, \quad k = 0, 1, \dots, 5$$

Evaluation of the determinate expressed by (2.294) yields the equivalent requirement for observability as

$$\det(\mathbf{H}^T \mathbf{H}) = \frac{1}{2} \sum_{k=1}^4 \sum_{l=1}^4 \left\{ \begin{aligned} & [(k+1)f_k f_{l+1} - (l+1)f_l f_{k+1}]^2 \\ & + 2[(k+1)f_k g_{k+1} - (l+1)g_l f_{k+1}]^2 \\ & + [(k+1)g_k g_{k+1} - (l+1)g_l g_{k+1}]^2 \end{aligned} \right\} \quad (2.296)$$

Requirement (2.294) is satisfied if at least one of the terms in the double summation in (2.296) does not vanish.

Sensor or target motion is required in order to evaluate the velocity terms. If more than one sensor is available so that multiple instantaneous LOBs are available, then geoposition determination is possible from a single set of LOBs, but velocity still cannot be determined.

Note that if the sensor as well as the target are both moving at a constant velocity, the system is unobservable [26]. When there is only one moving sensor involved, this requirement dictates that the sensor must perform some sort of maneuver so that the velocity vector of the sensor is changed. (It is tacitly assumed that the velocity vector of the target cannot be affected by the sensor.) Changing the velocity vector implies an acceleration/deceleration of the sensor. Thus, the fundamental requirement is for the sensor to accelerate/decelerate in some fashion in order for the target to be observable. It should be noted that some forms of acceleration are better than others.

The portions of the sensor path that are defined by different velocity vectors are referred to as *legs*. The target is unobservable in the first leg of the sensor path when there is only one sensor involved. If the motion of the single sensor is collinear<sup>5</sup> with the motion of the target, the system is not observable either. Acceleration is required in this case as well, that changes the direction of the velocity vector.

It is relatively easy and fast to maneuver an airborne sensor observing a target on the ground or on the sea. For surface sensors, such as a ship with sonar, this maneuvering can take considerably more time. The same can be said for sub-surface boats.

## 2.12 Sources of Error in Triangulation

There are several sources of error that can enter into the PF estimation by triangulation. Two have already been discussed at length: noise and measurement errors. There are others, however. Some of these are discussed in this section.

### 2.12.1 Geometric Dilution of Precision in Triangulation

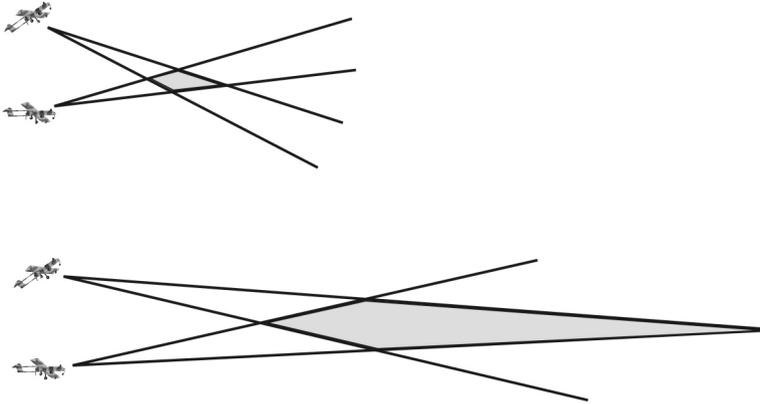
As explained in [27], the geometry of the baseline of sensors has an effect on the accuracy of the geopositions computed. Generally, the farther from the baseline, the less accurate the PF estimate will be. This is illustrated in Figure 2.44. The sensors shown at the top and bottom of Figure 2.44 have the same LOB accuracy, yet the area within which the target lies is substantially larger the farther it is from the baseline of the sensors. As can be seen, the farther from the baseline the target is, the LOBs become more and more parallel. This effect is called *geometric dilution of precision* (GDOP). These effects are quantified in the example shown in Figure 2.45 [28], where CEP/ $\sigma$  normalized by  $d$ , the length of the baseline, is plotted versus  $x$  and  $y$ , also normalized.

GDOP also has degradation effects when the target is moved off the perpendicular bisector of the baseline. These effects are also illustrated in Figure 2.45.

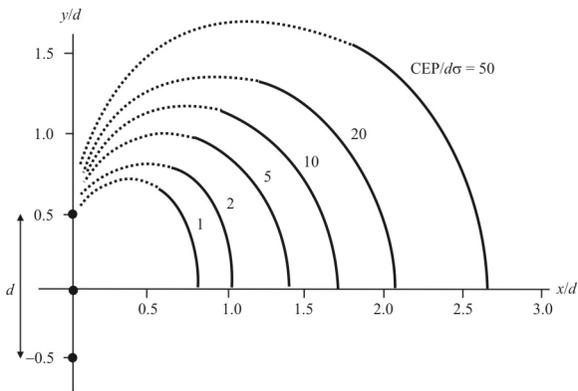
The shape of the baseline also has an effect. Off the end of a baseline between any two sensors, the LOBs become linearly dependent (they are collinear) and the fix accuracy explodes. This is illustrated in Figure 2.46 [28]. The effect is seen

---

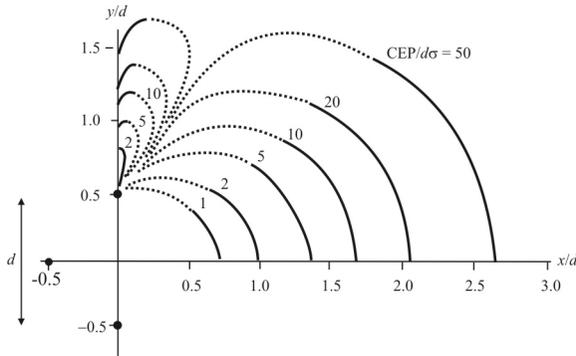
<sup>5</sup> A system moving with velocity vector  $\vec{v}_0(t)$  is moving *collinearly* with another system moving with velocity  $\vec{v}_1(t)$  if  $\vec{v}_0(t) = \vec{k}\vec{v}_1(t)$  for some constant vector  $\vec{k}$ . Expressed as 3-D vectors, this means that the two systems are moving in the same direction.



**Figure 2.44** Distance effects on PF calculation accuracy.



**Figure 2.45** The effects of GDOP when a linear baseline is employed. The dotted lines represent regions where the geolocation calculation is suspect (too close to the end of one or more baselines).



**Figure 2.46** GDOP V-baseline.

here by the range over which the CEP is some finite value. Off the ends of the baselines, the CEP increases without bound.

### 2.12.2 LOB Error

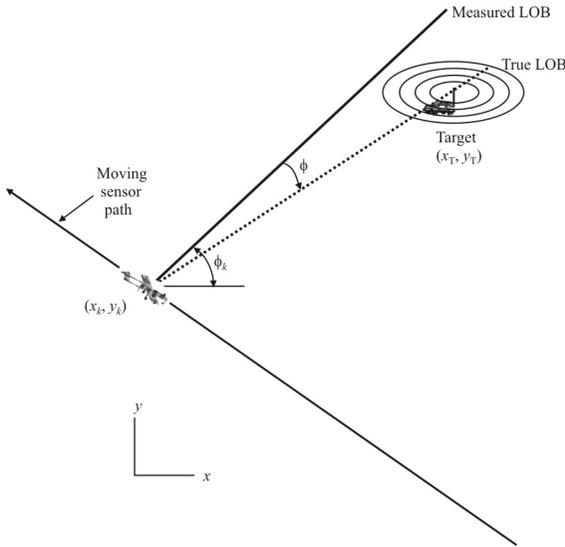
The constituent LOBs used for the PF estimation are never error free. Noise and systematic errors are always present. Some of the systematic error can be, and usually is, removed by calibration.

The LOB error contribution increases the farther the sensor array is from the target. This can be seen with the aid of Figure 2.44. The accuracy of LOB systems is usually specified in terms of LOB angular error. The LOBs in Figure 2.44 represent the  $1-\sigma$  limits and are the same for both parts of the figure. Notice that the area in the error zone in the lower part of the figure is much larger than that at the top. This is because in the lower case, the PF is farther from the sensor baseline.

### 2.12.3 Effects of Bias on Bearing-Only PF

The effects of bias errors on bearing-only PF calculations were examined by Gavish and Fogel [29]. The geometry is depicted in Figure 2.47. It is assumed that the angle measurements are corrupted by zero mean AWGN  $n_k$  with variance  $\sigma_n^2$  and bias  $\phi$ , which has zero mean and variance  $\sigma_\phi^2$ . Therefore,

$$\mathcal{E}\{n_j n_k\} = \sigma_n^2 \delta_{jk} \quad (2.297)$$



**Figure 2.47** Effects of LOB bias errors.

Thus, the measured bearings are characterized by

$$\phi_n = \tan^{-1} \left( \frac{\Delta_{y_n}}{\Delta_{x_n}} \right) + \phi + n_n, \quad n = 1, 2, \dots, N \quad (2.298)$$

where  $\Delta_{y_n} = y_T - y_k$  and  $\Delta_{x_n} = x_T - x_k$  and there are  $N$  observations (bearing measurements). Let

$$\vec{\theta} = [x_T \quad y_T \quad \phi]^T \quad (2.299)$$

denote the vector to be estimated. Since the bearing error bias is assumed  $\sim \mathcal{N}(0, \sigma_\phi^2)$  and the noise is assumed  $\sim \mathcal{N}(0, \sigma_n^2)$ , then  $\vec{\theta}$  has a multivariate normal distribution with mean vector  $\vec{m}$  and positive definite covariance matrix  $C_{\vec{\theta}}$ . Let  $\vec{h}(\vec{m}, \vec{\theta}) = \vec{0}$  be a vector of constraints. Also let

$$\vec{h}_m = \frac{\partial \vec{h}}{\partial \vec{m}}, \quad \vec{h}_\theta = \frac{\partial \vec{h}}{\partial \vec{\theta}} \quad (2.300)$$

With a priori information about  $\vec{\theta}$  given by the information matrix  $\mathbf{J}$ , the CRB of any unbiased estimator of  $\vec{\theta}$  is given by

$$\mathbf{S}_\phi = [\vec{h}_\phi^T (\vec{h}_m \mathbf{C}_\phi \vec{h}_m^T)^{-1} \vec{h}_\phi + \mathbf{J}]^{-1} \quad (2.301)$$

The variance of the bias,  $\sigma_\phi^2$ , is assumed to be known and nonzero. Therefore,

$$\mathbf{J} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1/\sigma_\phi^2 \end{bmatrix} \quad (2.302)$$

From (2.298),

$$\vec{h}_\theta = [\mathbf{G} \quad \vec{1}_N] \quad (2.303)$$

where

$$\mathbf{G} = \begin{bmatrix} \frac{-\Delta_{y_1}}{d_1^2} & \frac{-\Delta_{y_2}}{d_2^2} & \dots & \frac{-\Delta_{y_N}}{d_N^2} \\ \frac{-\Delta_{x_1}}{d_1^2} & \frac{-\Delta_{x_2}}{d_2^2} & \dots & \frac{-\Delta_{x_N}}{d_N^2} \end{bmatrix}^T \quad (2.304)$$

with, as before,

$$d_n^2 = \Delta_{x_n}^2 + \Delta_{y_n}^2, \quad n = 1, 2, \dots, N \quad (2.305)$$

and  $\vec{1}_N$  is a vector of  $N$  1's.

In this case  $\vec{h}_m$  and  $\sigma^{-2}\mathbf{R}$  are equal to the  $N \times N$  identity matrix, since from (2.297) the noise covariance is assumed to be diagonal.

Substituting (2.303) and (2.302) into (2.301) yields

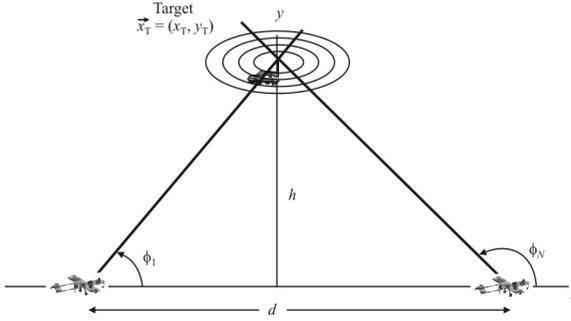


Figure 2.48 Geometry for examining the effects of bias.

$$\mathbf{S}_\theta = \{[\mathbf{G} \quad \vec{\mathbf{1}}_N]^T (\sigma_n^2 \mathbf{I}_N)^{-1} [\mathbf{G} \quad \vec{\mathbf{1}}_N] + \text{diag}(0, 0, \sigma_\phi^{-2})\}^{-1} \quad (2.306)$$

$$= \sigma_n^2 \begin{bmatrix} \mathbf{Y}^{-1} & \mathbf{f} \\ \mathbf{f}^T & \left(\frac{\sigma_n}{\sigma_\phi}\right)^2 + N \end{bmatrix}^{-1} \quad (2.307)$$

where

$$\vec{\mathbf{f}} = \mathbf{G}^T \vec{\mathbf{1}}_N, \quad \mathbf{Y} = (\mathbf{G}^T \mathbf{G})^{-1} \quad (2.308)$$

The CRB is the upper left  $2 \times 2$  block of (2.307), yielding

$$\mathbf{S} = \sigma_n^2 \begin{bmatrix} \mathbf{Y} + \frac{\mathbf{Y} \vec{\mathbf{f}} \vec{\mathbf{f}}^T \mathbf{Y}}{\left(\frac{\sigma_n}{\sigma_\phi}\right)^2 + N - \vec{\mathbf{f}}^T \mathbf{Y} \vec{\mathbf{f}}} \end{bmatrix} \quad (2.309)$$

The example from [29] is useful for illustrating the effects of bias in this analysis. There is a sensor (own-ship) moving along the  $x$ -axis ( $y_n = 0$ ), as illustrated in Figure 2.48. For large  $N$ , (2.308) can be expressed as

$$\vec{\mathbf{f}} = \frac{N-1}{L} \begin{bmatrix} -\phi_d & \ln \frac{\sin \phi_N}{\sin \phi_1} \end{bmatrix}^T \quad (2.310)$$

and

$$\mathbf{Y} = \frac{2dh}{N-1} \begin{bmatrix} \frac{\phi_d + \sin \phi_d \cos \phi_s}{\phi_d^2 - \sin 2\phi_d} & \frac{\sin \phi_d \sin \phi_s}{\phi_d^2 - \sin 2\phi_d} \\ \frac{\sin \phi_d \sin \phi_s}{\phi_d^2 - \sin 2\phi_d} & \frac{\phi_d - \sin \phi_d \cos \phi_s}{\phi_d^2 - \sin 2\phi_d} \end{bmatrix} \quad (2.311)$$

where

$$\phi_d = \phi_N - \phi_1$$

$$\phi_s = \phi_1 + \phi_N$$

The CEP for this case is given by

$$\text{CEP} \approx 0.75\sqrt{\text{TrS}} \quad (2.312)$$

which is accurate to approximately 10%.

### Example

For specificity, suppose that  $(x_T, y_T, z_T) = (0, 50, 0)$  and the sensor trajectory is given by  $x(t) = -50 + 0.15t$ ,  $y(t) = 0$ , and  $z(t) = 0$ . Suppose that a bearing is taken every five time units and that  $\sigma_n = 2^\circ$ . The resulting CEP is illustrated in Figure 2.49 [29] for the case of no bias and  $\sigma_\phi = 3^\circ$ . For the case of no bias, the CEP asymptotically approaches approximately 300 m while that with the bias approaches 400 m.

For this example, the CEP is shown in Figure 2.50 [29] versus the amount of bias in the bearing measurements for  $T = 500$ . The bias approaches

$$\mathbf{S}_{\max} = \lim_{\sigma_\phi \rightarrow \infty} \mathbf{S} = \sigma_n^2 \left\{ \mathbf{Y} + \frac{\mathbf{Y}\vec{f}\vec{f}^T\mathbf{Y}}{N - \vec{f}^T\mathbf{Y}\vec{f}} \right\}$$

### 2.12.4 Combining Noisy LOB Measurements

McCabe and Al-Samara [30] developed an approach to triangulation using a fusion methodology. The scenario considered is shown in Figure 2.51 [29] for

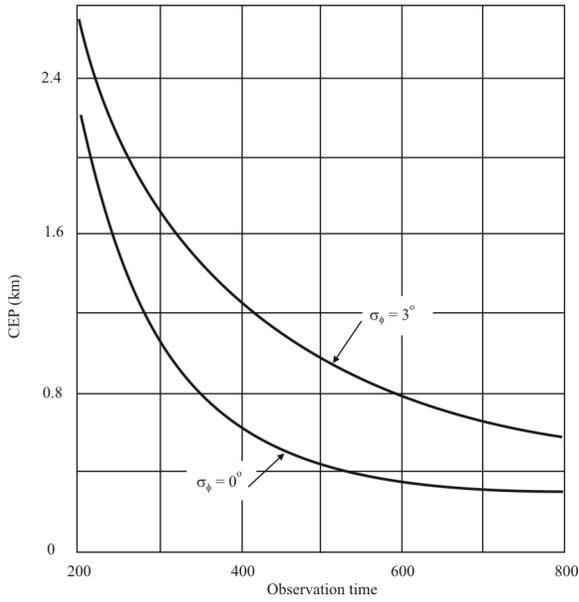


Figure 2.49 CEP of computed fixes for the bias example.

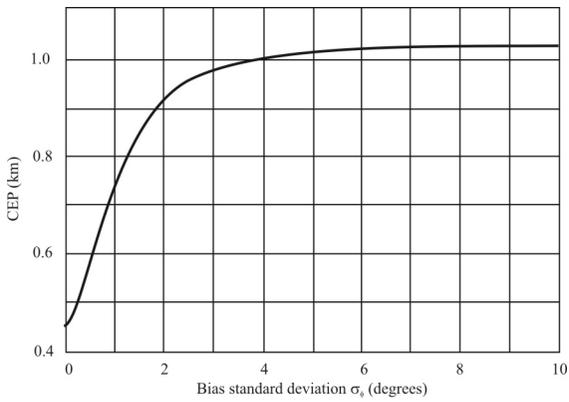
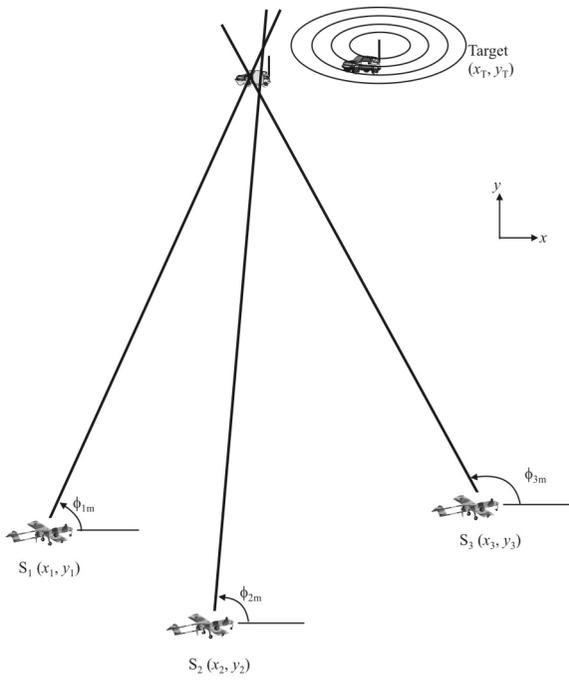


Figure 2.50 CEP versus the bias standard deviation for the example.



**Figure 2.51** Fusion geometry.

$M = 2$ . The position estimate obtained from combining the LOBs from sensors  $S_1$  and  $S_2$  is given by

$$x_{12} = \frac{x_1 \tan \phi_1 - x_2 \tan \phi_2 + y_2 - y_1}{\tan \phi_1 - \tan \phi_2} \quad (2.313)$$

$$y_{12} = (x_{12} - x_1) \tan \phi_1 + y_1 \quad (2.314)$$

Likewise, the PF estimate obtained by combining the data from sensors  $S_2$  and  $S_3$  is given by

$$x_{23} = \frac{x_2 \tan \phi_2 - x_3 \tan \phi_3 + y_3 - y_2}{\tan \phi_2 - \tan \phi_3} \quad (2.315)$$

$$y_{23} = (x_{23} - x_2) \tan \phi_2 + y_2 \quad (2.316)$$

with

$$\vec{\Delta}_{z_{12}} \triangleq \begin{bmatrix} \Delta_{x_{12}} \\ \Delta_{y_{12}} \end{bmatrix} \approx \begin{bmatrix} \frac{\partial x_{12}(\phi_1, \phi_2)}{\partial \phi_1} & \frac{\partial x_{12}(\phi_1, \phi_2)}{\partial \phi_2} \\ \frac{\partial y_{12}(\phi_1, \phi_2)}{\partial \phi_1} & \frac{\partial y_{12}(\phi_1, \phi_2)}{\partial \phi_2} \end{bmatrix} \begin{bmatrix} \Delta_{\phi_1} \\ \Delta_{\phi_2} \end{bmatrix} \quad (2.317)$$

$$\triangleq \mathbf{A}_{12} \begin{bmatrix} \Delta_{\phi_1} \\ \Delta_{\phi_2} \end{bmatrix} \quad (2.318)$$

the covariance matrix,  $\mathbf{C}_{12}$ , corresponding to (2.313) and (2.314) can be approximated by

$$\mathbf{C}_{12} \triangleq \mathcal{E}\{\vec{\Delta}_{z_{12}} \vec{\Delta}_{z_{12}}^T\} = \mathbf{A}_{12} \begin{bmatrix} \sigma_{\phi_1}^2 & 0 \\ 0 & \sigma_{\phi_2}^2 \end{bmatrix} \mathbf{A}_{12}^T \quad (2.319)$$

Likewise,

$$\mathbf{C}_{23} \triangleq \mathcal{E}\{\vec{\Delta}_{z_{23}} \vec{\Delta}_{z_{23}}^T\} = \mathbf{A}_{23} \begin{bmatrix} \sigma_{\phi_2}^2 & 0 \\ 0 & \sigma_{\phi_3}^2 \end{bmatrix} \mathbf{A}_{23}^T \quad (2.320)$$

The cross-covariance matrix,  $\mathbf{C}_{123}$ , corresponding to these measurements is

$$\mathbf{C}_{123} \triangleq \mathcal{E}\{\vec{\Delta}_{z_{12}} \vec{\Delta}_{z_{23}}^T\} = \mathbf{A}_{12} \begin{bmatrix} 0 & 0 \\ \sigma_{\phi_2}^2 & 0 \end{bmatrix} \mathbf{A}_{23} \quad (2.321)$$

The combining of these measurements is done with a fusion algorithm from [31]. That algorithm forms the linear combination

$$\hat{\mathbf{X}} = \sum_{m=1}^M \mathbf{A}_m \hat{\mathbf{X}}_m \quad (2.322)$$

For  $P = 2$  sets of measurements, as considered here,

$$\mathbf{A}_1 = \mathbf{C}'_{21} [\mathbf{C}']^{-1} \quad (2.323)$$

and

$$\mathbf{A}_2 = [\mathbf{I} - \mathbf{A}_1] \quad (2.324)$$

where

$$\begin{aligned} \mathbf{C}'_{21} &= \mathbf{C}_{23} - \mathbf{C}_{123}^T \\ \mathbf{C}' &= \mathbf{C}_{12} - \mathbf{C}_{123} - \mathbf{C}'_{123} \end{aligned}$$

The resulting fused covariance matrix is given by

$$\mathbf{C}_0 = \mathbf{C}_{23} - (\mathbf{C}_{23} - \mathbf{C}_{123}^T)(\mathbf{C}_{12} - \mathbf{C}_{123} - \mathbf{C}_{123}^T + \mathbf{C}_{23})^{-1}(\mathbf{C}_{23}^T - \mathbf{C}_{123}) \quad (2.325)$$

Although presented in two dimensions, this algorithm extends to three dimensions in the obvious way. Also, only three sensors were illustrated yielding  $P = 2$  sets of LOB measurements. This too extends in the natural way to  $P > 2$ .

### Example

Consider the case where the PDFs for the LOBs at the sensors are given as in Figure 2.52 [30]. Three sensors are located at  $S_1$ : (2, 3);  $S_2$ : (5, 4); and  $S_3$ : (9, 3). The PDF parameters are  $\delta_1 = 5^\circ$ ,  $\delta_2 = 4^\circ$ , and  $\delta_3 = 6^\circ$ . The fused  $x$ -data errors are shown in Figure 2.53(a) [30] and the fused  $y$ -data errors are shown in Figure 2.53(b).

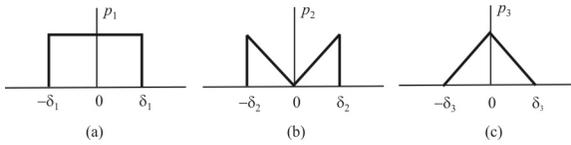


Figure 2.52(a-c) PDFs for the three sensors in the example.

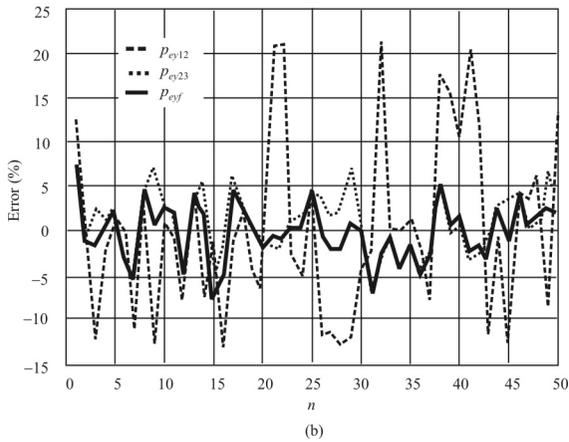
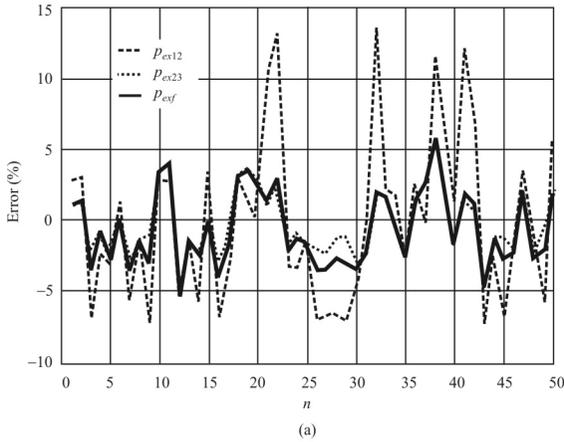


Figure 2.53(a, b) Results of the simulation for the example. Chart (a) illustrates the errors for the  $x$ -parameters, while chart (b) is for the  $y$ -parameters.

### 2.12.5 Effects of Navigation Error

Another source of error in triangulation estimation of the PF is errors associated with knowing the locations of the sensors. Manolakis and Cox [32] presented an analysis of sensor position error effects on the estimation of three-dimensional target PFs, although their analysis was actually based on calculating the effects of range difference estimation using *distance measuring equipment* (DME) to estimate the location of cooperating aircraft. This section follows [32].

The notation used here is as follows:

$\vec{x}_m = [x_m \ y_m \ z_m]^T$  is the vector of coordinates for sensor  $m$

$\vec{x}_T = [x_T \ y_T \ z_T]^T$  is the vector of target coordinates

$f_m(\vec{x}_T)$  is the system measurement equation for sensor  $m$  given by

$$f_m(\vec{x}_T) = 0 \quad (2.326)$$

For simplicity it is assumed that there is one master sensor where the PF calculations are performed and two slave stations configured similar to that illustrated in Figure 2.51. Thus,  $\vec{f}(\vec{x}_T) = [f_1(\vec{x}_T) \ f_2(\vec{x}_T) \ f_3(\vec{x}_T)]^T$ .

A Taylor series expansion of (2.326) is used and the terms higher than the linear ones are discarded, yielding

$$\vec{f}(\vec{x}_T) \approx \vec{f} \Big|_{\vec{x}_{T_0}} + \mathbf{J} \Big|_{\vec{x}_{T_0}} (\vec{x}_T - \vec{x}_{T_0}) \quad (2.327)$$

where the notation  $\mathbf{J} \Big|_{\vec{x}_{T_0}}$  denotes  $\mathbf{J}$  evaluated at  $\vec{x}_{T_0}$ . In (2.327),

$$\mathbf{J} = \frac{\partial \vec{f}}{\partial \vec{x}_T} \quad (2.328)$$

is the Jacobian matrix.

Calculation of the target position is an iterative process, with the initial guess denoted by  $\vec{x}_{T_0}$ . New position estimates are provided by (2.326) and (2.327) as

$$\vec{x}_T \approx \vec{x}_{T_0} - \mathbf{J}^{-1} \vec{f} \Big|_{\vec{x}_{T_0}} \quad (2.329)$$

and iteration continues until convergence is achieved.

Let  $[x_{s_m} \ y_{s_m} \ x_{s_m}]^T$  denote the measured position vector of sensor  $m$  and  $[\delta_{x_m} \ \delta_{y_m} \ \delta_{z_m}]^T$  the corresponding measurement errors for  $m = 1, 2, 2$ . Let  $\vec{p}_x$  denote the actual sensor vector along the  $x$ -axis, with corresponding definitions for measured sensor vectors along the  $y$ -axis and  $z$ -axis. Thus, using this notation,  $\vec{p}_x = [x_1 \ x_2 \ x_3]^T$ ,  $\vec{p}_y = [y_1 \ y_2 \ y_3]^T$ , and  $\vec{p}_z = [z_1 \ z_2 \ z_3]^T$ . Let  $\vec{p}_{sx}$  denote the vector of measured sensor  $x$ -coordinates so that  $\vec{p}_{sx} = [x_{s_1} \ x_{s_2} \ x_{s_3}]^T$ . Likewise,  $\vec{p}_{sy} = [y_{s_1} \ y_{s_2} \ y_{s_3}]^T$  and  $\vec{p}_{sz} = [z_{s_1} \ z_{s_2} \ z_{s_3}]^T$ . Last, let  $\delta\vec{p}_x$  denote the error vector containing the errors in measuring the coordinates, namely,  $\delta\vec{p}_{sx} = [\delta_{x_1} \ \delta_{x_2} \ \delta_{x_3}]^T$ , as well as  $\delta\vec{p}_{sy} = [\delta_{y_1} \ \delta_{y_2} \ \delta_{y_3}]^T$ , and  $\delta\vec{p}_{sz} = [\delta_{z_1} \ \delta_{z_2} \ \delta_{z_3}]^T$ . Obviously,

$$\vec{p}_{si} = \vec{p}_i + \delta\vec{p}_i \quad i \in \{x, y, z\} \tag{2.330}$$

Let  $\vec{p} = [\vec{p}_x \ \vec{p}_y \ \vec{p}_z]^T$  denote the vector consisting of all of the coordinates. That is, if  $\vec{p}_x$  is  $3 \times 1$ , then  $\vec{p}$  is  $9 \times 1$ . Likewise for  $\vec{p}_s$  and  $\delta\vec{p}$ .

The errors in estimating the PF of the target due to sensor positioning errors are given by  $\vec{\epsilon} = [\epsilon_x \ \epsilon_y \ \epsilon_z]^T$  and

$$\vec{\epsilon} = \vec{x}'_{\vec{x}_T} - \vec{x}_T \tag{2.331}$$

where

$$\vec{x}'_{\vec{x}_T} = \vec{x}_{T_0} - \mathbf{J}^{-1} \vec{f} \Big|_{(\vec{x}_{T_0}, \vec{p}_s)} \tag{2.332}$$

is the estimate of the PF when  $\mathbf{J}^{-1}$  and  $\vec{f}$  are evaluated at  $(\vec{x}_{T_0}, \vec{p}_s)$ .

Therefore, errors in sensor positioning can and should be included in (2.326), yielding

$$\vec{f}(\vec{x}_T, \vec{p}) \approx \vec{f} + \mathbf{J}(\vec{x}_T - \vec{x}_{T_0}) \Big|_{(\vec{x}_{T_0}, \vec{p}_s)} + \left( \frac{\partial \vec{f}}{\partial \vec{p}} \right)^T \delta\vec{p} \Big|_{(\vec{x}_{T_0}, \vec{p}_s)} \tag{2.333}$$

so that the new value of  $\vec{x}_T$  is given by

$$\vec{x}_T = \vec{x}_{T_0} - \mathbf{J}^{-1} \vec{f} \Big|_{(x_{T_0}, p_s)} - \mathbf{J}^{-1} \left( \frac{\partial \vec{f}}{\partial \vec{p}} \right)^T \delta \vec{p} \Big|_{(\vec{x}_{T_0}, \vec{p}_s)} \quad (2.334)$$

or

$$\vec{x}_T = \vec{x}'_{T_0} - \mathbf{J}^{-1} \left( \frac{\partial \vec{f}}{\partial \vec{p}} \right)^T \delta \vec{p} \Big|_{(\vec{x}_{T_0}, \vec{p}_s)} \quad (2.335)$$

yielding

$$\vec{e} = -\mathbf{J}^{-1} \left( \frac{\partial \vec{f}}{\partial \vec{p}} \right)^T \delta \vec{p} \Big|_{(\vec{x}_{T_0}, \vec{p}_s)} \quad (2.336)$$

Consider the case of errors in sensor positioning on PF estimation using triangulation with generalized bearings as discussed in Section 2.6. The geometry is shown in Figure 2.29. The system equations are given by

$$f_i(\vec{x}_T, \vec{p}_m) = 0 = \tan^{-1} \frac{\sqrt{(x_T - x_m)^2 + (z_T - z_m)^2}}{y_T - y_m} - \alpha_m \quad (2.337)$$

where  $\alpha_m$  is the measured generalized bearing from  $\vec{x}_T$  to  $\vec{x}_T$ . Now,

$$\begin{aligned} & f_i(x_{T_0} + \delta x, y_{T_0} + \delta y, z_{T_0} + \delta z, \vec{p}_m) \\ &= f_i(x_{T_0}, y_{T_0}, z_{T_0}, \vec{p}_m) + \left( \delta x \frac{\partial}{\partial x_T} + \delta y \frac{\partial}{\partial y_T} + \delta z \frac{\partial}{\partial z_T} \right) f_m(x_T, y_T, z_T, \vec{p}_m) \end{aligned} \quad (2.338)$$

In (2.338),

$$f_m(x_{T_0}, y_{T_0}, z_{T_0}, \vec{p}_m) = \tan^{-1} \frac{\sqrt{(x_{T_0} - x_m)^2 + (z_{T_0} - z_m)^2}}{y_{T_0} - y_m} - \alpha_m \quad (2.339)$$

$$\frac{\partial f_m}{\partial x_T} = \frac{(x_T - x_m)(y_T - y_m)}{(x_T - x_m)^2 + (y_T - y_m)^2 + (z_T - z_m)^2} \frac{1}{\sqrt{(x_T - x_m)^2 + (z_T - z_m)^2}} \quad (2.340)$$

$$\frac{\partial f_m}{\partial y_T} = -\frac{\sqrt{(x_T - x_m)^2 + (z_T - z_m)^2}}{\sqrt{(x_T - x_m)^2 + (y_T - y_m)^2 + (z_T - z_m)^2}} \quad (2.341)$$

$$\frac{\partial f_m}{\partial z_T} = \frac{(z_T - z_m)(y_T - y_m)}{(x_T - x_m)^2 + (y_T - y_m)^2 + (z_T - z_m)^2} \frac{1}{\sqrt{(x_T - x_m)^2 + (z_T - z_m)^2}} \quad (2.342)$$

which are all evaluated at  $x_T = x_{T_0}, y_T = y_{T_0}$ , and  $z_T = z_{T_0}$ .

The Jacobian matrix in this case is

$$\mathbf{J}|_{\vec{x}_{T_0}} = \frac{\partial \vec{f}}{\partial \vec{x}_T}|_{\vec{x}_{T_0}} \triangleq \begin{bmatrix} \left. \frac{\partial f_1}{\partial x_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=1}} & \left. \frac{\partial f_2}{\partial x_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=2}} & \left. \frac{\partial f_3}{\partial x_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=3}} \\ \left. \frac{\partial f_1}{\partial y_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=1}} & \left. \frac{\partial f_2}{\partial y_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=2}} & \left. \frac{\partial f_3}{\partial y_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=3}} \\ \left. \frac{\partial f_1}{\partial z_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=1}} & \left. \frac{\partial f_2}{\partial z_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=2}} & \left. \frac{\partial f_3}{\partial z_T} \right|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=3}} \end{bmatrix} \quad (2.343)$$

with  $\partial f_m / \partial x_T|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=k}}$  given by (2.340) with  $m = k$ ,  $\partial f_m / \partial y_T|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=k}}$  given by (2.341)

with  $m = k$ , and  $\partial f_m / \partial z_T|_{\substack{\vec{x}_T = \vec{x}_{T_0} \\ m=k}}$  given by (2.342) with  $i = k$ .

The derivative of  $\vec{f}$  with respect to  $\vec{p}$  is given by

$$\frac{\partial \vec{f}}{\partial \vec{p}} \triangleq \begin{bmatrix} \frac{\partial \vec{f}_1}{\partial \vec{p}_x} & \frac{\partial \vec{f}_2}{\partial \vec{p}_x} & \frac{\partial \vec{f}_3}{\partial \vec{p}_x} \\ \frac{\partial \vec{f}_1}{\partial \vec{p}_y} & \frac{\partial \vec{f}_2}{\partial \vec{p}_y} & \frac{\partial \vec{f}_3}{\partial \vec{p}_y} \\ \frac{\partial \vec{f}_1}{\partial \vec{p}_z} & \frac{\partial \vec{f}_2}{\partial \vec{p}_z} & \frac{\partial \vec{f}_3}{\partial \vec{p}_z} \end{bmatrix} \quad (2.344)$$

$$= \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & & \frac{\partial f_1}{\partial y_1} & & \frac{\partial f_1}{\partial z_1} & & \\ & \frac{\partial f_2}{\partial x_2} & & \frac{f_2}{\partial y_2} & & \frac{\partial f_2}{\partial z_2} & \\ & & \frac{\partial f_3}{\partial x_3} & & \frac{\partial f_3}{\partial y_3} & & \frac{f_3}{\partial z_3} \end{bmatrix}^T \quad (2.345)$$

with

$$\frac{\partial f_m}{\partial x_i} = \begin{cases} -\frac{(x_T - x_i)(y_T - y_i)}{(x_T - x_i)^2 + (y_T - y_i)^2 + (z_T - z_i)^2} \\ \quad \times \frac{1}{\sqrt{(x_T - x_i)^2 + (z_T - z_i)^2}}, & i = m \\ 0, & \text{otherwise} \end{cases} \quad (2.346)$$

$$\frac{\partial f_m}{\partial y_i} = \begin{cases} \frac{\sqrt{(x_T - x_i)^2 + (z_T - z_i)^2}}{(x_T - x_i)^2 + (y_T - y_i)^2 + (z_T - z_i)^2}, & i = m \\ 0, & \text{otherwise} \end{cases} \quad (2.347)$$

$$\frac{\partial f_m}{\partial z_i} = \begin{cases} -\frac{(y_T - y_i)(z_T - z_i)}{(x_T - x_i)^2 + (y_T - y_i)^2 + (z_T - z_i)^2} \\ \quad \times \frac{1}{\sqrt{(x_T - x_i)^2 + (z_T - z_i)^2}}, & i = m \\ 0, & \text{otherwise} \end{cases} \quad (2.348)$$

These derivatives are all evaluated at  $\vec{x}_T = \vec{x}_{T_0}$ . This process is iterated as long as necessary after updating the target location estimate with (2.335).

## 2.13 Concluding Remarks

The notion of geolocating emitting targets by triangulation was introduced in this chapter. Triangulation requires processing LOBs and determining the most likely target location based on where the LOBs intersect. When there is no noise or measurement error, this intersection is at a single point and the optimal PF is at

that point. Such is rarely the case, however, and noise and measurement error must be taken into account.

Triangulation is based on relatively simple trigonometric principles. Determination of the optimum PF, taking noise into consideration, complicates the calculations. Least-squared estimation techniques are quite common for calculating optimum locations, in a least-squares sense. Least-squares means that the error that is the square of the difference between the actual parameter being estimated and the estimate is minimized.

A novel PF calculation method based on the discrete probability density using the Tikhonov PDF was presented. This method deals with inaccurate data and tends to minimize the effects of highly inaccurate LOBs.

The Kalman filter realizations for PF determination are sequential methods for PF determination. They typically assume that the data is arriving at the sensor in a serial fashion, and the estimate of the PF is updated based on newly arriving data. The PF calculation problem is normally nonlinear in nature, and the standard Kalman filter does not apply. In that case, the extended Kalman filter can be used which linearizes the problem with a Taylor series expansion of the nonlinear equations.

The maximum likelihood methods attempt to find the optimal PF based on finding the location with maximizing probability. The a priori probabilities are assumed or measured, and the data is used to maximize those PDFs. Stansfield's method was perhaps the first of these techniques and is widely used for PF calculation. It is a biased technique, however, in that as the amount of data is increased, the covariance does not tend to zero.

There are many sources of error in triangulation. Some are due to the aforementioned noise and measurement error. Others are due to the geometry of the problem and can only be addressed by changing the geometry. GDOP is the primary cause for these types of errors.

A summary comparison of the methods introduced in this chapter is provided in Table 2.5. It should be noted that for most of these methods, special conditions may exist that lower the computational complexity. Indeed, it is the search for these special conditions that is the focus of much of the modern research in PF estimation.

**Table 2.5** Comparison of Triangulation Methods Discussed

Method	Biased	Covariance	Iterative	Complexity	Remarks
Least-squares — Brown	No	(2.49)	No	$\mathcal{O}(MN^2 \log N)$ [32]	Can be iterated and/or recursive
Least-squares — Hemispheric	No	(2.73)	Yes	$\mathcal{O}(MN^2 \log N)$ [32]	Can be iterated and/or recursive
Least-squares — Pages-Zamora	No	(2.93)	No	$\mathcal{O}(MN^2 \log N)$ [32]	Can be iterated and/or recursive
TLS	No	N/A	No	$\mathcal{O}(N^3)$ [33]	$\mathcal{O}(N)$ or $\mathcal{O}(N^2)$ using minor component analysis
MMSE	No	(2.118)	No	$\mathcal{O}(MN^2 \log N)$	
LMMSE	No	(2.153)	No	$\mathcal{O}(MN^2 \log N)$	
Kalman Filter	N/A	(2.192)	Yes	$\mathcal{O}(MN^2 \log N)$ [34]	
DPD	No	N/A	No	$\mathcal{O}(NL^2)$	Grid search required $L$ is the number of grid cells in each dimension
Generalized Bearings	N/A	N/A	Yes	$\mathcal{O}(MN^2 \log N)$	
MLE	No	(2.231)	Yes	N/A	
Stansfield	(2.268)	(2.269)	No	N/A	Widely used
Multiple Sample Correlation	N/A	N/A	Yes	N/A	Must know or estimate limits of LOB accuracy

All complexity estimates are per iteration when applicable.

$M$ : Number of sensors.

$N$ : Number of data points.

N/A: Not available.

In all cases zero mean AWGN is assumed.

## References

- [1] Stansfield, R. G., "Statistical Theory of DF Fixing," *Journal of IEE*, 14, Part IIIA, 15, 1947, pp. 762–770.
- [2] Jenkins, H. H., *Small Aperture Radio Direction Finding*, Norwood, MA: Artech House, 1991, p. 185.
- [3] Cadzow, J. A., "Signal Processing via Least-Squared Error Modeling," *IEEE ASSP Magazine*, October 1990, pp. 12–31.
- [4] Mendel, J. M., *Lessons in Estimation Theory for Signal Processing, Communications, and Control*, Englewood Cliffs, NJ: Prentice-Hall, 1995, pp. 115–117.
- [5] Golub, G. H., and C. F. Van Loan, "An Analysis of the Total Least-Squared Problem," *SIAM Journal on Numerical Analysis*, Vol. 17, 1980, pp. 883–892.
- [6] Brown, R. M., *Emitter Location Using Bearing Measurements from a Moving Platform*, NRL Report 8483, Naval Research Laboratory, Washington, D.C., June 1981.
- [7] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, Norwood, MA: Artech House, 2002, pp. 384–388.
- [8] Sage, A. P., and J. L. Melsa, *Estimation Theory with Applications to Communications and Control*, New York: McGraw-Hill, 1971, pp. 244–245.
- [9] Poirot, J. L., and M. S. Smith, "Moving Emitter Classification," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-12, No. 2, March 1976, pp. 255–269.
- [10] Poirot, J. L., and G. V. McWilliams, "Application of Linear Statistical Models to Radar Location Techniques," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-10, No. 6, November 1974, pp. 830–834.
- [11] Pages-Zamora, A., J. Vidal, and D. Brooks, "Closed-Form Solution for Positioning Based on Angle of Arrival Measurements," *PIMRC*, 2002.
- [12] Rao, K. D., and D. C. Reddy, "A New Method for Finding Electromagnetic Emitter Location," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 30, No. 4, October 1994, pp. 1081–1085.
- [13] Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME—Journal of Basic Engineering*, Vol. 82 (Series D), 1961, pp. 35–45.
- [14] Aidala, V. J., "Kalman Filter Behavior in Bearing-Only Tracking Applications," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 15, No. 1, January 1979, pp. 29–39.
- [15] Spingarn, K., "Passive Position Location Estimation Using the Extended Kalman Filter," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-23, No. 4, July 1987, pp. 558–567.
- [16] Julier, S. J., and J. K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defense Sensing, Simulation, and Controls*, 1997.
- [17] Elsaesser, D. S., and R. G. Brown, "The Discrete Probability Density Method for Emitter Geo-Location (U)," Defense Research & Development Canada, Ottawa Technical Memorandum 2003-068, June 2002.
- [18] Simon, M. K., and M. S. Alouini, *Digital Communication over Fading Channels*, New York: Wiley, 2000, p. 51.
- [19] Paradowski, L., "Unconventional Algorithm for Emitter Location in Three-Dimensional Space Using Data from Two-Dimensional Direction Finding," *Proceedings of the IEEE 1994 National Aerospace and Electronic Systems Conference, NAECON 1994*, May 23–27, 1994, pp. 246–250.

- [20] Levanon, N., "Interferometry against Differential Doppler: Performance Comparison of Two Emitter Location Airborne Systems," *IEE Proceedings*, Vol. 136, Pt. F., No. 2, April 1989, pp. 70–74.
- [21] Gavish, M., and A. J. Weiss, "Performance Analysis of Bearing-Only Target Location Algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 28, No. 3, July 1992, pp. 817–828.
- [22] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, Norwood, MA: Artech House, 2002, pp. 381–382.
- [23] Gething, P. J. D., *Radio Direction Finding and Resolution of Multicomponent Wave-Fields*, London: Peter Peregrinus, Ltd, 1978, pp. 271–275.
- [24] Fu, S. J., J. L. Vian, and D. L. Grose, "Determination of Ground Emitter Location," *IEEE Aerospace and Electronic Systems Magazine*, December 1988, pp. 15–18.
- [25] Nardone, S. C., A. G. Lindgren, and K. F. Gong, "Fundamental Properties and Performance of Conventional Bearings-Only Target Motion Analysis," *IEEE Transactions on Automatic Control*, Vol. AC-29, No. 9, September 1984, pp. 775–787.
- [26] Hammel, S. E., and V. J. Aidala, "Observability Requirements for Three-Dimensional Tracking via Angle Measurements," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-21, No. 2, March 1985, pp. 200–207.
- [27] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, Norwood, MA: Artech House, 2002, Chapter 11.
- [28] Torrieri, D. J., "Statistical Theory of Passive Location Systems," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-20, No. 2, March 1984, pp. 183–198.
- [29] Gavish, M., and E. Fogel, "Effect of Bias on Bearing-Only Target Location," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 26, No. 1, January 1990, pp. 22–26.
- [30] McCabe, H., and M. Al-Samara, "Application of a Sensor Fusion Algorithm to Nonlinear Data Containing Nonstationary, Multiplicative Noise of Unknown Distribution," *Proceedings of the 33rd Conference on Decision and Control*, Lake Buena Vista, FL, December 1994, pp. 1205–1206.
- [31] McCabe, H., "Minimum Trace Fusion of  $N$  Sensors with Arbitrarily Correlated Sensor-to-Sensor Fusion," *IFAC International Symposium on Distributed Intelligence Systems*, Arlington, VA, August 1991, pp. 229–234.
- [32] Manolakis, D. E., and M. E. Cox, "Effect in Range Difference Position Estimation Due to Stations' Position Errors," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 34, No. 1, January 1998, pp. 328–334.
- [33] Wang, Y., K. Ikeda, and K. Nakayama, "Computational Complexity Reduction of Predictor Based Least-squared Algorithm and Its Numerical Property," *Proceedings of the International Conference on Information, Communications, and Signal Processing, ICICS 1997*, Singapore, September 9–12, 1997, pp. 1203–1207.
- [34] Rao, Y. N., and J. C. Principe, "Efficient Total Least-squared Method for System Modeling Using Minor Component Analysis," *Proceedings of 2002 IEEE Workshop on Neural Networks for Signal Processing*, September 4–6, 2002, pp. 259–268.
- [35] Mendel, J. M., "Computational Requirements for a Discrete Kalman Filter," *IEEE Transactions on Automatic Control*, Vol. AC-16, No. 6, December 1971, pp. 748–758.

## Appendix 2A

### Least-Squares Error Estimation Program Listing

This appendix contains a source program listing in MathCAD® for the least-squared algorithm discussed in Section 2.2. This algorithm presented in Section 2.2 is intended to be a multipass, iterative procedure where the output of each pass is used in the subsequent pass to refine the PF calculation.

Initialize variables

$$\psi_T := 15 \quad i := 0..20$$

$$\lambda_T := 15$$

$$\lambda_{\text{hat}_0} := 13$$

$$\psi_{\text{hat}_0} := 13$$

$$\lambda_S(i) := i$$

$$\psi_S(t) := 0$$

$$\text{sigma} := 2 \cdot \frac{\pi}{180}$$

Calculate the bearing errors according to the normal distribution.

$$\text{BearingError} := \text{norm}(21, 0, \text{sigma})$$

Calculate the bearings to which the random errors will be added.

$$\phi_i := \text{acos} \left[ \frac{(\psi_T - \psi_S(i))}{\sqrt{(\psi_T - \psi_S(i))^2 + (\lambda_T - \lambda_S(i))^2}} \right]$$

	0
0	0.785
1	0.751
2	0.714
3	0.675
4	0.633
5	0.588
6	0.54
$\phi =$ 7	0.49
8	0.437
9	0.381
10	0.322
11	0.261
12	0.197
13	0.133
14	0.067
15	0

$\phi$ tilde is the vector of measured bearings.

$$\phi\text{tilde} := \phi + \text{BearingError}$$

Iteration

Initialize  $\lambda$ hat and  $m$ hat for this pass. With multiple iterations,  $\lambda$ hat.0 and  $\phi$ hat.0 are updated with the results from the previous pass.

$$\lambda\text{hat}_1 := \lambda\text{hat}_0 - \lambda_s(i)$$

Calculate the estimated bearing vector.

$$m\text{hat}_1 := \psi\text{hat}_0 - \psi_s(i)$$

$$\hat{\phi}_i := \arccos \left[ \frac{\hat{m}_i}{\sqrt{(\hat{m}_i)^2 + (\hat{n}_i)^2}} \right]$$

Calculate vector  $Y$ .

$$Y := \tilde{\phi} - \hat{\phi}$$

Determine the elements of two columns of  $X$ .

$$X_{t1_i} := \frac{\hat{n}_i}{(\hat{m}_i)^2 + (\hat{n}_i)^2}$$

$$X_{t2_i} := \frac{\hat{m}_i}{(\hat{m}_i)^2 + (\hat{n}_i)^2}$$

Convert these to vectors.

$$X_{i,0} := X_{t1_i}$$

$$X_{i,1} := X_{t2_i}$$

Calculate the error vector  $\hat{\beta}$ .

$$\hat{\beta} := (X^T \cdot X)^{-1} X^T \cdot Y$$

Calculate the estimated PF for this pass.

$$P_{T_0} := \hat{\psi}_0 - \hat{\beta}_0$$

$$P_{T_1} := \hat{\lambda}_0 - \hat{\beta}_1$$

$$P_T = \begin{pmatrix} 10.256 \\ 13.415 \end{pmatrix}$$

$$\hat{\psi}_0 := P_{T_0}$$

$$\hat{\lambda}_0 := P_{T_1}$$

Iterate

## Appendix 2B

### Generalized Bearing Program Listing

This appendix contains a source program listing in MathCAD® for the generalized bearing algorithm discussed in Section 2.7. This algorithm is intended to be a multipass, iterative procedure where the output of each pass is used in the subsequent pass to refine the PF calculation.

Initialize variables

$$m := 0$$

$$y_T := 0$$

$$i := 0..20$$

$$x_T := 10$$

$$\rho_{\theta e} := 0$$

$$\sigma_{\theta} := 2 \cdot \frac{\pi}{180}$$

$$\sigma_{\varepsilon} := 2 \cdot \frac{\pi}{180}$$

$$z_T := 0.5$$

Distances are in kilometers

Initial values of estimated location

$$\hat{x}_0 := 12$$

$$\hat{y}_0 := 5$$

$$\hat{z}_0 := 0.5$$

$$\hat{x}_{v0} := \begin{pmatrix} \hat{x}_0 \\ \hat{y}_0 \\ \hat{z}_0 \end{pmatrix}$$

Equations of sensor motion. An LOB is measured every unit of distance.

$$x_s(i) := i$$

$$y_s(i) := 0$$

$$z_s(i) := 3$$

Calculate the bearing errors according to the normal distribution.

$$\text{BearingError} := \text{rnorm}(21, 0, \sigma_\theta)$$

Calculate the generalized bearings to which the random errors will be added.

$$\alpha_i := \text{atan} \left[ \frac{\sqrt{(x_T - x_S(i))^2 + (z_T - z_S(i))^2}}{y_T - y_S(i)} \right]$$

The vector of observed bearings

$$\alpha_{\text{obs}} := \alpha + \text{BearingError}$$

Iteration

$$m := m + 1$$

Calculate the components of Gm.

$$r_m(i) := \sqrt{(\hat{x}_0 - x_S(i))^2 + (\hat{y}_0 - y_S(i))^2 + (\hat{z}_0 - z_S(i))^2}$$

$$s_m(i) := \sqrt{\left( \frac{r_m(i)}{\hat{y}_0 - y_S(i)} \right)^2 - 1}$$

$$g_{m_i,0} := \frac{-(\hat{x}_0 - x_S(i))}{(r_m(i))^2 \cdot s_m(i)} \quad g_{m_i,1} := \frac{(\hat{y}_0 - y_S(i))s_m(i)}{r_m(i)}$$

$$G_m := \begin{pmatrix} \xi_{m0,0} & \xi_{m0,1} & \xi_{m0,2} \\ \xi_{m1,0} & \xi_{m1,1} & \xi_{m1,2} \\ \xi_{m2,0} & \xi_{m2,1} & \xi_{m2,2} \\ \xi_{m3,0} & \xi_{m3,1} & \xi_{m3,2} \\ \xi_{m4,0} & \xi_{m4,1} & \xi_{m4,2} \\ \xi_{m5,0} & \xi_{m5,1} & \xi_{m5,2} \\ \xi_{m6,0} & \xi_{m6,1} & \xi_{m6,2} \\ \xi_{m7,0} & \xi_{m7,1} & \xi_{m7,2} \\ \xi_{m8,0} & \xi_{m8,1} & \xi_{m8,2} \\ \xi_{m9,0} & \xi_{m9,1} & \xi_{m9,2} \\ \xi_{m10,0} & \xi_{m10,1} & \xi_{m10,2} \\ \xi_{m11,0} & \xi_{m11,1} & \xi_{m11,2} \\ \xi_{m12,0} & \xi_{m12,1} & \xi_{m12,2} \\ \xi_{m13,0} & \xi_{m13,1} & \xi_{m13,2} \\ \xi_{m14,0} & \xi_{m14,1} & \xi_{m14,2} \\ \xi_{m15,0} & \xi_{m15,1} & \xi_{m15,2} \\ \xi_{m16,0} & \xi_{m16,1} & \xi_{m16,2} \\ \xi_{m17,0} & \xi_{m17,1} & \xi_{m17,2} \\ \xi_{m18,0} & \xi_{m18,1} & \xi_{m18,2} \\ \xi_{m19,0} & \xi_{m19,1} & \xi_{m19,2} \\ \xi_{m20,0} & \xi_{m20,1} & \xi_{m20,2} \end{pmatrix}$$

Calculate the correlation coefficients.

$$a_m(i) := \left[ \frac{-1 \cdot (\hat{x}_0 - x_s(i))}{s_m(i) \cdot (\hat{y}_0 - y_s(i))} \right]^2$$

$$b_m(i) := \left[ s_m(i) \cdot \sqrt{\left( \frac{r_m(i)}{\hat{z}_0 - z_s(i)} \right)^2 - 1} \right]^{-2}$$

$$c_m(i) := \sqrt{a_m(i) \cdot b_m(i)}$$

$$\sigma_m(i) := \sqrt{a_m(i) \cdot (\sigma_\theta)^2 + b_m(i) \cdot (\sigma_\epsilon)^2 + c_m(i) \cdot \rho_{\theta\epsilon} \cdot \sigma_\theta \cdot \sigma_\epsilon}$$

Convert  $\sigma$  to a vector.

$$h_i := \sigma_m(i)$$

$$C := \text{diag}(h)$$

$$\alpha_{\text{hat}_1} := \text{atan} \left[ \sqrt{\frac{(\hat{x}_0 - x_s(i))^2 + (\hat{z}_0 - z_s(i))^2}{\hat{y}_0 - y_s(i)}} \right]$$

$$w_m := \alpha_{\text{obs}} + G_m \cdot \hat{x}_{v0} - \alpha_{\text{hat}}$$

Calculate the updated location estimate.

$$\hat{x}_{\text{hat}_1} := (G_m^T C^{-1} G_m)^{-1} G_m^T C^{-1} w_m$$

$$\hat{x}_{\text{hat}_1} = \begin{pmatrix} 13.207 \\ 4.254 \\ 2.112 \end{pmatrix}$$

$$\hat{x}_{\text{hat}_0} := \hat{x}_{\text{hat}_1}_0$$

$$\hat{y}_{\text{hat}_0} := \hat{x}_{\text{hat}_1}_1$$

$$\hat{z}_{\text{hat}_0} := \hat{x}_{\text{hat}_1}_2$$

The above procedure is iterated until the difference in location estimate is small.

# Chapter 3

## DF Techniques

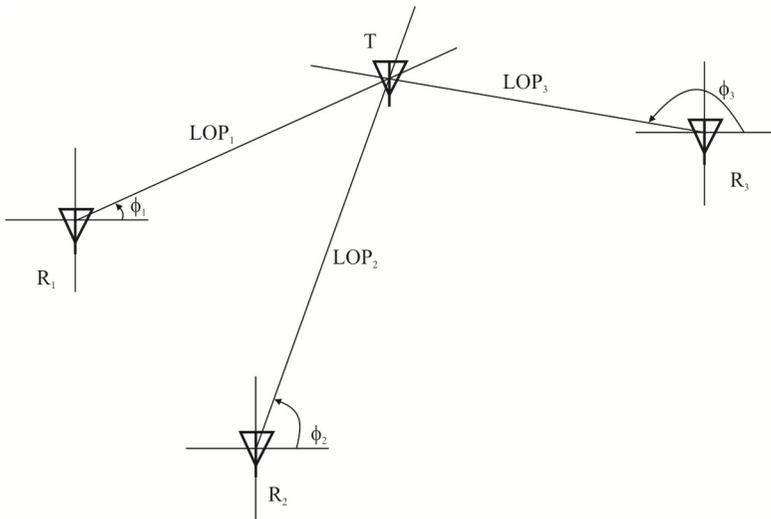
### 3.1 Introduction

In Chapter 2 we presented several approaches to computing an estimate of a target emitter based on estimates of the AOAs of the signal determined at two or more receiving sites. What we did not include there were any discussions on how those AOAs are determined in the first place. We address that issue in this chapter and the next. In this chapter we present several such methods, each with its own benefits and drawbacks. We devote Chapter 4 in its entirety to discussions of one of the most popular modern AOA determination algorithms called MUSIC.

As a reminder, Figure 3.1 illustrates the PF estimation using two or more DF sites (three are shown here). The  $\phi$ 's represent the AOAs of the signal at the various receive sites. These AOAs establish the LOPs, which, in turn, determine the PF. We discuss some methods of determining these LOPs in this chapter. For completeness, we also briefly include sections on phase interferometry, amplitude difference systems; monopulse systems based on the Butler matrix, which is a type of amplitude measurement system, using the array covariance to calculate the phase differences; beamforming; and maximum likelihood and least square error estimation processes. Many of the approaches in this last category of algorithms have been discussed at length elsewhere [1].

There is a considerable variety of approaches to calculation of the LOBs that have been developed. Each has its strong points and weak points. Those that we include here are not exhaustive; they are typical of some of the approaches that have been developed.

This chapter is structured as follows. We begin with a discussion of techniques that employ statistical methods for determining the AOA at an antenna array. We then briefly mention other, more hardware-oriented methods for making this estimation. Next we discuss a phase interferometer that employs MSE minimization to determine the AOA. DF with a Butler matrix is then discussed. The chapter concludes with a modern technique for estimating the phase



**Figure 3.1** Determining geoposition. Each of the antennas shown at the receiving sites is actually an array of antennas of some sort.

difference between two collocated antennas using especially configured SAW devices.

## 3.2 Array Processing Direction of Arrival Measurement Methods

### 3.2.1 Introduction

In this section, we present some of the more common algorithmic approaches for determining the AOAs of one or more signals impinging on an antenna array [2]. They all perform mathematical manipulations with the covariance matrix calculated from the samples of the signals. They differ in the specific steps taken to determine the AOAs.

These procedures differ from what could be referred to as hardware approaches to AOA determination. These methods usually rely on a particular antenna array configuration, such as the Adcock. Other examples are the phase interferometer and amplitude measuring and comparison techniques described in [3]. The Watson-Watt AOA determining technique is a further example of the hardware approaches. We include a summary of these techniques here, but do not discuss the details.

The AOAs thus determined can then be used in the PF algorithms described in Chapter 2 to estimate the point source locations of the targets from which the signals originate. In these algorithms it is assumed that techniques are available to properly associate which AOA corresponds to the AOAs measured from other receiving sites, when multiple AOAs are impinging on any one or more of the receiving systems. We take note that this may be a nontrivial task, but choose not to delve into that problem herein.

The approaches considered in this chapter are primarily associated with short baseline techniques. That is, time delays and corresponding phase differences at several antennas are all such that the phase differences are all on the order of  $\lambda/2$  or less. Baselines between antennas can be larger than this but there must be at least one baseline that meets the criteria in order to remove ambiguities in the measurements. Phase interferometry is one such technique.

In Chapter 5 we will discuss long baseline TDOA techniques. These baselines can be on the order of several kilometers. In such cases, phase unwrapping is normally necessary to remove ambiguities.

The coverage of array processing techniques in this chapter is limited by intent. For those wishing to delve more deeply into this fascinating topical area [4–7] are recommended.

### 3.2.2 The Model

We assume that one or more stochastic wavefields are generated by sources in the far field, denoted by  $s(t)$  [when there are more than one such signals then  $s(t)$  is a vector, denoted  $\vec{s}(t)$ ], are impinging on the antenna array. Random noise is assumed to accompany the signals, resulting in the signals at the antennas as

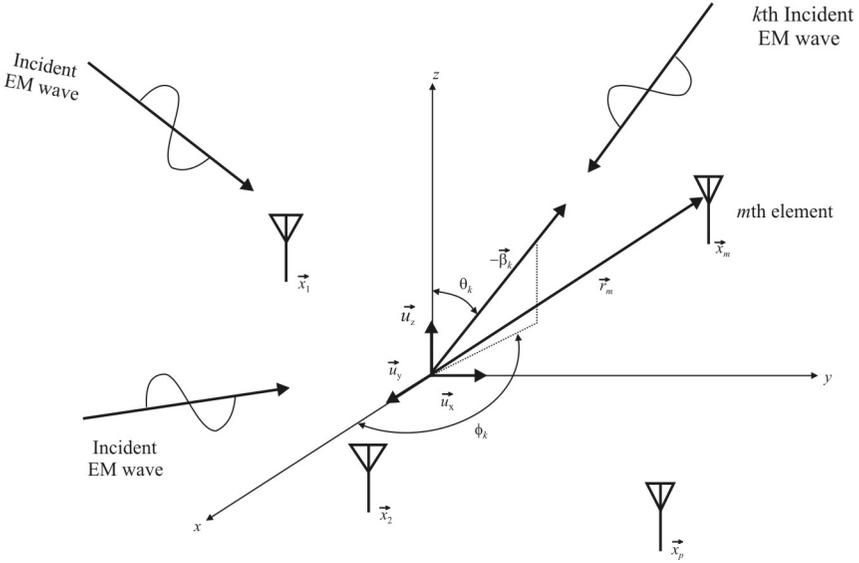
$$r(t, \vec{x}) = s(t, \vec{x}) + n(t) \quad (3.1)$$

where  $\vec{x}$  represents the vector of antenna locations, i.e.,  $\vec{x}_1 = [x_1 \ y_1 \ z_1]^T$ . Furthermore, we assume arbitrary antenna configurations. The antenna elements are assumed to have identical responses.

We consider only narrowband signals. A noiseless propagating signal can be represented by

$$s(t, \vec{r}) = A \cos[2\pi(f_0 t - \vec{\beta} \cdot \vec{x}) + \xi] \quad (3.2)$$

where  $A$  is its amplitude,  $f_0$  is the frequency, and  $\vec{\beta} = \vec{v}f_0 / |\vec{v}|^2$  is the wavenumber when  $\vec{v}$  is the three-dimensional velocity vector. Note that  $\vec{v}$  and therefore  $\vec{\beta}$  point in the direction of propagation. For example, in the  $x$ - $y$ -plane we have



**Figure 3.2** Arbitrary three-dimensional array.

$$\vec{\beta} = |\beta| [\cos(\angle \vec{v}) \quad \sin(\angle \vec{v})]^T \tag{3.3}$$

where  $\angle \vec{v}$  is the direction of propagation, defined counterclockwise relative the  $x$ -axis and  $\xi$  is a *random variable* (r.v.) uniformly distributed over  $(0, 2\pi)$ . We can also show that the magnitude of  $\vec{\beta}$  is given by  $|\beta| = 2\pi/\lambda$ .  $\vec{x} = [x \quad y \quad z]^T$  is an arbitrary spatial coordinate in the *three-dimensional* (3-D) space.

Suppose we have a set of  $M$  omnidirectional sensors located at the points  $\vec{x}_1, \vec{x}_2, \dots, \vec{x}_M$ , where  $\vec{x}_i = [x_i \quad y_i \quad z_i]^T$  in real 3-D space and that  $L$  narrowband plane waves impinge on the array (see Figure 3.2). The effect of these wavefields is to induce sensor signals which after conversion to baseband take the form of the  $M \times 1$  vector

$$\vec{r}(t) = \sum_{k=1}^M g_k(t) \vec{a}_k(\theta_k) + \vec{n}(t) \tag{3.4}$$

where the  $g_k(t)$  denote the envelope signals of the incident wavefields,  $\vec{n}(t)$  represents additive sensor noise, and the  $M \times 1$  steering vector associated with the  $k$ th incident wavefield is given by

$$\vec{a}_k(\vec{\theta}_k) = \begin{bmatrix} e^{-j\omega_c \tau_1(\phi_k)} \\ e^{-j\omega_c \tau_2(\phi_k)} \\ \vdots \\ e^{-j\omega_c \tau_M(\phi_k)} \end{bmatrix}, \quad 1 \leq k \leq L \quad (3.5)$$

where  $\omega_c$  is the carrier frequency. The steering vector parameters  $\tau_1(\phi_k), \dots, \tau_M(\phi_k)$  correspond to the delay times that appear across the array for the  $k$ th incident wavefield. These delay times depend on the array's geometry. Since we assume that the wavefields are planar, then these delay times are given by

$$\tau_m(\vec{\theta}) = \frac{1}{c} [x_m \cos(\phi) \cos(\theta) + y_m \sin(\phi) \cos(\theta) + z_m \sin(\theta)], \quad 1 \leq m \leq M \quad (3.6)$$

where  $\vec{x}_m = [x_m \ y_m \ z_m]^T$  corresponds to the location of the  $m$ th sensor and  $\phi$  and  $\theta$  designate the azimuth and elevation angles of the incident plane wave, respectively. The AOA parameters associated with the  $k$ th wavefield are designated by the AOA source parameter vector  $\vec{\theta}_k = [\phi_k \ \theta_k]^T$ .

Expressed in matrix/vector form, (3.4) becomes

$$\vec{r}(t) = \mathbf{A}(\phi) \vec{g}(t) + \vec{n}(t) \quad (3.7)$$

where  $\vec{g}(t)$  is the source signal vector whose components are given by  $g_k(t)$ . The  $M \times L$  composite steering matrix  $\mathbf{A}(\vec{\theta})$  has as its columns the steering vectors associated with the incident wavefields, that is,

$$\mathbf{A}(\vec{\theta}) = [\vec{a}_1(\vec{\theta}_1) \ \vec{a}_2(\vec{\theta}_2) \ \cdots \ \vec{a}_L(\vec{\theta}_L)] \quad (3.8)$$

This steering matrix is a function of the AOA parameter vector  $\vec{\theta}$  whose elements are composed of a concatenation of the  $m$  individual source AOA parameter vectors. We assume that  $\mathbf{A}(\vec{\theta})$  has full rank which is generally true in most multiple source applications that we are interested in for which  $L \leq M$ .

### 3.2.3 Array Covariance Modeling

We further assume that the source envelope signals and the additive noise are all mutually uncorrelated, wide-sense stationary stochastic processes. The snapshot vector then is also wide-sense stationary and its covariance matrix is given by

$$\begin{aligned} \mathbf{C}_{\pi} &= \mathcal{E}\{[\vec{r}(t) - \mathcal{E}\{\vec{r}(t)\}][\vec{r}(t) - \mathcal{E}\{\vec{r}(t)\}]^H\} \\ &= \mathcal{E}\{\vec{r}(t)\}\mathcal{E}\{\vec{r}^H(t)\} \end{aligned} \quad (3.9)$$

because we assume that the source signals  $r_k(t)$  and the sensor noise  $n(t)$  are zero mean [so that  $\mathcal{E}\{\vec{r}(t)\} = 0$  and  $\mathcal{E}\{\vec{n}(t)\} = 0$ ], *wide-sense stationary* (wss) stochastic processes. Since we assume that the source signals and the additive noise are mutually uncorrelated, we further get

$$\mathbf{C}_{\pi} = \mathbf{A}(\vec{\theta})\mathbf{C}_{\text{gg}}\mathbf{A}^H(\vec{\theta}) + \sigma^2\mathbf{C}_{\text{nn}} \quad (3.10)$$

where  $\mathcal{E}\{\cdot\}$  denotes the expected value operator and the <sup>H</sup> designates complex conjugate transpose. The  $L \times L$  source covariance matrix is given by  $\mathbf{C}_{\text{gg}} = \mathcal{E}\{\vec{g}(t)\vec{g}^*(t)\}$  and the  $M \times M$  noise covariance matrix  $\sigma^2\mathbf{C}_{\text{nn}} = \mathcal{E}\{\vec{n}(t)\vec{n}^H(t)\}$ .  $\mathbf{C}_{\text{gg}}$  is diagonal for incoherent sources, nondiagonal and nonsingular for partially coherent sources, and nondiagonal and singular if a subset of the sources is perfectly coherent. Finally,  $\mathbf{C}_{\text{nn}}$  is assumed to be known, but the noise power level  $\sigma^2$  is not.

It is possible to solve the fundamental AOA problem in the covariance domain. The concept of generalized eigenanalysis plays a prominent role in so doing. The (generalized) eigenanalysis of the matrix pair  $(\mathbf{C}_{\pi}, \mathbf{C}_{\text{nn}})$  is given by

$$\mathbf{C}_{\pi}\vec{e}_m = \lambda_m\mathbf{C}_{\text{nn}}\vec{e}_m, \quad 1 \leq m \leq M \quad (3.11)$$

The  $\lambda_m$  scalars are called (*generalized*) *eigenvalues*. If  $\vec{e}_m$  obeys (3.11), with some  $\lambda$ , then we call the  $M \times 1$  vector  $\vec{e}_m$  the *generalized eigenvector* of  $\mathbf{C}_{\pi}$  and  $\mathbf{C}_{\text{nn}}$ , and  $\lambda$  is called the generalized eigenvalue of  $\mathbf{C}_{\pi}$  and  $\mathbf{C}_{\text{nn}}$  which corresponds to the generalized eigenvector  $\vec{e}_m$ . The possible values of  $\lambda$  must obey the following<sup>1</sup>

<sup>1</sup> On a related side note, the set of matrices of the form  $\mathbf{A} - \lambda\mathbf{B}$ , where  $\lambda$  is a complex number, is called a *pencil*; the term *matrix pencil* can also refer to the pair  $(\mathbf{A}, \mathbf{B})$  of matrices. If  $\mathbf{B}$  is invertible, then the original problem can be written in the form

$$\mathbf{B}^{-1}\mathbf{A}\vec{v} = \lambda\vec{v}$$

which is a standard eigenvalue problem. However, in most situations it is preferable not to perform the inversion, but rather to solve the generalized eigenvalue problem as stated originally. This is especially

$$\det(\mathbf{C}_{rr} - \lambda \mathbf{C}_{nn}) = 0 \quad (3.12)$$

The following property provides a basis for employing the covariance domain description to estimate the AOA source parameters of the incident wavefields [8].

### Property 3.1

Let the snapshot vector be given by (3.7). Furthermore, let the corresponding  $M \times L$  steering matrix  $\mathbf{S}(\vec{\theta})$  have full rank  $L$  and the  $L \times L$  source covariance matrix  $\mathbf{C}_{gg}$  have rank  $r$  where  $r \leq L$ . Then the eigenvalues as specified by (3.11) have the property that

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_r > \sigma^2$$

and

$$\lambda_{r+1} = \lambda_{r+2} = \dots = \lambda_M = \sigma^2$$

The eigenvalue  $\sigma^2$  of multiplicity  $M - r$  is referred to as the *noise eigenvalue* while the  $r$  larger eigenvalues are called *signal eigenvalues*. With this eigenvalue classification, the associated generalized eigenvectors are characterized as follows.

*Noise Eigenvectors:* The eigenvectors associated with the noise eigenvalue  $\sigma^2$  are all contained in the null space of matrix  $\mathbf{C}_{gg} \mathbf{A}^H(\vec{\theta})$ , that is

$$\mathbf{C}_{gg} \mathbf{A}^H(\vec{\theta}) \vec{e}_k = \vec{0}, \quad k = r+1, r+2, \dots, M$$

Furthermore, if the sources are only partially coherent or incoherent so that  $r = L$ , the noise eigenvectors are orthogonal to

---

important if  $\mathbf{A}$  and  $\mathbf{B}$  are Hermitian matrices, since in this case  $\mathbf{B}^{-1} \mathbf{A}$  is not generally Hermitian and important properties of the solution are no longer apparent. If  $\mathbf{A}$  and  $\mathbf{B}$  are Hermitian and  $\mathbf{B}$  is a positive-definite matrix, the eigenvalues  $\lambda$  are real and eigenvectors  $\vec{v}_1$  and  $\vec{v}_2$  with distinct eigenvalues are  $\mathbf{B}$ -orthogonal ( $\vec{v}_1^* \mathbf{B} \vec{v}_2 = 0$ ). Also, in this case it is guaranteed that there exists a basis of generalized eigenvectors. This case is sometimes called a *Hermitian definite pencil* or *definite pencil*.

each of the steering vectors associated with the incident plane waves, that is

$$\mathbf{A}^H(\vec{\theta}_k)\vec{e}_m = 0, \quad k = r+1, r+2, \dots, M$$

If a subset of the plane waves is fully coherent, this orthogonality condition is generally invalid.

*Signal Eigenvectors:* The eigenvectors associated with the signal eigenvalues when premultiplied by  $\mathbf{C}_{nn}$  are each contained in the range space of the steering matrix, namely

$$\mathbf{C}_{nn}\vec{e}_m = \mathbf{A}(\vec{\theta})\vec{a}_m, \quad 1 \leq m \leq r$$

where  $\vec{a}_n$  are  $L \times 1$  vectors. The vectors  $\mathbf{C}_{nn}\vec{e}_m$  form a basis for the range space of  $\mathbf{A}(\vec{\theta})$  if and only if there are no coherent sources present.

### 3.2.4 Direction of Arrival

We use snapshot data to estimate the individual-source AOA parameter vectors that characterize the multiple wavefields incident on the array. The goal is to determine the AOA of these wavefields on the array. These AOAs are estimates of the direction to the point sources in the far field of the antenna array from which they originate.

Several direct methods have been developed for directly using samples of the raw snapshot data as specified by

$$\vec{r}(t_1), \vec{r}(t_2), \dots, \vec{r}(t_N) \quad (3.13)$$

to estimate the source AOA parameters. The sampling scheme is typically uniform. Some of these direct methods are classical beamforming algorithms [9], the ML method, and LSE modeling techniques. Several indirect methods of AOA estimation have been developed as well that depend on the eigencharacteristics of the array covariance matrix characterized by Property 3.1. These include noise subspace-based methods as exemplified by MUSIC that we discuss in the next chapter and other similar signal subspace-based methods.

In these eigen-based methods, the array covariance matrix must first be estimated from the sampled snapshot vectors. The standard estimate for the array covariance matrix is given by

$$\hat{\mathbf{C}}_{rr} = \frac{1}{N} \sum_{n=1}^N r(t_n) r^H(t_n) \quad (3.14)$$

where  $N$  is the number of snapshots. Indirect AOA algorithms use the eigencharacteristics of an array covariance matrix estimate such as this one to obtain the required AOA parameter vector estimate.

### 3.2.5 Subspace-Based Methods

The subspace-based methods of array processing first determine the signal and noise subspaces associated with the covariance matrix.<sup>2</sup> Manipulation using the properties of these subspaces is the unifying theme of these methods.

The subspace methods essentially form a beam in the “look direction” and rotate that beam to try to find from which direction the most energy is coming. What differentiates the methods is the way this beam is formed and therefore their resolution, accuracy, and bias.

The class of noise subspace methods are those that use the eigenvectors associated with the noise level eigenvalues. On the other hand, signals subspace methods employ the eigenvectors corresponding to the signal level eigenvalues. We will consider only the former of these approaches here.

#### 3.2.5.1 Eigenvector Decomposition

All of the subspace-based methods are based on the eigenvector decomposition of the covariance matrix

$$\mathbf{C}_{rr} = \mathcal{E}\{\vec{r}(t)\vec{r}^H(t)\} \quad (3.15)$$

Dropping the index of the steering matrix  $\mathbf{A}(\vec{\theta})$ , we get the covariance matrix

$$\mathbf{C}_{rr} = \mathbf{A} \mathcal{E}\{\vec{s}(t)\vec{s}^H(t)\} \mathbf{A}^H + \mathcal{E}\{\vec{n}(t)\vec{n}^H(t)\} \quad (3.16)$$

---

<sup>2</sup> The subspaces referred to here are mathematical constructs called *vector spaces*. A vector space  $V$  over a field  $F$  is a set of elements called vectors, such that any two vectors  $\alpha$  and  $\beta$  of  $V$  determine a (unique) vector  $\alpha + \beta$  as sum, and that any vector  $\alpha$  from  $V$  and any scalar  $c$  from  $F$  determine a scalar product  $c \cdot \alpha$  in  $V$ , with the properties

- $V$  is an Abelian group under addition,
- $c \cdot (\alpha + \beta) = c \cdot \alpha + c \cdot \beta$ ,  $(c + c') \cdot \alpha = c \cdot \alpha + c' \cdot \alpha$  (Distributive Laws),
- $(cc') \cdot \alpha = c \cdot (c' \cdot \alpha)$ ,  $1 \cdot \alpha = \alpha$ .

Denote the covariance matrix of  $s(t)$  as  $\mathcal{E}\{\vec{s}(t)\vec{s}^H(t)\} = \mathbf{C}_{ss}$ . Assuming the noise is i.i.d. Gaussian, the covariance matrix of  $\vec{n}(t)$  is  $\mathbf{C}_{nn} = \sigma^2\mathbf{I}$ . Therefore,  $\mathbf{C}_{rr}$  can now be written

$$\mathbf{C}_{rr} = \mathbf{A}\mathbf{C}_{ss}\mathbf{A}^H + \sigma^2\mathbf{I} \quad (3.17)$$

Since  $\mathbf{C}_{rr}$  is a positive definite, Hermitian matrix, we can use SVD to get

$$\mathbf{C}_{rr} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^H \quad (3.18)$$

with  $\mathbf{E}$  unitary<sup>3</sup> and  $\mathbf{\Lambda} = \text{diag}[\lambda_1 \ \lambda_2 \ \cdots \ \lambda_M]$ , a diagonal matrix of real eigenvalues ordered such that  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_M > 0$ .

If a vector  $\vec{r}$  is orthogonal to  $\mathbf{A}^H$  (so that  $\mathbf{A}^H\vec{r} = 0$ ), then it is an eigenvector of  $\mathbf{C}_{rr}$  with eigenvalue  $\sigma^2$ , because then

$$\mathbf{C}_{rr}\vec{r} = \mathbf{A}\mathbf{C}_{ss}\underbrace{\mathbf{A}^H\vec{r}}_{=0} + \sigma^2\vec{r} = \sigma^2\vec{r} \quad (3.19)$$

The eigenvector of  $\mathbf{C}_{rr}$  with eigenvalue  $\sigma^2$  lies in  $\mathcal{N}\{\mathbf{A}^H\}$ , the nullspace of  $\mathbf{A}^H$ . If and only if  $L < M$ , then

$$\mathcal{N}\{\mathbf{A}^H\} = \mathcal{R}\{\mathbf{Q}\}, \quad \mathbf{Q} \in \mathbb{C}^{M \times (M-L)}, \quad \text{rank}\{\mathbf{Q}\} = M - L \quad (3.20)$$

where  $\mathcal{R}\{\mathbf{Q}\}$  is the range of  $\mathbf{Q}$ .

Due to Property 3.1, it is possible to partition the eigenvectors into noise eigenvectors and signal eigenvectors and the covariance matrix  $\mathbf{C}_{rr}$  can be written as

$$\mathbf{C}_{rr} = \mathbf{E}_s\mathbf{\Lambda}_s\mathbf{E}_s^H + \mathbf{E}_n\mathbf{\Lambda}_n\mathbf{E}_n^H \quad (3.21)$$

The eigenvectors  $\mathbf{E} = [\mathbf{E}_s \ \mathbf{E}_n]$  form an orthonormal basis (i.e.,  $\mathbf{E}\mathbf{E}^H = \mathbf{E}^H\mathbf{E} = \mathbf{I}$ ).

The span of the  $L$  vectors  $\mathbf{E}_s$  defines the so-called signal subspace, and the

---

<sup>3</sup>An  $n \times n$  complex matrix  $\mathbf{U}$  is a *unitary matrix* if the condition

$$\mathbf{U}\mathbf{U}^H = \mathbf{U}^H\mathbf{U} = \mathbf{I}_n$$

is satisfied. This condition implies that  $\mathbf{U}^H = \mathbf{U}^{-1}$ .

orthogonal complement spanned by  $\mathbf{E}_n$  defines the noise subspace. This terminology is a consequence of the fact that  $\text{span}\{\mathbf{E}_s\} = \text{span}\{\mathbf{A}\} \perp \text{span}\{\mathbf{E}_n\}$ .

Furthermore, the range of  $\mathbf{Q}$  is the orthogonal complement to the range of  $\mathbf{A}$ , because

$$\mathcal{R}\{\mathbf{Q}\} = \mathcal{N}\{\mathbf{A}^H\} = \mathcal{R}^\perp\{\mathbf{A}\} \quad (3.22)$$

We therefore have

$$\mathcal{R}\{\mathbf{E}_s\} = \mathcal{R}\{\mathbf{A}\} \quad (3.23)$$

$$\mathcal{R}\{\mathbf{E}_n\} = \mathcal{R}^\perp\{\mathbf{A}^H\} \quad (3.24)$$

$\mathcal{R}\{\mathbf{E}_s\}$  is called the *signal subspace*, and  $\mathcal{R}\{\mathbf{E}_n\}$  is called the *noise subspace*. The projection operators onto these signal and noise subspaces are defined as

$$\mathbf{P}_A = \mathbf{A}\mathbf{A}^\dagger = \mathbf{E}_s(\mathbf{E}_s^H\mathbf{E}_s)^{-1}\mathbf{E}_s^H = \mathbf{E}_s\mathbf{E}_s^H \quad (3.25)$$

$$\mathbf{P}_S^\perp = \mathbf{I} - \mathbf{A}\mathbf{A}^\dagger = \mathbf{E}_n(\mathbf{E}_n^H\mathbf{E}_n)^{-1}\mathbf{E}_n^H = \mathbf{E}_n\mathbf{E}_n^H \quad (3.26)$$

where  $\mathbf{A}^\dagger$  is the pseudo-inverse<sup>4</sup> of  $\mathbf{A}$ .

### 3.2.6 Beamforming AOA Estimation

The snapshot vector (3.7) is composed of a time-varying linear combination of source steering vectors contaminated by noise. The  $k$ th steering vector is the response of the array to a signal arriving from direction specified by  $\vec{\theta}_k = [\phi_k \quad \theta_k]^T$ . Therefore, at any instant of time the snapshot vector generally has a nonzero component in each of the incident source steering vector directions. Beamforming techniques are based on fitting the sampled snapshot vector data by a single incident wavefront model. We can use the following squared error criterion for this fitting:

$$c_{\text{bf}}(\vec{\theta}_1, \{g_1(t_n)\}) = \sum_{n=1}^N [\vec{r}(t_n) - g_1(t_n)\vec{s}(\vec{\theta}_1)]^H [\vec{r}(t_n) - g_1(t_n)\vec{s}(\vec{\theta}_1)] \quad (3.27)$$

<sup>4</sup> When  $m \times n$  matrix  $\mathbf{F}$  is full rank, the Moore-Penrose pseudo-inverse is as follows:

$$m < n : \mathbf{F}^\dagger = \mathbf{F}^H(\mathbf{F}\mathbf{F}^H)^{-1}$$

$$m > n : \mathbf{F}^\dagger = (\mathbf{F}^H\mathbf{F})^{-1}\mathbf{F}^H$$

where  $g_1(t_n)$  and  $\vec{\theta}_1$  designates the sampled single source signal and the source AOA parameter vector, respectively. We want to find

$$\min_{\vec{\theta}_1, g_1(t_n)} c_{\text{bf}}(\vec{\theta}_1, \{g_1(t_n)\}) \quad (3.28)$$

When the minimum is sufficiently small, there is a high likelihood that an incident wavefront is present at that single source AOA parameter vector.

For any selection of this parameter vector, the corresponding optimum single source envelope signals are obtained by setting the derivatives of (3.27) with respect to the real and imaginary components of the single source signals to zero. This results in the optimum single source envelope sample selections

$$g_1^{\text{opt}}(t_n) = \frac{1}{M} \vec{s}^H(\vec{\theta}_1) \vec{r}(t_n), \quad 1 \leq n \leq N \quad (3.29)$$

Upon substitution of (3.29) into (3.27), we find that

$$c_{\text{bf}}(\vec{\theta}_1, \{g_1^{\text{opt}}(t_n)\}) = \sum_{n=1}^N \vec{r}^H(t_n) \vec{r}(t_n) - \frac{N}{M} \vec{s}^H(\vec{\theta}_1) \hat{\mathbf{C}}_{\text{r}} \vec{s}(\vec{\theta}_1) \quad (3.30)$$

where  $\hat{\mathbf{C}}_{\text{r}}$  is given by (3.14).

Since  $N$  and  $M$  are fixed, finding  $\vec{\theta}_1$  that minimizes (3.30) is equivalent to maximizing the beamforming function

$$bf(\vec{\theta}_1) = \vec{s}^H(\vec{\theta}_1) \hat{\mathbf{C}}_{\text{r}} \vec{s}(\vec{\theta}_1) \quad (3.31)$$

The nonlinear nature of (3.31) does not allow finding a closed form solution for  $bf(\vec{\theta}_1)$ . Instead of a closed form solution (3.31) is typically evaluated on a grid of relatively finely spaced values for  $\vec{\theta}_1$ . Values of  $\vec{\theta}_1$  that provide significant local maxima on this grid then serve as estimates of potential multiple incident sources. Interpolation is then used to refine these estimates. (We discuss interpolation later in Section 3.3.)

### 3.2.7 Maximum Likelihood AOA Estimation

When the additive sensor noises in (3.7) are Gaussian distributed, we have the LSE criterion whose minimization gives the MLE AOA estimate. We can show that the MLE estimate of the noise level eigenvalue in this case is given by

$$\hat{\sigma}^2 = \frac{1}{NM} \sum_{n=1}^N [\vec{r}(t_n) - \mathbf{A}(\vec{\theta})\vec{g}(t_n)]^H \mathbf{C}_{\text{nn}}^{-1} [\vec{r}(t_n) - \mathbf{A}(\vec{\theta})\vec{g}(t_n)] \quad (3.32)$$

The ML estimate of the  $\vec{\theta}$  is found by minimizing the ML criterion

$$c_{\text{ml}}(\vec{\theta}) = \sum_{n=1}^N [\vec{r}(t_n) - \mathbf{A}(\vec{\theta})\vec{g}(t_n)]^H \mathbf{C}_{\text{nn}}^{-1} [\vec{r}(t_n) - \mathbf{A}(\vec{\theta})\vec{g}(t_n)] \quad (3.33)$$

Again, a closed-form solution for a minimizing  $\vec{\theta}$  is generally unavailable because (3.33) is nonlinear. Although we could compute approximations of the minimum by direct evaluation as in beamforming, such an approach has a large computational cost due to the fact that the number of parameters used in this evaluation correspond to all the unknown parameters of the incident wavefields, a situation not encountered in beamforming, where a single wavefield was examined. Alternately we can use nonlinear programming techniques to iteratively find an optimum selection of  $\vec{\theta}^{\text{opt}}$ . We provide a description of this iterative approach in Section 3.2.9.

### 3.2.8 Least-Squares Error AOA Estimation

Snapshot vectors are used to estimate the individual source AOA parameters characterizing the multiple incident signals. The approach taken in many AOA algorithms is to select the AOA parameter vector so that the snapshot vector model is most compatible with these measurements in some sense. The following sum of squared errors criterion is often used for this purpose

$$c_{\text{lse}}(\vec{\theta}, \{g(t_n)\}) = \sum_{n=1}^N [\vec{r}(t_n) - \mathbf{A}(\vec{\theta})\vec{g}(t_n)]^H \mathbf{W} [\vec{r}(t_n) - \mathbf{A}(\vec{\theta})\vec{g}(t_n)] \quad (3.34)$$

where  $\mathbf{W}$  is a positive definite Hermitian matrix. We can see that the ML criterion is of this form in which  $\mathbf{W} = \mathbf{C}_{\text{nn}}^{-1}$ . This weighted criterion is a function of the AOA parameter vector and the sampled source signals  $\{\vec{g}(t_n)\}$ . We want to find values for these vector entities to render (3.34) a global minimum. Such a selection provides a multiple source model which is most compatible with the sensor data in the weighted LSE sense depicted by (3.34).

Again we encounter the difficulty of finding a closed-form solution for the minimum of (3.34) to yield parameters  $\vec{\theta}$  and  $\{\vec{g}(t_n)\}$ . As in the ML approach, it

is generally necessary to use nonlinear programming techniques to iteratively generate the optimum parameters.

### 3.2.9 Decoupling Sample Source Signals from AOA Parameters

The computational burden in using a nonlinear programming approach is proportional to the number of unknown parameters. It therefore makes sense to decrease this number whenever feasible. We will discuss such a technique in this section.

By examining the squared error criterion (3.34), we see that the source signals appear in a weighted quadratic fashion. We may therefore use standard calculus to determine the optimum choice for the source signal that is associated with any AOA parameter vector. If these optimum selections are then substituted back into the squared error criterion, the criterion becomes an exclusive function of the AOA parameter vector. Golub and Peryra first recognized this possibility in the unweighted case (i.e.,  $\mathbf{W} = \mathbf{I}$ ) and employed an AR decomposition to implement the decoupling operation [10]. This same objective may be reached in the weighted case by employing a generalized Gram-Schmidt orthogonalization procedure. The results of this approach are summarized in the following property [11].

#### Property 3.2

Let the  $M \times L$  steering matrix  $\mathbf{A}(\vec{\theta})$  have full rank  $L$ . This matrix can therefore be represented by the factorization  $\mathbf{A}(\vec{\theta}) = \mathbf{Q}(\vec{\theta})\mathbf{R}(\vec{\theta})$  in which  $\mathbf{Q}(\vec{\theta})$  is an  $M \times L$  orthogonal matrix so that  $\mathbf{Q}^*(\vec{\theta})\mathbf{W}\mathbf{Q}(\vec{\theta}) = \mathbf{I}$  and  $\mathbf{R}(\vec{\theta})$  is an  $L \times L$  nonsingular upper triangular matrix (this is the QR decomposition<sup>5</sup> [12]). Then the optimum source signals for any AOA parameter vector are given by

$$\vec{g}^{\text{opt}}(t_n) = \mathbf{R}^{-1}(\vec{\theta})\mathbf{Q}^{\text{H}}(\vec{\theta})\mathbf{W}\vec{r}(t_n), \quad 1 \leq n \leq N \quad (3.35)$$

---

<sup>5</sup>A QR decomposition (also called a QR factorization) of a matrix is a decomposition of a matrix  $\mathbf{A}$  into a product  $\mathbf{A} = \mathbf{Q}\mathbf{R}$  of an orthogonal matrix  $\mathbf{Q}$  and an upper triangular matrix  $\mathbf{R}$ . If  $\mathbf{A}$  has linearly independent columns (say,  $n$  columns), then the first  $n$  columns of  $\mathbf{Q}$  form an orthonormal basis for the column space of  $\mathbf{A}$ . More specifically, the first  $k$  columns of  $\mathbf{Q}$  form an orthonormal basis for the span of the first  $k$  columns of  $\mathbf{A}$  for any  $1 \leq k \leq n$ . The fact that any column  $k$  of  $\mathbf{A}$  only depends on the first  $k$  columns of  $\mathbf{Q}$  is responsible for the triangular form of  $\mathbf{R}$ .

The weighted squared error criterion (3.34) for these optimum source amplitudes is

$$c_{\text{lse}}(\vec{\theta}, \{\vec{g}^{\text{opt}}(t_n)\}) = \sum_{n=1}^N \vec{r}^{\text{H}}(t_n) \mathbf{W} \vec{r}(t_n) - \sum_{n=1}^N \vec{r}^*(t_n) \mathbf{W} \mathbf{Q}(\vec{\theta}) \mathbf{Q}^{\text{H}}(\vec{\theta}) \mathbf{W} \vec{r}(t_n) \quad (3.36)$$

The following procedure for solving the AOA problem ensues based on the above:

- (1) Find an AOA parameter vector that minimizes (3.36).
- (2) Determine the associated optimum sampled source signal using (3.35).

As previously, the first step of this procedure is the more difficult due to the nonlinear character of the process. It will therefore be computationally prudent to employ nonlinear programming techniques to obtain this solution. We will discuss such nonlinear programming methods in Section 3.2.11.

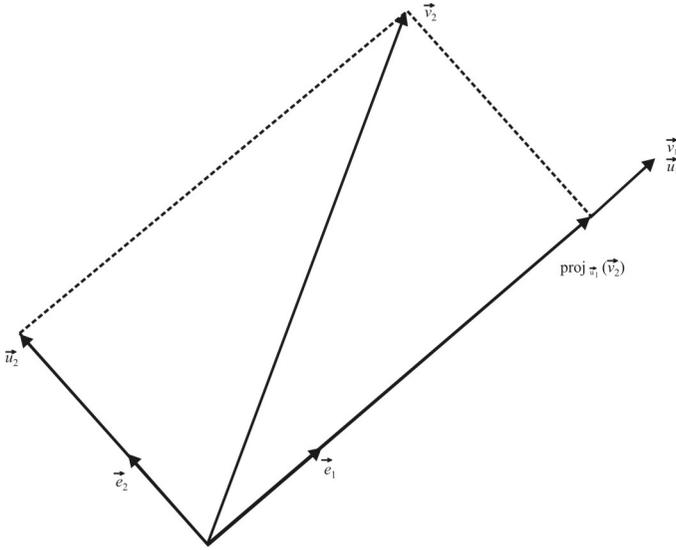
### 3.2.10 Gram-Schmidt Orthogonalization

Given two sets of vectors  $\{\vec{u}\}$  and  $\{\vec{v}\}$ , we define the projection operator by

$$\text{proj}_{\vec{u}}(\vec{v}) = \frac{\langle \vec{v}, \vec{u} \rangle}{\langle \vec{u}, \vec{u} \rangle} \vec{u} \quad (3.37)$$

where  $\langle \vec{u}, \vec{v} \rangle$  denotes the inner product of the vectors  $\vec{u}$  and  $\vec{v}$ . This operator projects the vector  $\vec{v}$  orthogonally onto the vector  $\vec{u}$ . The Gram-Schmidt process [13–17] then works as follows:

$$\begin{aligned} \vec{u}_1 &= \vec{v}_1, & \vec{e}_1 &= \frac{\vec{u}_1}{\|\vec{u}_1\|} \\ \vec{u}_2 &= \vec{v}_2 - \text{proj}_{\vec{u}_1}(\vec{v}_2), & \vec{e}_2 &= \frac{\vec{u}_2}{\|\vec{u}_2\|} \end{aligned}$$



**Figure 3.3** The first two steps of the Gram-Schmidt process.

$$\begin{aligned} \vec{u}_3 &= \vec{v}_3 - \text{proj}_{\vec{u}_1}(\vec{v}_3) - \text{proj}_{\vec{u}_2}(\vec{v}_3), & \vec{e}_3 &= \frac{\vec{u}_3}{\|\vec{u}_3\|} \\ & \vdots & & \\ \vec{u}_k &= \vec{v}_k - \sum_{j=1}^{k-1} \text{proj}_{\vec{u}_j}(\vec{v}_k), & \vec{e}_k &= \frac{\vec{u}_k}{\|\vec{u}_k\|} \end{aligned}$$

The first two steps of this process are illustrated in Figure 3.3. The sequence  $\vec{u}_1, \vec{u}_2, \dots, \vec{u}_k$  is the required system of orthogonal vectors, and the normalized vectors  $\vec{e}_1, \vec{e}_2, \dots, \vec{e}_k$  form an *orthonormal* set. The calculation of the sequence  $\vec{u}_1, \vec{u}_2, \dots, \vec{u}_k$  is known as *Gram-Schmidt orthogonalization*, while the calculation of the sequence  $\vec{e}_1, \vec{e}_2, \dots, \vec{e}_k$  is known as *Gram-Schmidt orthonormalization* as the vectors are normalized.

### 3.2.11 Nonlinear Programming

We now describe a nonlinear programming procedure for minimizing (3.34). We first seek an AOA parameter vector that minimizes (3.36), which is equivalent to maximizing the second term in (3.36) as the auxiliary function

$$d(\vec{\theta}) = \sum_{n=1}^N \vec{r}^H(t_n) \mathbf{W} \mathbf{Q}(\vec{\theta}) \mathbf{Q}^H(\vec{\theta}) \mathbf{W} \vec{r}(t_n) \quad (3.38)$$

Let  $\vec{\theta}^{\text{opt}}$  denote an optimum AOA parameter vector that maximizes this auxiliary function. Then the corresponding optimum source amplitude factors are obtained from (3.35).

These nonlinear programming techniques are based on incrementally perturbing the parameters being optimized in a systematic fashion so that the function being maximized takes on monotonically increasing values. The parameter sequences generated this way typically converge to a relative maximum of the function. In terms of the  $t \times 1$  AOA parameter vector [ $t = 2L$ , two entries (one azimuth angle and one elevation angle) for each incident wavefield], this incremental perturbation takes the form

$$\vec{\theta}^{(k+1)} = \vec{\theta}^{(k)} + \alpha_k \vec{p}^{(k)} \quad (3.39)$$

where  $\vec{\theta}^{(k)}$ ,  $\vec{p}^{(k)}$ , and  $\alpha_k$  denote the values of the AOA parameter vector, the perturbation vector, and the step size scalar, respectively, at the  $k$ th iteration. The various nonlinear programming algorithms differ, and therefore their characteristics (e.g., convergence rate) also differ, depending on how they select the perturbation vector and step size scalar. In all cases, an improvement in function value is obtained whenever the following improvement condition is met

$$d_m[\vec{\theta}^{(k+1)}] > d_m[\vec{\theta}^{(k)}] \quad (3.40)$$

Otherwise we are either on a nonincreasing slope and we are not converging.

The nonlinear programming algorithms used to solve these systems require the determination of the gradient vector and the Jacobian matrix of the function being maximized. For example, the perturbation vectors associated with the Gauss-Newton method is given by

$$\vec{p}_{\text{GN}}^{(k)} = [\text{Re}\{\mathbf{J}_d^H[\theta^{(k)}] \mathbf{J}_d[\theta^{(k)}]\}]^{-1} \nabla d_m[\vec{\theta}^{(k)}] \quad (3.41)$$

and for the Levenberg-Marquardt algorithm<sup>6</sup> by

$$\vec{p}_{\text{LM}}^{(k)} = [\text{Re}\{\mathbf{J}_d^H[\vec{\theta}^{(k)}] \mathbf{J}_d[\vec{\theta}^{(k)}] + \mu \mathbf{I}_p\}]^{-1} \nabla d_m[\vec{\theta}^{(k)}] \quad (3.42)$$

---

<sup>6</sup> See Appendix B for a brief description of these algorithms.

In these expressions, the gradient vector associated with the auxiliary function being maximized is denoted by  $\nabla d_m[\vec{\theta}^{(k)}]$  and the Jacobian matrix by  $\mathbf{J}_d[\vec{\theta}^{(k)}]$ . The  $NM \times t$  Jacobian matrix is given by

$$\mathbf{J}_d(\vec{\theta}) = \begin{bmatrix} \frac{\partial \mathbf{Q}(\vec{\theta})\mathbf{Q}^H(\vec{\theta})}{\partial \theta(1)} \vec{r}_1 & \cdots & \frac{\partial \mathbf{Q}(\vec{\theta})\mathbf{Q}^H(\vec{\theta})}{\partial \theta(t)} \vec{r}_1 \\ \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{Q}(\vec{\theta})\mathbf{Q}^H(\vec{\theta})}{\partial \theta(1)} \vec{r}_N & \cdots & \frac{\partial \mathbf{Q}(\vec{\theta})\mathbf{Q}^H(\vec{\theta})}{\partial \theta(t)} \vec{r}_N \end{bmatrix} \quad (3.43)$$

where  $\theta(k)$  are the azimuth and elevation angle entries in the  $t \times 1$  AOA parameter vector. Similarly, the  $t \times 1$  gradient vector is

$$\nabla_{\theta} d(\vec{\theta}) = \text{Re} \left\{ \mathbf{J}_d^*(\vec{\theta}) \begin{bmatrix} \mathbf{Q}(\vec{\theta})\mathbf{Q}^H(\vec{\theta})\vec{r}_1 \\ \mathbf{Q}(\vec{\theta})\mathbf{Q}^H(\vec{\theta})\vec{r}_2 \\ \vdots \\ \mathbf{Q}(\vec{\theta})\mathbf{Q}^H(\vec{\theta})\vec{r}_N \end{bmatrix} \right\} \quad (3.44)$$

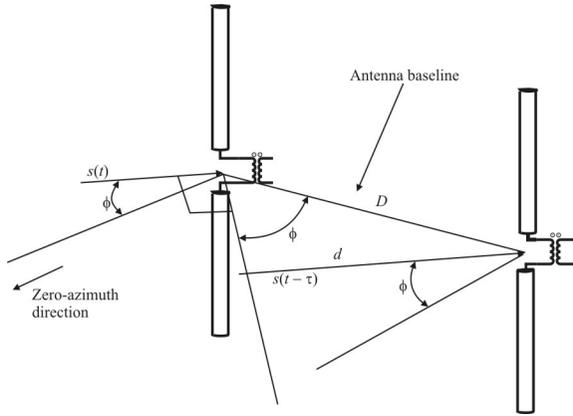
The Gauss-Newton method and the Levenberg-Marquardt algorithm for solving nonlinear least-squares problems are readily implemented using the above formulation of the Jacobian matrix and the gradient vector. The AOA parameter vector sequence generated by employing either of these algorithms converges in a rapid (e.g., quadratic) fashion.

### 3.3 Other Methods of Estimating the AOA

The next several techniques we will discuss are presented at length and analyzed in detail elsewhere [18–20]. We include a brief description of them here for completeness, but we will not duplicate the analyses provided there. The interested reader can get many more details from the references.

#### 3.3.1 Phase Interferometry

Phase interferometry uses the phase differences of a signal impinging on an array of antennas to determine an estimate of the AOA. The concept is illustrated in Figure 3.4 where a single baseline is formed by two antennas. In this case, signal  $s(t)$  impinges on the left antenna a time  $\tau$  before reaching the right antenna. These



**Figure 3.4** Interferometer (Source: [3]. © Artech House 2008. Reprinted with permission.)

time delays impose a phase shift between the antenna signals. The idea is to find the AOA  $\phi$  from measurements of these phase angle differences. Advantage is taken of the expression

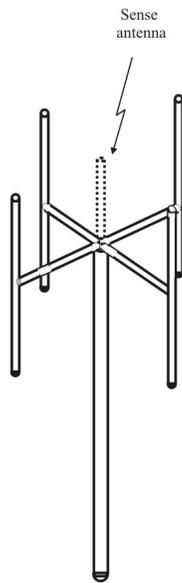
$$\tau = \frac{D}{c} \cos \phi \quad (3.45)$$

where  $c$  is the speed of propagation, to determine this AOA. With a single baseline there is a left-right ambiguity. This ambiguity is usually resolved by adding one or more antennas, forming at least one more baseline.

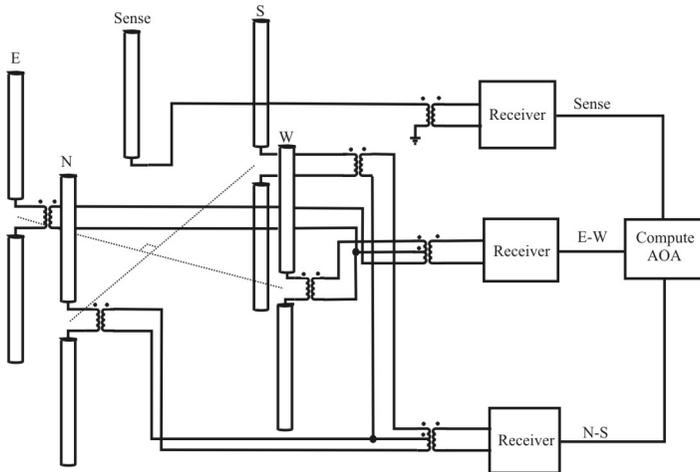
### 3.3.2 Amplitude Systems

Perhaps the most widely implemented DF technique over history has been the Watson-Watt amplitude DF systems with an Adcock antenna. One such four-element Adcock antenna is illustrated in Figure 3.5. This figure shows four dipoles mounted atop an antenna mast (monopoles or loops and several other antenna types are also possible). The sense antenna is rarely implemented as a separate antenna but is formed as the sum of the outputs of the other four antennas.

A simplified block diagram of this configuration is illustrated in Figure 3.6. The N and S antenna outputs are combined through an appropriately configured transformer as shown. So are the E and W antenna outputs. The two antenna patterns thus formed take the shape of orthogonal cardioids as shown in Figure 3.7. These two signals are fed to their own receivers where they are down-converted to baseband. The amplitudes of the two signals are compared. The ratio



**Figure 3.5** Four-element Adcock array. (Source: [3]. © Artech House 2008. Reprinted with permission.)



**Figure 3.6** Watson-Watt (Source: [3]. © Artech House 2008. Reprinted with permission.)

of the amplitudes provides an (ambiguous) indication of the AOA. The sense antenna is used to remove this ambiguity.

### 3.3.3 Doppler Direction Finder

As indicated in [21], the Doppler effect can be used to implement AOA estimators. Simply put, when a moving sensor is moving toward an emitting device, the frequency of the emission appears to be higher than when moving away. So, as long as the rotation rate is constant, the signal from a stationary antenna, when compared to the signal from an antenna that is rotating around the stationary antenna, will have frequency behavior that appears to be cyclic. See Figure 3.8. The amount of frequency shift is an indication of the AOA of the target signal.

## 3.4 MSE Phase Interferometer

### 3.4.1 Introduction

Direction finders for EW systems often employ compact circular arrays of dipoles or monopole antenna elements in the interest of mobile operation. One such system is the Adcock array shown in Figure 3.9. The N, E, S, W outputs from the four antenna elements are combined in such a way so as to provide an estimate of the AOA of a signal impinging on the array.

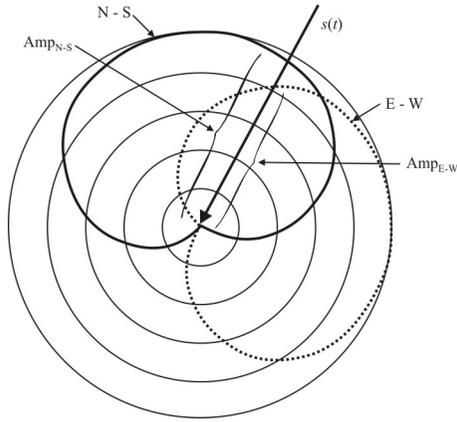


Figure 3.7 Cardioid.

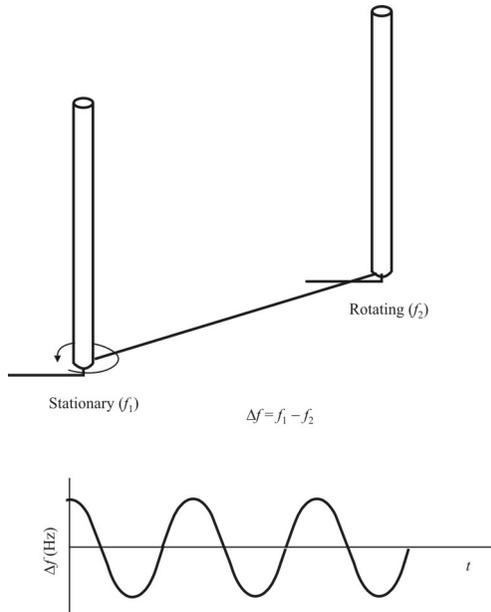


Figure 3.8 Doppler (Source: [3]. © Artech House 2008. Reprinted with permission.)

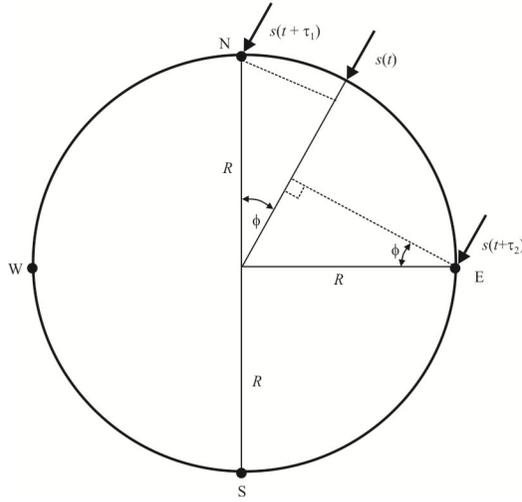


Figure 3.9 Adcock.

### 3.4.2 The Algorithm

Figure 3.9 defines the angle of arrival  $\phi$  of signal  $s(t)$  with respect to the array. Taking the center of the array as the reference, the output of the antennas are

$$\begin{aligned}
 N &= s(t + \tau_1) + n_1(t) = x_1(t) \\
 E &= s(t + \tau_2) + n_2(t) = x_2(t) \\
 S &= s(t - \tau_1) + n_3(t) = x_3(t) \\
 W &= s(t - \tau_2) + n_4(t) = x_4(t)
 \end{aligned}
 \tag{3.46}$$

where

$$\tau_1 = \frac{R \cos \phi}{c} \qquad \tau_2 = \frac{R \sin \phi}{c}
 \tag{3.47}$$

and  $R$  is the array radius and  $c$  the velocity of the signal. The additive noise  $n_i(t)$ ,  $i = 1, \dots, 4$  are stochastic processes mutually independent and independent of  $s(t)$ .

Sampling the outputs at  $f_s = 1/T$ , where without loss of generality we assume that  $T = 1$ , then the sampled outputs are for  $n = 0, 1, \dots, N - 1$ ,

$$\begin{aligned}
x_1(n) &= s(n + \tau_1) + n_1(n) \\
x_2(n) &= s(n + \tau_2) + n_2(n) \\
x_3(n) &= s(n - \tau_1) + n_3(n) \\
x_4(n) &= s(n - \tau_2) + n_4(n)
\end{aligned} \tag{3.48}$$

The noise samples  $n_i(t)$  are Gaussian, zero mean *band-limited white noise* (BLWN) of variance  $\sigma^2$ . The DFT of  $x_i(n)$  is

$$X_i(k) = \sum_{n=0}^{N-1} x_i(n) e^{-j\omega_k n} \tag{3.49}$$

where

$$\omega_k = \frac{2\pi k}{N} \tag{3.50}$$

Substituting (3.48) into (3.49) produces

$$\begin{aligned}
X_1(k) &= S(k) e^{j\omega_k \tau_1} + N_1(k) \\
X_2(k) &= S(k) e^{j\omega_k \tau_2} + N_2(k) \\
X_3(k) &= S(k) e^{-j\omega_k \tau_1} + N_3(k) \\
X_4(k) &= S(k) e^{-j\omega_k \tau_2} + N_4(k)
\end{aligned} \tag{3.51}$$

where

$$S(k) = \sum_{n=0}^{N-1} s(n) e^{-j\omega_k n} \tag{3.52}$$

is the DFT of  $s(n)$  and  $N_i(k)$  is the DFT of  $n_i(n)$ . It also follows from the time and phase shift relationship of the Fourier transform and (3.52) that

$$S(k) e^{\pm j\omega_k \tau_i} = \sum_{n=0}^{N-1} s(n \pm \tau_i) e^{-j\omega_k n}, \quad i = 1, 2 \tag{3.53}$$

$S(k)$  can be written in polar form as

$$S(k) = |S(k)| e^{j\tilde{\epsilon}_k} \tag{3.54}$$

where  $\xi_k$  is the phase of  $S(k)$ . The goal is to estimate  $\xi$  from the  $X_i(k)$ .

The process is an interferometer-type TDOA estimator that finds  $\tau_1$  and  $\tau_2$  by finding the LSE TDOA estimates between the N-S antenna outputs and the E-W antenna outputs. The N-S TDOA is  $2\tau_1$  and its estimate comes from an LSE fit of the phase of  $X_1(k)X_3^*(k)$  against frequency. Repeating the same procedure with the E-W output yields  $2\tau_2$ . Now from (3.47)

$$\tau_1^2 + \tau_2^2 = \frac{R^2}{c^2} \quad (3.55)$$

Incorporating (3.55) in the form of a constraint to the two LSE fits can improve the accuracy.

Consider first the LSE of  $\tau_1$  from  $X_1(k)$  and  $X_3(k)$ . The solution is

$$\hat{\tau}_1 = \frac{1}{2} \frac{\sum_{n=1}^{(N/2)-1} W_n \omega_n \varphi_n}{\sum_{n=1}^{(N/2)-1} W_n \omega_n^2} \quad (3.56)$$

where

$$\omega_n = \frac{2\pi n}{L} \quad (3.57)$$

$$W_n = \frac{|\hat{\gamma}(n)|^2}{1 - |\hat{\gamma}(n)|^2} \quad (3.58)$$

and

$$\varphi_k = \tan^{-1} \frac{\text{Im}\left\{\sum_{l=1}^{N_s} X_{1l}(k)X_{3l}^*(k)\right\}}{\text{Re}\left\{\sum_{l=1}^{N_s} X_{1l}(k)X_{3l}^*(k)\right\}} \quad (3.59)$$

is the cross-spectral phase between  $X_1(k)$  and  $X_3(k)$ . The magnitude squared coherence estimate is

$$|\hat{\gamma}(k)|^2 = \frac{\left|\sum_{l=1}^{N_s} X_{1l}(k)X_{3l}^*(k)\right|^2}{\sum_{l=1}^{N_s} |X_{1l}(k)| \sum_{l=1}^{N_s} |X_{3l}(k)|} \quad (3.60)$$

(not to be confused with the SNR,  $\gamma$ , used elsewhere herein). In (3.57) through (3.60), the spectral components  $X_{1l}(k)$  and  $X_{3l}(k)$  are the DFTs of the  $l$ th segments of  $x_1(n)$  and  $x_3(n)$ ,  $n = 0, 1, \dots, N - 1$ . These components come from breaking an  $N$ -point sequence into  $N_s$  segments of  $L$  points each, with a 50% overlap. Thus, if  $N = 1,024$ ,  $N_s = 7$  and  $L = 253$ .

Similarly, let  $\hat{\tau}_2$  be the TDOA obtained from  $x_3(n)$  and  $x_4(n)$ , with  $\tilde{\varphi}_k$  and  $\tilde{W}_k$  the corresponding phase and weight in (3.56).

To incorporate the constraint (3.55) with the joint estimation of  $\tau_1$  and  $\tau_2$ , we use the Lagrange multiplier technique [18]. Consider the cost function

$$J = [\mathbf{B}\bar{\tau} - \bar{b}]^T [\mathbf{B}\bar{\tau} + \bar{b}] + \ell \left[ \bar{\tau}^T \bar{\tau} - \frac{R^2}{c^2} \right] \quad (3.61)$$

where  $\ell$  is the Lagrange multiplier. The unknown vector we seek values for is

$$\bar{\tau} = [\hat{\tau}_1 \quad \hat{\tau}_2]^T \quad (3.62)$$

and

$$\mathbf{B} = \begin{bmatrix} \sqrt{W_1} \omega_1 & 0 \\ \vdots & \vdots \\ \sqrt{W_{(L/2)-1}} \omega_{(L/2)-1} & 0 \\ 0 & \sqrt{\tilde{W}_1} \tilde{\omega}_1 \\ \vdots & \vdots \\ 0 & \sqrt{\tilde{W}_{(L/2)-1}} \tilde{\omega}_{(L/2)-1} \end{bmatrix} \quad (3.63)$$

and

$$\bar{b} = \frac{1}{2} \begin{bmatrix} \sqrt{W_1} \Phi_1 \\ \vdots \\ \sqrt{W_{(L/2)-1}} \Phi_{(L/2)-1} \\ \cdots \\ \sqrt{\tilde{W}_1} \tilde{\Phi}_1 \\ \vdots \\ \sqrt{\tilde{W}_{(L/2)-1}} \tilde{\Phi}_{(L/2)-1} \end{bmatrix} \quad (3.64)$$

To find the  $\bar{\tau}$  that minimizes  $J$ , differentiate it with respect to  $\bar{\tau}$  and  $\ell$  and set the results to zero. This gives

$$\frac{\partial J}{\partial \bar{\tau}} = 2\mathbf{B}^T \mathbf{A} \bar{\tau} - 2\mathbf{B}^T \bar{b} + 2\ell \bar{\tau} = 0 \quad (3.65)$$

$$\frac{\partial J}{\partial \ell} = \bar{\tau}^T \bar{\tau} - \frac{R^2}{c^2} = 0 \quad (3.66)$$

or

$$[\mathbf{B}^T \mathbf{B} + \ell \mathbf{I}] \bar{\tau} = \mathbf{B}^T \bar{b} \quad (3.67)$$

Now

$$\mathbf{B}^T \mathbf{B} = \begin{bmatrix} \sum_{k=1}^{(L/2)-1} W_k \omega_k^2 & 0 \\ 0 & \sum_{k=1}^{(L/2)-1} \tilde{W}_k \omega_k^2 \end{bmatrix} \quad (3.68)$$

and

$$\mathbf{B}^T \bar{b} = \frac{1}{2} \begin{bmatrix} \sum_{k=1}^{(L/2)-1} W_k \omega_k \Phi_k \\ \sum_{k=1}^{(L/2)-1} \tilde{W}_k \omega_k \tilde{\Phi}_k \end{bmatrix} \quad (3.69)$$

Solving (3.65) yields

$$\begin{bmatrix} \hat{\tau}_1 \\ \hat{\tau}_2 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \frac{a_1}{b_1 + \ell} \\ \frac{a_2}{b_2 + \ell} \end{bmatrix} \quad (3.70)$$

where

$$\begin{aligned} a_1 &= \sum_{k=1}^{(L/2)-1} W_k \omega_k \varphi_k & b_1 &= \sum_{k=1}^{(L/2)-1} W_k \omega_k^2 \\ a_2 &= \sum_{k=1}^{(L/2)-1} \tilde{W}_k \omega_k \tilde{\varphi}_k & b_2 &= \sum_{k=1}^{(L/2)-1} \tilde{W}_k \omega_k^2 \end{aligned} \quad (3.71)$$

Putting  $\hat{\tau}_1$  and  $\hat{\tau}_2$  into constraint (3.55) yields

$$\frac{a_1^2}{(b_1 + \ell)^2} + \frac{a_2^2}{(b_2 + \ell)^2} = \frac{4R^2}{c^2} \quad (3.72)$$

which is a fourth-order equation in  $\ell$ . Let the root with the smallest magnitude be denoted as  $\ell^\diamond$ . Then

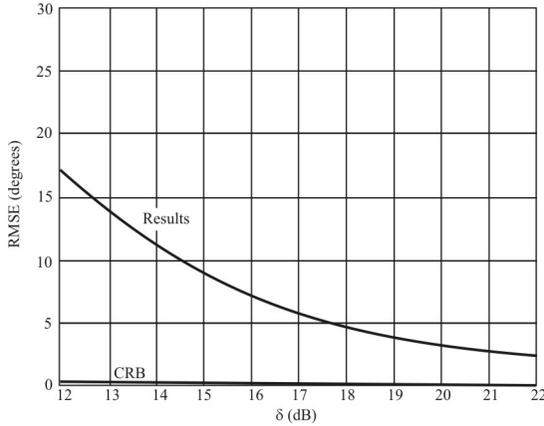
$$\hat{\tau} = \begin{bmatrix} \hat{\tau}_1 \\ \hat{\tau}_2 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \frac{a_1}{b_1 + \ell^\diamond} \\ \frac{a_2}{b_2 + \ell^\diamond} \end{bmatrix} = \begin{bmatrix} \frac{R}{c} \cos \hat{\phi} \\ \frac{R}{c} \sin \hat{\phi} \end{bmatrix} \quad (3.73)$$

and

$$\hat{\phi} = \tan^{-1} \frac{\hat{\tau}_2}{\hat{\tau}_1} \quad (3.74)$$

### 3.4.3 Simulation

This section describes two simulation experiments to evaluate the estimator described in the last section. The sampling frequency is  $f_s = 50$  MHz ( $T_s = 20$  ns) with the analog output signal digitized to form segments of  $N = 1,024$  samples. We break up  $N$  into 7, 50% overlapped segments of  $L = 256$  points each, before taking



**Figure 3.10** 15 MHz experiment.

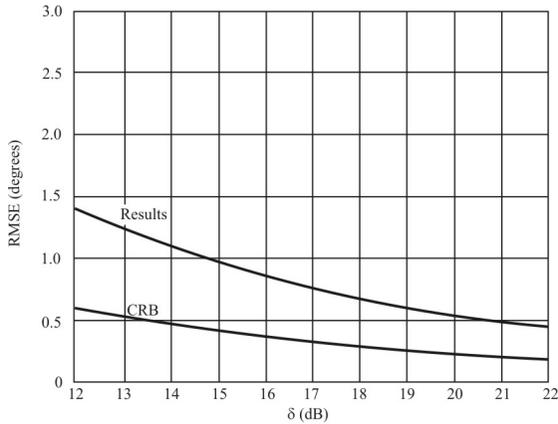
the Fourier transform. The independent noise sources are outputs of four separate Gaussian number generators of zero mean and variance  $\sigma^2$ . There were 250 independent trials to give the RMS error of the estimate

$$\text{RMSE} = \left[ \sum_{i=1}^{250} [\phi(i) - \phi]^2 / 250 \right]^{1/2} \quad (3.75)$$

where  $\phi(i)$  is the estimate of the  $i$ th trial. The array radius  $R = 2$  m and  $c = 2.99776 \times 10^8$  m s<sup>-1</sup>.

### *Results: Sinusoid Signals*

The cross-correlation function of two sinusoids exhibits multiple peaks, any one of which can easily be selected as the TDOA. This makes estimating the TDOA difficult. The signal in this experiment is at 15 MHz, which is an off-bin frequency. The results are shown in Figure 3.10 where  $\delta(\text{dB}) = \text{SNR}(\text{dB})$ . The loss of signal power in the gaps between bins reduces the effective SNR [23], and thus a higher error ensues than otherwise. The CRB is independent of the relationship between  $f_s$  and the frequency of the sinusoid, since it assumes analog processing. Zero filling a time sequence prior to taking the DFT can produce more frequency bins for processing. While this may ostensibly recover some of the off-bin energy of the sinusoid, the noise components between bins becomes correlated. As a result, there is no improvement in the TDOA estimate by zero filling.



**Figure 3.11** BLWN experiment.

### *Results: Stochastic Signals*

The signal is BLWN of variance  $\sigma_s^2$  and  $\text{SNR}(\text{dB}) = \delta(\text{dB}) = 10 \log_{10}(\sigma_s^2 / \sigma_n^2)$ . Figure 3.11 shows the results. In this case significantly better results are produced. The BLWN has a flat frequency spectrum within its bandwidth, which produces an approximate delta function TDOA estimate in the time domain.

## 3.5 DF with a Butler Matrix

### 3.5.1 Introduction

The  $N \times N$  Butler matrix consists of passive four-port hybrid power dividers and fixed phase shifters [24]. It has  $N$  input ports and  $N$  output ports. As a beamforming network, it is used to drive an array of  $N$  antenna elements. A beamformer is a reciprocal device so “drive” in this case means that they can be used to form beams for both transmitting and receiving. It can produce  $N$  orthogonally space beams overlapping at the  $-3.9$  dB level and having the full gain of the array. The network is used to produce (possibly 3-D, depending on the array geometry) beams in a pincushion deployment, where to be orthogonal  $(\sin x)/x$  patterns must be spaced so that the crossover is at about 4 dB down  $(2/\pi)$ , and the first sidelobes will be down 13.2 dB. Tapering changes these numbers; for a cosine taper, for example, the crossover level becomes approximately 9.5 dB for orthogonality.

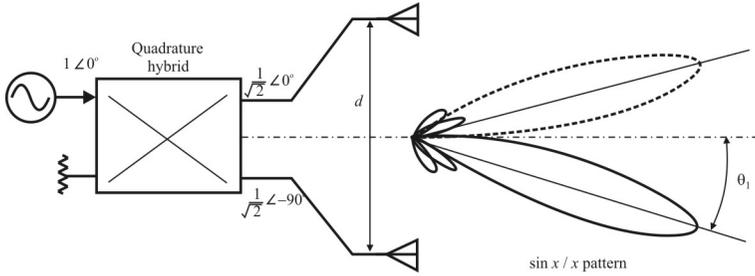


Figure 3.12 A two-beam Butler matrix (90°).

### 3.5.2 Beamforming Network

A generic version of the Butler matrix used as a beamforming network is shown in Figure 3.12. It consists of a 3 dB quadrature hybrid driving two antenna elements with separation  $d$ . Note the amplitude and phase relationships for the hybrid structure.

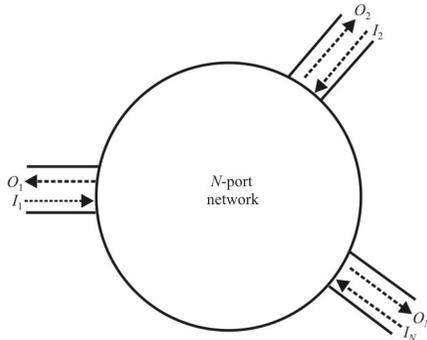
When used in a receiving role, the AOA of a signal impinging on the array causes variable voltage levels out of the input ports, the levels of which are determined by that AOA. By measuring and comparing these voltage levels, the angle of arrival of the signal can be estimated. The biggest single advantage of using a Butler matrix in this mode is its speed of operation. There are no servo loops that must settle. The estimated AOA appears immediately and continuously at the matrix output. This has advantages when the target set of concern includes short duration signals such as frequency-hopping push-to-talk communication networks.

#### 3.5.2.1 Scattering Matrix of the Quadrature Hybrid

For the  $N$ -port network shown in Figure 3.13, we have the inputs and outputs related through the scattering matrix  $\mathbf{S}$  as

$$\begin{bmatrix} O_1 \\ \vdots \\ O_N \end{bmatrix} = \begin{bmatrix} s_{11} & \cdots & s_{1N} \\ \vdots & \ddots & \vdots \\ s_{N1} & \cdots & s_{NN} \end{bmatrix} \begin{bmatrix} I_1 \\ \vdots \\ I_N \end{bmatrix} \tag{3.76}$$

The scattering matrix for a four-port network is



**Figure 3.13** *N*-port network.

$$\mathbf{S} = \begin{bmatrix} s_{11} & s_{12} & s_{13} & s_{14} \\ s_{21} & s_{22} & s_{23} & s_{24} \\ s_{31} & s_{32} & s_{33} & s_{34} \\ s_{41} & s_{42} & s_{43} & s_{44} \end{bmatrix} \quad (3.77)$$

In a theoretical ideal device, certain ports are isolated from each other. For example, in the quadrature hybrid, port 1 is isolated from port 2, and port 3 from 4. Therefore, these elements in the matrix are

$$s_{12} = s_{21} = s_{34} = s_{43} = 0$$

If all the ports are matched there are no reflections (VSWR = 1 or  $\Gamma = 0$  dB), and since the diagonal elements are reflection coefficients, they equal zero

$$s_{11} = s_{22} = s_{33} = s_{44} = 0$$

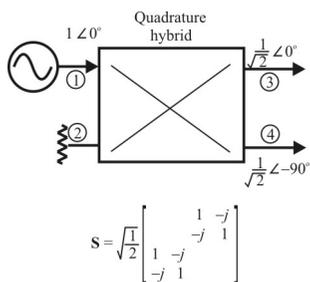
Off-diagonal elements are the transmission coefficients. The hybrid is reciprocal making the elements of the matrix symmetrical about the diagonal. Therefore

$$s_{12} = s_{21}, s_{23} = s_{32}, s_{41} = s_{14}$$

$$s_{42} = s_{24}, s_{43} = s_{34}, s_{23} = s_{32}$$

The quadrature hybrid and the scattering matrix shown in Figure 3.14 reflect the comments made above.

If we perform the matrix operation, we get



**Figure 3.14** Quadrature hybrid for producing equal signals, but with  $90^\circ$  phase differential.

$$\underbrace{\begin{bmatrix} O_1 \\ O_2 \\ O_3 \\ O_4 \end{bmatrix}}_{\text{Outputs}} = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 0 & 1 & -j \\ 0 & 0 & -j & 1 \\ 1 & -j & 0 & 0 \\ -j & 1 & 0 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix}}_{\text{Inputs}} \quad (3.78)$$

Therefore,

$$O_1 = \frac{1}{\sqrt{2}}(I_3 - jI_4) \quad (3.79)$$

$$O_2 = \frac{1}{\sqrt{2}}(-jI_3 + I_4) \quad (3.80)$$

$$O_3 = \frac{1}{\sqrt{2}}(I_1 - jI_2) \quad (3.81)$$

$$O_4 = \frac{1}{\sqrt{2}}(-jI_1 + I_2) \quad (3.82)$$

but

$$I_2 = I_3 = I_4 = 0$$

therefore

$$O_1 = 0$$

$$O_2 = 0$$

$$O_3 = \frac{I_1}{\sqrt{2}} = \frac{I_1}{\sqrt{2}} \angle 0^\circ$$

$$O_4 = \frac{-jI_1}{\sqrt{2}} = \frac{I_1}{\sqrt{2}} \angle -90^\circ = \frac{I_1}{\sqrt{2}} \exp\left(-j\frac{\pi}{2}\right)$$

that is, two outputs from ports 3 and 4 but in phase quadrature.

### Example

Feeding port 1 in the Butler matrix in Figure 3.12 results in the antenna array being uniformly illuminated and differential phased to point the resulting beam peak to the right of boresight. With the phasing indicated, the relationship becomes

$$\frac{2\pi d}{\lambda} \sin \theta_1 - \frac{\pi}{2} = 0$$

or

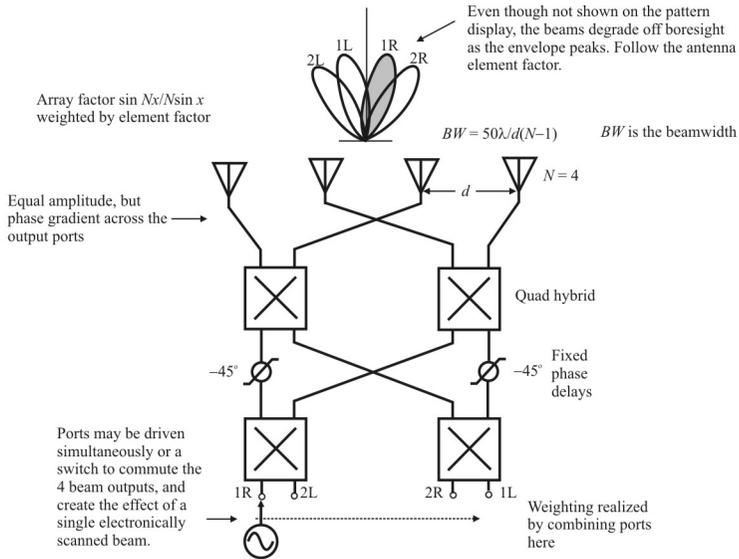
$$\sin \theta_1 = \frac{\lambda}{4d}$$

Feeding port 2 results in a beam pointing to the left of boresight at an angle

$$\theta_2 = \sin^{-1}\left(-\frac{\lambda}{4d}\right)$$

If both ports 1 and 2 are driven, two beams will be produced at angles  $\theta = \pm\lambda/4d$ . The matrix produces  $N$  ( $= 2$  here) orthogonally spaced beams overlapping at the  $-3.9$  level.

A four-beam version is shown in Figure 3.15. This consists of several quadrature hybrids and fixed phase shifters. As we trace a phasor through the network, we note that no boresight beam is formed and the beams are symmetrically deployed about the array axis. For the port driven in Figure 3.15 we can see that the phase front across the aperture elements is  $-45^\circ$ ,  $-90^\circ$ ,  $-135^\circ$ , and  $-180^\circ$ . Therefore, the shaded beam is produced.



1. No broadside beam is produced if quad hybrids are used (use of E/H tees or hybrid rings will produce broadside beams).
2. For a lossless system, crossover is at  $-3.9$  dB (uniformly illuminated) for orthogonality or  $-3.9$  dB for tapered illumination for higher crossover, loss must be accepted.
3. For orthogonality, beams must change pointing direction with frequency (or a loss must be accepted).

**Figure 3.15** Four-beam Butler matrix.

The four-beam matrix may also have a sequencing switch on the input ports to scan to any one of the four positions shown. A more elaborate network can scan or produce many more beams.

In the basic array, the number of beams is equal to the number of elements, and the normalized array factor is of the form  $\sin(Mx)/(M\sin x)$ , where  $M$  is the number of elements. The (normalized by  $M$ ) magnitude of the field intensity in the far field of a linear array of  $N$  uniformly excited isotropic radiators is given by [25]

$$E(\theta) = E_c(\theta) \frac{\sin(M\phi/2)}{M \sin(\phi/2)} \quad (3.83)$$

where

$E_c(\theta)$  is the element factor (weights the array factor [26]). Since we are assuming isotropic antennas for this analysis,  $E_c(\theta) = 1$ .

$\phi = (2\pi d / \lambda) \sin \theta - \delta$  is a measure of the angle off boresight of the beam.

$\delta$  is the progressive phase difference generated by the matrix and is equal to

$$\delta_k = (2k - 1)\pi / M, k = 1, 2, \dots, M / 2.$$

$k$  is the beam number.

Note in the two-element array previously shown

$$\delta_k = (2k - 1)\pi / M$$

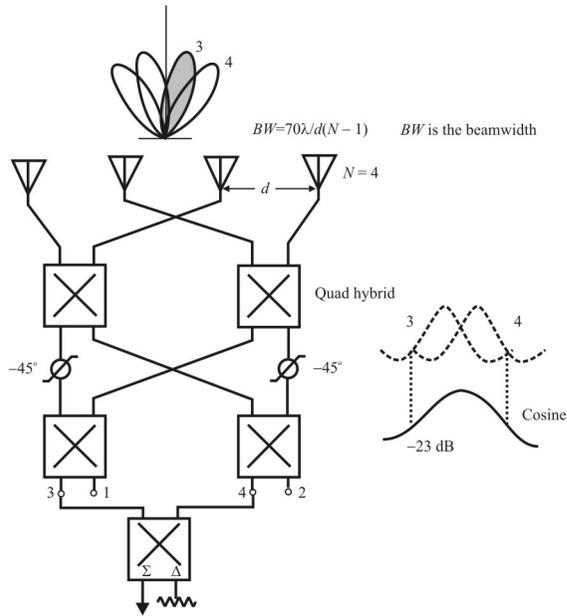
$$\delta = \pi / M = \pi$$

The location of the beams can be found using

$$\sin \theta = \frac{\pi}{Md} \left[ k - \frac{1}{2} \right] \quad (3.84)$$

The first sidelobe is down 13.5 dB, which is typical for a linear array with equal excitations and equal spacing. Interestingly, the beams cross over at the 3.9 dB points, which suggest that the beams are orthogonal and the network is lossless.

The sidelobes can be significantly improved by coherently combining two output ports to give a cosine variation with the sidelobes down by 23 dB, but with a beam distention of 35%. This also is quite common for beams: along with greater sidelobe suppression comes a broadening of the beam. Figure 3.16 illustrates the addition of two beams each having uniform illumination (with  $\sin x/x$  far field) to obtain a cosine illumination. In general,  $n + 1$  beams can be added to form a cosine( $n$ ) illumination.



Notes:

1.  $\Delta$  port can be used to provide difference between ports 3 and 4.
2. Crossover for two adjacent cosine beams is 2.1 dB.
3. 0.9 dB gain loss 1.35 increase in BW.

**Figure 3.16** Cosine illumination achieved by combining two adjacent beams.

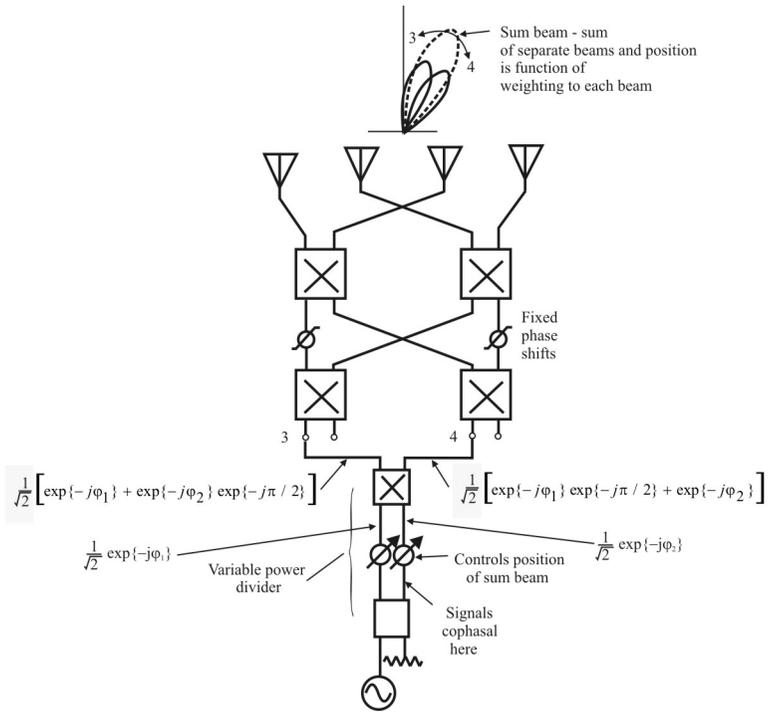


Figure 3.17 Beam scanning in a Butler matrix using a variable power divider.

There are other uses for the Butler matrix. The network shown in Figure 3.17 can be used to provide a limited scan function. Power is varied to ports 3 and 4 (for example) and the variable power divider controls the power to each port. Varying the division of power steers the beam between individual beam axes.

Note that  $180^\circ$  hybrids are also possible. They are used mostly when forming beams with nonlinear arrays, such as circular arrays. As the name implies, the outputs from such a device can be  $180^\circ$  out of phase. The  $\mathbf{S}$  matrix of the  $180^\circ$  hybrid is

$$\mathbf{S} = \frac{-j}{\sqrt{2}} \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \end{bmatrix} \quad (3.85)$$

with reference to the port definitions in Figure 3.18. With inputs at ports 2 and 3 and using columns 2 and 3 of  $\mathbf{S}$ , we can deduce that both ports 2 and 3 are matched, port 1 will provide the sum of the two input signals, and port 4 will provide the difference. Because of this, ports 1 and 4 are sometimes called the sum and difference ports, respectively. We will discuss an application of the  $180^\circ$  hybrid next.

A Butler matrix configuration that works with a circular Adcock antenna array is shown in Figure 3.19 [3]. This is a phase comparison architecture. The output port labeled  $n = 0$  is the reference signal. The signals from all four antennas arrive at output port 0 with the same phase. The north antenna output passes through the two bottom  $180^\circ$  hybrids with no phase shift to port zero. The east antenna output passes through the  $+90^\circ$  phase shifter, is shifted  $-90^\circ$  through the  $90^\circ$  hybrid, and passes through the bottom right hybrid with no phase shift for a net total of  $0^\circ$ . The south antenna's signals passes through both the bottom two hybrids with no phase shift. The west antenna's signal reaches output port 0 bypassing through both the  $90^\circ$  hybrid and the bottom right hybrid with no phase shift.

Next consider the phase at the  $n = 1$  output port. The signal from the north antenna reaches this port with no phase shift, passing directly through the bottom left hybrid and upper right hybrid. The signal from the east antenna reaches the  $n = 1$  port by passing through the  $+90^\circ$  phase shift and then directly through the upper two hybrids for a net  $+90^\circ$  phase shift. The south antenna's signal reaches the  $n = 1$  port via  $180^\circ$  phase shift in the bottom left hybrid and then directly through the upper right hybrid for a net  $180^\circ$  phase shift. The west antenna's signal reaches the  $n = 1$  port with  $-90^\circ$  ( $270^\circ$ ) phase shift through the  $90^\circ$  hybrid and direct connection through the upper right hybrid for a total of  $270^\circ$ . The results are tabulated in Table 3.1.

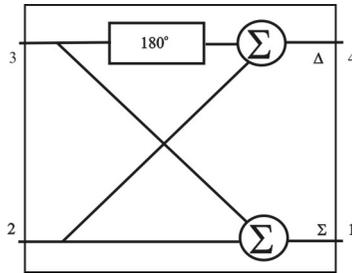


Figure 3.18 180° hybrid.

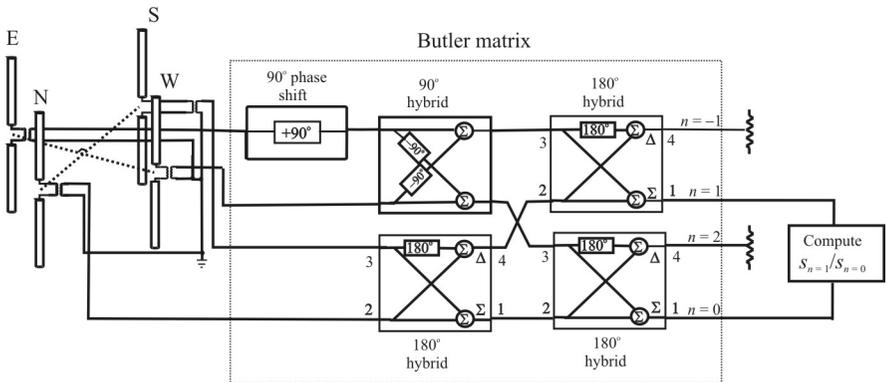


Figure 3.19 Butler matrix. (Source: [3]. © Artech House 2008. Reprinted with permission.)

**Table 3.1** Phase Shifts through a Butler Matrix

Antenna	Spatial Angle (°)	Electrical Phase Shift at Port			
		$n = 0$	$n = 1$	$n = 2$	$n = -1$
N	0	0°	0°	0°	0°
E	90	0°	90°	180°	270°
S	180	0°	180°	0°	180°
W	270	0°	270°	180°	90°

Source: [27].

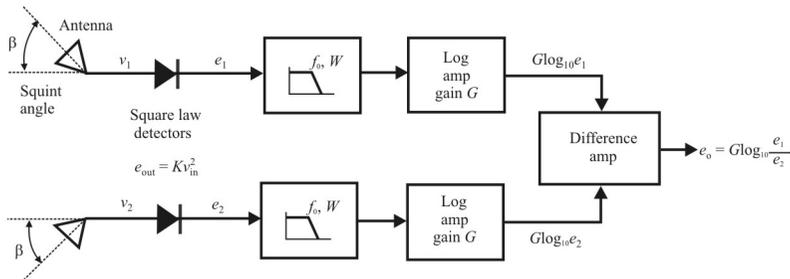
As shown, the phase difference between the  $n = 0$  and  $n = 1$  ports is equal to the spatial angle around the circular array. Note that only one-fourth of the power from any one antenna is available at a given output port due to the power-dividing properties of the hybrids. Also note that the phase difference at the  $n = 0$  and  $n = 2$  ports is exactly twice the phase difference between the  $n = 0$  and  $n = 1$  ports (modulo 360°).

By measuring the  $n = 0$  to  $n = 1$  phase difference, the bearing angle is obtained. This is true even if the signal is, let us suppose, at a 45° bearing angle. Then the north and east antennas have equal phase and the result is a 45° phase difference at the output (i.e., equal components at 0° and 90° produce a 45° phase shift). As long as the antennas produce equal amplitudes, the proper bearing angle can be found regardless of AOA. If the hybrids and antenna elements are broadband, the AOA measurement can be made over a broad band.

This type of receiver measures the AOA of a single pulse and thus the name monopulse. Because of this, the Butler matrix-based monopulse receiver has been adopted in several applications for DF of short duration communication signals, such as those associated with frequency hopping targets. On these targets, the signal dwells at a single frequency for only a short time (~1 ms or so) before moving on to the next channel. Speed is the critical factor in this application.

Another type of monopulse receiver uses directional antennas to form the beams for estimating the AOA. This type of receiver is fairly inaccurate, using the amplitudes of the received signals as adjusted by the antenna gain patterns.

These monopulse direction finders had their debut in radar warning receivers for aircraft self-protect applications. A block diagram showing two channels (there are a total of four or more) of one implementation is shown in Figure 3.20 [28]. The antennas are squinted away from each other and they have directive response patterns (not omnidirectional). The radar signals impinging on the antennas are first amplitude detected in the square-law detectors, lowpass filtered, and fed to a pair of logarithmic amplifiers. The difference of the log signals emerging from these amplifiers contains relative amplitude information and the ratio of these signals is an indication of the AOA of the signal at the two antennas. Since the amplitudes are used to measure the AOA, this is a version of an amplitude-based DF system. Amplitude measurements tend to be not as accurate as phase



**Figure 3.20** Amplitude monopulse receiver. (Source: [1]. © Artech House 2011. Reprinted with permission.)

measurements, so the accuracy of such a monopulse receiver is not very high. Due to their primary application, that is, as ASE receivers, high accuracy is normally not required. They are quite simple to implement, however, and therefore inexpensive. Cost is the primary driver for the ASE application.

Another type of monopulse DF uses phase interferometry to make the AOA measurements. The accuracy of this receiver is much higher than the amplitude DF technique. It is quite a bit more complicated, however, and therefore more expensive to implement.

### 3.5.3 Summary

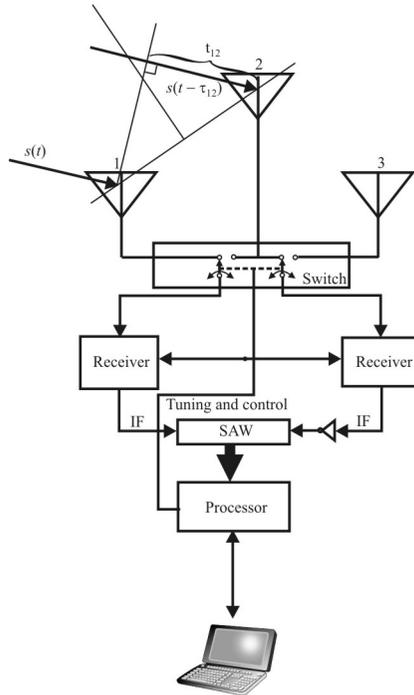
The Butler matrix is a versatile device. It can serve as a beamforming network permitting each port to have the gain of the full array. Being passive and reciprocal, they can be used for both reception and transmission in an antenna array. The beams may be deployed simultaneously or sequentially depending on the application.

## 3.6 Phase Difference Estimation Using SAW Devices<sup>7</sup>

### 3.6.1 Introduction

In this section we describe an approach for rapidly determining the TDOA of a signal at two separated antennas using a *surface acoustic wave* (SAW) device in a distinctive way, coupled with algorithms for processing the output of the SAW [29, 30]. From this time difference we can readily determine the AOA. In this analysis, we will initially assume that there are multiple signals present.

<sup>7</sup> U.S. patent pending.



**Figure 3.21** Block diagram of triple antenna direction-finding system.

As a refresher, the triangulation system being considered was shown in Figure 3.1. LOPs from two or more receiving sites are determined, and an estimate of the location of the emitter is where these lines intersect. With three or more systems, due to noise, they rarely intersect at the same point.

A block diagram of such a system for measuring the LOPs is illustrated in Figure 3.21. A switch determines which antennas are selected for input to each end of the SAW, two at a time. In this way a single SAW can be used. A more complicated switch could also be used to reverse the inputs to the SAW to reduce the effects of component errors. It is also possible to implement a SAW between each pair of antenna elements so that simultaneous measurements are possible, making the determination of the TDOAs even faster, at the expense, of course, of more complicated and therefore costlier hardware.

The advantages of this SAW processor are:

1. Speed. The SAW device determines the time difference very rapidly, thereby enabling determination of the AOA of short duration signals.

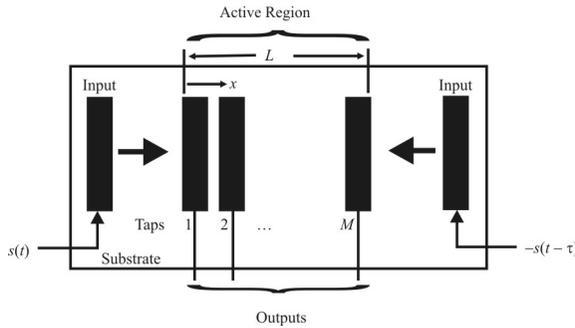


Figure 3.22 SAW device.

2. The potential for operation at low SNR levels.

The AOA,  $\phi$ , is related to the time difference  $t_{12}$  through

$$\phi = 2\pi \frac{c}{\lambda} t_{12} \tag{3.86}$$

so if we determine the time difference between the arrival times at the two antennas, we can determine  $\phi$ . We demonstrate how this time can be estimated with a properly configured SAW device.

### 3.6.2 SAW Characteristics

Due to superposition, counterpropagating signals injected into the two ends of a SAW delay line, the second the negative and delayed version of the first, as shown in Figure 3.22, will form standing waves within the SAW device. These standing waves are of the form

$$\alpha_i(x) = A_i \sin(\beta x - \pi f \tau_i) \tag{3.87}$$

where  $x$  represents the lateral distance from one end of the active region of length  $L_A$  within the SAW,  $\beta$  is a constant associated with the SAW,  $f$  is the operating frequency of the SAW,  $A_i$  is the amplitude of the  $i$ th waveform, and  $\tau_i$  is the time delay difference of signal  $i$  at the two ends of the SAW device. At any given time there will be a finite number of standing waveforms present which will be denoted herein by  $L$ . In general  $L$  is unknown and the task at hand is to determine the parameters of these waveforms from the measurement of the output of the SAW device.

$M$  will denote the number of taps in the SAW. Typical values for  $M$  are 16 to 64. Shown in Figure 3.23 is a typical standing wave in such a device when there is only one signal present and no noise added. It is more informative to plot the standing wave versus the time delay. For the parameters in the caption, such a curve is illustrated in Figure 3.24.

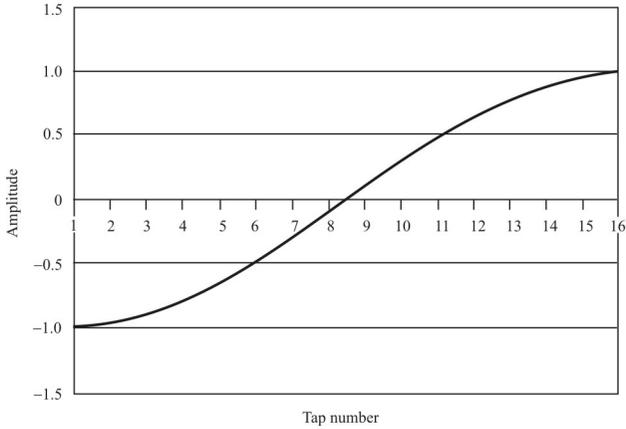
With 16 taps, the angular resolution that results is  $360/16 = 22.5^\circ$ . Except in some specific applications, such as ASE, like the monopulse receiver discussed previously, such a resolution would typically not be adequately precise. Instead, an interpolation scheme, such as parabolic interpolation shown in Figure 3.25(a) (if looking for peaks) or Figure 3.25(b) (if looking for zero-crossings) would be employed. With such a technique, resolution on the order of  $1^\circ$  is possible. (Note that this is the achievable resolution, not the accuracy.)

When there is more than one signal present, then the signals interact within the SAW. Three signals and their sum, which is the standing wave observed at the output taps, are shown in Figure 3.26. Notice that the sum of the three standing waveforms is of the same functional shape as the constituent waveforms. It is also worth noting that the amplitudes of the constituent waveforms affect the zero crossing of the sum waveform.

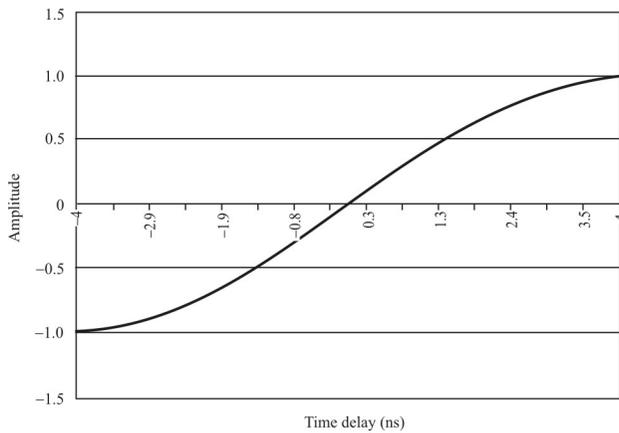
### 3.7 Concluding Remarks

Some of the more commonly used procedures for solving the AOA problem have been presented. Beamforming constitutes the technique most often employed in applications where low resolution is tolerable. It provides robust behavior and is effective in cases where coherent sources are present. MLE, LSE modeling, and signal subspace AOA methods share some of the same attributes of beamforming and are capable of determining AOAs with high resolution. Noise subspace-based methods also give high-resolution performance but are adversely affected by highly coherent sources (such as multipath).

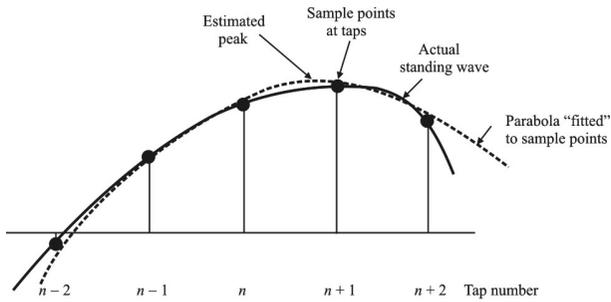
We also included a novel technique for AOA estimation using a properly configured SAW device. Signals from two antennas (one inverted) are inserted into the ends of the SAW. They interact within the SAW in such a way that the phase difference between the two signals can be determined. From this phase difference the AOA can be determined.



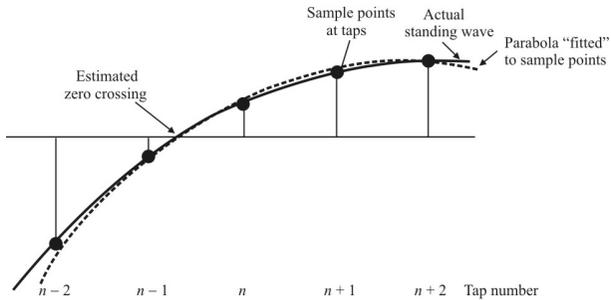
**Figure 3.23** Standing wave in a SAW device caused by counterpropagating like signals, amplitude versus tap number. For this example,  $A_1 = 1$  V,  $\tau_1 = 0$  nanoseconds,  $\beta = 180 \times 10^3$  degrees/meter,  $L = 1$  mm, and  $f = 125$  MHz.



**Figure 3.24** Standing wave in a SAW device caused by counterpropagating like signals, amplitude versus time delay. For this example,  $A_1 = 1$  V,  $\tau_1 = 0$  nanoseconds,  $\beta = 180 \times 10^3$  degrees/meter,  $L = 1$  mm, and  $f = 125$  MHz.

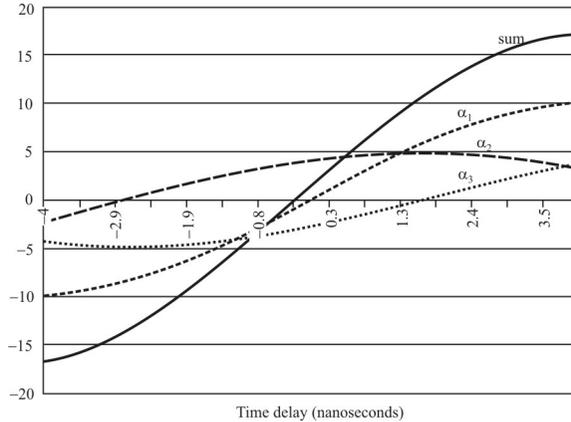


(a)



(b)

**Figure 3.25**(a, b) Parabolic interpolation. In these cases, five points are used to define a parabola. Minimum MSE is often used to estimate the parabola, where the square of the distances between the parabolic model and the actual data points is minimized.



**Figure 3.26** Multiple input signals cause multiple standing waves in the SAW. These constituent waves sum to produce a single standing wave shown here as “sum.” In this example,  $A_1 = 10$  V,  $A_2 = 5$  V,  $A_3 = 5$  V,  $\tau_1 = 0$  ns,  $\tau_2 = -3.5$  ns,  $\tau_3 = 1.5$  ns,  $\beta = 180 \times 10^3$  degrees/meter,  $L = 1$  mm, and  $f = 125$  MHz.

## References

- [1] Poisel, R. A., *Antenna Systems and Electronic Warfare Applications*, Norwood, MA: Artech House, 2012.
- [2] Cadzow, J. A., “Methods of Direction of Arrival,” *Proceedings IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing*, May 9–10, 1991.
- [3] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, 2nd ed., Norwood, MA: Artech House, 2008.
- [4] Johnson, D. H., and D. E. Dudgeon, *Array Signal Processing Concepts and Techniques*, Upper Saddle River, NJ: Prentice Hall, 1993.
- [5] Van Trees, H. L., *Detection, Estimation, and Modulation Theory, Part IV: Optimum Array Processing*, New York: Wiley, 2002.
- [6] Rappaport, T. S., (ed.), *Smart Antennas: Adaptive Arrays, Algorithms, and Wireless Position Location*, Piscataway, NJ: IEEE, 1994.
- [7] Gething, P. J. D., *Radio Direction Finding and Superresolution*, London: Peregrinus Press, 1991.
- [8] Cadzow, J. A., “Direction of Arrival Estimation Using Signal Subspace Modeling,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 28, No. 1, January 1992, pp. 64–79.
- [9] Poisel, R. A., *Principles of Electronic Warfare Antenna Systems*, Norwood, MA: Artech House, 2011, Ch. 10.
- [10] Golub, G. H., and V. Pereyra, “The Differentiation of Pseudo-Inverses and Nonlinear Least Squares Problems Whose Variables Separate,” *SIAM Journal on Numerical Analysis*, Vol. 10, April 1973, pp. 413–432.
- [11] Cadzow, J. A., “Direction of Arrival Estimation Using Signal Subspace Modeling,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 28, No. 1, January 1992, pp. 64–79.

- [12] Strange, G., *Introduction to Applied Mathematics*, Wellesley, MA: Wellesley-Cambridge Press, 1986, pp. 78–79.
- [13] Indritz, J., *Methods in Analysis*, New York: Macmillan, 1963, p. 16.
- [14] Gantmacher, F. R., *The Theory of Matrices*, Vol. I, New York: Chelsea, 1959.
- [15] Curtis, C. W., *Linear Algebra, an Introductory Approach*, 3rd ed., Boston: Allyn and Bacon, 1974.
- [16] Herstein, I. N., *Topics in Algebra*, Waltham, MA: Zeros College Publishing, 1964.
- [17] Birkhoff, G., and S. MacLane, *A Survey of Modern Algebra*, Revised Edition, New York: Macmillan, 1953.
- [18] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, 2nd ed., Norwood, MA: Artech House, 2008, Ch. 8.
- [19] Poisel, R. A., *Antenna Systems and Electronic Warfare Applications*, Norwood, MA: Artech House, 2011.
- [20] Jenkins, H. H., *Small-Aperture Radio Direction-Finding*, Norwood, MA: Artech House, 1991.
- [21] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, 2nd ed., Norwood, MA: Artech, 2008.
- [22] Poisel, R. A., *Target Acquisition in Communication Electronic Warfare Systems*, Norwood, MA: Artech House, 2004, Appendix A.
- [23] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, Norwood, MA: Artech House, 2002, Section 9.3.2.
- [24] Patten, B., “The Versatile Butler Matrix,” *Microwave Journal*, Vol. 47, No. 11, November 2004, p. 126, [http://www.mwjjournal.com/article.asp?HH\\_ID=AR\\_919&title=The+Versatile+Butler+Matrix](http://www.mwjjournal.com/article.asp?HH_ID=AR_919&title=The+Versatile+Butler+Matrix).
- [25] Stutzman, W. L., and G. A. Thiele, *Antenna Theory and Design*, New York: Wiley, 1981, p. 125.
- [26] Poisel, R. A., *Principles of Electronic Warfare Antenna Systems*, Norwood, MA: Artech, 2011.
- [27] Wiley, R. G., *Electronic Intelligence: The Interception of Radar Signals*, Norwood, MA: Artech House, 1985, pp. 101–103.
- [28] Poisel, R. A., *Antenna Systems and Electronic Warfare Applications*, Norwood, MA: Artech House, 2011.
- [29] Ward, R. B., “Continuous-Wave Surface-Acoustic-Wave Delay-Difference Device,” *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, Vol. 36, No. 3, May 1989, pp. 300–303.
- [30] Minter, T. C., M. Hankamer, and R. E. Reid, “Precision Direction Finding Using the Surface Acoustic Wave Device,” *Proceedings MILCOM*, 1992, San Diego, CA, 1992.



# Chapter 4

## MUSIC

### 4.1 Introduction

In this chapter we discuss one of the more popular methods in modern use for AOA determination called *multiple signal classification* (MUSIC). The MUSIC algorithm was first reported in [1] by Schmidt. That reference, however, had rather limited distribution so the paper was republished in [2] because of its wide interest and referencing in the literature.

According to Schmidt:

The term MULTIPLE signal classification (MUSIC) is used to describe experimental and theoretical techniques involved in determining the parameters of multiple wavefronts arriving at an antenna array from measurements made on the signals received at the array elements.

As mentioned in Chapter 3, MUSIC is one technique out of several in a class of algorithms known as subspace methods that can determine the AOAs of several signals impinging on an array at the same time. We discuss MUSIC in this chapter as the premier method of processing multiple signals simultaneously. We describe MUSIC here and examine some of its performance characteristics as an example of the subspace techniques. It has been shown, however, that it is one of the more robust techniques available and it has been widely implemented. Its major shortcoming is that it does not perform with fully correlated signals.

After describing the algorithm in its vanilla form, we delve into some error models that can perturb the amplitudes and phases of the signals collected at a plurality of antennas in an array. The performance of MUSIC in the presence of these error sources is estimated. We finish the chapter with a comparison of MUSIC with some other superresolution techniques that determine the AOAs of multiple signals.

## 4.2 MUSIC Overview

The AOA  $\vec{\theta}_0$  of a signal is not calculated directly from the data in superresolution DF systems, in general, but by finding which of the stored array gain vectors [ $\vec{a}(\vec{\theta})$  for direction  $\vec{\theta}$ ] corresponds to the gain vector  $\vec{b}$  seen by the signal arriving from direction  $\vec{\theta}_0$ . [We represent  $\vec{\theta}_0$  as a vector with two components  $\vec{\theta}_0 = [\phi \ \varphi]^T$ , the azimuth and elevation angles of the signal.<sup>1</sup>] Having determined which vector, say  $\vec{a}(\vec{\theta}'_0)$ , is “nearest” in some sense to  $\vec{b}$  we take  $\vec{\theta}'_0$  to be our estimate of the signal AOA. (The array gain vector in direction  $\vec{\theta}$  is the set of complex amplitude factors, relative to a unit gain element at some reference position, applied to the waveform of any signal in this direction by the elements of the array. Beamforming using the conjugate of this vector, or a scaled version of it, will maximize the array gain in this direction and may be termed the *steering vector* for direction  $\vec{\theta}$ .)

In MUSIC the data from the  $M$  array elements is processed to give not the  $L$  signal vectors  $\vec{b}_1, \vec{b}_2, \dots, \vec{b}_L$  directly, but a set of vectors which form a basis for the  $L$ -dimensional vector space spanned by the signal vectors—a subspace of the  $M$ -dimensional Hilbert space containing the gain vectors (assuming  $L < M$ ). This vector space is termed the *signal space*, or *subspace*. We have a stored set of array gain vectors for all AOAs of interest, at some resolution (thus forming a set of samples rather than a continuum), or alternatively with a mathematical description of the array from which these vectors can be calculated as required. The full (continuous) set of such vectors is referred to as the array manifold.

Each manifold vector is taken in turn to determine how close it lies to the signal space. The vectors which most nearly lie in this space are taken to be the signal gain vectors, thereby giving the signal AOAs. In fact, the square magnitudes of the projection of the (normalized) manifold vectors orthogonal to the signal space is calculated to give the MUSIC functional  $f(\vec{\theta})$  and the positions of the  $L$  lowest minima are taken.

The signal space basis may be found in various ways, but the most common is eigenanalysis of the (estimated) data covariance matrix  $\hat{\mathbf{C}}$ .

---

<sup>1</sup> Note that this is not the typical use elevation angle, which is normally measured from zenith. We use  $\varphi$  here as the elevation angle as measured from the horizontal plane.

### 4.3 MUSIC

The MUSIC algorithm was devised as a form of superresolution DF technique for processing the signals from many antennas to obtain estimates of the AOA of multiple signals [3]. It is in a family of processes called subspace-based<sup>2</sup> processing and it is based on an eigendecomposition of the covariance matrix derived from data samples (vectors).

In most cases of interest the covariance matrix  $\mathbf{C}_{rr}$  is singular as noted above since normally  $R < N-1$ . That makes inverting this matrix for ML processing impossible in the absence of noise and measurement errors. A different technique for determining the time differences is thus required. One method is based on the singular value decomposition on the data matrix.

#### 4.3.1 The MUSIC Algorithm

The simplest of the algorithms that are based on the subspace decomposition is the MUSIC algorithm. Assume that  $L$  signals are impinging on the sensor array. Now  $\vec{a}(\vec{\theta})$  is projected onto the noise subspace  $\mathcal{R}\{\mathbf{E}_n\}$ . The projection vector gives the vector

$$\vec{z} = \mathbf{P}_A^\perp \vec{a}(\vec{\theta}) \quad (4.1)$$

The magnitude squared of  $\vec{z}$  can be written as

$$f(\vec{\theta}) = \vec{z}^H \vec{z} = \vec{a}^H(\vec{\theta}) (\mathbf{P}_A^\perp)^H \mathbf{P}_A^\perp \vec{a}(\vec{\theta}) = \vec{a}^H(\vec{\theta}) \mathbf{E}_n \mathbf{E}_n^H \vec{a}(\vec{\theta}) \quad (4.2)$$

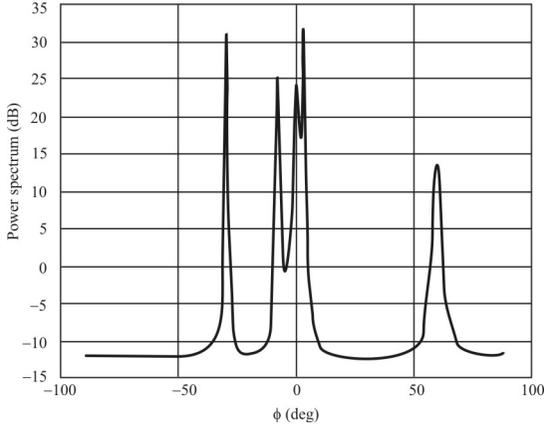
Obviously  $f(\vec{\theta}) = 0$  when  $\vec{\theta} \in \{\vec{\theta}_1, \vec{\theta}_2, \dots, \vec{\theta}_L\}$ . Therefore we search the array manifold, that is,  $f(\vec{\theta})$  is evaluated for all  $\vec{\theta}$ , and we select the AOA estimates as the points which satisfy  $f(\vec{\theta}) = 0$ .

For coherent or fully correlated signals the autocovariance matrix  $\mathbf{C}_{ss}$  is rank deficient. Therefore, (3.29) must be replaced with

$$\mathcal{R}\{\mathbf{E}_s\} \in \mathcal{R}\{\mathbf{A}\} \quad (4.3)$$

---

<sup>2</sup> The “space” involved here is a mathematical construct consisting of one or more  $n$ -tuples (vectors). A subspace is a subset of the vectors from the space.



**Figure 4.1** MUSIC spectrum. Example of a MUSIC power spectrum plot for a ULA.

This condition is the major drawback of the MUSIC algorithm (i.e., it breaks down for correlated or fully coherent signals). As we will show, however, it performs reasonably well until the fractional coherence approaches 0.8 to 0.9, and in all real situations there is a certain amount of noise on signals, even multipath signals.

Sharper peaks ensue if the MUSIC functional is the inverse of (4.2) yielding the power spectrum evaluated at all  $\vec{\theta}_i \in \{\vec{\theta}_1, \vec{\theta}_2, \dots, \vec{\theta}_D\}$ :

$$P_{\text{MU}}(\vec{\theta}_i) = \frac{1}{\vec{a}^H(\vec{\theta}_i) \mathbf{E}_n \mathbf{E}_n^H \vec{a}(\vec{\theta}_i)} \quad (4.4)$$

where  $D$  is the number of possible AOA arrival angles (size of the array manifold). A typical example of the MUSIC spectrum when only azimuth AOAs are considered is illustrated in Figure 4.1.

If the additive noise is not spatially white, its covariance  $\mathcal{E}\{\vec{n}(t)\vec{n}^*(t)\} = \sigma^2 \mathbf{\Sigma}$  is not the identity ( $\mathbf{\Sigma} \neq \mathbf{I}$ ),<sup>3</sup> and the minimization of  $f(\vec{\theta})$  must be carried out in the  $\mathbf{\Sigma}$ -metric. Assuming that  $\mathbf{\Sigma}$  is known, this minimization can be done either via a generalized eigendecomposition of the pair  $(\mathbf{C}_{\text{rr}}, \mathbf{\Sigma})$  or via an eigendecomposition of the prewhitened covariance

$$\begin{aligned} \mathbf{\Sigma}^{-1/2} \mathbf{C}_{\text{rr}} \mathbf{\Sigma}^{1/2} &= \mathbf{\Sigma}^{-1/2} \mathbf{A} \mathbf{C}_{\text{ss}} \mathbf{A}^* \mathbf{\Sigma}^{1/2} + \sigma^2 \mathbf{I} \\ &= \sum_{m=1}^M \lambda_m \vec{e}_m \vec{e}_m^H = \mathbf{E}_s \mathbf{\Lambda}_s \mathbf{E}_s^H + \sigma^2 \mathbf{E}_n \mathbf{E}_n^H \end{aligned} \quad (4.5)$$

where  $\mathbf{E}_s = [\vec{e}_1 \cdots \vec{e}_L]$  and  $\mathbf{E}_n = [\vec{e}_{L+1} \cdots \vec{e}_M]$ . The subspace relationships are  $\text{span}\{\mathbf{E}_s\} = \text{span}\{\boldsymbol{\Sigma}^{-1/2} \mathbf{A}\} \perp \text{span}\{\mathbf{E}_n\}$ , and the MUSIC functional becomes

$$f(\vec{\theta}) = \text{Tr}(\mathbf{P}_{\Sigma, \theta} \hat{\mathbf{E}}_n \hat{\mathbf{E}}_n^H) \quad (4.6)$$

where  $\mathbf{P}_{\Sigma, \theta} = \boldsymbol{\Sigma}^{-1/2} \vec{a}(\vec{\theta}) [\vec{a}^H(\vec{\theta}) \boldsymbol{\Sigma}^{-1} \vec{a}(\vec{\theta})]^{-1} \vec{a}^H(\vec{\theta}) \boldsymbol{\Sigma}^{-1/2}$ . This more general formulation of MUSIC will be used in the error analysis in the next section.

## 4.4 Performance of MUSIC in the Presence of Modeling Errors

Errors in MUSIC AOA estimates can arise from a number of sources. The most important of these include:

- Finite sample effects;
- An imprecisely known noise covariance matrix  $\boldsymbol{\Sigma}$ ;
- A perturbed array manifold  $\{\vec{a}(\vec{\theta})\}$  (amplitudes, phases, or both).

Finite sample effects occur since a perfect covariance measurement  $\mathbf{C}_{rr}$  cannot be obtained. In practice, the sample covariance  $\hat{\mathbf{C}}_{rr}$  defined by

$$\hat{\mathbf{C}}_{rr} = \frac{1}{N} \sum_{n=1}^N \vec{r}(t_n) \vec{r}^*(t_n) \quad (4.7)$$

is used to estimate  $\mathbf{C}_{rr}$ . For finite  $N$ , the signals and noise have not had time to decorrelate, and the noise covariance has not yet converged to its limiting value. When  $N$  is large or the SNR is high, finite sample effects may usually be neglected, however. There are many applications for which the limiting factor in performance is not due to finite sample effects, but rather to the model errors.

### 4.4.1 Model Errors

To isolate the effects of these model errors on the AOA estimates, we will assume that the finite sample effects due to additive noise are negligible and that an exact measurement of the perturbed covariance  $\hat{\mathbf{C}}_{rr}$  is available. A model for  $\hat{\mathbf{C}}_{rr}$  when it contains errors is

$$\hat{\mathbf{C}}_{\text{tr}} = (\mathbf{I} + \Delta) \left[ (\mathbf{A} + \tilde{\mathbf{A}}) \mathbf{C}_{\text{ss}} (\mathbf{A} + \tilde{\mathbf{A}})^{\text{H}} + \sigma^2 (\boldsymbol{\Sigma} + \tilde{\boldsymbol{\Sigma}}) \right] (\mathbf{I} + \Delta)^{\text{H}} \quad (4.8)$$

where the matrices  $\Delta$ ,  $\tilde{\mathbf{A}}$ , and  $\tilde{\boldsymbol{\Sigma}}$  are the results of various types of model perturbations. The matrix  $\Delta$  contains errors that affect both the signal and noise components. Such errors include gain imbalances in the receivers, channel crosstalk, and antenna mutual coupling effects. The matrix  $\tilde{\mathbf{A}}$  represents the error in the nominal array response, and incorporates the effects of imprecisely known sensor locations, perturbations in the antenna amplitude and phase patterns, and signal-only mutual coupling. Finally, deviations of the additive noise statistics from  $\boldsymbol{\Sigma}$  are denoted by the Hermitian matrix  $\tilde{\boldsymbol{\Sigma}}$ .

We are primarily interested in how  $\Delta$ ,  $\tilde{\mathbf{A}}$ , and  $\tilde{\boldsymbol{\Sigma}}$  affect the noise subspace since that is the subspace we use for MUSIC. Let  $\hat{\mathbf{E}}_{\text{n}} = \mathbf{E}_{\text{n}} + \tilde{\mathbf{E}}_{\text{n}}$  represent the perturbed noise subspace eigenvectors, and assume that  $\hat{\mathbf{E}}_{\text{n}}$  has been normalized so that  $\hat{\mathbf{E}}_{\text{n}}^{\text{H}} \hat{\mathbf{E}}_{\text{n}} = \mathbf{I}$ . The perturbation  $\tilde{\mathbf{E}}_{\text{n}}$  will, in general, have components in both the true signal and noise subspaces.<sup>3</sup> As we will see shortly, however, only those components of  $\tilde{\mathbf{E}}_{\text{n}}$  in the signal subspace will contribute to the estimation error.

Next we examine the noise eigenvectors of  $\boldsymbol{\Sigma}^{-1/2} \hat{\mathbf{C}}_{\text{tr}} \boldsymbol{\Sigma}^{-1/2}$ , defined by

$$\boldsymbol{\Sigma}^{-1/2} \hat{\mathbf{C}}_{\text{tr}} \boldsymbol{\Sigma}^{-1/2} \hat{\mathbf{E}}_{\text{n}} = \hat{\mathbf{E}}_{\text{n}} (\sigma^2 \mathbf{I} + \tilde{\boldsymbol{\Lambda}}) \quad (4.9)$$

to establish a link between the error terms of (4.8) and  $\tilde{\mathbf{E}}_{\text{n}}$ . In (4.9)  $\tilde{\boldsymbol{\Lambda}}$  represents the perturbed noise eigenvalues. Expanding this equation using the model of (4.8) and eliminating second-order error terms [e.g., terms of order  $\mathcal{O}(\|\tilde{\mathbf{A}}\|^2)$ ,  $\mathcal{O}(\|\tilde{\mathbf{A}}\| \|\Delta\|)$ , and so forth] and terms involving  $\tilde{\mathbf{A}}^{\text{H}} \boldsymbol{\Sigma}^{-1/2} \mathbf{E}_{\text{n}} = 0$  we get

$$\begin{aligned} \boldsymbol{\Sigma}^{-1/2} \tilde{\mathbf{A}} \mathbf{C}_{\text{ss}} \left[ (\tilde{\mathbf{A}} + \Delta \mathbf{A})^{\text{H}} \boldsymbol{\Sigma}^{-1/2} \mathbf{E}_{\text{n}} + \mathbf{A}^{\text{H}} \boldsymbol{\Sigma}^{-1/2} \tilde{\mathbf{E}}_{\text{n}} \right] \\ + \sigma^2 \boldsymbol{\Sigma}^{-1/2} (\tilde{\boldsymbol{\Sigma}} + \Delta \boldsymbol{\Sigma} + \boldsymbol{\Sigma} \Delta^{\text{H}}) \boldsymbol{\Sigma}^{-1/2} \mathbf{E}_{\text{n}} = \mathbf{E}_{\text{n}} \tilde{\boldsymbol{\Lambda}} \end{aligned}$$

Multiplying on the left by  $\mathbf{A}^{\text{H}} \boldsymbol{\Sigma}^{-1/2}$  and transposing the result then yields

$$\mathbf{E}_{\text{n}}^{\text{H}} \boldsymbol{\Sigma}^{-1/2} \mathbf{A} = -\mathbf{E}_{\text{n}}^{\text{H}} \boldsymbol{\Sigma}^{-1/2} \boldsymbol{\Xi} \quad (4.10)$$

<sup>3</sup> This phenomenon may also result from the fact that, in general,  $\hat{\mathbf{E}}_{\text{n}}$  is unique while  $\mathbf{E}_{\text{n}}$  is not (i.e., it may be replaced by  $\mathbf{E}_{\text{n}} \mathbf{U}$  for any unitary  $\mathbf{U}$ ).

where we have defined the error matrix  $\Xi$  as

$$\Xi = \tilde{\mathbf{A}} + \Delta\mathbf{A} + \sigma^2(\tilde{\Sigma} + \Sigma\Delta^H + \Delta\Sigma)\Sigma^{-1/2}(\Sigma^{-1/2}\mathbf{A})^H\mathbf{C}_{ss}^{-1} \quad (4.11)$$

It is clear from (4.10) that only those model errors that give rise to components outside the signal subspace (i.e., those that result in errors in the AOA estimates) will produce a noise eigenvector perturbation  $\tilde{\mathbf{E}}_n$  with components outside the noise subspace.

Multiplying (4.10) on the left by  $\hat{\mathbf{E}}_n$  and using the fact that  $\hat{\mathbf{E}}_n^H\Sigma^{-1/2}\mathbf{A} = \tilde{\mathbf{E}}_n^H\Sigma^{-1/2}\mathbf{A}$  leads to

$$\hat{\mathbf{E}}_n\hat{\mathbf{E}}_n^H\Sigma^{-1/2}\vec{a}(\vec{\theta}_i) = \mathbf{E}_n\tilde{\mathbf{E}}_n^H\Sigma^{-1/2}\vec{a}(\vec{\theta}_i) = -\mathbf{E}_n\mathbf{E}_n^H\Sigma^{-1/2}\vec{c}_i, \quad i = 1, \dots, L \quad (4.12)$$

where the error vector  $\vec{c}_i$  for the  $i$ th source is defined as the  $i$ th column of  $\Xi$ :

$$\vec{c}_i = \vec{a}(\vec{\theta}_i) + \Delta\vec{a}(\vec{\theta}_i) + \sigma^2(\tilde{\Sigma} + \Sigma\Delta^H + \Delta\Sigma)\Sigma^{-1/2}(\Sigma^{-1/2}\mathbf{A})^H\mathbf{C}_{ss,i}^{-1} \quad (4.13)$$

and  $\mathbf{C}_{ss,i}^{-1}$  denotes the  $i$ th column of  $\mathbf{C}_{ss}^{-1}$ . Thus, the projection of the true steering vector  $\Sigma^{-1/2}\vec{a}(\vec{\theta}_i)$  into the perturbed noise subspace is approximately equal to the projection of the perturbation vector  $\Sigma^{-1/2}\vec{c}_i$  into the true noise subspace. This is the key relationship used in the following to develop error expressions for the AOA estimates.

#### 4.4.2 Error Expressions

Expanding  $\hat{f}'$  about the estimate  $\hat{\vec{\theta}}_i$ , for small enough error, we may write

$$0 = \hat{f}'(\hat{\vec{\theta}}_i) + \hat{f}''(\hat{\vec{\theta}}_i)(\hat{\vec{\theta}}_i - \vec{\theta}_i) \quad (4.14)$$

where  $\hat{f}'(\vec{\theta}) = \partial\hat{f}/\partial\vec{\theta}$  and

$$\hat{f}''(\vec{\theta}_i) \triangleq \hat{f}''(\vec{\theta})\Big|_{\vec{\theta}=\vec{\theta}_i}$$

Before solving for the error  $\hat{\vec{\theta}}_i - \vec{\theta}_i$  in (4.14), we will find approximations for each term of the sum.

To evaluate  $\hat{f}'$ , we note that

$$\frac{\partial \mathbf{P}_{\Sigma, \theta}}{\partial \vec{\theta}} = \mathbf{P}_{\Sigma, \theta}^{\perp} \Sigma^{-1/2} \vec{d}(\vec{\theta}) [\Sigma^{-1/2} \vec{d}(\vec{\theta})]^{\dagger} + \{ [\Sigma^{-1/2} \vec{d}(\vec{\theta})]^{\dagger} \}^H \vec{d}^H(\vec{\theta}) \Sigma^{-1/2} \mathbf{P}_{\Sigma, \theta}^{\perp}$$

where

$$\begin{aligned} \vec{d}(\vec{\theta}) &= \frac{\partial \vec{a}(\vec{\theta})}{\partial \vec{\theta}} \\ \mathbf{P}_{\Sigma, \theta}^{\perp} &= \mathbf{I} - \mathbf{P}_{\Sigma, \theta} \end{aligned}$$

With these expressions and the relationship  $\mathbf{E}_n^H \Sigma^{-1/2} \vec{a}(\vec{\theta}_i) = 0$ , the first derivative may be written as

$$\begin{aligned} \hat{f}'(\vec{\theta}_i) &= 2 \operatorname{Re} \left\{ \vec{d}^H(\vec{\theta}_i) \Sigma^{-1/2} \mathbf{P}_{\Sigma, \theta}^{\perp} \hat{\mathbf{E}}_n \hat{\mathbf{E}}_n^H \{ [\Sigma^{-1/2} \vec{a}(\vec{\theta}_i)]^{\dagger} \}^H \right\} \\ &= 2 \operatorname{Re} \left\{ \vec{d}^H(\vec{\theta}_i) \Sigma^{-1/2} \mathbf{E}_n \tilde{\mathbf{E}}_n^H \{ [\Sigma^{-1/2} \vec{a}(\vec{\theta}_i)]^{\dagger} \}^H \right\} + O(\|\mathbf{E}_n\|^2) \\ &= \frac{2 \operatorname{Re} \left\{ \vec{d}^H(\vec{\theta}_i) \Sigma^{-1/2} \mathbf{E}_n \tilde{\mathbf{E}}_n^H \Sigma^{-1/2} \vec{a}(\vec{\theta}_i) \right\}}{\vec{a}^H(\vec{\theta}_i) \Sigma^{-1} \vec{a}(\vec{\theta}_i)} \end{aligned} \quad (4.15)$$

Since the second derivative appearing in (4.14) is multiplied by the error term  $\hat{\vec{\theta}}_i - \vec{\theta}_i$ , which is assumed to be small, we make the following approximation:

$$\hat{f}''(\vec{\theta}_i)(\hat{\vec{\theta}}_i - \vec{\theta}_i) = f''(\vec{\theta}_i - \vec{\theta}_i)$$

which, together with (4.14), produces

$$\hat{\vec{\theta}}_i - \vec{\theta}_i = -\frac{\hat{f}'(\vec{\theta}_i)}{f''(\vec{\theta}_i)} \quad (4.16)$$

It is straightforward to show that

$$f''(\vec{\theta}_i) = 2 \operatorname{Re} \left\{ \frac{\vec{d}^H(\vec{\theta}_i) \Sigma^{-1/2} \mathbf{E}_n \mathbf{E}_n^H \Sigma^{-1/2} \vec{d}(\vec{\theta}_i)}{\vec{d}^H(\vec{\theta}_i) \Sigma^{-1} \vec{d}(\vec{\theta}_i)} \right\} \quad (4.17)$$

So using (4.12), (4.15), (4.16), and (4.17) leads to the following first-order expression for the estimation error

$$\begin{aligned} \hat{\theta}_i - \vec{\theta}_i &= - \frac{\operatorname{Re} \left\{ \vec{d}^H(\vec{\theta}_i) \Sigma^{-1/2} \mathbf{E}_n \tilde{\mathbf{E}}_n^H \Sigma^{-1/2} \vec{d}(\vec{\theta}_i) \right\}}{\vec{d}^H(\vec{\theta}_i) \Sigma^{-1/2} \mathbf{E}_n \mathbf{E}_n^H \Sigma^{-1/2} \vec{d}(\vec{\theta}_i)} \\ &= \frac{\operatorname{Re} \left\{ \vec{d}^H(\vec{\theta}_i) \Sigma^{-1/2} \mathbf{E}_n \tilde{\mathbf{E}}_n^H \Sigma^{-1/2} \vec{\epsilon}_i \right\}}{\vec{d}^H(\vec{\theta}_i) \Sigma^{-1/2} \mathbf{E}_n \mathbf{E}_n^H \Sigma^{-1/2} \vec{d}(\vec{\theta}_i)} \end{aligned} \quad (4.18)$$

We assume that  $\vec{\epsilon}_i$  is not known precisely, but instead is a realization of some known perturbation model. We are not interested in a particular value of the estimation error  $\hat{\theta}_i - \vec{\theta}_i$  but rather some averaged measure of it. Here we will assume that the perturbation model is specified in probabilistic terms (i.e.,  $\vec{\epsilon}_i$  is random) and develop expressions for the resulting bias and variance of  $\hat{\theta}_i - \vec{\theta}_i$ .

For cases where  $\vec{\epsilon}_i$  is zero mean, from (4.18) we see that  $\mathcal{E}\{\hat{\theta}_i - \vec{\theta}_i\} = 0$ . This condition will be assumed for all cases that we consider here since we have no reason to factor a bias term in and the bias is a systematic type of error that usually can be calibrated out, so the bias term will not be addressed further.

As for the variance (or, in this case, RMS value) of the estimation error for MUSIC

$$\begin{aligned} \mathbf{C}_{\text{MUSIC}} &\triangleq \mathcal{E} \left\{ (\hat{\theta}_i - \vec{\theta}_i)(\hat{\theta}_k - \vec{\theta}_k) \right\} \\ &= \frac{\operatorname{Re} \left\{ \vec{f}_i^H \mathbf{E}_n^H \Sigma^{-1/2} \left( \mathbf{C}_{\xi,1}^{ik} \Sigma^{-1/2} \mathbf{E}_n \vec{f}_k + \mathbf{C}_{\xi,2}^{ik} \overline{\Sigma^{-1/2} \mathbf{E}_n \vec{f}_k} \right) \right\}}{2(\vec{f}_i^H \vec{f}_i)(\vec{f}_k^H \vec{f}_k)}, \quad i, k = 1, \dots, L \end{aligned} \quad (4.19)$$

where

$$\begin{aligned} \vec{f}_i &= \mathbf{E}_n^H \Sigma^{-1/2} \vec{d}(\vec{\theta}_i) \\ \mathbf{C}_{\xi,1}^{ik} &= \mathcal{E} \{ \vec{\epsilon}_i \vec{\epsilon}_k^H \} \\ \mathbf{C}_{\xi,2}^{ik} &= \mathcal{E} \{ \vec{\epsilon}_i \vec{\epsilon}_k^T \} \end{aligned}$$

Thus, the error covariance of MUSIC's AOA estimates can be computed for any scenario where the perturbation covariances  $\mathbf{C}_{\xi,1}^{ik}$  and  $\mathbf{C}_{\xi,2}^{ik}$  may be evaluated. This will, in turn, be possible provided that the second-order statistics of the elements of  $\tilde{\mathbf{A}}, \mathbf{\Lambda}$ , and  $\tilde{\mathbf{\Sigma}}$  are known.

#### Example [4]

There are a variety of models that could be used to describe  $\tilde{\mathbf{A}}$ . A particularly simple model that has been widely used is to assume that the columns of  $\tilde{\mathbf{A}}$  are independent zero-mean complex Gaussian random vectors with known covariance

$$\tilde{\mathbf{a}}(\vec{\theta}_i) \sim \mathcal{N}_{\mathbb{C}}(0, \mathbf{B}_i), \quad \mathcal{E}\{\tilde{\mathbf{a}}(\vec{\theta}_i)\tilde{\mathbf{a}}^T(\vec{\theta}_i)\} = 0, \quad i = 1, \dots, L \quad (4.20)$$

If the errors are independent from sensor to sensor,  $\mathbf{B}_i$  is clearly diagonal. Off-diagonal terms indicate sensor to sensor correlations that result, for example, if there are uncalibrated mutual coupling effects, or if some sensors tend to perturb uniformly (such as identical or adjacent elements).

If under (4.20) the errors are i.i.d., then  $\mathbf{C}_{\xi,2}^{ik} = 0$  and  $\mathbf{C}_{\xi,1}^{ik} = \mathbf{B} = \sigma_a^2 \mathbf{I} \delta_{ik}$ , where  $\sigma_a^2$  represents the variance of the perturbation at each sensor and  $\delta_{ik}$  is the Kronecker delta function. This corresponds to adding an independent, circular complex Gaussian r.v. of variance  $\sigma_a^2$  to the response at each element of the array and in each signal direction. Though the sensor errors themselves are angle-dependent, their statistics under this model are independent of  $\phi$ . For this simple case, the covariance of the MUSIC estimates as given by (4.19) can be written as

$$\mathbf{C}_{\text{MU}} = \frac{\sigma_a^2}{2} (\mathbf{H} \odot \mathbf{I})^{-1} (\mathbf{F}^H \mathbf{E}_n^H \mathbf{\Sigma}^{-1} \mathbf{E}_n \mathbf{F} \odot \mathbf{I}) (\mathbf{H} \odot \mathbf{I})^{-1} \quad (4.21)$$

where  $\mathbf{X} \odot \mathbf{Y}$  is the Schur (Hadamard) product:

$$\begin{aligned} (\mathbf{X} \odot \mathbf{Y})_{ik} &= \mathbf{X}_{ik} \mathbf{Y}_{ik} \\ \mathbf{F} &= \begin{bmatrix} \vec{f}_1 & \dots & \vec{f}_L \end{bmatrix} = \mathbf{E}_n^H \mathbf{\Sigma}^{-1/2} \mathbf{D} \end{aligned}$$

$$\mathbf{D} = \begin{bmatrix} \vec{d}(\vec{\theta}_1) & \cdots & \vec{d}(\vec{\theta}_L) \end{bmatrix}$$

and

$$\mathbf{H} = \mathbf{F}^H \mathbf{F} = \mathbf{D}^H \boldsymbol{\Sigma}^{-1/2} \mathbf{E}_n \mathbf{E}_n^H \boldsymbol{\Sigma}^{-1/2} \mathbf{D}$$

Note that if  $\boldsymbol{\Sigma} = \mathbf{I}$ , (4.21) reduces to

$$\mathbf{C}_{\text{MU}} = \frac{\sigma_a^2}{2} (\mathbf{D}^H \mathbf{P}_A^\perp \mathbf{D} \odot \mathbf{I})^{-1}$$

where  $\mathbf{P}_A^\perp \triangleq \mathbf{I} - \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H = \mathbf{E}_n \mathbf{E}_n^H$ .

The variance  $\sigma_a^2$  of the perturbation determines the amount of deviation of the gain and phase response from their nominal values. To quantify this deviation, the *relative gain error amplitude* (RGEA) and *phase error amplitude* (PEA) for the response of the  $k$ th antenna in direction  $\vec{\theta}_i$  are defined to be  $20 \log_{10}[\sigma_a / a_k(\vec{\theta}_i)]$  and  $180(\sigma_a / \pi)$  degrees, respectively.

### 4.4.3 Results

The reader interested in derivation of more of the various perturbation models should consult [4]. We will present in this section some of the simulation results from that reference.

#### 4.4.3.1 Phase Error Amplitude

First we consider the effects of perturbations to the amplitude of the phase error. We assume that the array is a 12-element  $\lambda/2$  *uniform linear array* (ULA). Two uncorrelated emitters at true AOAs of  $10^\circ$  and  $15^\circ$  relative to broadside were simulated, each with a power level of 0 dB relative to the additive noise, that is,  $\text{SNR} = \gamma = 0$  dB. The signal at  $10^\circ$  is the SOI. An angle- and sensor-independent Gaussian phase perturbation was made to the array response, and the performance of MUSIC was evaluated as a function of the variance of the perturbation. The results of the simulations are plotted in Figure 4.2. The solid line is the sample standard deviation of the MUSIC algorithm while the dashed line represents the CRB for this case. For this scenario, MUSIC compares very favorably to the CRB.

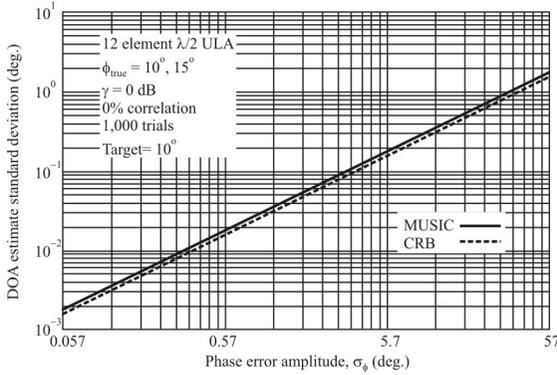


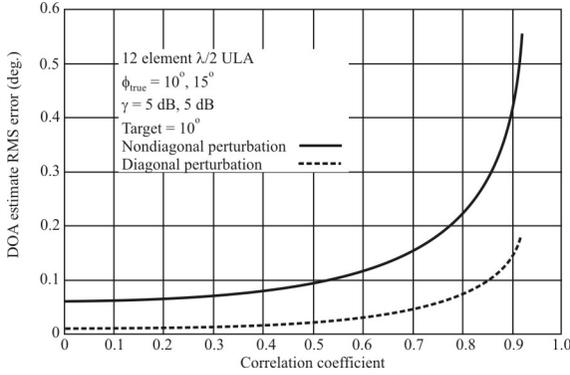
Figure 4.2 MUSIC versus phase error amplitude; 12-element  $\lambda/2$  ULA.

#### 4.4.3.2 Noise Covariance Perturbation

Next we consider the effects of perturbations to the noise covariance as a function of signal correlation. This array and source configuration was identical to that of the previous case, except that the SNR was increased to 5 dB. The nominal covariance of the noise was assumed to be identity, and two types of perturbations were simulated. The first was a diagonal perturbation of covariance  $\mathbf{B}_\sigma = \sigma_s^2 \mathbf{I}$ , and the second was random errors of variance  $\sigma_s^2$  were added to all elements of the noise covariance. Figure 4.3 shows the performance of MUSIC for these two types of perturbations as a function of the correlation between the two signals when  $\sigma_s = 0.2$ . The solid line corresponds to the case of nondiagonal perturbation, while the dashed line corresponds to the case for diagonal perturbation. Even though in this case the level of uncertainty in  $\tilde{\Sigma}$  results in diagonal elements between 0.6 and 1.4 and the off diagonal terms of magnitude as large as 0.3 to 0.5, at low values of correlation the error in the AOA estimates is quite small. Performance does, however, degrade rapidly as the correlation coefficient approaches 0.8–0.9.

#### 4.4.3.3 Mutual Coupling

The performance degradation due to uncalibrated mutual coupling is examined in the next example. The sources are at  $10^\circ$  and  $15^\circ$  with 0 dB SNRs. The antenna is a



**Figure 4.3** MUSIC correlation.

UCA with a radius of  $3\lambda$ . Immediately adjacent sensors on the perimeter of the circle were coupled with angle- and sensor-independent complex Gaussian coupling coefficients of variance  $\sigma_m^2$ .

An angle-independent gain perturbation with RGEA  $\sigma_g = 0.01 \sim -40 \text{ dB}$  and an angle-independent phase perturbation with PEA  $\sigma_\phi = 0.06^\circ$  were also made to the nominal array response in addition to the mutual coupling errors. The results of the simulations are shown in Figure 4.4. Notice that, as we would expect, the mutual coupling perturbation deteriorates only when its magnitude exceeds that of the gain and phase errors and that the theoretical expressions accurately rack the effects of increasing  $\sigma_m$ .

## 4.5 Determining the Number of Wavefields

Estimating the number of incident signals ( $\hat{L}$ ) plays a critical role in most array processing eigenanalysis algorithms. Perhaps the simplest way to estimate the number of waves present is to set a threshold on the eigenvalues by some criterion, and declaring all those smaller than the threshold as noise eigenvalues and all those larger than the threshold as signal eigenvalues. Setting the threshold may not always be easy, however.

Wax and Kailath [5] developed a particularly simple process for making this determination. It makes direct use of the array covariance matrix estimate eigenanalysis. They showed that the criterion can be expressed as

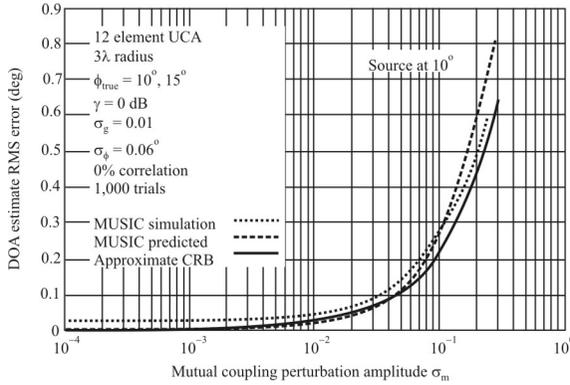


Figure 4.4 MUSIC performance versus mutual coupling perturbation.

$$\text{MDL} = -\log_{10} \left[ \frac{\prod_{n=m+1}^M \hat{\lambda}_n^{1/(M-m)}}{1} \right]^{(M-m)N} + \frac{1}{2} m(2M - m + 1) \log_{10} N \quad (4.22)$$

In (4.22),  $\hat{\lambda}_i$  are the eigenvalues of the sample covariance matrix  $\hat{R}$ . The number of incident sources is then determined by finding that integer  $m \in \{0, 1, 2, \dots, M - 1\}$  that minimizes this criterion. Thus,

$$\hat{L} = m_{\min} \quad (4.23)$$

Wax and Kailath showed that this criterion yields a consistent estimated of the number of sources [5].

**Example [5]**

Consider an array with seven sensors ( $M = 7$ ) and two sources ( $L = 2$ ) with AOAs  $20^\circ$  and  $25^\circ$ . The SNR for both signals was 10 dB. Using  $N = 100$  samples, the resulted eigenvalues of the sample-covariance matrix were 21.2359, 2.1717, 1.4279, 1.0979, 1.0544, 0.9432, and 0.7324. Observe the gradual decrease of the eigenvalues; it is clear that the separation of the three “smallest” eigenvalues from the two “large” ones is a difficult task. However, applying the approach described here yielded the

**Table 4.1** Eigenvalues for the Example.

$L$	0	1	2	3	4	5	6
$\lambda$	590.4	67.2	64.9	80.7	94.5	104.2	110.5

Source: [5].

values in Table 4.1. The minimum of the MDL is obtained, as expected, for  $L = 2$ , although the difference between  $L = 1$  and  $L = 2$  is not large.

## 4.6 Effect of Phase Errors on the Accuracy of MUSIC

Brandwood documented an investigation of the effects of phase errors on the MUSIC algorithm [6]. This investigation included the effects of both azimuth and elevation angle measurement distortions. It was assumed that the effects of the phase errors were so significant that the usual effects of noise (both amplitude and phase) could be ignored. The results of that investigation are presented in this section.

### 4.6.1 Introduction

In practical parameter estimation systems it is often calibration and other system errors, rather than receiver noise, which limit the performance. In antenna systems used for AOA measurements the dominant errors are most likely to be phase errors—the difference between the stored (computed or calibrated) phase values of the array description and the values actually seen by the signal. Phase errors may arise, for example, as a result of local array site multipath effects, from small errors in the knowledge of element positions or from mismatches in receiver channels. With tactical ES systems where the antenna resides on an erectable mast, raising and lowering the mast can cause shifts in the relative positions of the antenna in the array, which, in turn, invalidates any calibration to some degree. Of course, amplitude errors will also arise, which also limits the performance, but generally they are less serious and more easily controlled. (For example, element position errors have little or no effect on amplitude response.)

Clearly it is desirable to know how the performance depends on the errors and what error tolerances may be allowed for a given required performance. This is particularly the case for DF operation at HF where both azimuth and elevation measurements are often required, or at all frequencies when the sensor platform is elevated, such as on UAS aircraft, helicopter platforms or higher flying fixed winged aircraft. Again, at HF, frequently the arrays are planar, with groundwave signals arriving in the plane of the array, and other signals arriving with *near vertical incidence* (NVI). The elevation accuracy for this type of array is relatively

very poor for the ground wave signals and the azimuth accuracy is poor for the NVI signals.

In this section expressions are obtained for the measurement variance, for both azimuth and elevation, in terms of the phase error variance and the array geometry, using MUSIC as the DF processing method. MUSIC is a widely known and used superresolution method, often used as a standard for comparison with other methods, which we will do at the end of this chapter. The results are derived here only for the case of a single signal present; for multiple, well-separated signals (with the number of signals considerably less than the number of array elements), the accuracies should be close to these results. However, for close signals the problems of accuracy and resolution are related [1] and the analysis presented in this section is not appropriate. The derived expressions are plotted for a number of arrays, showing the expected variation in accuracy with azimuth or elevation. Simulation results are shown that agree well with the theoretical curves.

#### 4.6.2 Accuracy

Even with a perfect manifold containing the exact gain vectors seen by the signal, the system performance will be limited because the ever-present receiver noise will perturb the estimate of the signal space basis. Here we ignore the effect of noise, taking the case of a single large signal, with random phase errors, rather than noise, limiting the performance. We assume the gain vector seen by the signal  $\vec{r}$  becomes the single signal space basis vector. We assume that  $\vec{r}$  has phase errors in each component relative to  $\vec{a}(\vec{\theta}_0)$ , the manifold vector for the signal direction, and we ignore inaccuracies due to the finite sampling of the manifold. We want to determine the value of  $\vec{\theta}$ , say,  $\vec{\theta}_0 + \vec{\delta}_0$ , that gives the manifold vector closest to  $\vec{r}$ , as this is what will be given by MUSIC;  $\vec{\delta}_0$  is then the error in the measurement of  $\vec{\theta}_0$ .

In the case of a single signal, with signal space bases  $\vec{r}$ , the MUSIC function, in the form of the square modulus of the projection of  $\vec{a}(\vec{\theta})$  orthogonal to  $\vec{r}$  is

$$f(\vec{\theta}) = 1 - \left| \vec{r}^H \vec{a}(\vec{\theta}) \right|^2 \quad (4.24)$$

where, without loss of generality,  $\vec{a}$  and  $\vec{r}$  are taken to be normalized. Let the components of  $\vec{a}$  and  $\vec{r}$  be represented by

$$\begin{aligned} a_k &= A_k \exp(j\alpha_k) \\ r_k &= \rho_k \exp(j\beta_k) \end{aligned}$$

where  $x_k$  and  $y_k$  are real amplitude factors and  $\alpha_k$  and  $\beta_k$  are real phase values. Then, from (4.24),

$$f(\vec{\theta}) = 1 - \left| \sum_k x_k y_k \exp[j\varepsilon_k(\vec{\theta})] \right|^2 \quad (4.25)$$

where

$$\varepsilon_k(\vec{\theta}) = \alpha_k(\vec{\theta}) - \beta_k(\vec{\theta})$$

We now consider the case where the elements all have the same gain magnitude in any given direction. (This is a common case in practice, when the elements are similar and oriented in the same way—vertical monopoles for example.) In this case  $A_k = \rho_k = 1/\sqrt{M}$  for all  $k$ , giving

$$f(\vec{\theta}) = 1 - \left| \sum_k \frac{1}{M} \exp[j\varepsilon_k(\vec{\theta})] \right|^2 \quad (4.26)$$

The first few terms of the series expansion for  $\exp(x)$  are given by  $\exp(x) \approx 1 + x + x^2/2$ , so for small  $x = \exp[j\gamma_k(\vec{\theta})]$  we have for variables  $v_k$

$$\begin{aligned} \left| \frac{1}{M} \sum \exp(jv_k) \right|^2 &\approx \left| 1 + \frac{1}{M} \sum (jv_k - v_k^2) \right|^2 \\ &= \left| j\bar{v} - \frac{\bar{v}^2}{2} \right|^2 = \left( 1 - \frac{\bar{v}^2}{2} \right)^2 + \bar{v}^2 \\ &= -(\bar{v}^2 - \bar{v}^2) + \left( \frac{\bar{v}^2}{2} \right)^2 \approx -(\bar{v}^2 - \bar{v}^2) = 1 - \text{var}(v) \end{aligned}$$

As such, (4.26) becomes

$$f(\vec{\theta}) \approx \text{var}[\varepsilon_k(\vec{\theta})] \quad (4.27)$$

Thus, we see that minimizing  $f$  is the same as minimizing  $\text{var}[\varepsilon(\vec{\theta})]$ .

### 4.6.3 Solutions for Errors

We now express  $\vec{\epsilon}$  in terms of the measurement error  $\vec{\delta}_\theta$  (assumed small) and the phase errors,  $\vec{\delta}_\alpha$ , such that  $\vec{\beta}(\vec{\theta}_0) = \vec{\alpha}(\vec{\theta}_0) + \vec{\delta}_\alpha$ . The phase of an element of a manifold vector at  $\vec{\theta} = \vec{\theta}_0 + \vec{\delta}_\theta$  is, again using the first order (linear) Taylor approximation (recall that  $\vec{\theta} = [\phi \quad \varphi]^T$ ),

$$\alpha_k(\vec{\theta}) \approx \alpha_k(\vec{\theta}_0) + \frac{\partial \alpha_k(\vec{\theta}_0)}{\partial \phi} \delta_\phi + \frac{\partial \alpha_k(\vec{\theta}_0)}{\partial \varphi} \delta_\varphi \quad (4.28)$$

or

$$\alpha_k(\vec{\theta}) \approx \alpha_k(\vec{\theta}_0) + [\vec{p} \quad \vec{q}] \begin{bmatrix} \delta_\phi \\ \delta_\varphi \end{bmatrix} = \alpha(\vec{\theta}_0) + \mathbf{P} \vec{\delta}_\theta \quad (4.29)$$

where

$$\begin{aligned} \mathbf{P} &= [\vec{p} \quad \vec{q}] \\ \vec{p} &= \frac{\partial \alpha_k(\vec{\theta}_0)}{\partial \phi} \\ \vec{q} &= \frac{\partial \alpha_k(\vec{\theta}_0)}{\partial \varphi} \\ \vec{\delta}_\theta &= [\delta_\phi \quad \delta_\varphi]^T \end{aligned}$$

so

$$\vec{\epsilon}(\vec{\theta}) = \vec{\alpha}(\vec{\theta}) - \vec{\beta}(\vec{\theta}_0) = \vec{\alpha}(\vec{\theta}_0) + \mathbf{P} \vec{\delta}_\theta - [\vec{\alpha}(\vec{\theta}_0) + \vec{\delta}_\alpha] = \mathbf{P} \vec{\delta}_\theta - \vec{\delta}_\alpha \quad (4.30)$$

Expressing  $\text{var}[\vec{\epsilon}(\vec{\theta})]$  in terms of  $\vec{\delta}_\theta$  and  $\vec{\delta}_\alpha$ , we get

$$\begin{aligned} \text{var}[\vec{\epsilon}(\vec{\theta})] &= \frac{1}{M} \sum \epsilon_k^2 - \left( \frac{1}{M} \sum \epsilon_k \right)^2 = \frac{1}{M} \|\vec{\epsilon}\|^2 - \left[ \frac{1}{M} \vec{1}^T \vec{\epsilon} \right]^2 \\ &= \frac{1}{M} \vec{\epsilon}^T \vec{\gamma} - \frac{1}{M^2} \vec{\epsilon}^T \vec{1} \vec{1}^T \vec{\epsilon} = \vec{\epsilon}^T \mathbf{V} \vec{\epsilon} \end{aligned} \quad (4.31)$$

where

$$\mathbf{V} = \frac{1}{N^2} (\mathbf{M}\mathbf{I} - \vec{\mathbf{1}}\vec{\mathbf{1}}^T)$$

is a variance matrix and the vector  $\vec{\mathbf{1}}$  is given by  $\vec{\mathbf{1}} = [1 \ 1 \ \dots \ 1]^T$  of appropriate size. Substituting for  $\vec{\mathbf{e}}$ , we have

$$\begin{aligned} \text{var}[\vec{\mathbf{e}}(\vec{\theta})] &= (\mathbf{P}\vec{\delta}_\theta - \vec{\delta}_\alpha)^T \mathbf{V} (\mathbf{P}\vec{\delta}_\theta - \vec{\delta}_\alpha) \\ &= \vec{\delta}_\theta^T \mathbf{P}^T \mathbf{V} \mathbf{P} \vec{\delta}_\theta - 2\vec{\delta}_\theta^T \mathbf{P}^T \mathbf{V} \vec{\delta}_\alpha + \vec{\delta}_\alpha^T \mathbf{V} \vec{\delta}_\alpha \end{aligned} \quad (4.32)$$

Note that for any two variables  $z$  and  $c$ , two vectors  $\vec{x}$  and  $\vec{m}$ , and matrix  $\mathbf{B}$ , if  $z = \vec{x}^T \mathbf{B} \vec{x} - 2\vec{x}^T \vec{m} + c$  with  $\mathbf{B}$  symmetric, then, differentiating with respect to  $\vec{x}$ ,  $\vec{\nabla}_z = 2\mathbf{B}\vec{x} - 2\vec{m}$ , and we see that the function is stationary when  $\vec{\nabla}_z = \vec{0}$ , that is, when  $\vec{x} = \mathbf{B}^{-1}\vec{m}$ .

So the value of  $\vec{\delta}_\theta$  that minimizes  $\text{var}[\vec{\mathbf{e}}(\vec{\theta})]$  is given by

$$\vec{\delta}_\theta = (\mathbf{P}^T \mathbf{P})^{-1} \mathbf{P}^T \mathbf{V} \vec{\delta}_\alpha = \mathbf{S} \vec{\delta}_\alpha \quad (4.33)$$

where

$$\mathbf{S} = \begin{bmatrix} \vec{p}^T \mathbf{V} \vec{p} & \vec{p}^T \mathbf{V} \vec{q} \\ \vec{q}^T \mathbf{V} \vec{p} & \vec{q}^T \mathbf{V} \vec{q} \end{bmatrix}^{-1} \begin{bmatrix} \vec{p}^T \mathbf{V} \\ \vec{q}^T \mathbf{V} \end{bmatrix}$$

Putting  $\mathbf{S} = \begin{bmatrix} s_1^T & s_2^T \end{bmatrix}^T$  and  $\vec{\delta}_\theta = \begin{bmatrix} \delta_\phi & \delta_\varphi \end{bmatrix}^T$ , we get

$$\delta_\phi = s_1^T \vec{\delta}_\alpha = \sum_k s_{1k} \delta_{\alpha_k} = s_{11} \delta_{\alpha_1} + s_{12} \delta_{\alpha_2} + \dots + s_{1N} \delta_{\alpha_N} \quad (4.34)$$

$$\delta_\varphi = s_2^T \vec{\delta}_\alpha = \sum_k s_{2k} \delta_{\alpha_k} = s_{21} \delta_{\alpha_1} + s_{22} \delta_{\alpha_2} + \dots + s_{2N} \delta_{\alpha_N} \quad (4.35)$$

#### 4.6.4 Statistics

The results provide the measurement errors, given the array geometry, that defines  $\mathbf{S}$ , and the actual phase errors present,  $\vec{\delta}_\alpha$ . If the errors are not known, but their

statistics can be estimated, then we can obtain statistics for the measurement errors. The mean of  $\delta_\phi$  is  $\mathcal{E}\{\delta_\phi\} = \overline{\delta_{\phi_k}} = \sum_k s_{1k} \delta_{\alpha_k} \overline{\delta_{\alpha_k}}$  and if we assume that the mean phase errors in all the channels have the same value (a reasonable first order assumption),  $\overline{\delta_\alpha}$ , then we have

$$\overline{\delta_{\phi_k}} = s_1^T \vec{\delta}_\alpha = \sum_k s_{1k} \delta_{\alpha_k} = s_1^T \vec{1} \overline{\delta_\alpha} \quad (4.36)$$

Similarly

$$\overline{\delta_{\phi_k}} = s_2^T \vec{\delta}_\alpha = s_2^T \vec{1} \overline{\delta_\alpha} \quad (4.37)$$

The value of  $\delta_\phi^2$  is given by  $\delta_\phi^2 = \sum_i \sum_j s_{1i} s_{1j} \delta_{\alpha_i} \delta_{\alpha_j}$  so the mean square value of  $\delta_\phi$  depends on the cross terms  $\mathcal{E}\{\delta_{\alpha_i} \delta_{\alpha_j}\}$  as well as the square terms  $\mathcal{E}\{\delta_{\alpha_i}^2\}$ . If we assume that the mean cross-terms are zero and the square terms are all equal,  $\overline{\delta_\alpha^2}$ , then

$$\overline{\delta_\phi^2} = \sum_k s_{1k}^2 \overline{\delta_\alpha^2} = \vec{s}_1^T \vec{s}_1 \overline{\delta_\alpha^2} = \|\vec{s}_1\|^2 \overline{\delta_\alpha^2} \quad (4.38)$$

and

$$\overline{\delta_\phi^2} = \vec{s}_2^T \vec{s}_2 \overline{\delta_\alpha^2} = \|\vec{s}_2\|^2 \overline{\delta_\alpha^2} \quad (4.39)$$

The variances of  $\delta_\phi$  and  $\delta_\phi$  are therefore given by

$$\text{var}(\delta_\phi) = \overline{\delta_\phi^2} - \overline{\delta_\phi}^2 = \|\vec{s}_1\|^2 \overline{\delta_\alpha^2} - (\vec{s}_1^T \vec{1} \overline{\delta_\alpha})^2 \quad (4.40)$$

and

$$\text{var}(\delta_\phi) = \overline{\delta_\phi^2} - \overline{\delta_\phi}^2 = \|\vec{s}_2\|^2 \overline{\delta_\alpha^2} - (\vec{s}_2^T \vec{1} \overline{\delta_\alpha})^2 \quad (4.41)$$

If the phase error distributions have zero means ( $\overline{\delta_\alpha} = 0$ ) then  $\overline{\delta_\phi} = 0$ ,  $\overline{\delta_\phi} = 0$ ,

$$\text{var}(\delta_\phi) = \overline{\delta_\phi^2} = \|\vec{s}_1\|^2 \overline{\delta_\alpha^2} \quad (4.42)$$

and

$$\text{var}(\delta_\varphi) = \overline{\delta_\varphi^2} = \|\vec{s}_2\|^2 \delta_\alpha^2 \quad (4.43)$$

To determine  $\mathbf{S}$ , we need the vectors  $\vec{p}$  and  $\vec{q}$ . The phase at element  $k$  for a signal in direction  $\vec{\theta} = [\phi \ \varphi]^T$  is given by

$$\alpha_k = \frac{2\pi}{\lambda} \vec{x}_k^T \vec{u}(\vec{\theta}) \quad (4.44)$$

where  $\vec{x}_k$  is the position vector of element  $k$  and  $\vec{u}(\vec{\theta})$  is the unit vector in direction  $\vec{\theta}$  and  $\lambda$  is the wavelength. Using cylindrical coordinates  $(\tilde{r}_k, \psi_k, z_k)$  for antenna  $k$ , we have

$$\begin{aligned} \alpha_k(\vec{\theta}) &= \frac{2\pi}{\lambda} [\tilde{r}_k \cos \psi_k \quad \tilde{r}_k \sin \psi_k \quad z_k] [\cos \phi \cos \phi \quad \cos \phi \sin \phi \quad \sin \phi]^T \\ &= \frac{2\pi}{\lambda} [\tilde{r}_k \cos \phi \cos(\psi_k - \phi) + z_k \sin \phi] \end{aligned} \quad (4.45)$$

producing the components of  $\vec{p}$  and  $\vec{q}$  as

$$p_k = \frac{\partial \alpha_k(\vec{\theta}_0)}{\partial \phi} = \frac{2\pi}{\lambda} \tilde{r}_k \cos \phi_0 \sin(\psi_k - \phi_0) \quad (4.46)$$

$$q_k = \frac{\partial \alpha_k(\vec{\theta}_0)}{\partial \varphi} = \frac{2\pi}{\lambda} [-\tilde{r}_k \sin \phi_0 \cos(\psi_k - \phi_0) + z_k \cos \phi_0] \quad (4.47)$$

#### 4.6.5 Horizontal Planar Arrays

When considering the important case of horizontal planar arrays (a typical case),  $z_k = 0$ , and for groundwave signals, for which  $\varphi = 0$ , we see that  $\vec{q} = 0$  for these arrays. Although the expressions hold to quite low elevations, for this case the elevation error expressions are unbounded because the assumptions are violated. The expressions for azimuth errors for groundwave signals, however, are obtained by replacing  $\mathbf{P}$  by  $\vec{p}$  and  $\mathbf{S}$  by  $\vec{s} = \vec{p}^T \mathbf{V} / \vec{p}^T \mathbf{V} \vec{p}$ , producing

$$\delta_\phi = \frac{\vec{p}^T \mathbf{V} \vec{\delta}_\alpha}{\vec{p}^T \mathbf{V} \vec{p}} \quad (4.48)$$

We note that  $\vec{1}^T \vec{\delta}_\alpha = M \bar{\delta}_\alpha$ , which is zero if the errors have zero means, and in this case

$$\mathbf{V} \vec{\delta}_\alpha = \frac{1}{M^2} [\mathbf{M} \mathbf{I} - \vec{1} \vec{1}^T] = \frac{1}{M} \vec{\delta}_\alpha \quad (4.49)$$

Also  $\vec{p}^T \mathbf{V} \vec{p}$  is the variance of the elements of  $\vec{p}$  or  $\vec{p}_{\text{RMS}}^2$  where  $\vec{p}_{\text{RMS}}$  is the mean square value of the components of  $\vec{p}$ , assuming the mean value is small.<sup>4</sup> Thus we have

$$\vec{\delta}_\theta = \frac{\vec{p}^T}{M \vec{p}_{\text{RMS}}^2} \vec{\delta}_\alpha \quad (4.50)$$

and

$$\overline{\delta_\phi^2} = \frac{1}{M^2 \vec{p}_{\text{RMS}}^4} \sum_k p_k^2 \overline{\delta_\alpha^2} \quad (4.51)$$

if the covariance matrix of  $\delta_\alpha$  is  $\overline{\delta_\alpha^2} \mathbf{I}$  as assumed above. Putting  $\sum_k p_k^2 = M \vec{p}_{\text{RMS}}^2$ , we have

$$\overline{\delta_\phi^2} = \frac{1}{M \vec{p}_{\text{RMS}}^2} \overline{\delta_\alpha^2} \quad (4.52)$$

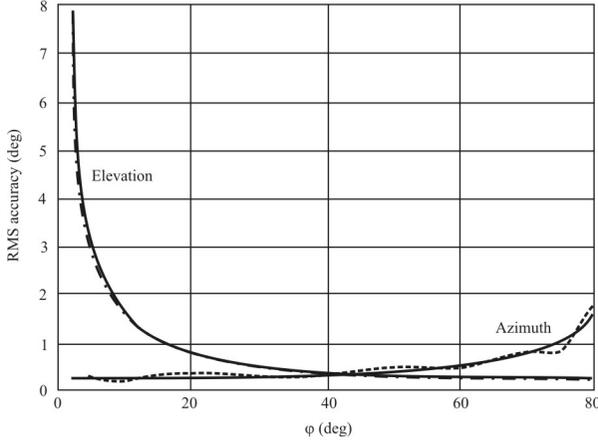
or

$$\phi_{\text{RMS}} = \frac{1}{\sqrt{M} \vec{p}_{\text{RMS}}} \delta_{\alpha_{\text{RMS}}} \quad (4.53)$$

At  $\phi = 0$  we have  $p_k = (2\pi/\lambda) \tilde{r}_k \sin(\psi_k - \phi) = (2\pi/\lambda) y_k(\phi)$  where  $y_k(\phi)$  is the length of the projection of the element position vector orthogonal to the signal direction. Thus, finally we have

---

<sup>4</sup> In fact, the mean is zero if the centroid of the array is its coordinate origin.



**Figure 4.5** RMS accuracy for UCA. Solid lines = theory, dashed lines = simulation.

$$\phi_{\text{RMS}} = \frac{1}{\sqrt{M} (2\pi/\lambda) y_{\text{RMS}}(\phi)} \delta_{\sigma_{\text{RMS}}} \quad (4.54)$$

#### 4.6.6 Simulations

We present the results of some typical simulations to verify the above theoretical results. In the figures the solid curves give the theoretical results and the broken curves are the results of simulation for the same case. For the simulations a single large noise-like signal with the element SNR =  $\gamma = 40$  dB, with integration over 50 snapshots and 40 runs were used to give the RMS errors. Antenna phase errors were extracted from a zero mean normal distribution with RMS value of  $3^\circ$ .

Figures 4.5 and 4.6 are for a six-element *uniform circular array* (UCA) of one wavelength radius. Therefore the array elements are  $60^\circ$  apart. Figure 4.5 shows the azimuth and elevation errors against elevation, and we see a very good agreement between theory and simulation. Figure 4.6 shows no variation with azimuth (at an elevation of  $30^\circ$ ) because of the circular symmetry of the UCA.

For the less symmetrical case, a five-element array with a length of two wavelengths and a width of a half wavelength was used for the remaining figures. Figure 4.7 shows the errors with elevation varied at an azimuth of  $0^\circ$ , along the length of the array, its maximum aperture for the elevated signals. The agreement again is very good. At an azimuth angle of  $90^\circ$  (Figure 4.8), when we are looking across the narrow dimension of the array, the elevation errors are rather large, especially for the lower elevation angles. The theory curve is in error at low angles as the small error approximations are no longer valid. At an azimuth of  $30^\circ$  (Figure

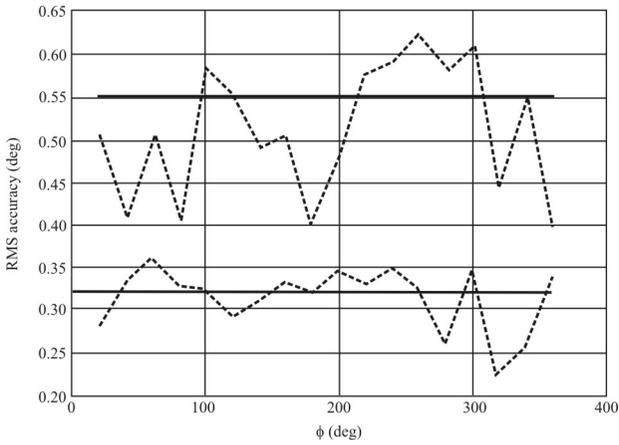


Figure 4.6 RMS accuracy for UCA. Solid lines = theory, dashed lines = simulation.

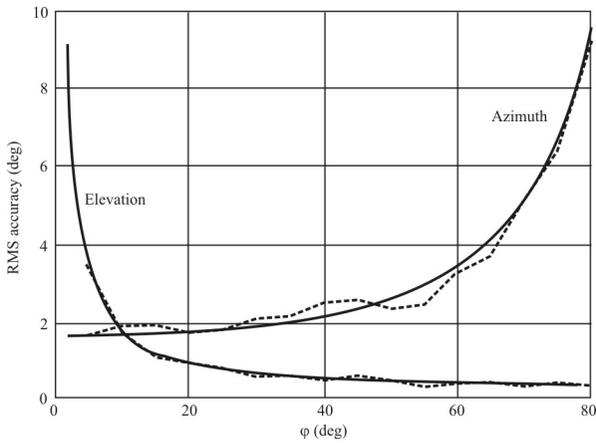
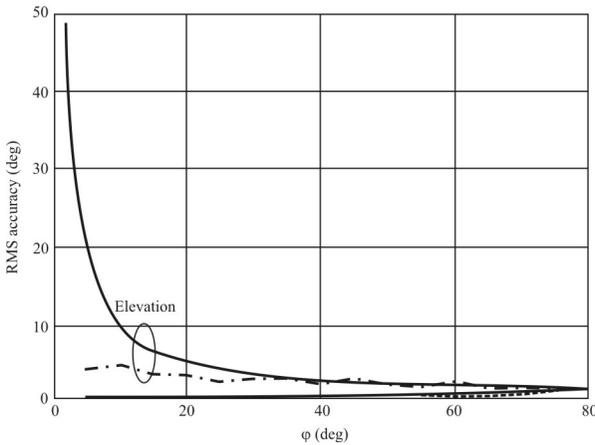


Figure 4.7 RMS accuracy for 5 element linear array,  $\phi = 0^\circ$  (along the longest length). Solid lines = theory, dashed lines = simulation.



**Figure 4.8** RMS accuracy for 5 element linear array  $\phi = 90^\circ$  (along the shortest length). Solid lines = theory, broken lines = simulation.

4.9), the theory seems consistently to overestimate the errors found in simulation by a factor of about 2. Finally Figure 4.10 shows the errors with azimuth, and the variation with aperture, over a factor of 4, as expected. The simulation results approximately follow the theoretical curves, but tend to be about a degree lower.

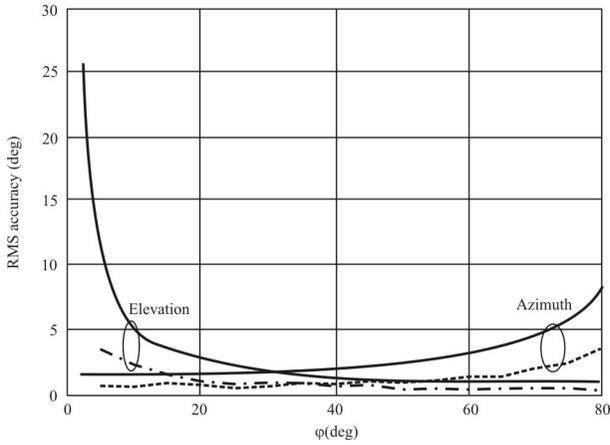
#### 4.6.7 Summary

Theoretical expressions estimating the accuracy of AOA measurement with MUSIC, when limited by phase error have been derived and these results have been generally confirmed by simulations of some specific cases.

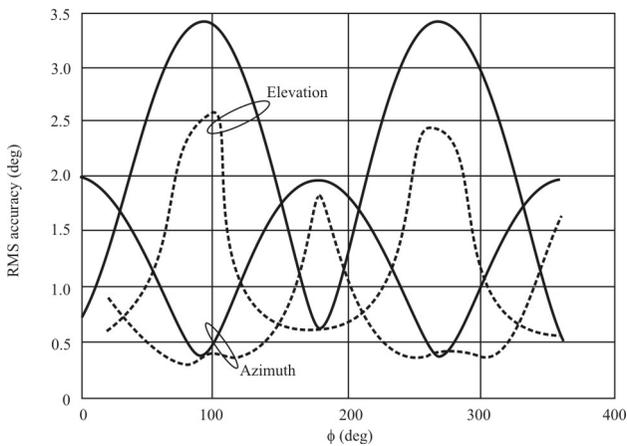
Phase errors are perhaps the most dominant form in error in many systems and knowledge of how the performance of the system depends on these errors and assists system designers by indicating either what performance may be expected, given an error level, or what level errors should be reduced to in order to achieve a required performance.

## 4.7 Other Superresolution Algorithms

Some other algorithms for high-resolution AOA estimation are described in this section. We do not go into the derivation of these approaches and simply briefly describe them. The ones identified here were included in a comparison against each other and MUSIC by Johnson and Miner [7].



**Figure 4.9** RMS accuracy for five-element linear array,  $\phi = 30^\circ$ . Solid lines = theory, dashed lines = simulation.



**Figure 4.10** RMS accuracy for five-element linear array,  $\phi = 30^\circ$ . Solid lines = theory, dashed lines = simulation.

### 4.7.1 Maximum Likelihood Method

The *maximum likelihood method* (MLM) estimates the AOAs from a given set of array samples by maximizing the log-likelihood function [8, 9]. The likelihood function is the joint PDF of the sampled data given the AOAs and viewed as a function of the desired variables, which are the AOAs in this case. The method searches for those directions that maximize the log of this function. The MLM criterion signifies that plane waves from these directions are most likely to cause the given samples to occur.

The MLM power spectrum is given by

$$P_{\text{MLM}}(\vec{\theta}_k) = \frac{1}{\vec{a}^H(\vec{\theta}_k)\mathbf{C}_{\text{rr}}^{-1}\vec{a}(\vec{\theta}_k)}, \quad k = 1, 2, \dots, \mathcal{D} \quad (4.55)$$

Based on (4.55), we search the array manifold for those azimuth and elevation AOAs where peaks in the power spectrum occur. Those are the assumed AOAs of the impinging signals.

### 4.7.2 Adaptive Angular Response

The *adaptive angular response* (AAR) method is an adaption of the MLM described above. The algorithm normalizes the output power of the array to a thermal noise term [7, 8, 10]. The power spectrum is given by

$$P_{\text{AAR}}(\vec{\theta}_k) = \frac{\vec{a}^H(\vec{\theta}_k)\mathbf{C}_{\text{rr}}^{-1}\vec{a}(\vec{\theta}_k)}{\vec{a}^H(\vec{\theta}_k)\mathbf{C}_{\text{rr}}^{-1}\vec{a}(\vec{\theta}_k)}, \quad k = 1, 2, \dots, \mathcal{D} \quad (4.56)$$

### 4.7.3 Thermal Noise Algorithm

The *thermal noise algorithm* (TNA) is also derived from the MLM. It uses only the thermal noise term in (4.56) (the denominator) so the power spectrum is of the form [7, 8]

$$P_{\text{TNA}}(\vec{\theta}_k) = \frac{1}{\vec{a}^H(\vec{\theta}_k)\mathbf{C}_{\text{rr}}^{-2}\vec{a}(\vec{\theta}_k)}, \quad k = 1, 2, \dots, \mathcal{D} \quad (4.57)$$

### 4.7.4 Maximum Entropy Method

The *maximum entropy* (ME) method finds a power spectrum such that its Fourier transform equals the measured correlation subjected to the constraint that its

entropy is maximized.<sup>5</sup> The entropy of a Gaussian band-limited time series with power spectrum  $S(f)$  is defined as

$$H(S) = \int_{-f_N}^{f_N} \ln S(f) df \quad (4.58)$$

where  $f_N$  is the Nyquist frequency.

For estimating AOAs from the measurements using an array of sensors, the ME method finds a continuous function  $P_{ME}(\vec{\theta}) > 0$  such that it maximizes the entropy function; that is

$$\max_{\vec{\theta}} H[P(\vec{\theta})] = \max_{\vec{\theta}} \int_0^{2\pi} \ln P_{ME}(\vec{\theta}) d\vec{\theta} \quad (4.59)$$

subject to the constraint that the measured correlation between the  $i$ th and the  $j$ th elements  $\rho_{ij}$  satisfies

$$\rho_{ij} = \int_0^{2\pi} P_{ME}(\vec{\theta}) \cos[2\pi f \tau_{ij}(\vec{\theta})] d\vec{\theta} \quad (4.60)$$

where  $\tau_{ij}(\vec{\theta})$  denotes the differential delay between elements  $i$  and  $j$  due to a source in  $\vec{\theta}$  direction. The solution to this problem requires an infinite dimensional search. The problem may be transformed to a finite dimensional search using the duality principle [11] leading to

$$P_{ME}(\vec{\theta}) = \frac{1}{\hat{w} \vec{q}(\vec{\theta})} \quad (4.61)$$

In (4.61),  $\hat{w}$  is obtained by minimizing

$$\min_{\vec{w}} H(\vec{w}) = \min_{\vec{w}} \int_0^{2\pi} \ln[\vec{w}^T \vec{q}(\vec{\theta})] d\vec{\theta} \quad (4.62)$$

subject to

---

<sup>5</sup> The appellation entropy is derived from thermodynamics—the entropy of a process is the amount of “randomness” of that process and therefore is a measure of the information in the process.

$$\vec{w}^T \vec{\rho} = 2\pi \quad (4.63)$$

and

$$\vec{w}^T \vec{q}(\vec{\theta}) = 0 \quad \forall \vec{\theta} \quad (4.64)$$

where  $\vec{q}(\vec{\theta})$  and  $\vec{\rho}$ , respectively, are defined as

$$\vec{q}(\vec{\theta}) = \left[ 1 \quad \sqrt{2} \cos[2\pi f \tau_{12}(\vec{\theta})] \quad \sqrt{2} \cos[2\pi f \tau_{13}(\vec{\theta})] \quad \dots \right]^T \quad (4.65)$$

and

$$\vec{\rho} = \left[ \rho_{11} \quad \sqrt{2}\rho_{12} \quad \sqrt{2}\rho_{13} \quad \dots \right]^T \quad (4.66)$$

It should be noted that the dimension of these vectors depends on the array geometry and is equal to the number of known correlations  $\rho_{ij}$  for every possible  $i$  and  $j$ .

The minimization problem defined above may be solved iteratively using the standard gradient LMS algorithm.

The variation of the ME method here was developed by Lang and McClellan [12]. The ME spectrum is specified by the positive polynomial  $P$  defined as

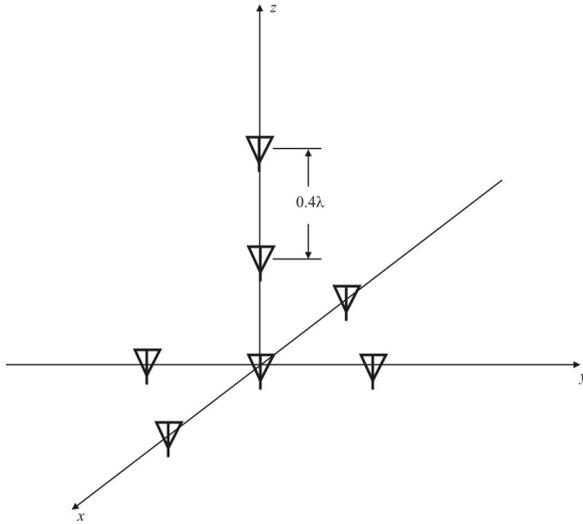
$$P_{\text{MEM}} = \frac{1}{P} = \frac{1}{\sum p(x) e^{-j\beta \cdot x}} \quad (4.67)$$

### 4.7.5 Comparisons

Johnson and Miner investigated a comparison of these techniques with each other as well as with MUSIC [7]. In particular, they were interested in comparing the performance for application to HF skywave DF. This implies that both azimuth and elevation angles need to be determined. As such, they simulated the three-dimensional, seven-element array shown in Figure 4.11 at 4 MHz. The distance between elements was 30 m, which yields an array spacing of  $0.4\lambda$  as shown.

#### 4.7.5.1 Signal Separation

The first comparison evaluated the algorithm's ability to separate multiple signals. These results are shown in Figure 4.12. From left to right, the algorithms are

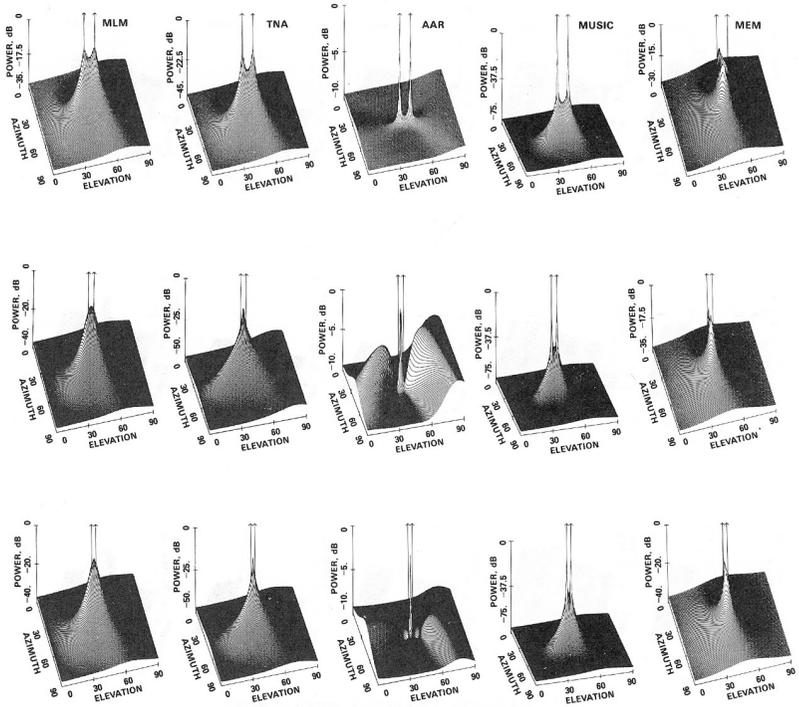


**Figure 4.11** Seven-element array.

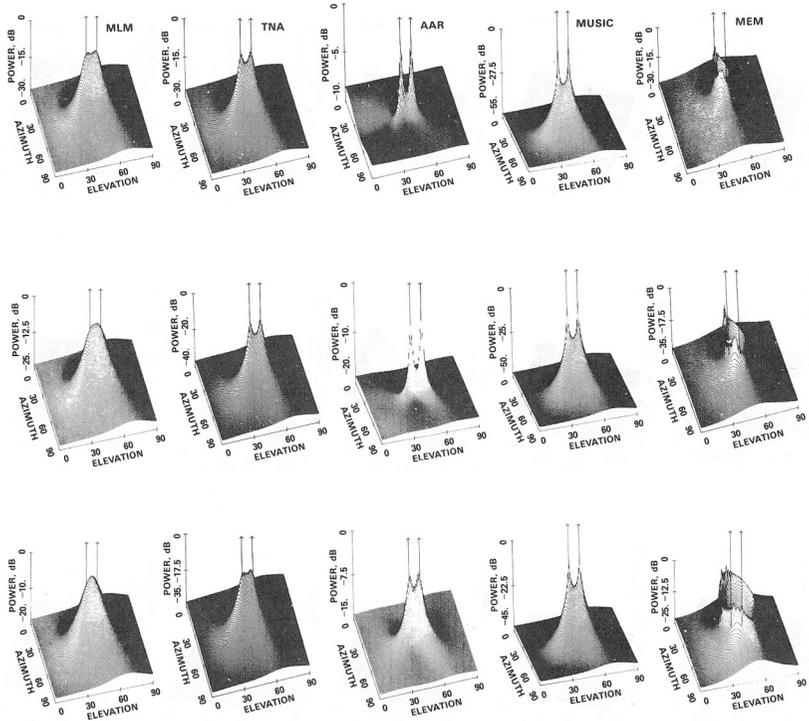
MLM, TNA, AAR, MUSIC, and the ME method. The two signals are indicated by the vertically arrowed lines. In all cases except for MEM, all algorithms correctly separated the signals when the elevation angle was  $10^\circ$ . For ME, the results were biased  $1^\circ$  away from the correct result. For  $5^\circ$ , the MLM spectrum shows only one broad peak. The TNA and AAR spectra show two narrow peaks separated by  $1^\circ$  in elevation and biased  $2^\circ$  from the correct value. The MUSIC spectrum shows two unbiased narrow peaks. For  $4^\circ$  separation, the MLM, TNA, and AAR spectra have a single peak at  $42^\circ$  elevation, while the MUSIC peaks are correct.

#### 4.7.5.2 SNR Simulation

Figure 4.13 shows the results of varying the SNR of the signals. At  $\gamma = 10$  dB, all the procedures except ME resolved the signals correctly. The peaks for MUSIC are the most clearly defined, followed by the AAR, TNA, and MLM. The ME spectrum is a hollow with the larger peaks displaced  $\pm 3^\circ$  in azimuth and at  $45^\circ$  elevation. At 5 dB, the MLM algorithm produces a single broad peak. The TNA, AAR, and MUSIC spectra results are correct. At 0 dB the MLM spectrum shows a single broad peak, just as at 5 dB while the TNA spectrum shows two peaks that are barely resolvable. The AAR and MUSIC peaks are resolvable and at the correct locations.



**Figure 4.12** Signal separation. Equal amplitudes, mutually incoherent,  $\gamma = 30$  dB,  $\phi = 45^\circ$ , nominal  $\phi = 45^\circ$ ,  $\Delta\phi = 10^\circ, 5^\circ, 4^\circ$ . (Source: [7]. © IEEE 1986. Reprinted with permission.)



**Figure 4.13** SNR simulation. Equal amplitude, mutually incoherent,  $\phi = 45^\circ$ ,  $\varphi = 40^\circ$  and  $50^\circ$ ,  $\gamma = 10$ , 5, 0 dB. (Source: [7]. © IEEE 1986. Reprinted with permission.)

#### 4.7.5.3 Coherent Signals

While [7] contains other comparisons of these algorithms, the last one we will include here is the performance when the signals are varying degrees of coherence. The results are shown in Figure 4.14.

Although these algorithms are not confined to the HF range for their performance, the HF range can certainly produce coherent cochannel interference due to the multiple reflections possible off the ionosphere.

The signals are not resolved by MLM, TNA, and AAR when the signals have equal amplitude and  $\text{coh} = 0.8$  (top row). The MUSIC algorithm correctly resolves the peaks, while the ME procedure shows two incorrectly resolved peaks. In the second row, the signal at  $45^\circ$  azimuth and  $40^\circ$  elevation has unit amplitude while the signal at  $50^\circ$  elevation has an amplitude of 0.1. The mutual coherence in this case was 0.8. The MLM, TNA, and AAR spectra show a single resolved peak corresponding to the stronger signal with a  $1^\circ$  bias in the direction of the weaker signals. MUSIC produced the correct spectra, while ME shows a single peak midway between the signals. The third row corresponds to equal amplitude signals, with mutual coherence of 0.3. The MLM produces two barely resolved peaks, while TNA, AAR, and ME did not resolve the signals. The MUSIC algorithm correctly found the peaks.

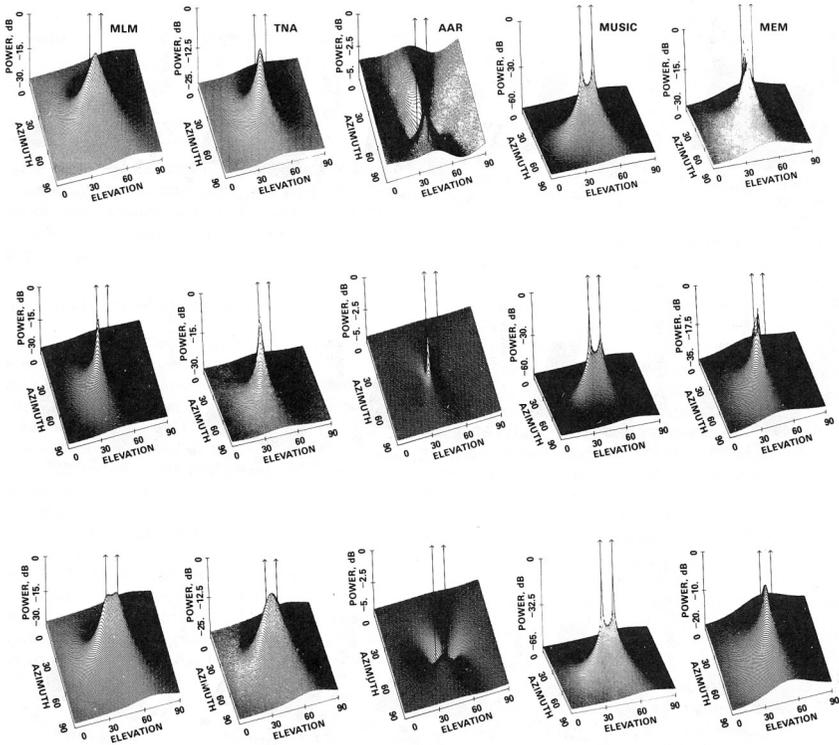
MUSIC's greatest shortcoming is frequently stated to be its performance with correlated signals. We can see from these examples that the signals must be significantly correlated ( $\text{coh} > 0.9$ ) before MUSIC starts having difficulties resolving two correlated signals.

#### 4.7.5.4 Summary

MUSIC performed far better than any of the other algorithms for the cases considered. The same can be said for the other results in [7] that are not included here. Even in the case of relatively high coherence, where MUSIC is susceptible to degraded performance, it performed remarkably well. Two signals must be 80% to 90% coherent for the performance to degrade noticeably.

## 4.8 Concluding Remarks

We discussed the MUSIC superresolution algorithm in this chapter and examined its performance when noise and measurement errors are present. We concluded that it is fairly resilient to such errors. We also presented a brief comparison with some other available, nonsubspace-based superresolution algorithms. That comparison showed that MUSIC is the most robust of the methods evaluated.



**Figure 4.14** Coherent signals.  $\gamma = 30$  dB,  $\phi = 45^\circ$ ,  $\varphi = 40^\circ$  and  $50^\circ$ . Top: equal amplitude, coh = 0.8; middle: 10:1 amplitude, coh = 0.8; bottom: equal amplitude, coh = 0.3. (Source: [7].) © IEEE 1986. Reprinted with permission.)

The singular largest issue with MUSIC in its role of finding the AOAs of multiple signals on an array is its tendency to fail when fully coherent signals are present. We showed, however, that it is fairly robust as long as the degree of coherence is 0.8 to 0.9 or less. All real signals have noise present that tends to decrease the degree of coherence.

## References

- [1] Schmidt, R. O., "Multiple Emitter Location and Signal Parameter Estimation," *Proceedings of the RADC Spectrum Estimation Workshop*, Griffiss Air Force Base, Rome, NY, October 1979.
- [2] Schmidt, R. O., "Multiple Emitter Location and Signal Parameter Estimation," *IEEE Transactions on Antennas and Propagation*, Vol. AP-34, No. 3, March 1986, pp. 276–280. Also reprinted in Rappaport, T. S., (ed.), *Smart Antennas*, Piscataway, NJ: IEEE, 1998, pp. 195–199.
- [3] Schmidt, R., "Multiple Emitter Location and Signal Parameter Estimation," *IEEE Transactions on Antennas and Propagation*, Vol. AP-34, March 1986, pp. 276–290.
- [4] Swindlehurst, A., and T. Kailath, "A Performance Analysis of Subspace-Based Methods in the Presence of Model Errors—Part I: The MUSIC Algorithm," *IEEE Transactions on Signal Processing*, Vol. 40, July 1992, pp. 1758–1774.
- [5] Wax, M., and T. Kailath, "Detection of Signals by Information Theoretic Criteria," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-33, No. 2, 1985, pp. 387–392.
- [6] Brandwood, D. H., "Effects of Phase Errors on the Accuracy of MUSIC," *Proceedings IEE*, London: IEE, 1994.
- [7] Johnson, R. L., and G. E. Miner, "Comparison of Superresolution Algorithms for Radio Direction Finding," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-22, No. 4, July 1986, pp. 432–441.
- [8] Gabriel, W. F., "Spectral Analysis and Adaptive Array Superresolution Techniques," *Proceedings of the IEEE*, Vol. 68, No. 6, June 1980, pp. 654–664.
- [9] Lagunas-Hernandez, M. A., and A. Gasull-Llampallas, "An Improved Maximum Likelihood Method for Power Spectral Density Estimation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-32, No. 1, February 1984, pp. 170–173.
- [10] Borgiotti, G. V., and L. J. Kaplan, "Superresolution of Uncorrelated Interference Sources by Using Adaptive Array Techniques," *IEEE Transactions on Antennas and Propagation*, Vol. AP-27, No. 6, November 1979, pp. 842–844.
- [11] McClellan, J. H., and S. W. Lang, "Duality for Multidimensional MEM Spectral Analysis," *Proceedings IEE*, Vol. 130, Part F, 1983, pp. 230–234.
- [12] Lang, S. W., and J. H. McClellan, "Multidimensional MEM Spectral Estimation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-30, No. 6, December 1982, pp. 880–887.



# Chapter 5

## Quadratic Position-Fixing Methods

### 5.1 Introduction

Techniques for determining PF estimates of targets based on measuring the TOA, TDOA, differential frequency [otherwise known as differential Doppler (DD)], and RD are discussed in this chapter.

When the TOA or the time difference of arrival (denoted here by  $\tau$ ) measured at two or more widely dispersed sensors are used to geolocate emitters, then quadratic LOPs result. Generally, the intersection of these LOP curves is taken as the estimated emitter location. Calculation of this intersection can be tedious and complicated, requiring a search over a large parameter space.

The principal advantages of these forms of processing are the following:

- Frequently a single antenna per sensor is required as opposed to an array for interferometric and similar processing methods that rely on the intersection of LOBs to determine the PF.
- Normally, higher precision and more accuracy can be obtained with quadratic processing.

On the other hand, there is one primary drawback:

- Typically, preprocessed data samples are required for nonpulsed modulations. This requires interlinking data links with considerable bandwidth between sensor platforms or the sensor platform and some processing site. This is not normally a problem in pulsed signal environments where signal events can be identified (for example, the leading edge of a pulse), such as radar geolocation or digital data signals.

A technique that yields the PF in closed form using TDOA measurements was presented in [1]. Presented here are some alternative algorithms. Two of these

algorithms would be expected to perform better with noisy or erroneous measurements because of their iterative search behavior.

This chapter begins with a presentation of some TDOA techniques of PF estimation. After a presentation of the definitions involved, a nonlinear least-squared technique is discussed. This is followed by a presentation of estimation of the TDOA from phase data collected at widely separated sensors. Next, a discussion of TDOA measurement accuracy is included. The independent variables in this discussion are the SNR and integration time, as these two parameters are the primary determining factors of TDOA estimation performance. Three algorithms are presented next and compared for TDOA estimation when noise is included. The effects of GDOP are present in TDOA estimation, just as they were for triangulation in Chapter 2. The last topics discussed in the TDOA section are these GDOP effects.

This chapter also talks about DD methods for PF estimation. After basic definitions are presented, a brief discussion of measurement accuracy is included. This is followed by a section in which a maximum likelihood algorithm is given, along with performance estimates. After a presentation of the *cross-ambiguity function* (CAF), which uses both TDOA and DD data, a section is included that discusses the very practical case of estimating the DD of a sinusoid in AWGN using phase data.

The last major topic presented in this chapter is PF estimation using range data. After definitions are presented, three least-squared distance error algorithms are discussed and compared—the spherical interpolation method, the spherical intersection method, and the plane intersection method. Lastly, an RD method for estimation of the PF using mathematical constructs from Grassmann algebra, called feasible bivectors, is presented.

## 5.2 TDOA Position-Fixing Techniques

### 5.2.1 Introduction

The *time of arrival* (TOA) measured at several sensors can be used for geoposition determinations. Although these arrival times can be used directly for this purpose, described in this chapter are techniques for calculating PFs using the time differences of the TOA at several sensors compared with a reference sensor.

TOA techniques can also be, and have been for years, used for geoposition calculations. Navigation systems such as Loran use TOA to determine the location of aircraft and ships at sea. It has been shown by Rusu [2] that TDOA and TOA processing are exactly equivalent. It has been shown by Shin and Sung [3] that the error characteristics of TOA and TDOA, as manifest in the *dilution of precision*

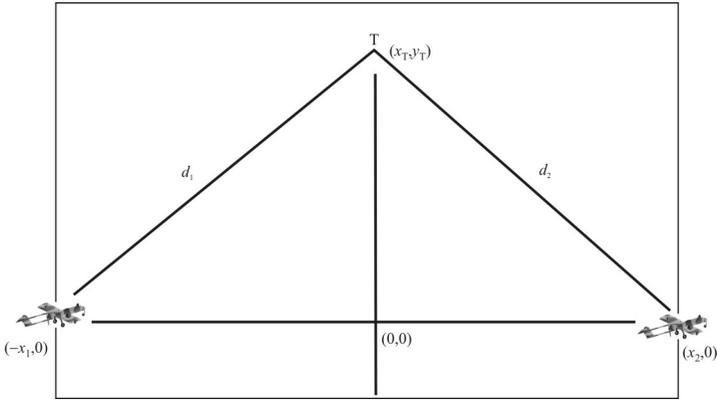


Figure 5.1 TDOA computation scenario.

(DOP), are exactly equivalent. We will be concerned only with TDOA processing here.

### 5.2.2 TDOA

The geometry is shown in Figure 5.1, where, for simplicity, only two receiving systems are shown. Since the transmitter and/or receiving systems can be elevated, in general,  $d_1$  and  $d_2$  are slant ranges between the transmitter and the sensor systems. Initially here, only two dimensions will be considered, however.

In a constant velocity medium, these distances can be expressed as

$$d_i = ct_i, \quad i = 1, 2 \tag{5.1}$$

where  $c$  is the speed of propagation of the signal, normally for communication signals assumed to be the speed of light, and  $t_i$  is the time between when the signal leaves the transmitter and when it arrives at the sensor.

The  $\tau$  is the time difference between when the signal arrives at one receiving site and the other:

$$\tau = t_2 - t_1 = \frac{d_2}{c} - \frac{d_1}{c} = \frac{1}{c}(d_2 - d_1) \tag{5.2}$$

From Figure 5.1,

$$d_i = \sqrt{(x_T - x_i)^2 + y_T^2}, \quad i = 1, 2 \quad (5.3)$$

so that

$$\begin{aligned} \Delta d &= d_2 - d_1 \triangleq h(x_T, y_T, \bar{x}_R, \bar{y}_R) \\ &= \sqrt{(x_T - x_1)^2 + y_T^2} - \sqrt{(x_T - x_2)^2 + y_T^2} \end{aligned} \quad (5.4)$$

where  $\bar{x}_R$  and  $\bar{y}_R$  are vectors of the  $x$ - and  $y$ -coordinates of the sensors. Let  $d$  denote the length of the baseline, which is the distance between the sensors given by

$$d = x_1 + x_2 \quad (5.5)$$

This can be rewritten in the form of a hyperbola as

$$\frac{x^2}{a} - \frac{y^2}{b} = \frac{x^2}{(\Delta d)^2 / 4} - \frac{y^2}{d^2 / 4 - (\Delta d)^2 / 4} = 1 \quad (5.6)$$

The resulting LOPs are called isochrones, since they characterize points in space where  $\tau$  is a constant. The target can lie at any point on the isochrone.

Denote  $(X, Y)$  as the global coordinates for sensor positions that are located off the axis as in Figure 5.1. If  $(x, y)$  are the local coordinates, then

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix} + \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (5.7)$$

where

$$X_0 = \frac{X_i + X_j}{2} \quad Y_0 = \frac{Y_i + Y_j}{2} \quad (5.8)$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} X - X_0 \\ Y - Y_0 \end{bmatrix} \quad (5.9)$$

and

$$\alpha = \tan^{-1} \left( \frac{Y_i - Y_j}{X_i - X_j} \right) \quad (5.10)$$

The  $\tau$  contour for the scenario shown in Figure 5.1 is shown in Figure 5.2 when  $d = 80$  km. The  $\tau$  varies between about  $-130 \mu\text{sec}$  to  $+130 \mu\text{s}$ . Also shown is a constant  $\tau$  curve; in this case,  $\tau$  is  $90 \mu\text{s}$ . The intersection of these curves forms a hyperbola as shown. As visualized from above, several of these curves are shown in Figure 5.3.

Let  $M$  denote the number of sensors. Then there are  $\kappa = \binom{M}{2}$  sensor pairs, or hyperbolas, available for geolocation.

### 5.2.3 Calculating the PF with TDOAs

Consider the 3-D geometry shown in Figure 5.4. It will be assumed that, without loss of generality, a reference sensor is placed at the origin. It is also assumed without loss of generality that the reference sensor is sensor 1. This section follows [4] closely; a similar derivation was documented by Mellen et al. [5].

Measurement at the  $i$ th sensor is given by the expression

$$(x_T - x_i)^2 + (y_T - y_i)^2 + (z_T - z_i)^2 = c^2(\tau_i + t_0) \quad (5.11)$$

where  $t_0$  is the unknown arrival time at the reference sensor placed at the origin.

The target is located at  $\vec{x} = [x_T \quad y_T \quad z_T]^T$ . At the reference site,

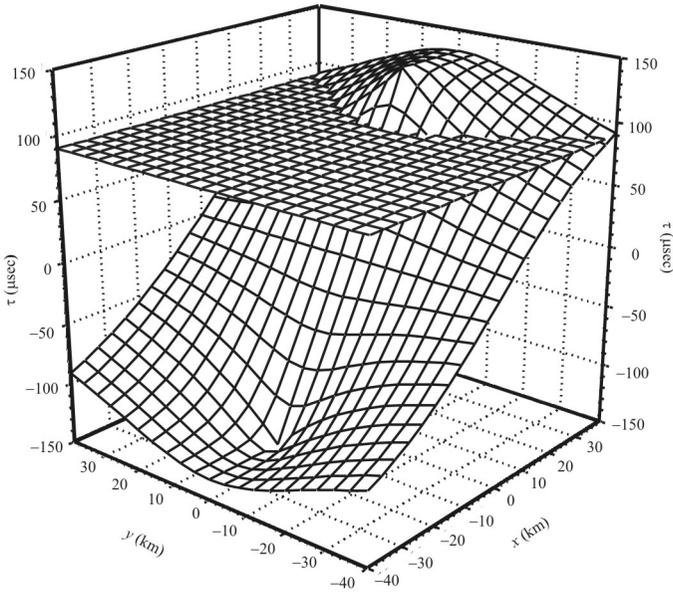
$$x_T^2 + y_T^2 + z_T^2 = (ct_0)^2 = d_0^2 = \|\vec{x}\|_2^2 = \vec{x}^T \vec{x} \quad (5.12)$$

where  $\|\vec{x}\|_2$  denotes the  $\mathcal{L}_2$  norm of  $\vec{x}$ . Combining (5.11) and (5.12) yields

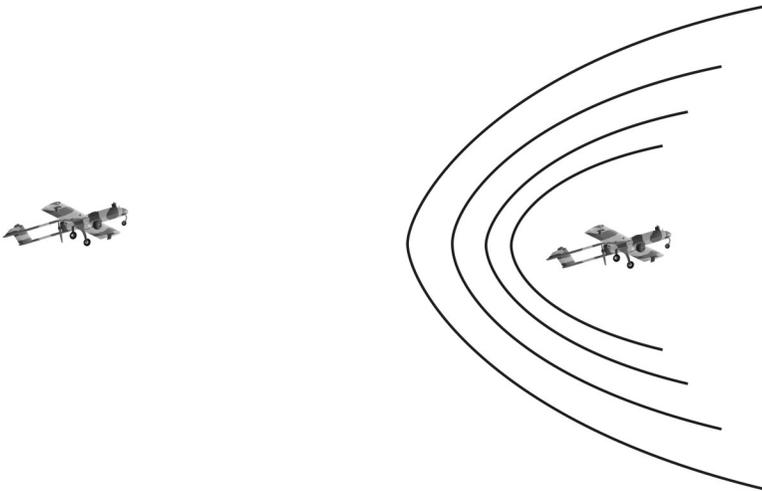
$$\begin{aligned} & [x_i \quad y_i \quad z_i] \begin{bmatrix} x_T \\ y_T \\ z_T \end{bmatrix} + c\tau_i \sqrt{x_T^2 + y_T^2 + z_T^2} \\ &= -\frac{1}{2}c^2\tau_i^2 + \frac{1}{2}(x_i^2 + y_i^2 + z_i^2), \quad i = 1, 2, \dots, M-1 \end{aligned} \quad (5.13)$$

In matrix form, (5.13) is represented as

$$\mathbf{A}\vec{x} + \vec{c} \|\vec{x}\|_2 = \vec{m} \quad (5.14)$$



**Figure 5.2** The intersection of the  $\tau$  curve with the plane defined by a constant  $\tau$  curve of  $90 \mu\text{sec}$  forms a hyperbolic curve. (Source: [1]. © Horizon House 2002. Reprinted with permission.)



**Figure 5.3** TDOA isochrones with no error. (Source: [1]. Horizon House © 2002. Reprinted with permission.)

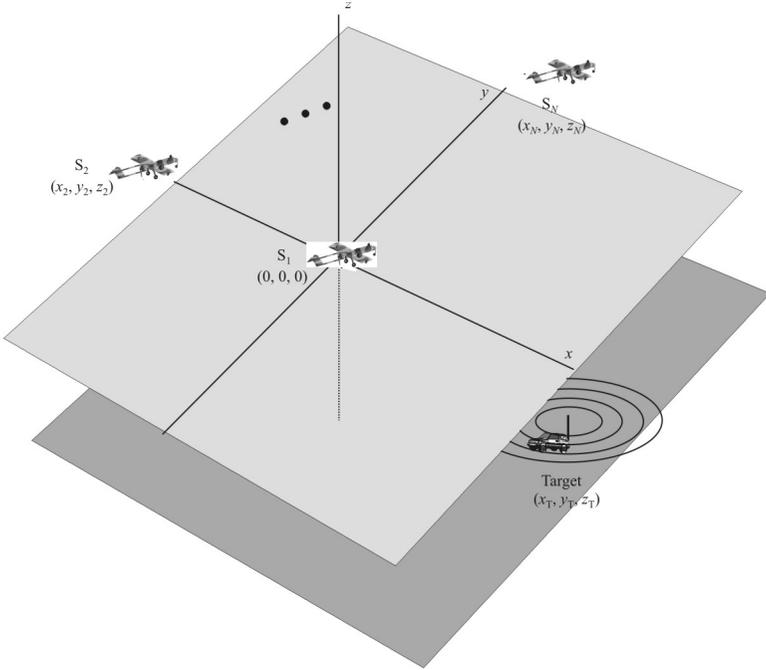


Figure 5.4 TDOA geometry.

where

$$\mathbf{A} = \begin{bmatrix} \bar{\mathbf{x}}_1^T \\ \bar{\mathbf{x}}_2^T \\ \vdots \\ \bar{\mathbf{x}}_{M-1}^T \end{bmatrix} \in \mathfrak{R}^{M-1 \times 3}$$

$$\bar{\mathbf{x}}_i = [x_i \quad y_i \quad z_i]^T$$

$$\bar{\mathbf{c}} = c\bar{\boldsymbol{\tau}} \in \mathfrak{R}^{M-1 \times 1}$$

$$\bar{\boldsymbol{\tau}} = [\tau_1 \quad \tau_2 \quad \cdots \quad \tau_{M-1}]^T$$

$$\bar{\mathbf{m}} = \left[ \frac{1}{2} (\mathbf{A}\mathbf{A}^T - c^2 \bar{\boldsymbol{\tau}}\bar{\boldsymbol{\tau}}^T) \right]_{mm}, \quad m = 1, 2, \dots, M \in \mathfrak{R}^{M-1 \times 1}$$

$\mathfrak{R}^{m \times n}$  denotes an  $m \times n$  array of real numbers (an  $m \times n$  matrix).

In general (5.14) will be overdetermined in that there are more measurements than the minimum number required; that is  $\mathbf{A} \in \mathfrak{R}^{m \times n}$ , where  $m > n$ . Equation (5.14) can be rearranged as

$$\mathbf{A}\bar{x} = \bar{m} - \bar{c} \|\bar{x}\|_2 = \bar{f}(p) \quad (5.15)$$

The vector  $\bar{f} \in \mathfrak{R}^{m \times 1}$  has the parameter  $p$  such that, as discussed in Section 2.3, for any  $p$ , the pseudoinverse of  $\mathbf{A}$  given by

$$\mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \quad (5.16)$$

solves (5.15) in the least-squared sense; thus

$$\hat{x} = \mathbf{A}^\dagger (\bar{m} - \bar{c}p) \quad (5.17)$$

solves (5.15) such that the residual

$$\epsilon = \left\| \mathbf{A}\hat{x} - (\bar{m} - \bar{c}p) \right\|_2 \quad (5.18)$$

is minimized.

The measurements are used in the computation of the scalar equation

$$(\bar{c}^T \Phi \bar{c} - 1)p^2 - 2\bar{m}^T \Phi \bar{c}p + \bar{m}^T \Phi \bar{m} = 0 \quad (5.19)$$

where

$$\Phi = \mathbf{A}(\mathbf{A}^T \mathbf{A} \mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \quad (5.20)$$

With noise present, the measurements are corrupted so that

$$\mathbf{A}\bar{x} = \bar{f}(p) + \bar{n} \quad (5.21)$$

with

$$\mathcal{E}\{\bar{n}\} = 0 \quad \mathcal{E}\{\bar{n}\bar{n}^T\} = \mathbf{Q} \quad (5.22)$$

The pseudoinverse in this case is

$$\mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{Q}^{-1} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{Q} \quad (5.23)$$

If the noise is uncorrelated among sensors with equal variances given by  $\sigma_n^2$ , then

$$\mathbf{Q} = \sigma_n^2 \mathbf{I} \quad (5.24)$$

In this case, (5.20) must be changed to account for the noise as

$$\Phi = (\mathbf{A}^*)^{-1} \mathbf{A}^* [(\mathbf{A}^*)^T \mathbf{Q}^{-1} \mathbf{A}^* (\mathbf{A}^*)^T \mathbf{Q}^{-1} \mathbf{A}^*]^{-1} (\mathbf{A}^*)^T \mathbf{Q}^{-1} \quad (5.25)$$

The solution of (5.19) determines  $p$ , which is equal to  $d_0$ , the range from the reference sensor to the target. Thus,

$$p = d_0 = \|\bar{\mathbf{x}}\|_2 = \sqrt{\bar{\mathbf{x}}^T \bar{\mathbf{x}}} \quad (5.26)$$

and is used to compute the estimate of the geolocation of the target as

$$\hat{\mathbf{x}}(d_0) = [x_T \quad y_T \quad z_T]^T = \mathbf{A}^\dagger (\bar{\mathbf{m}} - \bar{c}d_0) \quad (5.27)$$

The Taylor series gradient descent algorithm described in Chapter 1 is one of the most accurate methods for estimating a PF (discussed more fully in Appendix B). If the PF computed as indicated here is not accurate enough, the PF can at least be used as a starting point for the descent algorithm to improve upon the accuracy. As mentioned in Chapter 1, the gradient descent algorithm can use any type of measurement.

### 5.2.4 Nonlinear Least-Squares

Equation (5.4) is highly nonlinear and we want to solve it in generalized coordinates, which is a nonlinear minimization problem. Note that this expression could be overdetermined by the measurements. In two dimensions, the nonlinear least-squares estimate of the target location  $(x_T, y_T)$  is given by

$$(\hat{x}_T, \hat{y}_T) = \underset{(x,y)}{\operatorname{argmin}} \sum_{i>j} [\Delta d_{i,j} - h(x_T, y_T, x_i, y_i, x_j, y_j)]^2 \quad 1 \leq i < j \leq N \quad (5.28)$$

In matrix form, the target location being sought is denoted by  $\bar{\mathbf{x}}_T = [x_T \quad y_T]^T$ , and this minimization becomes

$$\hat{\vec{x}}_T = \underset{\vec{x}_T}{\operatorname{argmin}} [\Delta\vec{d} - \vec{h}(\vec{x}_T)]^T \mathbf{C}_{\Delta\vec{d}\Delta\vec{d}}^{-1} [\Delta\vec{d} - \vec{h}(\vec{x}_T)] \quad (5.29)$$

where

$$\Delta\vec{d} = [\Delta d_{1,2} \quad \Delta d_{2,3} \quad \cdots \quad \Delta d_{N-1,N}] \quad (5.30)$$

and

$$\mathbf{C}_{\Delta\vec{d}\Delta\vec{d}} = \mathbf{Cov}(\Delta\vec{d}) \quad (5.31)$$

Assuming the calculated position is close to the true position,  $\vec{x}_{T,0}$ , then  $\Delta\vec{d} = \vec{h}(\vec{x}_{T,0}) + \vec{\epsilon}$ , and if  $\mathbf{C}_{\epsilon\epsilon} = \mathbf{Cov}(\vec{\epsilon})$ , then, finding the Taylor series and retaining only the linear terms we get

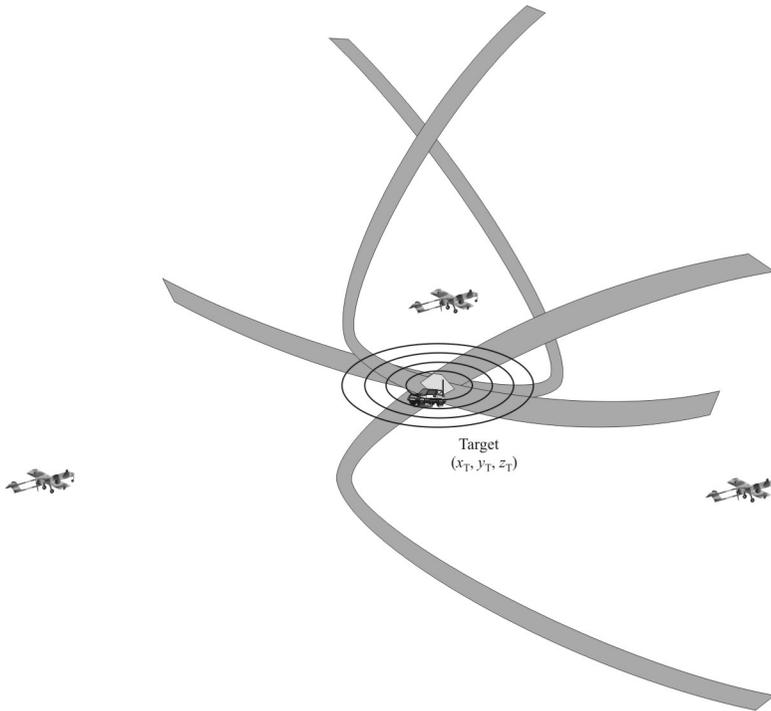
$$\mathbf{C}_{\hat{\vec{x}}_T} = \{[\vec{h}_p^H(\vec{x}_{T,0})]^T \mathbf{C}_{\epsilon\epsilon}^{-1} \vec{h}_p'(\vec{x}_{T,0})\} \quad (5.32)$$

where  $^H$  denotes the Hermitian or conjugate transpose, and  $\vec{h}'$  denotes differentiation of  $\vec{h}$  with respect to  $\vec{x}$ . When  $\vec{\epsilon} \sim \mathcal{N}(\vec{0}, \sigma^2)$ , then (5.32) describes the CRB.

### 5.2.5 TDOA Measurement Accuracy

When errors are taken into consideration, the TDOA isochrones are no longer clean functions, as illustrated in Figure 5.3, but form regions or areas within which the target should lie, as illustrated in Figure 5.5.

All of the methods of computing the geolocation of targets with measurement of the TDOAs are subject to errors in the measurements. The accuracy of TDOA measurements is discussed in this section in terms of the standard deviation. The noise and measurement errors are the two primary sources of error. Included here are the effects of noise that determine the SNR at the receiver. Measurement errors are systematic and unique to each implementation. They will not be discussed here.



**Figure 5.5** TDOA isochrones with errors included. The target indicated location could be anywhere in the light area.

### 5.2.5.1 Expected Accuracy of TDOA PF Estimates

As indicated in [1], the Cramer-Rao bound on parameter estimation is a frequently used measure of how well such a parameter can be measured, as long as the parameter is unbiased. Under some reasonable assumptions it represents the best that can be obtained under those assumptions. The CRB for estimating the TOA of a signal at a sensor is given by [6]

$$\sigma_{\tau} = \frac{1}{\beta} \frac{1}{\sqrt{WT\gamma}} \tag{5.33}$$

where  $W$  is the noise bandwidth of the receivers,  $T$  is the integration time, and  $\gamma$  is the effective input SNR at the two sensor sites.

The RMS radian frequency is given by  $\beta$ , which is a measure of the bandwidth of the signal and is given by

$$\beta = 2\pi \left[ \frac{\int_{-\infty}^{\infty} f^2 |S(f)|^2 df}{\int_{-\infty}^{\infty} |S(f)|^2 df} \right]^{1/2} \quad (5.34)$$

where  $S(f)$  is the spectrum of the signal. For an ideal signal with sharp edges in the frequency spectrum,  $\beta = \pi W / \sqrt{3}$  (where  $W$  is in hertz), for example.

Variable  $\gamma$  is a composite SNR at the two sensors. If  $\gamma_i$  and  $\gamma_j$  are the SNRs at the two sensors, then  $\gamma$  is given by

$$\frac{1}{\gamma} = \frac{1}{2} \left[ \frac{1}{\gamma_i} + \frac{1}{\gamma_j} + \frac{1}{\gamma_i \gamma_j} \right] \quad (5.35)$$

### 5.2.5.2 Low SNR

Quazi compared the theoretical measurement accuracy of several TDOA computation models [7]. For low SNR levels, the standard deviation computed with the Knapp and Carter [8], Schultheiss [9], Hahn [10], and Tomlinson and Sorokowsky [11] techniques all evaluated to

$$\sigma_\tau \geq \sqrt{\frac{3}{8\pi^2 T}} \frac{1}{\gamma} \frac{1}{\sqrt{f_2^3 - f_1^3}} \quad (5.36)$$

where  $T$  is the integration time,  $\gamma$  is the SNR (not dB), and it was assumed that both the signal and noise were constant levels over the bandwidth specified by  $f_1$  to  $f_2$ . This function is illustrated in Figures 5.7 and 5.8 for a 25 kHz bandwidth for several values of  $T$ . This expression can be represented in terms of the center frequency  $f_0$  and bandwidth  $W = f_2 - f_1$  as

$$\sigma_\tau \approx \sqrt{\frac{1}{8\pi^2}} \frac{1}{\gamma} \frac{1}{\sqrt{TW}} \frac{1}{f_0} \frac{1}{\sqrt{1 + \frac{W^2}{12f_0^2}}} \quad (5.37)$$

### 5.2.5.3 High SNR

The standard deviation when the SNR is high,  $\gamma \gg 1$ , is given by

$$\sigma_\tau \approx \sqrt{\frac{3}{4\pi^2 T}} \frac{1}{\sqrt{\gamma}} \frac{1}{\sqrt{f_2^3 - f_1^3}} \quad (5.38)$$

which is the same as (5.36) with the square root of the SNR in the denominator rather than just the SNR. This function is plotted in Figures 5.8 and 5.9.

## 5.2.6 TDOA PFs with Noisy Measurements

Gustafsson and Gunnarsson [12] presented three algorithms for determining the PF of a target based on TDOA measurements that may contain measurement errors. Those algorithms are discussed and compared in this section.

### 5.2.6.1 Isochrone Intersection Algorithm

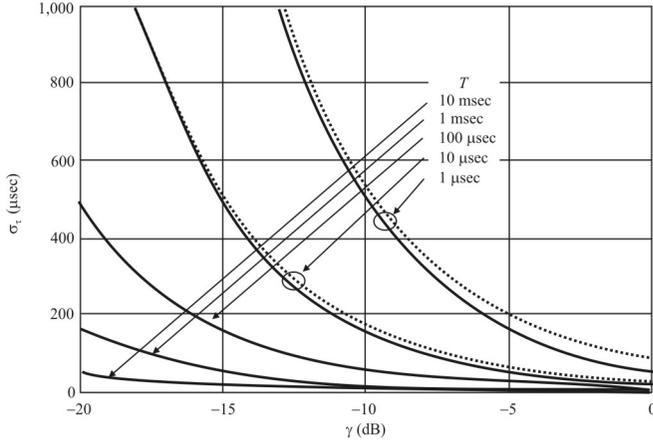
The first of these computes the intersection point of each pair of isochrones. With  $K$  pairs of sensors, this results in

$$\binom{K}{2} = \frac{K!}{2!(K-2)!} \quad (5.39)$$

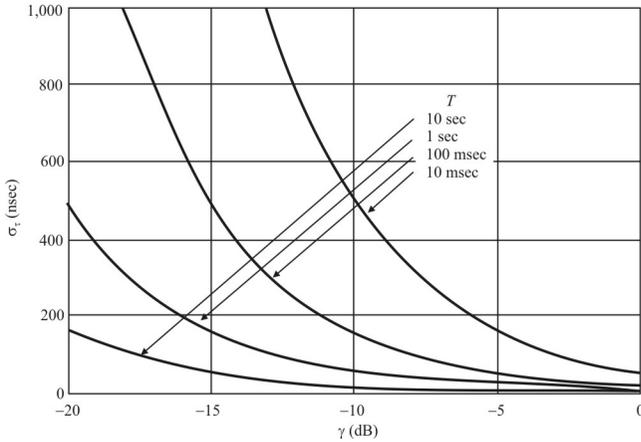
points to estimate. The PF estimate is given by the point where all  $\binom{K}{2}$  of these isochrones intersect (or perhaps point where the largest number of them intersect). Each pair of isochrones can have zero, one, or two intersections. Determining such intersections can be difficult. The authors claim that this approach leads to inferior results compared to the other two algorithms when the measurements are corrupted with noise. With noise present, it could be that no more than two isochrones intersect at any point.

### 5.2.6.2 Stochastic Gradient Algorithm

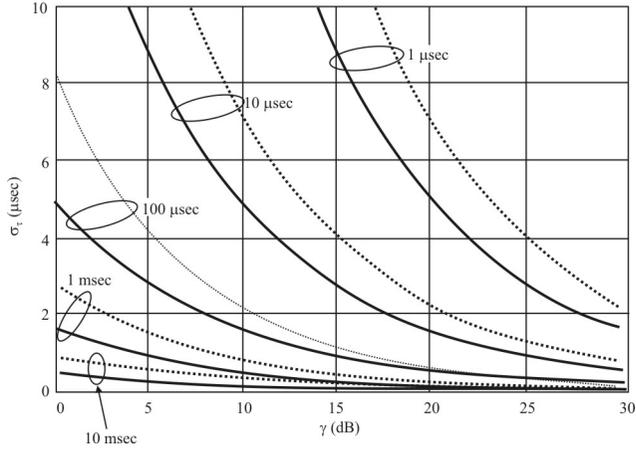
The second algorithm applies the stochastic gradient algorithm to the nonlinear least-squares problem described above. Let  $\bar{x}_T = [x_T \ y_T]^T$  denote the target position and  $\hat{x}_{T,i}$  the estimate of  $\bar{x}_T$  on the  $i$ th iteration. In this algorithm,



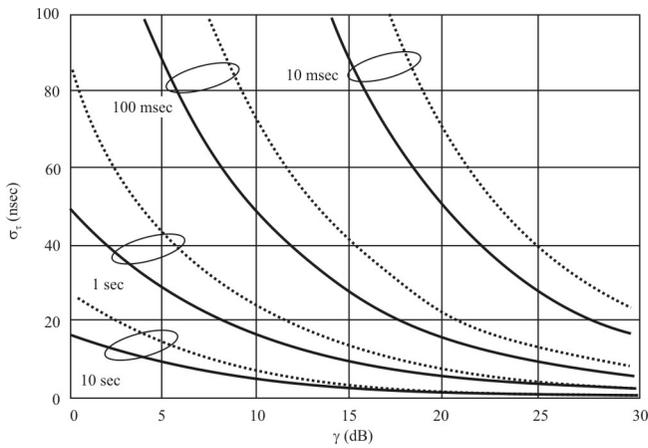
**Figure 5.6** TDOA standard deviation for low SNRs and short integration times. The dotted lines are (5.33) while the solid lines are (5.36). For  $T > 10 \mu\text{s}$ , the differences in the lines are not discernible.



**Figure 5.7** TDOA standard deviations for longer integration times,  $W = 25 \text{ kHz}$ . Both (5.33) and (5.36) are plotted with (5.33) given by dotted lines and (5.36) by solid lines, but, as with Figure 5.8, the differences in the curves are not discernible because  $T > 10 \mu\text{s}$ .



**Figure 5.8** TDQA standard deviation for higher SNRs and short integration times,  $W = 25$  kHz. The dotted lines are (5.33) and the solid lines are (5.38).



**Figure 5.9** TDQA standard deviation for higher SNRs and long integration times,  $W = 25$  kHz. The dotted lines are (5.33) and the solid lines are (5.38).

$$\hat{\mathbf{x}}_{T,i+1} = \hat{\mathbf{p}}_i - \mu_i \vec{h}'_p(\hat{\mathbf{x}}_{T,i}) [\Delta \vec{d} - \vec{h}(\hat{\mathbf{x}}_{T,i})] \quad (5.40)$$

The step size  $\mu^{(i)}$  can be found using

$$\mu_i = \frac{\mu}{[\vec{h}'_p(\hat{\mathbf{x}}_{T,i})]^T \vec{h}'_p(\hat{\mathbf{x}}_{T,i})} \quad (5.41)$$

where  $\mu$  is the *least-mean square* (LMS) step size.

Step size  $\mu$  is selected based on stability considerations. The sequence (5.30) will converge if and only if  $\mu$  satisfies

$$0 < \mu < \frac{1}{\lambda_{\max}} \quad (5.42)$$

where  $\lambda_{\max}$  is the largest eigenvalue of the covariance matrix  $\mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}$ . Of course, the larger the value of  $\mu$ , the faster the convergence, typically.

### 5.2.6.3 Static Particle Filter Algorithm

The third algorithm in [12] is based on the *static particle filter* [13–15]. The particle filter is a method of implementing a recursive Bayesian filter by Monte Carlo simulations. The a posteriori density function is represented by a set of random samples with associated weights and estimates based on these samples and weights are computed. The static particle filter algorithm is an approximation of the optimal Bayesian filter described here. Note that the Kalman filter (not the EKF) described in Section 2.9.1 is an implementation of the optimal Bayesian filter.

Given the system state equation that describes the evolution of the state sequence of the target

$$\vec{\mathbf{x}}_k = \vec{f}_k(\vec{\mathbf{x}}_{k-1}, \vec{\mathbf{n}}_{k-1}) \quad (5.43)$$

where  $\{\vec{\mathbf{x}}_k, k \in \mathfrak{N}\}$ . The set  $\{\vec{\mathbf{x}}_k, k \in \mathfrak{N}\}$  describes the state of the system at time step  $k$ ,  $\vec{f}_k : \mathfrak{R}^{N_x} \times \mathfrak{R}^{N_n} \rightarrow \mathfrak{R}^{N_x}$  describes the state update from  $k - 1$  to  $k$  and may be nonlinear,  $\{\vec{\mathbf{n}}_k, k \in \mathfrak{N}\}$  is an i.i.d. noise process sequence,  $N_x$  and  $N_n$  are the dimensions of the state and noise vectors, respectively, and  $\mathfrak{N}$  is the set of natural numbers. The measurements at time  $k$  are given by

$$\bar{z}_k = \bar{h}_k(\bar{x}_k, \bar{\eta}_k) \tag{5.44}$$

where  $\bar{h}_k : \mathfrak{R}^{N_x} \times \mathfrak{R}^{N_\eta} \rightarrow \mathfrak{R}^{N_z}$  is a (possibly nonlinear) function,  $\{\bar{\eta}_k, k \in \mathfrak{N}\}$  is an i.i.d. measurement noise sequence, and  $N_\eta$  is the dimension of that noise process. The goal is to estimate the states  $\bar{x}_k$  based on the set of measurements up to time  $k$ :  $\bar{z}_{1:k} = \{\bar{z}_i, i = 1, \dots, k\}$ .

The PDF  $p(\bar{x}_k | \bar{z}_{1:k})$  is the desired function to be maximized to provide the best estimate of  $\bar{x}_k$ . This PDF is obtained recursively by prediction and updating. The initial PDF  $p(\bar{x}_0 | \bar{z}_0)$  is assumed to be known. If the PDF  $p(\bar{x}_{k-1} | \bar{z}_{1:k-1})$  is known, then (5.43) and the Chapman-Kilmogorov equation

$$p(\bar{x}_k | \bar{z}_{1:k-1}) = \int p(\bar{x}_k | \bar{x}_{1:k-1}) p(\bar{x}_{k-1} | \bar{z}_{1:k-1}) d\bar{x}_{k-1} \tag{5.45}$$

which uses the fact that (5.43) is a first-order Markov process so that  $p(\bar{x}_k | \bar{x}_{1:k-1}, \bar{z}_{1:k-1}) = p(\bar{x}_k | \bar{x}_{1:k-1})$ , are used to obtain the prior PDF of the state at time  $k$ . The PDF  $p(\bar{x}_k | \bar{x}_{1:k-1})$  is determined from (5.43) and the noise characteristics that are assumed to be known.

The PDF is updated for time  $k$  using Bayes' rule

$$p(\bar{x}_k | \bar{z}_{1:k}) = \frac{p(\bar{z}_k | \bar{x}_k) p(\bar{x}_k | \bar{z}_{1:k-1})}{p(\bar{z}_k | \bar{z}_{1:k-1})} \tag{5.46}$$

where

$$p(\bar{z}_k | \bar{z}_{1:k-1}) = \int p(\bar{z}_k | \bar{x}_{1:k}) p(\bar{x}_k | \bar{z}_{1:k-1}) d\bar{x}_k \tag{5.47}$$

which is a function of (5.44) and noise process  $\bar{\eta}_k$ .

There are four principal concepts of the particle filter algorithm:

1. Bayesian inference;
2. Monte Carlo samples;
3. Importance sampling;
4. Resampling.

Based on the above development, the Bayesian inference consists of  $\bar{x}_k$ , which is an unknown vector of random variables. The set of observations is given by  $\bar{z}_k$ , which is also a set of random variables. We wish to estimate  $\bar{x}_k$  by knowing  $\bar{z}_k$ . The a priori PDF  $p(\bar{x}_k)$  contains our prior knowledge of  $\bar{x}_k$ . The conditional PDF  $p(\bar{z}_k | \bar{x}_{1:k})$  describes the relationship between  $\bar{z}_k$  and  $\bar{x}_k$ . The a posteriori PDF  $p(\bar{x}_k | \bar{z}_{1:k})$  represents the information about  $\bar{x}_k$  given measurement  $\bar{z}_k$ .

The a posteriori PDF  $p(\bar{x}_k | \bar{z}_{1:k})$  may be difficult or impossible to compute in closed form. Thus, Monte Carlo samples, also called *particles*, are used to approximate this PDF. Each particle has a value and an associated weight. Ideally, each particle would represent a sample from the PDF  $p(\bar{x}_k | \bar{z}_{1:k})$ . Since in general this PDF is not known in closed form, *importance sampling* is used. The particles are weighted with weights drawn from an importance distribution denoted  $q(\bar{x}_{0:k}^i | \bar{z}_{1:k})$  for weight  $i$ . Selection of this distribution is typically based on some optimality criterion [14].

In estimation with the particle filter, most weights tend to zero except a few, which become large. The few that grow large are those that closely match the observations. Resampling is done to concentrate particles in regions where  $p(\bar{x}_k | \bar{z}_{1:k})$  is larger.

These steps are shown graphically in Figure 5.10. The steps to the particle filter algorithm for computing the PF estimate based on TDOA measurements are as follows:

1. Randomize  $L$  possible target locations as  $\bar{x}_T^{(i)}$  from the a priori distribution  $p(\bar{x}_{T,0})$ .
2. Choose jittering constants  $K_n$  and  $K_w$  and let the position random walk covariance  $C_w = K_w / k^2$  and jittering measurement noise  $C_{nn} = C_{\Delta\bar{d}\Delta\bar{d}}^{-1} + K_n / k^2$ .
3. Iterate for  $k = 1, 2, \dots$  until  $\hat{\bar{x}}_k$  has converged.
  - a. Compute the location weights  $\bar{w}^{(i)}$  using the likelihood

$$\bar{w}^{(i)} = \exp\{[\Delta\bar{d} - \bar{h}(\bar{x}_T^{(i)})]^T C_{\Delta\bar{d}\Delta\bar{d}}^{-1} [\Delta\bar{d} - \bar{h}(\bar{x}_T^{(i)})]\} \quad (5.48)$$

where  $\Delta\bar{d}$  is given by (5.30),  $C_{\Delta\bar{d}\Delta\bar{d}}$  is given by (5.31), and  $\bar{w}^{(i)}$  is normalized to

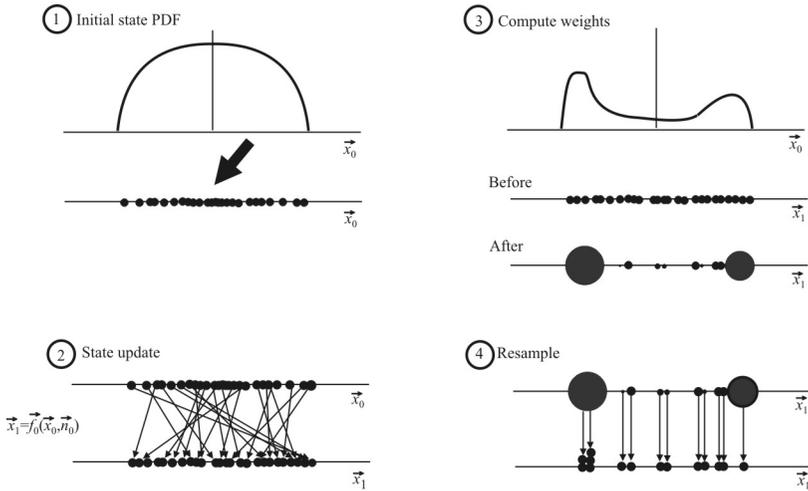


Figure 5.10 Steps in the particle filter algorithm.

$$\bar{w}^{(i)} = \frac{\tilde{w}^{(i)}}{\sum \tilde{w}^{(i)}} \tag{5.49}$$

b. Compute the estimate

$$\hat{\vec{x}}_{T,k} = \sum_i \bar{w}^{(i)} \vec{x}_T^{(i)} \tag{5.50}$$

c. Resample with the replacement locations, where the probability to pick one location is proportional to its weight. After the resampling, the weights are reset to  $w^{(i)} = 1/L$ .

d. Spread out the locations as  $\vec{x}_T^{(i)} = \bar{\vec{x}}_T^{(i)} + \vec{w}$ , where  $\vec{w} \in \mathcal{N}(\vec{0}, \mathbf{C}_{ww})$ .

The idea with jittering noise in step (2) is to explore a smaller and smaller neighborhood more accurately.

A comparison example of these algorithms is shown in Figure 5.11 [11]. The receiver and target layout are shown in Figure 5.11(a), along with the results of the

stochastic gradient algorithm with no noise. The receivers are placed at  $(-1, 0)$ ,  $(0, -1)$ ,  $(1, 0)$ , and  $(0, 1)$ , while the target is located at  $(1.2, 1.2)$ . In the noise-free case, all six hyperbolas intersect at the target location. With noise added, the results are as shown in Figure 5.11(b). There is no unique target location computed in this case. The results for the static particle filter algorithm are shown in Figure 5.11(c) with the contour plots of the nonlinear least-squares criterion:

$$\sum_{i < j} [\Delta d_{i,j} - h(x_T, y_T)]^2 \quad (5.51)$$

The algorithm is seen to converge on the correct target location. The results for the gradient search algorithm using a normalized least-mean squares method with the static particle filter algorithm are illustrated in Figure 5.11(d). Again, convergence to the correct target location occurs. In this case there are no local minima and convergence to the correct target location is assured.

### 5.2.7 TDOA Dilution of Precision

TDOA geoposition estimating suffers from another type of error caused by long ranges from the sensor baseline. Consider Figure 5.12, where three sensors are included and the target is a considerable distance from the baselines between the sensors. The hyperbolic LOPs are nearly parallel in this region, and noise or small measurement errors can cause considerable errors in computing the geolocation. This is called GDOP or DOP. The farther away from the sensor baselines, the worse GDOP gets.

Combining (5.11) and (5.12) yields (5.13). Expressing (5.13) in vector form,

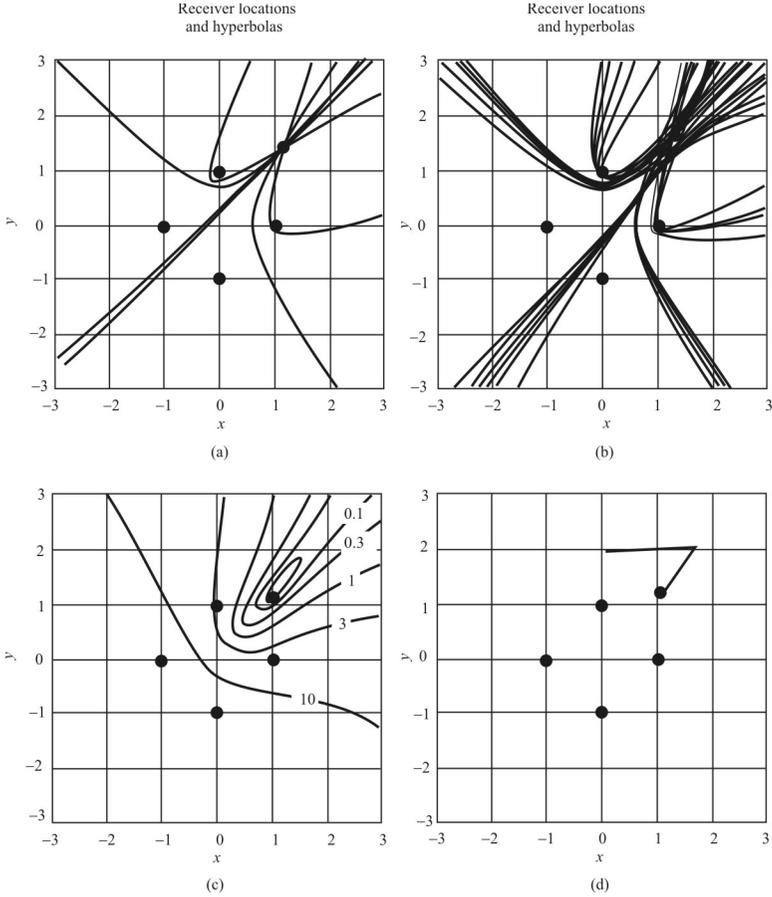
$$\bar{x}_i \bar{x} + c \tau_i \|\bar{x}\|_2 = -\frac{1}{2} c^2 \tau_i^2 + \frac{1}{2} \|\bar{x}_i\|_2^2 \quad (5.52)$$

where  $\bar{x}_i = [x_i \quad y_i \quad z_i]^T$ .

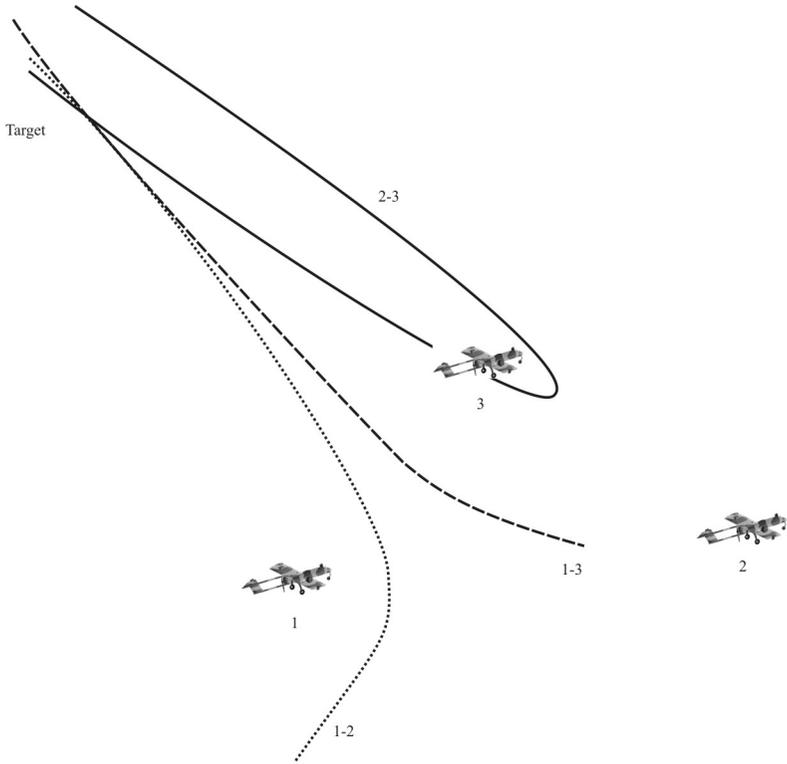
The sensitivity of the measurements is given by the gradient of (5.52) as

$$\frac{\partial \bar{x}_i \bar{x}}{\partial \bar{x}} + c \frac{\partial \tau_i \|\bar{x}\|_2}{\partial \bar{x}} = -\frac{1}{2} c^2 \frac{\partial \tau_i^2}{\partial \bar{x}} \quad (5.53)$$

since the last term in (5.52) is a constant. This can be written as



**Figure 5.11(a–d)** Algorithm comparison example. The test scenario is illustrated in (a), which also shows the PF estimate for the first algorithm, with noise-free hyperbolas; (b) also shows the results of the stochastic gradient algorithm but with noisy data; (c) is the contour plot of the nonlinear least-squares criterion  $\sum_{i < j} [\Delta d_{ij} - h(x_T, y_T)]^2$ ; and (d) is the gradient search with the static particle filter algorithm.



**Figure 5.12** An illustration of the errors caused by the GDOP effect with TDOA processing.

$$\bar{x}_i^T + c \left( \tau_i \frac{\bar{x}}{\|\bar{x}\|_2} + \|\bar{x}\|_2 \frac{\partial \tau_i}{\partial \bar{x}} \right) = -c^2 \tau_i \frac{\partial \tau_i}{\partial \bar{x}} \quad (5.54)$$

The solution to this partial differential equation is given by [16]

$$\frac{\partial \tau_i}{\partial \bar{x}} = \frac{1}{c} \left( \frac{\bar{x} - \bar{x}_i^T}{\|\bar{x} - \bar{x}_i^T\|_2} - \frac{\bar{x}}{\|\bar{x}\|_2} \right) \quad (5.55)$$

A gradient vector with entries from (5.55) is defined as

$$\bar{\mathbf{k}} = \left[ \frac{\partial \tau_1}{\partial \bar{x}} \quad \frac{\partial \tau_2}{\partial \bar{x}} \quad \dots \quad \frac{\partial \tau_N}{\partial \bar{x}} \right]^T \quad (5.56)$$

Let  $g$  denote the GDOP, which is given by [17]

$$g = \sqrt{\text{tr}(\bar{\mathbf{k}}^T \bar{\mathbf{k}})^{-1}} \quad (5.57)$$

If the variances of the measurements are not the same at all sensors, then this is taken into consideration by weighting matrix  $\mathbf{W}$ . In this case,

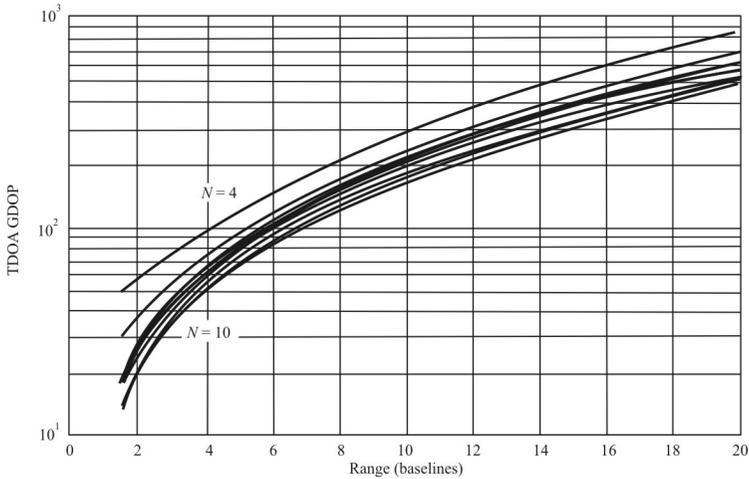
$$\mathbf{C}_{\Delta \bar{r} \Delta \bar{r}} = \begin{bmatrix} \sigma_1^2 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_M^2 \end{bmatrix} \quad (m^2) \quad (5.58)$$

Then the GDOP becomes

$$g = \sqrt{\text{tr}(\bar{\mathbf{k}}^T \mathbf{W} \bar{\mathbf{k}})^{-1}} \quad (5.59)$$

where  $\mathbf{W} = \sigma_b^2 \mathbf{C}_{\Delta \bar{r} \Delta \bar{r}}^{-1}$  and  $\sigma_b^2$  is a selected normalizing variance. The spherical uncertainty is then given by

$$\sigma_{3D} = \sigma_b g \quad (5.60)$$



**Figure 5.13** TDOA GDOP for  $N = 4$  to 10 sensor elements.

In [16], an example of the effects of GDOP is illustrated. The sensors consist of a near-planar circular array of a certain radius; the radius is referred to as the baseline. The results of GDOP-induced error with this array are illustrated in Figure 5.13 [16]. At a range of only two baselines, the accuracy degrades by a factor of 20 to 60.

### 5.2.8 Bias Effects of TDOA PF Estimation

Just as LOB measurement bias errors cause errors in estimating a PF with triangulation, errors are caused by measurement bias in TDOA calculations as well. An analysis and simulation was presented by Koorapaty, Grubeck, and Cedervall [18] to examine such effects when TDOA and TOA measurements were used to estimate the location of 911 callers. Their results are based on the 911 emergency call requirements and the specific results are couched in the vernacular of this requirement. Nevertheless, they illustrate the effects of bias on TDOA PF estimation.

The system equations are given by

$$d_m = f_m(\bar{x}) + b_m + n_m, \quad m = 1, 2, \dots, M \quad (5.61)$$

where  $b_i$  represents the bias in measurement  $i$  and  $\bar{x} = [x_T \ y_T \ z_T]^T$  is the location of the target. In general,  $f_i(\bar{x})$  is nonlinear, so a Taylor expansion is determined and only the linear terms are retained. In matrix form, this becomes

$$\bar{d} = \bar{f}(\bar{x}) + \bar{b} + \bar{n} \tag{5.62}$$

The least-squares estimator finds the  $\hat{\bar{x}}$  that minimizes

$$Q(\bar{x}) = [\bar{d} - \bar{f}(\bar{x}) - \bar{b}]^T \mathbf{C}_{\Delta\bar{d}\Delta\bar{d}}^{-1} [\bar{d} - \bar{f}(\bar{x}) - \bar{b}] \tag{5.63}$$

When the elements of  $\bar{n}$  are jointly Gaussian with covariance matrix  $\mathbf{C}_m$ , this process is also the maximum likelihood estimator.

It is assumed that the biases were unknown but nonrandom parameters, constant over the measurement interval. It was also assumed that the target remained fixed over the measurement interval.

The Taylor series approximation around reference point  $\bar{x}_0$  is given by

$$\bar{f}(\bar{x}) \approx \bar{f}(\bar{x}_0) + \mathbf{G}_0(\bar{x} - \bar{x}_0) \tag{5.64}$$

where

$$\mathbf{G}_0 = \begin{bmatrix} \left. \frac{\partial f_1(\bar{x})}{\partial x_1} \right|_{\bar{x}=\bar{x}_0} & \dots & \left. \frac{\partial f_1(\bar{x})}{\partial x_k} \right|_{\bar{x}=\bar{x}_0} \\ \vdots & \ddots & \vdots \\ \left. \frac{\partial f_M(\bar{x})}{\partial x_1} \right|_{\bar{x}=\bar{x}_0} & \dots & \left. \frac{\partial f_M(\bar{x})}{\partial x_k} \right|_{\bar{x}=\bar{x}_0} \end{bmatrix} \tag{5.65}$$

Therefore, from (5.63) and (5.64),

$$Q(\bar{x}) = [\bar{d}_1 - \mathbf{G}_0\bar{x} - \bar{b}]^T \mathbf{C}_{\Delta\bar{d}\Delta\bar{d}}^{-1} [\bar{d}_1 - \mathbf{G}_0\bar{x} - \bar{b}] \tag{5.66}$$

is the function to be minimized where

$$\bar{d}_1 = \bar{d} - \bar{f}(\bar{x}_0) + \mathbf{G}_0\bar{x}_0 \tag{5.67}$$

To ascertain where the minimum lies, the gradient vector is calculated,

$$\nabla_{\vec{x}} Q(\vec{x}) = \left[ \frac{\partial Q}{\partial x_1} \quad \frac{\partial Q}{\partial x_2} \quad \dots \quad \frac{\partial Q}{\partial x_k} \right]^T \quad (5.68)$$

and that value of  $\vec{x}$  such that  $\nabla_{\vec{x}} Q(\vec{x}) = \vec{0}$  is determined.

If we assume that the biases  $\vec{b}$  are not a function of  $\vec{x}$  in the region around the sensor yielding  $\nabla_{\vec{x}} \vec{b} = \vec{0}$  and that  $\mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} \mathbf{G}_0$  is nonsingular, then that value of  $\vec{x}$  that yields the minimum  $Q(\vec{x})$  is given by

$$\hat{\vec{x}} = (\mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} \mathbf{G}_0)^{-1} \mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} \vec{d}_1 - (\mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} \mathbf{G}_0)^{-1} \mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} \vec{b} \quad (5.69)$$

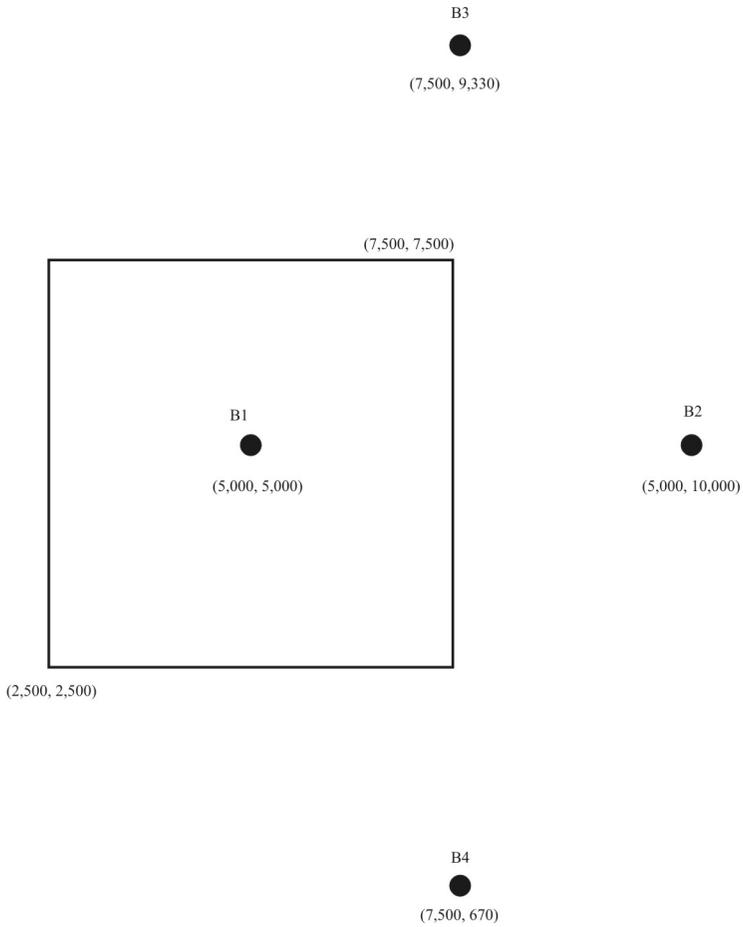
If the biases are known, then (5.69) can be used to determine the best estimate of the target location. If the biases are not known, then they are assumed to be zero and, using (5.67),

$$\hat{\vec{x}} = (\mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} \mathbf{G}_0)^{-1} \mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} [\vec{d} - \vec{f}(\vec{x}_0)] + \vec{x}_0 \quad (5.70)$$

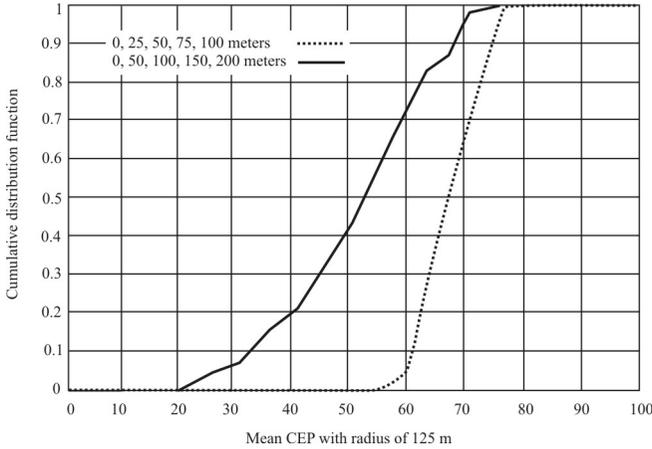
and, using (5.62),

$$\hat{\vec{x}} = (\mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} \mathbf{G}_0)^{-1} \mathbf{G}_0^T \mathbf{C}_{\Delta \vec{d} \Delta \vec{d}}^{-1} [\vec{f}(\vec{x}) - \vec{f}(\vec{x}_0) - \mathbf{G}_0 (\vec{x} - \vec{x}_0) + \vec{b} + \vec{n}] + \vec{x} \quad (5.71)$$

The indication of how the bias affects the PF estimate is given by the CEP. Assuming that the measurement errors are Gaussian, a simulation was performed using the geometry illustrated in Figure 5.14 [18]. The base stations are indicated by the black dots. The CEP was calculated at 100 equally spaced points within the square and the average value over the area was used as the figure of merit. The standard deviation of the measurements at the base stations was assumed to be 50 meters. The mean values of bias were assumed to be 0, 50, 100, 150, and 200 meters as one set and 0, 25, 50, 75, and 100 meters as another set. One hundred evenly spaced points within the square were selected and the mean CEP computed for each of them assuming the bias values given. A comparison of the results is shown in Figure 5.15 [18]. This cumulative distribution function shows the fraction of cells with CEPs less than the percentage shown on the abscissa, which is the percentage of the 125 m CEP. The rest of the CEPs were larger than 125 m.



**Figure 5.14** Geometry to simulate the effects of measurement bias on TDOA PF estimation in a cellular phone system.



**Figure 5.15** Cumulative distribution function of the CEP considering the effects of bias. The abscissa is the percentage within 125 m. The others were larger than 125 m. The mean measurement variance = 2,500 m<sup>2</sup>.

Clearly, the performance with smaller biases produced significantly more CEPs within the 125 m limit imposed by the FCC.

**5.2.9 Effects of Movement on TDOA PF Estimation**

When there is relative motion between the target and sensors, which would normally be the case for airborne sensors, then errors in TDOA measurements arise. Chan and Ho call this effect the *scale difference of arrival* (SDOA) [19] and provide a method of estimating its effects. The TDOA error induced by this motion is proportional to the SDOA and the data record length.

The signals received at the two sensors are given by

$$r_1(t) = s(t) + n_1(t) \tag{5.72}$$

$$r_2(t) = s\left(\frac{t+D}{a}\right) + n_2(t) \tag{5.73}$$

where  $s(t)$  is the transmitted signal,  $n_1(t)$  and  $n_2(t)$  are noise sources independent of each other and of the signal,  $D$  is the TDOA sought, and  $a$  is the SDOA. The joint estimate of the TDOA and SDOA can be obtained from the CAF between the two signals given by

$$\text{CAF} = \int_0^T r_1(t)r_2(\alpha t - \tau)dt \quad (5.74)$$

by finding the values of  $\alpha$  and  $\tau$  that maximize (5.74). Note that when there is no noise present,

$$r_2(\alpha t - \tau) = s\left(\frac{\alpha t - \tau + D}{a}\right) \quad (5.75)$$

so that (5.74) is maximized when  $\alpha = a$  and  $\tau = D$ .

The TDOA estimate from maximizing (5.74) *neglecting SDOA* is given by

$$\tau = (1 - a)t + D \quad (5.76)$$

so the bias error is

$$t - D = (1 - a)t \quad (5.77)$$

Averaging (5.77) from 0 to  $N$ , where  $N$  is the number of data samples, yields

$$\epsilon(a, N) = \frac{1}{N} \int_0^N (1 - a)t dt = \frac{(1 - a)N}{2} \quad (5.78)$$

The effects of the data record length on the TDOA bias are illustrated in Figure 5.16 [18]. Here the bias, neglecting the SDAO when there is no noise, with  $a = 1.001$  and  $D = 0$ , is plotted versus the record length. Clearly from (5.78) the bias is a linear function of the record length  $N$ .

The reference contains a fast algorithm for computing the TDOA MSE and SDOA MSE from the CAF. The results for one case are illustrated in Figures 5.17 and 5.18 [19]. Figure 5.17 illustrates the MSE estimation error on the TDOA versus the SNR given by

$$\gamma = 10 \log \frac{\sigma_s^2}{\sigma_n^2}, \quad \text{dB} \quad (5.79)$$

where  $\sigma_n^2 = \sigma_{n_1}^2 = \sigma_{n_2}^2$  and band-limited white noise is assumed. Likewise, Figure 5.18 shows the MSE estimation error of the SDOA versus SNR. At low SNR the

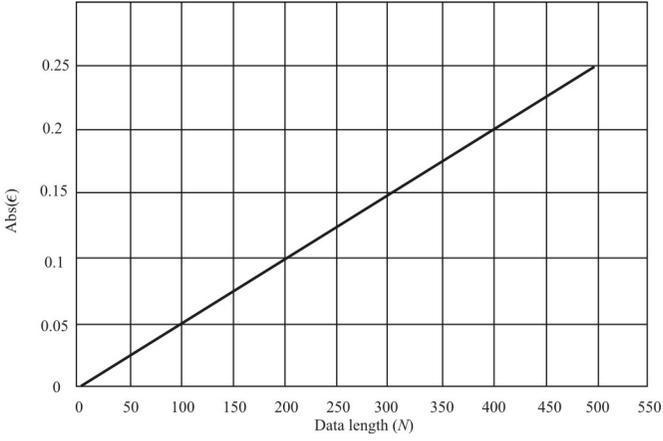


Figure 5.16 TDOA bias error caused by data record length, ignoring the SDOA.

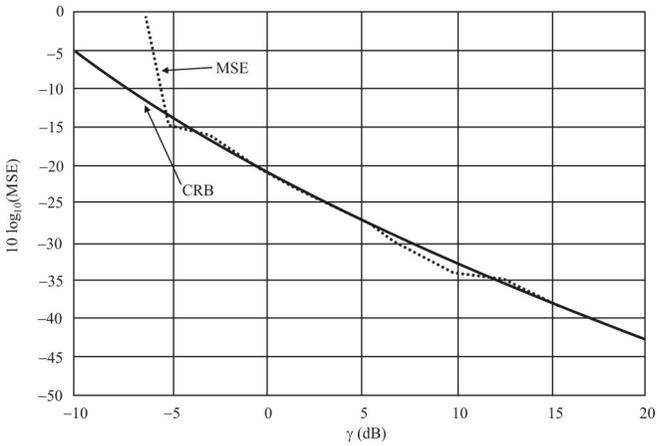


Figure 5.17 TDOA bias error caused by motion.

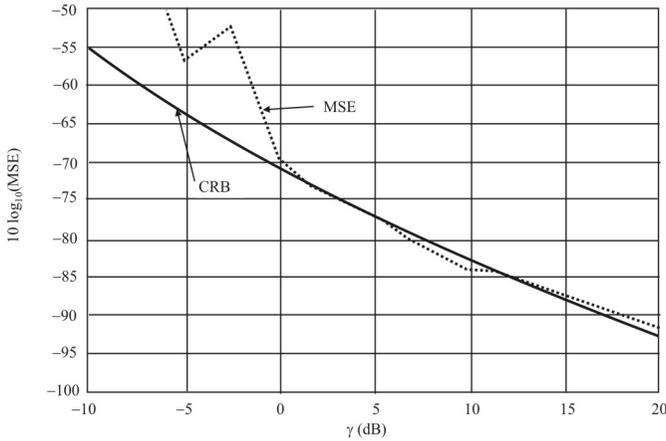


Figure 5.18 SDOA error caused by relative motion.

effects on the TDOA MSE are significantly more pronounced than those on the SDOA. At  $\gamma = 0$  dB, the SDOA MSE is 45 dB less than that for the TDOA.

### 5.3 Differential Doppler

Utilization of frequency difference information at two or more sensors for PF calculation is presented in this section. This follows that in [1] closely, and is included here for completeness.

#### 5.3.1 Introduction

The signals emitted by a moving object exhibit an effect called Doppler shift that manifests itself in a frequency difference depending on the direction of travel of the moving object relative to a receiver. When the frequency of a signal is measured at two (or more) sensor sites, the frequency difference between the signals at the two sensors can be computed. This frequency difference, called DD, is a measurement that can be used for target geolocation estimation. To utilize the DD effect, only one object needs to be moving—one or more sensors or the target. The higher the velocity the greater the frequency difference, which makes the DD easier to measure accurately.

### 5.3.2 DD

The DD, where the two sensors and/or target velocities are much smaller than the speed of light, is given by

$$\dot{\tau} = \frac{v_2}{c} f_0 - \frac{v_1}{c} f_0 = \frac{f_0}{\lambda} (v_2 - v_1) \quad (5.80)$$

where  $v_i$  represents the instantaneous velocity of the sensors relative to the transmitter in the radial direction and  $f_0$  is the frequency of the transmitted signal. The DD is frequently denoted by  $\dot{\tau}$  because it represents the difference in two velocities, which are, in turn, the time derivatives of the distances, to wit,

$$\dot{\tau} = \frac{f_0}{c} \left( \frac{dd_2}{dt} - \frac{dd_1}{dt} \right) \quad (5.81)$$

where  $d_i$  is the range between the transmitter and the sensor, as shown in Figure 5.1. Thus,  $dd_i/dt$  is the rate of change of the range in the radial direction. The distances are given by

$$d_i = \sqrt{(x_T - x_i)^2 + y_T^2} \quad (5.82)$$

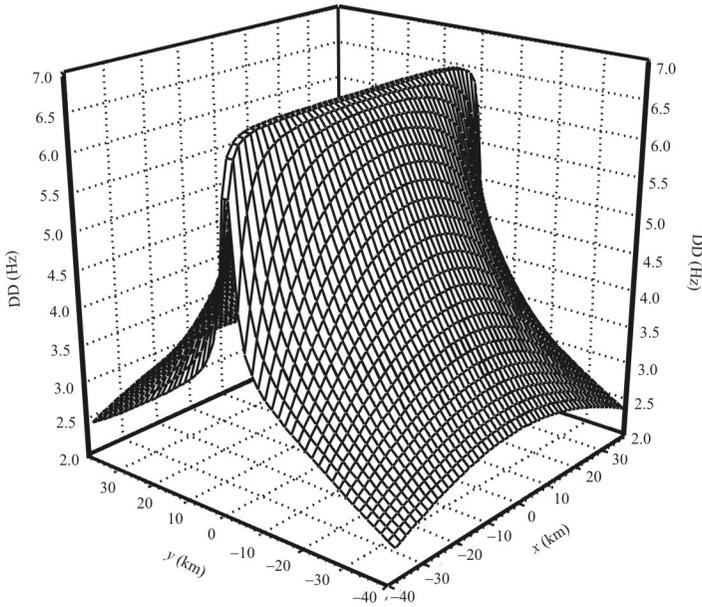
yielding

$$\frac{dd_i}{dt} = \frac{d \left[ (x_T - x_i)^2 + y_T^2 \right]^{1/2}}{dt} \quad (5.83)$$

$$= \frac{(x_T - x_i)}{\left[ (x_T - x_i)^2 + y_T^2 \right]^{1/2}} \frac{dx_i}{dt}, \quad i = 1, 2 \quad (5.84)$$

where it is assumed that the aircraft are flying at the same speed parallel to the  $x$ -axis so that  $dy_i/dt = 0$ . Denoting  $v = v_1 = dx_1/dt = v_2 = dx_2/dt$ , then

$$\dot{\tau} = \frac{f_0 v}{c} \left\{ \frac{(x_T - x_1)}{\left[ (x_T - x_1)^2 + y_T^2 \right]^{1/2}} - \frac{(x_T - x_2)}{\left[ (x_T - x_2)^2 + y_T^2 \right]^{1/2}} \right\} \quad (5.85)$$



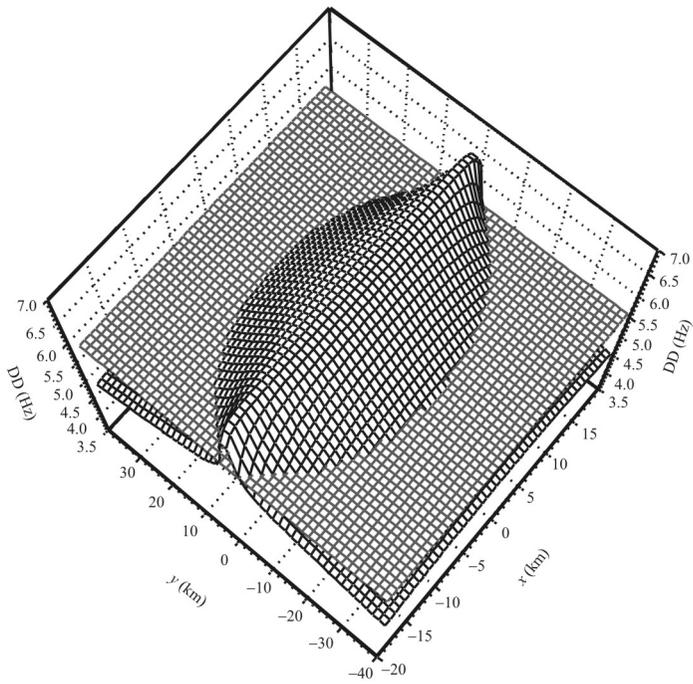
**Figure 5.19** Surface of the DD for the example. In this case  $v = 10$  m/sec and  $f = 100$  MHz ( $\lambda = 3$ m). (Source: [1]. © Horizon House 2002. Reprinted with permission.)

The surface formed by this expression for the example here is shown in Figure 5.19. The DD curves are complex quadratic functions, not simple hyperbolas, as was the case for TDOA processing. Viewed from the top they look as seen in Figure 5.20. The intersection of these curves for the appropriate value of  $\hat{\tau}$  will yield candidate geolocations for the emitter, but with only two sensors there is a left-right ambiguity as seen in Figures 5.21 and 5.22. As with TDOA, adding a third sensor can resolve the ambiguity and produce a unique solution, as shown in Figure 5.23.

### 5.3.3 DD Measurement Accuracy

Just as for TDOA, at extreme ranges from the baselines the iso-Doppler curves become nearly parallel, as illustrated in Figure 5.26. For accurate PF estimates it is desirable for the Doppler isochrone lines to cross at right angles. When they are nearly parallel, as in Figure 5.24, poor PF estimates result.

The CRB for estimating  $\hat{\tau}$  is given by



**Figure 5.20** Top view of the  $\tau$  contour. The shape resembles an ellipse, but actually it is a complex quadratic. (Source: [1]. © Horizon House 2002. Reprinted with permission.)

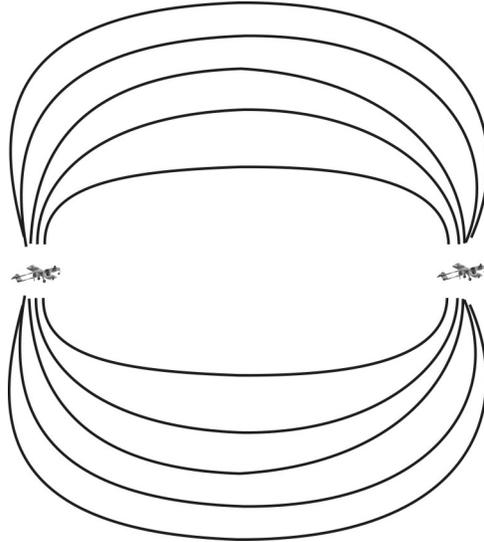


Figure 5.21 Lines of constant  $\tau$  are complex quadratic curves.

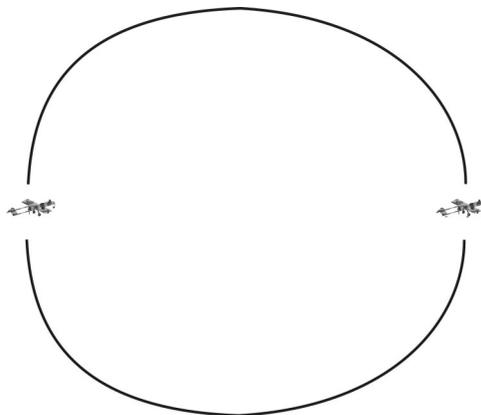
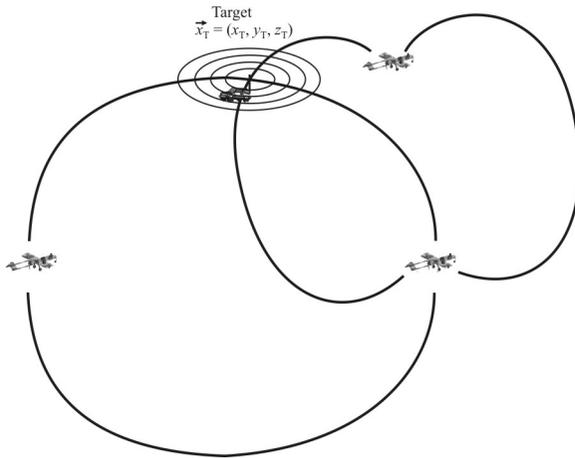


Figure 5.22 DD contours corresponding to an emitting target. The contours have a left-right ambiguity.



**Figure 5.23** Iso-Doppler contours for three sensors can uniquely locate the target.

$$\sigma_{\tau} = \frac{1}{T_e} \frac{1}{\sqrt{WT\gamma}} \quad (5.86)$$

where  $W$ ,  $T$ , and  $\gamma$  are as above,  $\sigma_{\tau}$  is the standard deviation of the  $\tau$  measurement, and  $T_e$  is the RMS integration time given by

$$T_e = 2\pi \left[ \frac{\int_{-\infty}^{\infty} t^2 |u(t)|^2 dt}{\int_{-\infty}^{\infty} |u(t)|^2 dt} \right]^{1/2} \quad (5.87)$$

where  $u(t)$  is the PDF of the integration time. Again, for example, if the actual integration time is  $T$  and the PDF for the integration time has sharp edges, then  $T_e = \pi T / \sqrt{3}$ . Equation (5.86) is plotted in Figure 5.25 for a few values of integration time versus SNRs. As indicated in Figure 5.21, the DD values for representative values of parameters are in the single-to-double digit hertz range. The standard deviation values indicated in Figure 5.25 for low SNRs are only about 1/100 those values, so the possible accuracy at these integration times may be suspect. Higher SNRs produce better fixes.

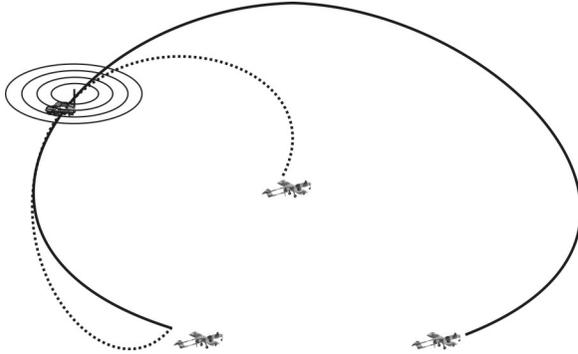


Figure 5.24 Iso-Doppler contours for a distant target are nearly parallel.

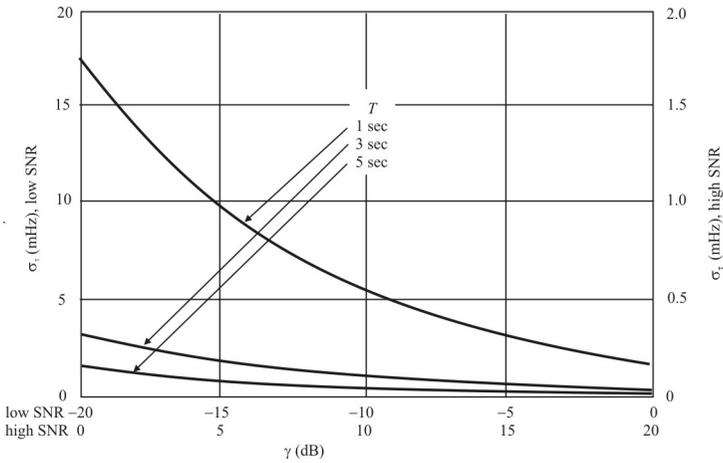


Figure 5.25 The CRB for DD processing versus SNRs assuming a uniform distribution for  $T_c$  and  $W = 25$  kHz.

### 5.3.4 Maximum Likelihood DD Algorithms

A DD MLE algorithm similar to the LOB MLE algorithm described in Section 2.6 has been developed by Levanon [20]. A PF system that uses DD measurements avoids the ambiguity problem inherent in interferometer PF systems. The antennas on the same platform can be separated by more than  $\lambda/2$  and effective measurements can be obtained. On an airborne sensor platform, however, relatively dense measurements are required. For an airborne sensor, the Doppler measurements must be made often enough so that at least one measurement is taken during each interval defined by the wavelength of the signal.

An interferometer, on the other hand, must have the sensor receivers spaced less than  $\lambda/2$  to avoid ambiguous results.

The DD is determined by utilizing many measurements of the phase, and, therefore, range differences along the path. Each time the phase jumps by  $2\pi$ , one more wavelength in range has been traversed. These jumps are counted and used as a measurement of the differential frequency.

#### 5.3.4.1 DD MLE Algorithm

The unknowns in the geometry shown in Figure 2.32, repeated here as Figure 5.26 [20], are  $y_c$ , the center of the observation interval, and  $x_T$ , the location of the target (it is assumed for now that  $y_T = 0$ ). Therefore, the vector of unknowns is

$$\bar{\theta} = [x_T \quad y_c]^T \quad (5.88)$$

Denote the noiseless relationship between the unknowns and the measurement during the  $k$ th observation as  $\tilde{z}_k(\bar{\theta})$ . For the case of the Doppler measurements here,

$$\tilde{z}_k(\bar{\theta}) = R_{1k} - R_{2k} - (R_{10} - R_{20}) \quad (5.89)$$

where, for  $k = 0, \pm 1, \pm 2, \dots, \pm K$ ,  $N = 2K + 1$ ,

$$R_{2k} = \sqrt{\left(y_c - \frac{b}{2} + k\Delta L\right)^2 + x_T^2}$$

$$R_{1k} = \sqrt{\left(y_c + \frac{b}{2} + k\Delta L\right)^2 + x_T^2}$$

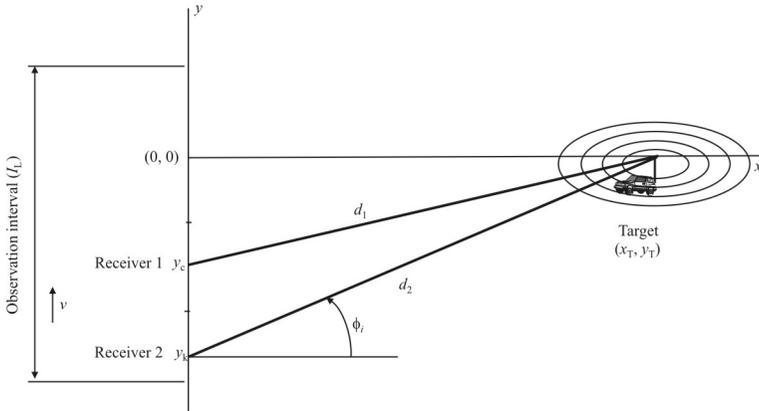


Figure 5.26 MLE geometry.

$\Delta L$  is the distance traveled between measurements

With these values of  $k$ , the reference is at the center of the observation interval  $I_L$ . The  $2K + 1$  noisy measurements, denoted by  $z_k$ , corrupted by samples of the Gaussian noise process denoted by  $n_k$ , are given by

$$z_k(\vec{\theta}) = \tilde{z}_k(\vec{\theta}) + n_k, \quad k = 0, \pm 1, \pm 2, \dots, \pm K \tag{5.90}$$

The final covariance matrix for the variables gives the PF error. It is given by

$$\mathbf{C}_{\vec{\theta}\vec{\theta}} = \text{Cov}(\vec{\theta}) = \sigma_n^2 (\mathbf{G}^T \mathbf{G})^{-1} \tag{5.91}$$

The error in the 0th range difference measurement  $\epsilon$  must be accommodated in the noiseless measurement (5.89), changing it to

$$\tilde{z}_k(\vec{\theta}) = R_{1k} - R_{2k} - (R_{10} - R_{20}) + \epsilon \tag{5.92}$$

where  $\epsilon$  is an unknown and therefore must be estimated along with  $x_T$  and  $y_c$ . Thus, the unknown vector to be estimated is

$$\vec{\theta} = [x_T \quad y_c \quad \epsilon]^T \tag{5.93}$$

With two simplifying assumptions that  $y_c = 0$  and  $K\Delta L \ll x_T$ , then

$$\partial_{kx} = \frac{\partial \bar{z}_k(\bar{\theta})}{\partial x_T} \tag{5.94}$$

$$\approx -\frac{bk\Delta L}{x_T^2} \tag{5.95}$$

and

$$\partial_{ky} = \frac{\partial \bar{z}_k(\bar{\theta})}{\partial y_c} \tag{5.96}$$

$$\approx -\frac{3b(k\Delta L)^2}{2x_T^3} \tag{5.97}$$

The **G** matrix now is

$$\mathbf{G} = \begin{bmatrix} \partial_{-Kx} & \cdots & \partial_{kx} & \cdots & \partial_{Kx} \\ \partial_{-Ky} & \cdots & \partial_{ky} & \cdots & \partial_{Ky} \\ 1 & \cdots & 1 & \cdots & 1 \end{bmatrix}^T$$

which, using (5.94) and (5.96), generates

$$\begin{aligned} \mathbf{G}^T \mathbf{G} &= \begin{bmatrix} \partial_{-Kx} & \cdots & \partial_{kx} & \cdots & \partial_{Kx} \\ \partial_{-Ky} & \cdots & \partial_{ky} & \cdots & \partial_{Ky} \\ 1 & \cdots & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} \partial_{-Kx} & \partial_{-Ky} & 1 \\ \vdots & \vdots & \vdots \\ \partial_{ky} & \partial_{ky} & 1 \\ \vdots & \vdots & \vdots \\ \partial_{Kx} & \partial_{Ky} & 1 \end{bmatrix} \\ &= \begin{bmatrix} \sum_{k=-M}^M (\partial_{kx})^2 & \sum_{k=-M}^M \partial_{kx} \partial_{ky} & \sum_{k=-M}^M \partial_{kx} \\ \sum_{k=-M}^M \partial_{kx} \partial_{ky} & \sum_{k=-M}^M (\partial_{ky})^2 & \sum_{k=-M}^M \partial_{ky} \\ \sum_{k=-M}^M \partial_{kx} & \sum_{k=-M}^M \partial_{ky} & \sum_{k=-M}^M 1 \end{bmatrix} \end{aligned}$$

Assuming that  $\Delta L$  is small enough that it can be approximated by a differential, then converting these sums to integrals, dividing (outside the sum) by  $dL$ , and multiplying (inside the sum) by  $dL$  yields

$$\mathbf{G}^T \mathbf{G} = \begin{bmatrix} \frac{Nb^2 I_L^2}{12x_T^4} & 0 & 0 \\ 0 & \frac{9Nb^2 I_L^4}{320x_T^6} & \frac{NbI_L^2}{8x_T^3} \\ 0 & \frac{NbI_L^2}{8x_T^3} & N \end{bmatrix} \quad (5.98)$$

and

$$(\mathbf{G}^T \mathbf{G})^{-1} = \begin{bmatrix} \frac{12x_T^4}{Nb^2 I_L^2} & 0 & 0 \\ 0 & \frac{80x_T^6}{Nb^2 I_L^4} & -\frac{10x_T^3}{NbI_L^2} \\ 0 & -\frac{10x_T^3}{NbI_L^2} & \frac{9}{4N} \end{bmatrix} \quad (5.99)$$

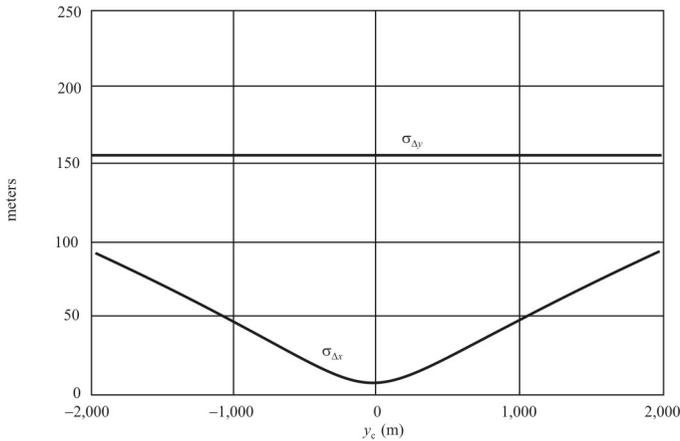
Now from (5.91) and (5.99),

$$\sigma_x \approx \sqrt{\frac{12}{N} \frac{\sigma_{\Delta R}}{b} \frac{x_T^2}{I_L^2}} \quad (5.100)$$

$$\sigma_y \approx \sqrt{\frac{80}{N} \frac{\sigma_{\Delta R}}{b} \frac{x_T^3}{I_L^2}} \quad (5.101)$$

When  $y_c \neq 0$ , (5.100) becomes more complicated while (5.101) remains the same. Thus,

$$\sigma_x \approx \sqrt{\frac{12}{N} \frac{\sigma_{\Delta R}}{b} \frac{x_T^2}{I_L^2} \sqrt{1 + 60 \frac{y_c^2}{I_L^2}}} \quad (5.102)$$



**Figure 5.27** Differential Doppler results. For this example  $N = 40$ ,  $x_T = 10$  km,  $I_L = 1.6$  km, and  $\sigma_{\Delta R} = 0.0141$ m.

### 5.3.4.2 Performance

The DD results are portrayed graphically in Figure 5.27 [20]. These values of the parameters would correspond to a UAV platform that is prosecuting a fixed-frequency target that is transmitting a considerable length of time. A UAV flying at 100 kph would require about 1 minute to fly 1.6 km, so the target must be transmitting for at least that length of time.

Comparing Figure 5.27 with Figure 2.36 for the interferometric algorithm, it can be seen that the DD performance is better in the down-range estimation, while the interferometer is better in the cross-range results. The DD performance would be expected to be good, especially when  $y_c = 0$ , because at that point the quadratic is intersecting with the  $x$ -axis at right angles, yielding the highest sensitivity possible from the DD measurements (see Figure 5.28). The fact that the variance is a minimum can be established by finding the derivative of (5.102) with respect to  $y_c$  and setting it to zero.

The highest sensitivity in the interferometric measurement also occurs at  $y_c = 0$ , as seen in Figure 5.29. At that point, the slope of the  $\Delta\phi$  versus  $y_c$  curve is maximized because it is an inflection point. The difference in phase of the signals impinging on the two antennas is zero since the arrival azimuths are orthogonal to the  $y$ -axis. Again, the minimum variance point can be shown to be at  $y_c = 0$  by taking the derivative of (2.125) with respect to  $y_c$  and setting it equal to zero.

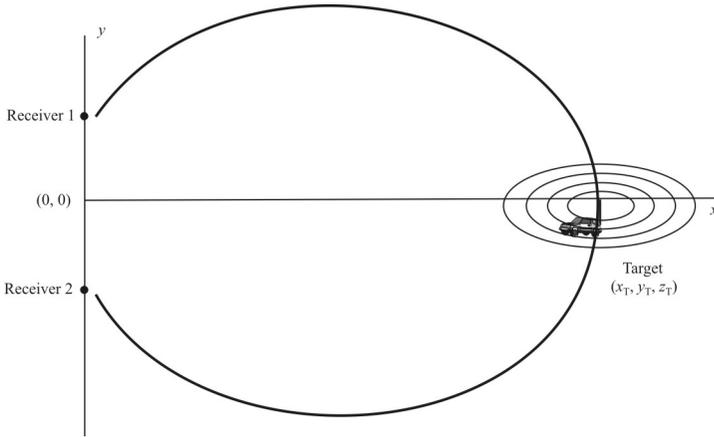


Figure 5.28 Intersection of the iso-Doppler contour with the  $x$ -axis when  $y_c = 0$ .

### 5.3.5 Cross-Ambiguity Function

TDOA and DD calculations can be combined to take advantage of the benefits of each, as well as possibly increase the accuracy of the PF estimates. This is typically accomplished with the CAF. This computation simultaneously yields the  $\tau$  and  $\hat{\tau}$  for two sensors, and, just as for TDOA isochrones and iso-Doppler curves, the CAF must be computed for each pair of sensors in order to yield sufficient information to calculate an estimate. The CAF is a generalization of the cross-correlation function and is given by

$$\text{CAF}(\tau, t) = \int_0^T s_1(t) s_2(t + \tau) e^{-j\omega t} dt \tag{5.103}$$

Under ideal conditions, a plot of the amplitude of the CAF versus  $\tau$  and  $\hat{\tau}$  displays a peak at the proper value of these variables.

As indicated in [1], the standard deviation of the measurements of the PF coordinates, calculated on the perpendicular bisector of the baseline shown in Figure 5.1, depends on these equations as follows:

$$\sigma_x = \frac{c\sigma_\tau \sqrt{\left(\frac{b}{2}\right)^2 + d^2}}{d} \tag{5.104}$$

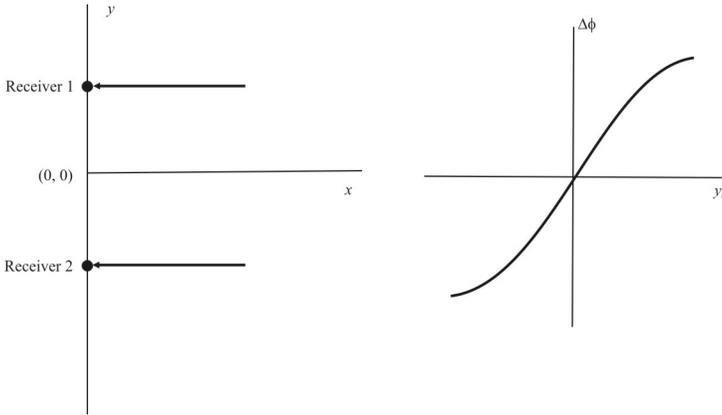


Figure 5.29 Least-squared LOB geometry when  $y_c = 0$ .

and

$$\sigma_y = \frac{\lambda \sigma_{\tau} \left[ \left( \frac{b}{2} \right)^2 + d^2 \right]^{3/2}}{vbd} \tag{5.105}$$

where  $\lambda$  is the wavelength,  $v$  is the velocity of the sensors,  $b$  is the baseline length, and  $d$  is the distance from the baseline.

As with TDOA and DD processing alone, as the target deviates from the perpendicular bisector, GDOP comes into play and the accuracy degrades.

### 5.3.6 Estimating the DD of a Sinusoid in Noise Using Phase Data

Otnes derived a method of estimating the DD of a sinusoidal signal in noise received at two or more platforms using regression techniques [21]. In other words, the DD is estimated as the slope of the line that is the linear regression of the DD data. This is a similar approach to that described in Section 5.1.8, which discussed a regression matching technique for estimating the TDOA derived by Piersol [22].

The signal transmitted by the target is given by

$$s(t) = Ae^{j\omega_0 t + \phi_0} \tag{5.106}$$

where  $\omega_0$  is the frequency and  $\phi_0$  is the phase offset. The signals received at two geographically dispersed sensors  $j$  and  $k$  are given by

$$r_j(t) = s(t) + n_j(t) \quad (5.107)$$

$$r_k(t) = s(t - D_{jk}) + n_k(t) \quad (5.108)$$

where it is assumed that the noises are uncorrelated to each other or to the transmitted signal. It is also assumed that there is relative motion between one or more sensors and the target. Normalizing the received signals so the time delay can be ignored and using relative phase delays so  $\phi_0 = 0$  results in the received signals given by

$$r_j(t) = A_j e^{j(\omega_j t + \phi_j)} + n_j(t) \quad (5.109)$$

$$r_k(t) = A_k e^{j(\omega_k t + \phi_k)} + n_k(t) \quad (5.110)$$

The *mixing product* [6] of these two signals is given by

$$p_{jk}(t) = r_j(t)r_k^*(t) \quad (5.111)$$

Because of the assumed independence of the noise from the signals,

$$\mathcal{E}\{p_{jk}(t)\} = A_j A_k^* e^{j(\omega_{jk} t + \phi_{jk})} \quad (5.112)$$

where  $\omega_{jk} = \omega_j - \omega_k$  and  $\phi_{jk} = \phi_j - \phi_k$ .

Using (5.109) and (5.110) in (5.111) gives

$$p_{jk}(t) = A_j A_k^* e^{j(\omega_{jk} t + \phi_{jk})} + n_{jk}(t) \quad (5.113)$$

$$= A e^{j\omega_{jk} t} + n_{jk}(t) \quad (5.114)$$

where  $A = A_j A_k^* e^{j\phi_{jk}}$  and  $n_{jk}(t)$  is the noise term consisting of the remaining products resulting from (5.111) incorporating (5.109) and (5.110).

Ignoring noise, the phase of the mixing product after the phase has been unwrapped is given by

$$y(t) = \tan^{-1} \frac{\text{Im}[p_{jk}(t)]}{\text{Re}[p_{jk}(t)]} = \tan^{-1} \frac{|A| \sin(\omega_{jk}t + \varphi_{jk})}{|A| \cos(\omega_{jk}t + \varphi_{jk})} = \omega_{jk}t + \varphi_{jk} \quad (5.115)$$

Thus, ignoring noise, the phase function is a linear function of time. With noise included and assuming a sufficiently high SNR, the phase function can be expressed as

$$y_{jk}(t) = \omega_{jk}t + \varphi_{jk} + n(t) \quad (5.116)$$

and  $\omega_{jk}$  and  $\varphi_{jk}$  can be determined using linear regression as discussed in Section 2.2, resulting in the MLSE estimate provided by (2.31).

Considering discrete time processing, during time interval  $i$ ,

$$y_i = \omega_0 x_i + \varphi_0 + n_i = \omega_0 iT + \varphi_0 + n_i \quad (5.117)$$

because  $x_i = iT$  with  $T$  the sampling interval. The optimum DD is given by

$$\omega_{jk}^* = \frac{\sum_{i=1}^N y_i (x_i - \bar{x})}{\sum_{i=1}^N (x_i - \bar{x})^2} \quad (5.118)$$

where  $\bar{x}$  denotes the average value of  $x$ .

We can reduce (5.118) to

$$\omega_{jk}^* = \frac{\omega_0 TQ(N) + \sum_{i=1}^N n_i \left( i - \frac{N+1}{2} \right)}{TQ(N)} \quad (5.119)$$

where

$$Q(N) = \frac{2N(N-1)(N+1)}{24} \quad (5.120)$$

The corresponding variance is given by

$$\sigma_{\omega_{jk}}^2 = \frac{\sum_{i=1}^N y_i(x_i - \bar{x}) - \frac{\left[ \sum_{i=1}^N y_i(x_i - \bar{x}) \right]^2}{\sum_{i=1}^N (x_i - \bar{x})^2}}{(N-2) \sum_{i=1}^N (x_i - \bar{x})^2} \quad (5.121)$$

The expected value of the variance for large  $N$  is given by

$$\mathcal{E}\{\sigma_{\omega_{jk}}^2\} \approx \frac{12\sigma_n^2}{T^2 N^3} \quad (5.122)$$

where  $\sigma_n^2$  is the variance of the phase noise. Thus, the variance in the estimation of the DD decreases as the square of the sampling interval and the cube of the number of samples ( $N$ ). This indicates that the accuracy in estimating the DD improves by these factors, illustrating the significant advantage of using a large number of samples.

Utilizing some assumptions, this method of estimating the DD will work when there is *amplitude modulation* (AM) applied to the sinusoidal carrier. In particular, when the DD value being sought is the fine value (as opposed to the coarse value which just gets close), then the amplitude of the modulation can be assumed to be equal to one and (5.109) and (5.110) still apply. This amplitude can be made to equal to one by clipping the signal, for example.

Because the parameter being sought is the differential frequency received at two sensor sites, this estimation method would not work well for signals with *frequency modulation* (FM) or *phase modulation* (PM).

### 5.3.7 Effects of Motion on DD PF Estimating

As discussed in Section 5.1.8, relative motion between the target and sensor can cause errors in the measurement of TDOA unless such motion can be taken into account. Even though it is necessary for there to be relative motion between the target and a receiver to be able to observe Doppler shift, assuming the transmitter to be stationary when it is in fact moving can lead to considerable error estimating the DD. It can be very difficult to take out the effects of a moving target on calculating the DD. An alternative is to attempt to ascertain that the target is moving or stationary, and, if moving, do not attempt to estimate the PF because it will contain substantial error.

Ullman and Geraniotis [23] developed two techniques for estimating when the transmitter is moving so that the data obtained can be ignored for PF computation purposes. Both techniques assume that the target position is confined to some surface and its velocity is constrained to a tangent plane. Also, both techniques form and test the hypotheses

$$H_0 : \vec{v}_e = \vec{0} \quad (5.123)$$

$$H_1 : \vec{v}_e \neq \vec{0} \quad (5.124)$$

where  $\vec{v}_e$  is the actual velocity vector of the target.

The first technique is based on postulating that the transmitter is stationary. After several samples of TDOA/DD data are collected, the maximum likelihood stationary target position is found using these measurements and the probability of the measurements occurring given this position is determined. Let  $\vec{q}(\vec{p}_e, \vec{v}_e)$  denote the TDOA/DD measurement vector. Let  $\mathbf{C}$  denote the covariance matrix of measurements. The observed measurement vector,  $\vec{q}_m$ , consisting of noise corrupted measurements is assumed to be  $\vec{q}_m \sim \mathcal{N}[\vec{q}(\vec{p}_e, \vec{v}_e), \mathbf{C}]$ , where  $\vec{p}_e$  is the actual position vector of the target. The maximum likelihood estimate is then given by

$$\tilde{\vec{p}}_e = \underset{p}{\operatorname{argmin}} [\vec{q}(\vec{p}, \vec{0}) - \vec{q}_m]^T \mathbf{C}^{-1} [\vec{q}(\vec{p}, \vec{0}) - \vec{q}_m] \quad (5.125)$$

The second technique is based on a *likelihood ratio test* (LRT) (LRT-like actually, since it is not a true LRT) given by

$$\text{LRT} = \frac{\max_{\vec{p}_e, \vec{v}_e} \Pr_1 \{ \vec{q}_m \mid \vec{p}_e, \vec{v}_e \}}{\max_{\vec{p}_e} \Pr_0 \{ \vec{q}_m \mid \vec{p}_e, \vec{v}_e = \vec{0} \}} \quad (5.126)$$

Both of these techniques yield similar results, although the specific results are highly scenario dependent, as we would expect, since the performance of the algorithms used for estimating the PF using TDOA and DD measurements are scenario dependent (due to GDOP, for example).

A comparison of the two techniques for specific examples illustrates that they produce similar results at detection of target motion. The test for stationarity is relatively simple compared to the LRT test, for which the computation complexity is substantially larger. Based on this result, the test for stationarity is the recommended approach in most situations.

An alternative to ignoring the data when target motion is detected is to estimate the velocity vector along with the target location. One such technique was developed by Rusu [24]. In that work, the TDOA and FDOA data are transformed into TOA and FOA data and closed-form expressions for the target location and velocity are used to determine the unknowns by using the implicit function theorem. The algebraic results are compared with simulation results.

Another technique for estimating the target velocity along with the location was developed by Ho and Xu [25]. Least squares minimizations are used to determine the unknowns. When the TDOA and FDOA errors are Gaussian, the technique is claimed to achieve the CRB. The analytical results are compared favorably with the algebraic results.

## 5.4 Range Difference Methods

### 5.4.1 Introduction

The time of arrival of a signal at sensor  $i$  is given by

$$t_i = \frac{d_i}{c} \quad (5.127)$$

where  $d_i$  is the range from the target to sensor  $i$ . The corresponding TDOA of that signal arriving at two sensors is given by

$$(t_i - t_j) = \tau_{ij} = \frac{d_i - d_j}{c} \quad (5.128)$$

In the absence of errors due to inaccurate measurements or noise, the target must lie at the intersection of hyperboloids that are formed as the locus of constant RD as determined by (5.128). Therefore, finding this intersection is a method for establishing where the target lies.

Noise and measurement errors are always present, however, and normally these hyperboloids do not intersect at a single point but at several points. LSE techniques can then be used to determine the point where the total distance from that point to each of the hyperboloids is minimized.

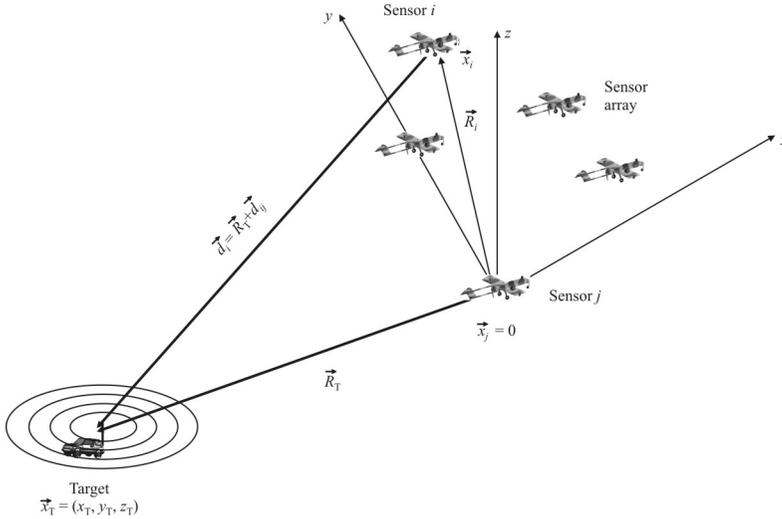


Figure 5.30 Geometry of the problem.

### 5.4.2 Least-Squares Range Difference Methods

Let  $N$  represent the number of sensors and let  $\Delta_{ij}$  represent the range difference between sensor  $i$  and the target and sensor  $j$  and the target. The location of sensor  $i$  is given by the coordinate vector  $\vec{x}_i = [x_i \ y_i \ z_i]^T$  and the location of the target is given by the vector  $\vec{x}_T = [x_T \ y_T \ z_T]^T$ . The distance between the target and sensor  $i$  is given by

$$d_i = \|\vec{x}_i - \vec{x}_T\|_2 \tag{5.129}$$

while the distance from the origin to point  $i$  is denoted by  $R_i$ .

Smith and Abel [26] formulated a method for minimizing the PF estimation error in an LS sense. They define the *equation error* as follows. Consider the geometry indicated in Figure 5.30 [24] where, without loss of generality, sensor  $j$  is placed at the origin. Also without loss of generality,  $j = 1$ . Thus,

$$x_j = 0 \quad \text{and} \quad R_j = 0, d_j = R_T \tag{5.130}$$

Now

$$(R_T + d_{ij})^2 = R_i^2 - 2\vec{x}_i^T \vec{x}_T + R_T^2 \tag{5.131}$$

so

$$R_i^2 - d_{ij}^2 - 2R_T d_{ij} - 2\bar{x}_i^T \bar{x}_T = 0 \tag{5.132}$$

yielding  $N - 1$  equations in three unknowns.

With measurement errors and noise in the data, (5.132) is not identically zero, but it is equal to some residual amount of error denoted by  $\epsilon_i$ . That is,

$$R_i^2 - d_{ij}^2 - 2R_T d_{ij} - 2\bar{x}_i^T \bar{x}_T = \epsilon_i, \quad i = 1, 2, \dots, N \tag{5.133}$$

In matrix notation, (5.133) can be expressed as

$$\bar{\epsilon} = \bar{\delta} - 2R_T \bar{d} - 2\mathbf{S}\bar{x}_T \tag{5.134}$$

where

$$\bar{\delta} = \begin{bmatrix} R_2^2 - d_{21}^2 \\ R_3^2 - d_{31}^2 \\ \vdots \\ R_N^2 - d_{N1}^2 \end{bmatrix} \quad \bar{d} = \begin{bmatrix} d_{21} \\ d_{31} \\ \vdots \\ d_{N1} \end{bmatrix} \quad \mathbf{S} = \begin{bmatrix} x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ \vdots & \vdots & \vdots \\ x_N & y_N & z_N \end{bmatrix}$$

The least-squares solution for  $\bar{x}_T$  when  $R_T$  is known with weighted equation error, where the weights  $\mathbf{W}$  on the range differences are determined a priori, is

$$\bar{x}_T = \frac{1}{2} \mathbf{S}_w^\diamond (\bar{\delta} - 2R_T \bar{d}) \tag{5.135}$$

where  $\mathbf{S}_w^\diamond$  is given by (2.16),

$$\mathbf{S}_w^\diamond = (\mathbf{S}^T \mathbf{W} \mathbf{S})^{-1} \mathbf{S}^T \mathbf{W} \tag{5.136}$$

Unfortunately,  $R_T$  is unknown if the location of the target is unknown, which it is in this case. In addition, expression (5.135) is nonlinear and some nonlinear minimization technique must be used to solve it.

Smith and Abel proposed a technique they referred to as the *spherical interpolation* (SI) method for solving (5.135). Furthermore, they compare three types of PF techniques based on range difference computations:

- The SI method;
- The *spherical intersection* (SX) method;
- The *plane intersection* (PX) method.

The SX method was developed by Schau and Robinson [27] and the PX method is based on a development by Schmidt described later.

#### 5.4.2.1 Spherical Interpolation Method

The SI method is based on substituting (5.135) into (5.134) and minimizing  $\bar{\epsilon}$  with respect to  $R_T$ . First, two matrices are defined as

$$P_{s,N-1 \times N-1} \triangleq \mathbf{S}(\mathbf{S}^T \mathbf{W} \mathbf{S})^{-1} \mathbf{S}^T \mathbf{W} \quad (5.137)$$

and

$$\mathbf{P}_{s,N-1 \times N-1}^\perp = \mathbf{I} - \mathbf{P}_s \quad (5.138)$$

yielding

$$\tilde{R}_T = \frac{1}{2} \frac{\bar{d}^T \mathbf{P}_s^\perp \mathbf{Y} \mathbf{P}_s^\perp \bar{\delta}}{\bar{d}^T \mathbf{P}_s^\perp \mathbf{Y} \mathbf{P}_s^\perp \bar{d}} \quad (5.139)$$

where  $\mathbf{Y}$  is a symmetric positive definite weighting matrix.

The resulting estimate for  $\bar{x}_T$  that minimizes the error in an LS sense is given by

$$\hat{\bar{x}}_s = \frac{1}{2} (\mathbf{S}^T \mathbf{P}_d^\perp \mathbf{W} \mathbf{P}_d^\perp \mathbf{S})^{-1} \mathbf{S}^T \mathbf{P}_d^\perp \mathbf{W} \mathbf{P}_d^\perp \bar{\delta} \quad (5.140)$$

where

$$\mathbf{P}_d^\perp \triangleq \mathbf{I} - \frac{\bar{d} \bar{d}^T}{\bar{d}^T \bar{d}} \quad (5.141)$$

is a projection matrix.

### 5.4.2.2 Spherical Intersection Method

The spherical intersection solution to find  $\bar{x}_T$  is found by substituting (5.135) into  $R_T^2 = \bar{x}_T^T \bar{x}_T$ , which yields

$$aR_T^2 + bR_T + c = 0 \tag{5.142}$$

where

$$a = 4 - 4\bar{d}^T \mathbf{S}_W^{\circ T} \mathbf{S}_W^{\circ} \bar{d} \tag{5.143}$$

$$b = 4\bar{d}^T \mathbf{S}_W^{\circ T} \mathbf{S}_W^{\circ} \bar{\delta} \tag{5.144}$$

$$c = -\bar{\delta}^T \mathbf{S}_W^{* T} \mathbf{S}_W^{*} \bar{\delta} \tag{5.145}$$

Then (5.142) yields two solutions as given by the roots

$$R_T = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{5.146}$$

Given  $R_T$ , it is substituted into (5.135) to obtain  $\bar{x}_T$ .

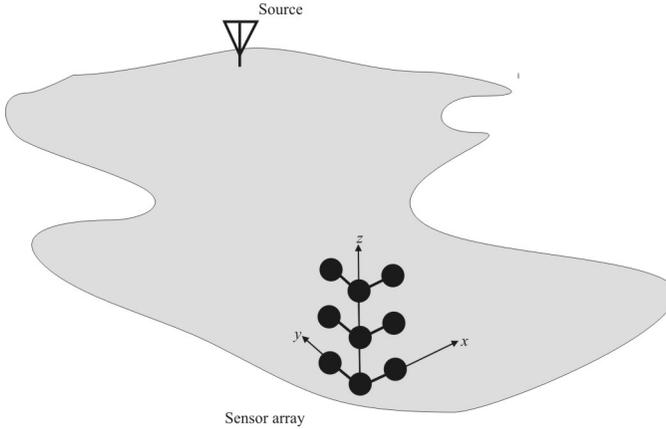
### 5.4.2.3 Plane Intersection Method

This technique is based on a development due to Schmidt [28]. In this case,

$$\epsilon_{ijk} = 2\bar{\Delta}_{ijk}^T \mathbf{S}_{ijk} \bar{x}_T - [d_{ji} d_{kj} d_{ik} + R_i^2 d_{kj} + R_j^2 d_{ik} + R_k^2 d_{ji}] \tag{5.147}$$

where  $\epsilon_{ijk}$  is the equation error,  $d_{jk}$  is the range difference measured between sensors  $j$  and  $k$ , and

$$R_i^2 = x_i^2 + y_i^2 \tag{5.148}$$



**Figure 5.31** Range difference example.

where  $\bar{x}_i = [x_i \ y_i]^T$  when  $x_i$  and  $y_i$  are the coordinates of sensor  $S_i$ . The three coordinate terms in (5.147) are given by

$$\bar{\Delta}_{ijk} = [d_{kj} \ d_{ik} \ d_{ji}]^T \quad (5.149)$$

$$\mathbf{S}_{ijk} \triangleq [\bar{x}_i^T \ \bar{x}_j^T \ \bar{x}_k^T]^T \quad (5.150)$$

Expression (5.147) is linear in  $\bar{x}_T$  and therefore can be used as the basis for an LS solution, just as (5.133) was used above.

#### 5.4.2.4 Performance Comparison

To compare the performance of these three methods, Smith and Abel conducted Monte Carlo simulation experiments with the configuration shown in Figure 5.31 [24] and Table 5.1. One hundred runs with four configurations were performed. The four cases were according to the parameters shown in Table 5.1. In all cases the sensors were located at (0, 0, 0), (0, 0, 100), (0, 0, 200), (100, 0, 0), (100, 0, 100), (100, 0, 200), (0, 100, 0), (0, 100, 100), and (0, 100, 200).

The bias results are given in Table 5.2, while the sample standard deviations are given in Table 5.3. The sample RMS values are given in Table 5.4. The spherical intersection is seen to have much higher errors than the other two methods. The spherical interpolation has the lowest RMS error, followed not too distantly by the plane intersection method.

**Table 5.1** Simulation Cases

Case	Target Location $x_T$	Range $R_T$	Bearing Cosines $\Omega_s$	RD Noise Standard Deviation $\sigma_{dl}$
1	(390, 160, 170)	454.5	(0.858, 0.352, 0.374)	0.1
2	(390, 160, 170)	454.5	(0.858, 0.352, 0.374)	1.0
3	(540, 1,360, 110)	1,467	(0.368, 0.927, 0.750)	0.1
4	(540, 1,360, 110)	1,467	(0.368, 0.927, 0.750)	1.0

Source: [24].

**Table 5.2** Biases in the Computed PFs

Method	Run	Source Location Bias			Range	Bearing
		$\hat{x}_T - x_T$	$\hat{y}_T - y_T$	$\hat{z}_T - z_T$	$\hat{R}_T - R_T$	$\hat{\theta}_T - \theta_T$
Spherical Interpolation	1	0.239	0.60	0.026	0.225	0.010
	2	5.16	1.47	0.837	5.42	0.174
	3	1.87	4.60	0.003	4.95	0.014
	4	62.7	165	1.40	178	0.500
Plane Intersection	1	0.195	0.052	-0.035	0.183	0.010
	2	4.24	1.22	0.602	4.19	0.150
	3	0.961	2.18	-0.035	2.37	0.009
	4	26.9	72.5	1.8	76.9	0.2
Spherical Intersection	1	0.341	0.090	0.048	0.341	0.012
	2	0.316	-0.129	-0.057	0.129	0.035
	3	-2.0	-5.76	-0.090	-6.10	0.017
	4	*	*	*	*	*

Source: [24].

\* Results too large to be useful.

**Table 5.3** Standard Deviations of the Simulations

Method	Run	Source Location Error			Range	Bearing
		$\sigma_{\hat{x}_T}$	$\sigma_{\hat{y}_T}$	$\sigma_{\hat{z}_T}$	$\sigma_{\hat{R}_T}$	$\sigma_{\hat{\theta}_T}$
Spherical Interpolation	1	2.07	0.698	0.445	2.27	0.042
	2	20.4	6.91	4.39	22.4	0.41
	3	9.98	26.2	0.600	28.2	0.07
	4	82	219	5	234	0.7
Plane Intersection	1	4.17	1.33	0.898	4.37	0.07
	2	40.3	12.8	8.732	42.2	0.66
	3	15.8	42.5	0.817	45.2	0.11
	4	159	421	7	449	1.0
Spherical Intersection	1	5.12	1.10	0.513	5.24	0.066
	2	31.1	11.1	5.13	32.4	0.64
	3	39.4	105	0.958	112	0.28
	4	*	*	*	*	*

Source: [24].

\*Results too large to be useful.

**Table 5.4** RMS Values Simulation Results

Method	Run	Source Location Error			Range	Bearing
		$\bar{\sigma}_{\hat{x}_T}$	$\bar{\sigma}_{\hat{y}_T}$	$\bar{\sigma}_{\hat{z}_T}$	$\bar{\sigma}_{\hat{R}_T}$	$\bar{\sigma}_{\hat{\theta}_T}$
Spherical Interpolation	1	2.08	0.700	0.446	2.28	0.043
	2	21	7.06	4.47	23	0.45
	3	10.1	26.6	0.600	28.6	0.071
	4	103	274	5.2	294	0.7
Plane Intersection	1	4.18	1.33	0.898	4.37	0.07
	2	40.5	12.9	8.75	42.4	0.68
	3	15.8	42.6	0.818	45.3	0.11
	4	161	427	7	456	1
Spherical Intersection	1	5.12	1.1	0.515	5.25	0.067
	2	31.1	11.1	5.13	32.4	0.64
	3	39.4	105	0.962	112	0.28
	4	*	*	*	*	*

Source: [24].

\*Results too large to be useful.

#### 5.4.2.5 Geometrical Interpretations

The geometrical arguments put forth by Smith and Abel are particularly interesting and point out some physical reasons why one technique performs better than another.

The SI geometry is shown in Figure 5.32 [24]. The RDs in this case are spread over a reasonable range. This will make the technique relatively insensitive to noise-induced errors.

The SX geometry is shown in Figure 5.33 [24]. The target is located at the intersection of spheres with radius  $\hat{R}_T + r_{i1}$  around each sensor. When the target is far away from the sensor array, these spheres have large radii and their intersection is close to the intersection of parallel lines—the intersection could be anywhere on a long circle segment. This leads to inherent inaccuracies as the SNR is decreased.

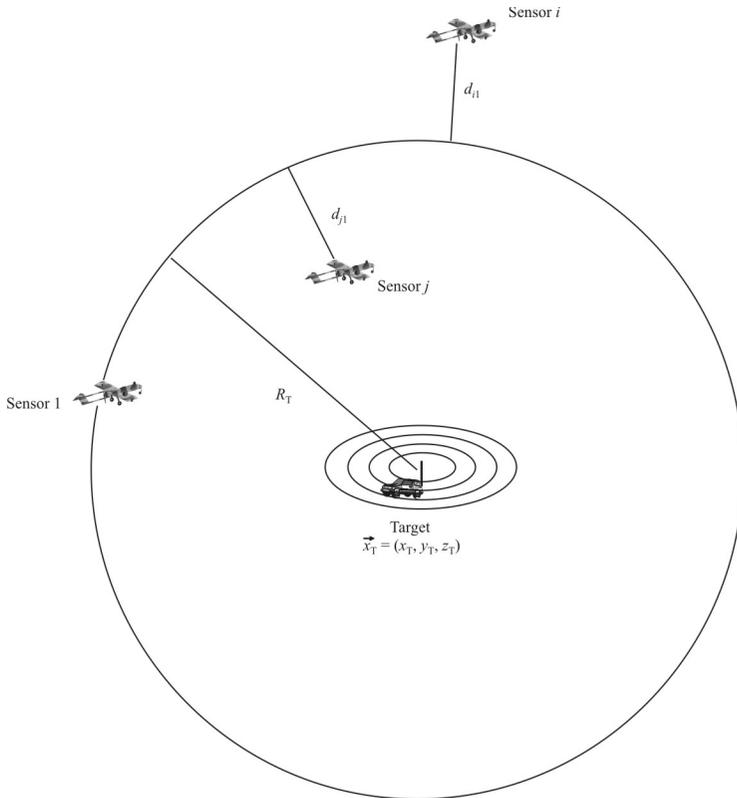
For the PX method, the target lies on the major axis of some conical shape. Shown in Figure 5.34 are two ellipses. Considering three sensors at a time, the three sensors define a plane. The conic shape is on this plane and, therefore, so is its major axis.

With two such ellipses, the major axes intersect at a point and that, of necessity, is where the target lies. With more than two ellipses and including measurement error and noise, these conic axes do not intersect at a point and therefore the LS process is required. Nevertheless, the intersection of the axes indicating the location of the target produces reasonable crossing angles, so higher noise immunity would be expected.

#### 5.4.3 Range Difference Using Feasible Bivectors

A novel approach to geolocation with range differences was developed by Schmidt [28]. The development included use of Grassmann algebra, some details of which are given in Appendix A.

The approach is based on modifying each set of TDOAs from three sensors so that their sum is zero by subtracting their average. Each of these sets defines a plane which may or may not (mostly will not) contain the location of the target. The planes will not normally contain the target due to noise in the process and the intersection of the planes (the intersection of two planes defines a straight line and the intersection of three planes defines a point). The target will, however, be close to these planes. Once these TDOAs are thus modified, a least-squares computation finds the point that is closest to all the planes. That is the estimated location of the target.



**Figure 5.32** The SI configuration of the sensors and target geometry.

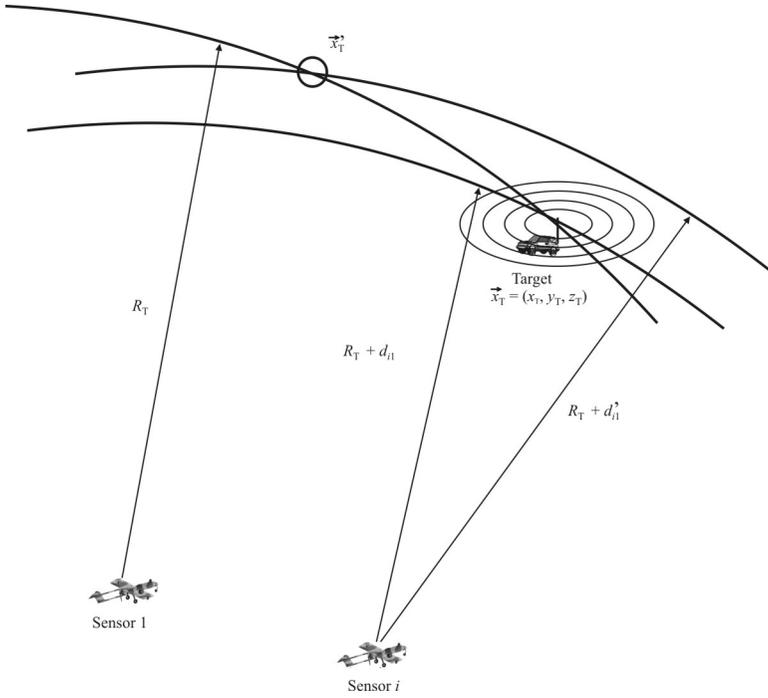


Figure 5.33 SX geometry.

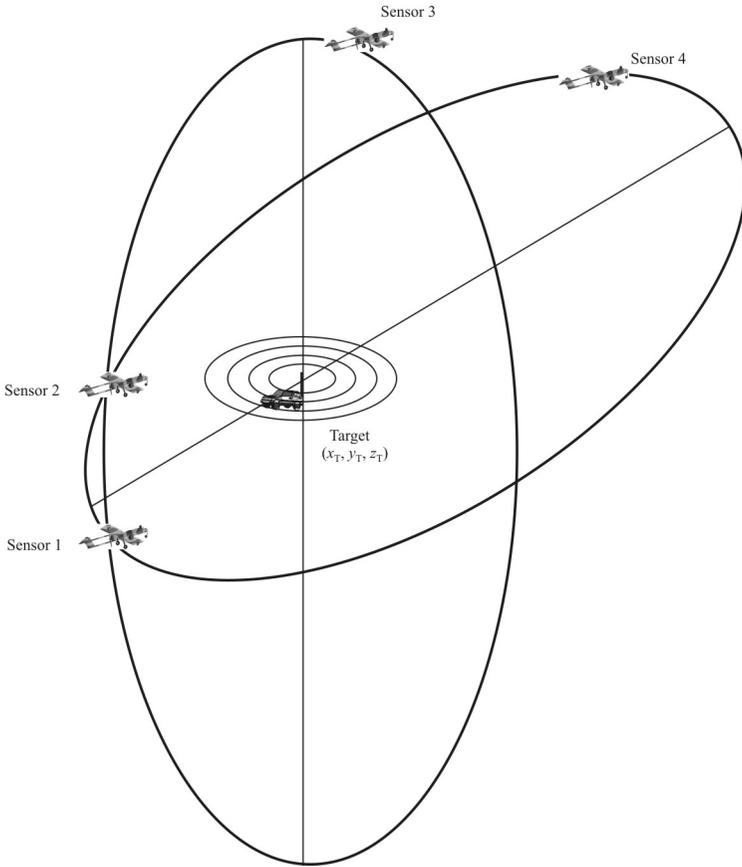


Figure 5.34 PX geometry.

Given that there are  $N$  sensors with the coordinates of the  $i$ th sensor being  $(x_i, y_i, z_i)$ , then

$$d_i = \sqrt{(x_T - x_i)^2 + (y_T - y_i)^2 + (z_T - z_i)^2} \tag{5.151}$$

is the range between the  $i$ th sensor and the target when  $(x_T, y_T, z_T)$  are the coordinates of the target emitter. The range differences are given by

$$\Delta_{ij} = d_j - d_i, \quad 1 \leq i, j \leq N \tag{5.152}$$

and  $\Delta_{ji} = -\Delta_{ij}$ , as well as

$$\Delta_{ii} = 0$$

A differencing matrix  $\mathbf{D}$  is defined such that

$$\bar{\Delta} = \mathbf{D}\bar{d} \tag{5.153}$$

For  $N$  sensors it is denoted  $\mathbf{D}_N$ . Thus, for  $N = 4$ ,

$$\begin{bmatrix} \Delta_{12} \\ \Delta_{13} \\ \Delta_{14} \\ \Delta_{23} \\ \Delta_{24} \\ \Delta_{34} \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d \\ d_4 \end{bmatrix} \tag{5.154}$$

Vector  $\Delta$  is referred to as a *bivector*, since each element has two subscripts, not just one as in a linear vector. As noted in [29],  $\Delta$  cannot be just any bivector; it must lie in the range space of  $\mathbf{D}$ . If  $\Delta$  can be expressed in the form of (5.153), it is said to be *feasible*, otherwise, it is *infeasible*.

Given a real (measured) bivector  $\bar{\Delta}$ , the nearest feasible bivector  $\hat{\Delta}$  is obtained in an LS sense by projecting  $\bar{\Delta}$  onto the range space of  $\mathbf{D}$ . That is,

$$\hat{\Delta} = \mathbf{P}\bar{\Delta} \tag{5.155}$$

The projection vector is

$$\mathbf{P} = \mathbf{D}(\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \quad (5.156)$$

where  $\mathbf{D}^{-}$  is the generalized inverse of  $\mathbf{D}$ .<sup>1</sup> In addition, in this case,

$$\mathbf{D}^T \mathbf{D} = N \left( \mathbf{I} - \frac{1}{N} \bar{\mathbf{1}} \bar{\mathbf{1}}^T \right) \quad (5.157)$$

where  $\bar{\mathbf{1}}$  is a vector of all 1s. The term in the brackets in (5.157) is its own generalized inverse, so

$$\begin{aligned} \mathbf{P} &= \mathbf{D}(\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T = \mathbf{D} \left( N \left( \mathbf{I} - \frac{1}{N} \bar{\mathbf{1}} \bar{\mathbf{1}}^T \right) \right)^{-1} \mathbf{D}^T \\ &= \frac{1}{N} \mathbf{D} \left( \mathbf{I} - \frac{1}{N} \bar{\mathbf{1}} \bar{\mathbf{1}}^T \right) \mathbf{D}^T \\ &= \frac{1}{N} \mathbf{D} \mathbf{D}^T - \frac{1}{N^2} \mathbf{D} \bar{\mathbf{1}} \bar{\mathbf{1}}^T \mathbf{D}^T \end{aligned} \quad (5.158)$$

$$= \frac{1}{N} \mathbf{D} \mathbf{D}^T \quad (5.159)$$

because the second term in (5.158) is zero.

Any *algebra* is a mathematical construct that includes vectors and an operation called the product that is closed under that operation. In Grassmann algebra, the product operation is called the *wedge product* and is normally denoted by  $\wedge$ . “Closed” means that if  $\vec{a}$  and  $\vec{b}$  are vectors in the algebra, then so is  $\vec{a} \wedge \vec{b}$ . Note that the set of bivectors generated by a set of vectors is not in the same space as the vectors and therefore does not form an algebra. Extending the results, however, to include all higher forms does form an algebra. If  $N = 4$ , then the wedge product of  $\vec{x} = [x_1 \ x_2 \ x_3 \ x_4]^T$  and  $\vec{y} = [y_1 \ y_2 \ y_3 \ y_4]^T$  is given by

---

<sup>1</sup> A generalized inverse of matrix  $\mathbf{A}$  is any matrix  $\mathbf{B}$  satisfying  $\mathbf{A} \mathbf{B} \mathbf{A} = \mathbf{A}$ .

$$\vec{x} \wedge \vec{y} = \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} = \begin{bmatrix} x_1 y_2 - x_2 y_1 \\ x_1 y_3 - x_3 y_1 \\ x_1 y_4 - x_4 y_1 \\ x_2 y_3 - x_3 y_2 \\ x_2 y_4 - x_4 y_2 \\ x_3 y_4 - x_4 y_3 \end{bmatrix} = \begin{bmatrix} z_{12} \\ z_{13} \\ z_{14} \\ z_{23} \\ z_{24} \\ z_{34} \end{bmatrix} \quad (5.160)$$

Note that

$$\vec{1} \wedge \vec{d} = \begin{vmatrix} 1 & d_1 \\ 1 & d_2 \\ 1 & d_3 \\ 1 & d_4 \end{vmatrix} = \begin{bmatrix} d_2 - d_1 \\ d_3 - d_1 \\ d_4 - d_1 \\ d_3 - d_2 \\ d_4 - d_2 \\ d_4 - d_3 \end{bmatrix} = \begin{bmatrix} \Delta_{21} \\ \Delta_{31} \\ \Delta_{41} \\ \Delta_{32} \\ \Delta_{42} \\ \Delta_{43} \end{bmatrix} = \vec{\Delta} \quad (5.161)$$

Adding a second product to (5.161) forces the product to zero as

$$\vec{1} \wedge \vec{\Delta} = \vec{1} \wedge \vec{1} \wedge \vec{d} = \vec{0} \quad (5.162)$$

This expression is true because any bivector multiplied by itself is zero; to wit,

$$\vec{1}_3 \wedge \vec{1}_3 = \begin{vmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{vmatrix} = \begin{vmatrix} 1-1 \\ 1-1 \\ 1-1 \end{vmatrix} = \vec{0}_3 \quad (5.163)$$

where  $\vec{1}_N$  is a vector of all 1s of length  $N$ . Expression (5.162) is called a *trivector equation*. Expression (5.162) is true if and only if  $\vec{\Delta}$  is a *feasible bivector*. With measurement errors and noise included, (5.162) will not be identically true and the closest feasible bivector  $\hat{\Delta}$  is sought.

The trivector equation  $\vec{1} \wedge \vec{\Delta} = \vec{s}$ , including measurement errors and noise, can be expressed in terms of vectors and matrices as  $\vec{S}\vec{\Delta} = \vec{s}$ . For  $N = 4$ , for example,

$$\mathbf{S}_4 \bar{\Delta} = \begin{bmatrix} 1 & -1 & 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 1 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} \Delta_{12} \\ \Delta_{13} \\ \Delta_{14} \\ \Delta_{23} \\ \Delta_{24} \\ \Delta_{34} \end{bmatrix} = \begin{bmatrix} s_{123} \\ s_{124} \\ s_{134} \\ s_{234} \end{bmatrix} \quad (5.164)$$

The matrix product  $\mathbf{S}_n \Delta$  equals zero if  $\Delta$  is feasible and equals the trivector of residuals if not.

The matrices  $\mathbf{D}_n$  and  $\mathbf{S}_n$  can be found recursively by

$$\mathbf{D}_{N+1} = \begin{bmatrix} -\bar{\mathbf{I}}_N & \bar{\mathbf{I}}_N \\ \bar{\mathbf{0}} & \mathbf{D}_N \end{bmatrix} \quad \mathbf{S}_{N+1} = \begin{bmatrix} -\mathbf{D}_N & \mathbf{I}_{\binom{N}{2}} \\ \bar{\mathbf{0}} & \mathbf{S}_N \end{bmatrix} \quad (5.165)$$

Schmidt defines TDOA (range difference) averaging as

$$\hat{\Delta}_{ij} = \Delta_{ij} - \frac{1}{n} \sum_{k \neq i, j} (\Delta_{ij} + \Delta_{jk} + \Delta_{ki}) \quad 1 \leq i, j \leq n \quad (5.166)$$

which simply subtracts from each triad measured range difference the average of the range differences involved. The term in brackets is referred to as *circuitual sum* and will equal zero if  $\Delta_{ij}$  is feasible (lies on a plane).

It is also proven in [28] that TDOA (range difference) averaging produces the same result as minimum least-squared estimating. The algorithm for emitter geolocation is given as the following:

1. Given the measured range difference bivector  $\bar{\Delta}$ , calculate the closest feasible range difference bivector  $\hat{\Delta}$  using either (5.166) or (5.155), since they are equivalent.
2. Calculate the point nearest to the  $\binom{N}{3}$  planes of position, one for each of the triads of elements of  $\hat{\Delta}$ .

The second step can be executed using the *location on the conic axis* (LOCA) algorithm developed in [29]. It is the algorithm upon which (5.147) is based and is summarized by the following:

Given that sensor  $i$  is located at  $(x_i, y_i, z_i)$  and there are  $N$  sensors total, then the target lies in planes described by

$$Ax + By + Cz = D \tag{5.167}$$

where

$$\begin{aligned} A &= x_1\Delta_{23} + x_2\Delta_{31} + x_3\Delta_{12} \\ B &= y_1\Delta_{23} + y_2\Delta_{31} + y_3\Delta_{12} \\ C &= z_1\Delta_{23} + z_2\Delta_{31} + z_3\Delta_{12} \\ D &= \frac{1}{2}(\Delta_{12}\Delta_{23}\Delta_{31} + \tilde{d}_1^2\Delta_{23} + \tilde{d}_2^2\Delta_{31} + \tilde{d}_3^2\Delta_{12}) \\ \tilde{d}_i &= \sqrt{x_i^2 + y_i^2 + z_i^2} \end{aligned}$$

For  $M$  sensors there are  $\binom{M}{3}$  such planes. Thus,

$$\begin{bmatrix} A_{123} & B_{123} & C_{123} \\ \vdots & \vdots & \vdots \\ A_{ijk} & B_{ijk} & C_{ijk} \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} D_{123} \\ \vdots \\ D_{ijk} \\ \vdots \end{bmatrix} \tag{5.168}$$

The point closest to the intersection of these planes provides the least-squares estimate of the target location.

Schmidt presents simulation results that are based on the same geometry as that used by Smith and Abel. Those results are given in Tables 5.5 and 5.6, but the SX results have been omitted, since those results are much worse than the rest.

Comparing the results in Table 5.6 with those in Tables 5.2 through 5.4 indicates that the feasible bivector method is the best of those analyzed. The improvements range from a factor of just a few to over 10 times better.

**Table 5.5** Least-Squares and Weighted Least-Squares Simulation Results

Run	$\epsilon_x$	$\sigma_x$	$\epsilon_y$	$\sigma_y$	$\epsilon_z$	$\sigma_z$	RMS
LS Plane Intersection							
1	-0.03	1.6	0.0005	0.63	-0.002	0.32	1.2
2	-7.7	15.6	-2.8	5.8	-1.2	5.5	18.3
3	-6.2	7.4	-16.7	20	-0.06	0.66	17.5
4	-226	104	-610	283	-4.1	5.0	690
Weighted LS Plane Intersection							
1	0.06	1.1	0.01	0.37	0.0002	0.23	1.30
2	-8.3	11.7	-2.7	4.3	-1.6	2.6	15.2
3	-1.2	5.1	-5.4	15.8	-0.05	0.30	14.5
4	-102	27	-276	74.5	-1.6	2.4	156

Source: [28].

**Table 5.6** Feasible Bivector Simulation Results

Run	$\epsilon_x$	$\sigma_x$	$\epsilon_y$	$\sigma_y$	$\epsilon_z$	$\sigma_z$	RMS
1	-0.02	0.53	-0.01	0.19	0.004	0.12	0.678
2	0.15	5.6	0.10	2.0	0.08	1.3	6.79
3	0.55	5.7	1.7	10.0	-0.01	0.20	0.37
4	-9.4	35	-25	95	-0.13	2.2	95.75

Source: [28].

## 5.5 Concluding Remarks

Calculating target geolocation estimates by methods other than triangulation were presented in this chapter. These techniques are based on calculating the TDOA, DD, and range differences between several sensors. These methods invariably lead to nonlinear (quadratic) equations to solve for the PF.

The principal advantage of these techniques over triangulation is the antenna configuration—quadratic methods typically require only a single antenna, while triangulation typically requires an antenna array. The biggest drawback is that for many communication signals there is no feature that can be used to correlate arrival times between sensors. This leads to the requirement to compute the pre-detect cross-correlation function of the received signals, which requires exchanging large amounts of data leading to the requirement for significant data link capacity, if real-time operation is a requirement. Of course, if real-time is not a requirement, then the data can be stored for later processing.

The quadratic methods generally produce more accurate PFs than triangulation. Among the quadratic methods discussed here, the RD method using feasible bivector calculations based on Grassmann algebra is the most accurate.

The quadratic methods are still subject to GDOP-induced errors, however. They are also subject to the errors produced by noise and inaccurate measurements.

## References

- [1] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, 2nd ed., Norwood, MA: Artech House, 2008, Chapter 9.
- [2] Rusu, P., "The Equivalence of TOA and TDOA RF Transmitter Location," Applied Research Laboratories, the University of Texas at Austin, 2002.
- [3] Shin, D.-H., and T.-K. Sung, "Comparison of Error Characteristics Between TOA and TDOA Positioning," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 38, No. 1, January 2002, pp. 307–310.
- [4] Bard, J. D., F. M. Ham, and W. L. Jones, "An Algebraic Solution to the Time Difference of Arrival Equations," *Proceedings of the IEEE SouthEastcon Conference*, Tampa, FL, April 11–14, 1996, pp. 313–319.
- [5] Mellen, G. M. II, M. Pachter, and J. Raquet, "Closed-Form Solution for Determining Emitter Location Using Time Difference of Arrival Measurements," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 39, No. 3, July 2003, pp. 1056–1058.
- [6] Stein, S., "Algorithms for Ambiguity Functions Processing," *IEEE Transactions on Acoustic, Speech, and Signal Processing*, Vol. ASSP-29, No. 3, June 1981, pp. 588–599.
- [7] Quazi, A. H., "An Overview on the Time Delay Estimate in Active and Passive Systems for Target Localization," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-29, No. 3, June 1981, pp. 527–535.
- [8] Knapp, C. H., and G. C. Carter, "The Generalized Correlation Method for Estimation of Time Delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-24, August 1976, pp. 320–327.

- [9] Schultheiss, P. M., "Locating a Passive Source with Array Measurements—A Summary Result," *Proceedings ICASSP 1979*, Washington, D.C., April 24, 1979, pp. 967–970.
- [10] Hahn, W. R., "Optimum Signal Processing for Passive Sonar Range and Bearing Estimation," *Journal Acoustic Society of America*, Vol. 58, July 1975, pp. 201–207.
- [11] Tomlinson, G., and J. Sorokowsky, "Accuracy Prediction Report, Raploc Error Model," IBM, Manassas, VA, June 30, 1977, pp. B1–B7.
- [12] Gustafsson, F., and F. Gunnarsson, "Positioning Using Time-Difference of Arrival Measurements," *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 03*, Vol. 6, Hong Kong, April 6–10, 2003, pp. VI-553–556.
- [13] Nordlund, P. J., F. Gunnarsson, and F. Gustafsson, "Particle Filters for Positioning in Wireless Networks," *EUSIPCO*, Toulouse, France, September 2002.
- [14] Arulampalam, M. S., et al., "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, February 2002, pp. 174–188.
- [15] Gustafsson, F., et al., "Particle Filters for Positioning, Navigation, and Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, February 2002, pp. 425–437.
- [16] Bard, J. D., and F. M. Ham, "Time Difference of Arrival Dilution of Precision and Applications," *IEEE Transactions on Signal Processing*, Vol. 47, No. 2, February 1999, pp. 521–525.
- [17] Hofman-Wellenhopf, B., H. Lichtenegger, and J. Collins, *GPS Theory and Practice*, 3rd ed., New York: Springer-Verlag, 1994, pp. 235–237, 249–255.
- [18] Koorapaty, H., H. Grubeck, and M. Cedervall, "Effect of Biased Measurement Errors on Accuracy of Position Location Methods," *IEEE GLOBECOM 98*, Vol. 3, November 8–12, 1998, pp. 1497–1502.
- [19] Chan, Y. T., and K. C. Ho, "TDOA-SDOA Estimation with Moving Source and Receivers," *IEEE Transactions ICASSP 2003*, 2003, pp. V-153–V-156.
- [20] Levanon, N., "Interferometry against Differential Doppler: Performance Comparison of Two Emitter Location Airborne Systems," *IEE Proceedings*, Vol. 136, Pt. F., No. 2, April 1989, pp. 70–74.
- [21] Otnes, R. K., "Frequency Difference of Arrival Accuracy," *IEEE Transactions on Acoustic, Speech, and Signal Processing*, Vol. 37, No. 2, February 1989, pp. 306–308.
- [22] Piersol, A. G., "Time Delay Estimation Using Phase Data," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-29, No. 3, June 1981, pp. 471–477.
- [23] Ullman, R. J., and E. Geraniotis, "Motion Detection Using TDOA and FDOA Measurements," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 37, No. 2, April 2001, pp. 759–764.
- [24] Rusu, P. I., "A TOA-FOA Solution for the Position and Velocity of RF Emitters," Applied Research Laboratories Report #SGL-GR-99-15, The University of Texas at Austin, 1999.
- [25] Ho, K. C., and W. Xu, "An Accurate Algebraic Solution for Moving Source Location Using TDOA and FDOA Measurements," *IEEE Transactions on Signal Processing*, Vol. 52, No. 9, September 2004, pp. 2453–2465.
- [26] Smith, J. O., and J. S. Abel, "Closed-Form Least-Squares Source Location Estimation from Range-Difference Measurements," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-35, No. 2, December 1987, pp. 1661–1669.
- [27] Schau, H. C., and A. Z. Robinson, "Passive Source Localization Employing Intersecting Spherical Surfaces from Time-of-Arrival Differences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-35, No. 8, August 1987, pp. 1223–1225.
- [28] Schmidt, R. O., "Least-Squares Range Difference Location," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 32, No. 1, January 1996, pp. 234–242.
- [29] Schmidt, R. O., "A New Approach to Geometry of Range Difference Location," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-8, November 1972, pp. 821–835.

# Chapter 6

## Time Delay Estimation

### 6.1 Introduction

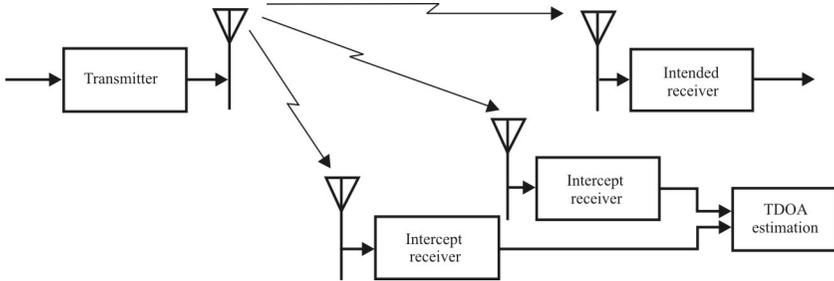
In Chapter 3 we discussed PF estimation using quadratic algorithms that include time delay estimates. Again, we did not mention in that chapter how these time delay estimates are obtained. In this chapter some of the techniques for estimating time delay are presented. We denote time differences by  $D$  in this chapter.

In this chapter we examine cross-correlation methods for determining the time delay (the TDOA) between a stochastic signal arriving at two sensor elements. Classic cross-correlation is covered in the first section while generalized cross correlation is considered in the second [1–3]. Using the phase of the cross correlation function to determine the TDOA is analyzed in the third.

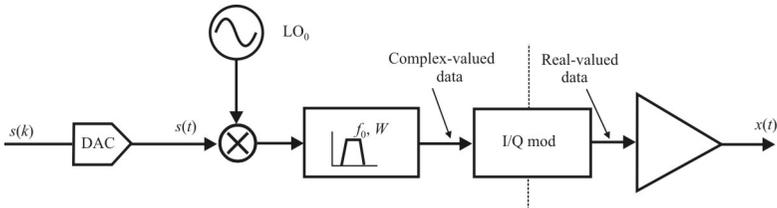
### 6.2 System Overview

Consider a military communication system with a transmitter and an intended receiver. The transmitter will most likely employ measures to avoid detection and/or location by an adversary, such as stealth waveforms (LPI), power control, directional antennas, and so forth. The EW system using two intercept receivers is eavesdropping on this communication with no knowledge of waveforms or position, as illustrated in Figure 6.1.

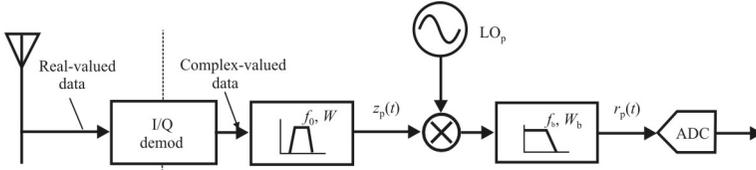
As shown in Figure 6.2, the transmitted signal,  $x(t)$ , is assumed to have unknown characteristics, and accordingly the message  $s(t)$  is modeled as a complex-valued zero-mean wss process, characterized by its autocorrelation function  $\rho_{ss}(\tau)$ . Furthermore, it is assumed that  $s(t)$  is band-limited into the frequency range  $(-W, W)$  Hz and that its PSD is continuous in frequency  $f$ , that is, the transmitted signal is assumed to be band-limited, but broadband with center frequency  $f_0$  and phase  $\varphi_0 = 0$ .



**Figure 6.1** A communication system with a transmitter and an intended receiver is shown at the top. The two intercept receivers noncooperatively intercept the signal sent by the transmitter. Using the outputs of the intercept receivers, an estimate of the TDOA is obtained.



**Figure 6.2** The transmitted signal  $x(t)$  is a bandpass signal with carrier frequency  $f_0$  (determined by the oscillator  $LO_0$ ) and bandwidth  $W$ .



**Figure 6.3** The nonideal receiver systems introduce both a frequency error  $\epsilon$  and a phase error  $\kappa$  between the mixing oscillators,  $LO_1$  and  $LO_2$ , in the two intercept receivers depicted in Figure 6.2.

The signal is transmitted through nondispersive channels and is received by two sensors whose outputs [say,  $z_1(t)$  and  $z_2(t)$ , respectively] after quadrature mixing contain two noisy and differently delayed versions of the real-valued bandpass signal  $x(t)$ . See Figure 6.3.

Now,  $z_1(t)$  and  $z_2(t)$  are complex-valued bandpass RF signals. We model the receivers as simple superheterodyne receivers (filter, mixer, filter) where the filter bandwidths are  $W$  Hz and the LO frequencies of the two receivers are  $f_1$  and  $f_2$  with phases  $\phi_1$  and  $\phi_2$ , respectively. The outputs after mixing are

$$\begin{aligned} r_1(t) &= z_1(t)e^{-j2\pi f_1 t - j\phi_1} + n_1(t) \\ &= s(t)e^{j2\pi(f_0 - f_1)t - j\phi_1} + n_1(t) \end{aligned} \tag{6.1}$$

and

$$\begin{aligned} r_2(t) &= z_2(t)e^{-j2\pi f_2 t - j\phi_2} + n_2(t) \\ &= s(t - D)e^{j2\pi(f_0 - f_2)t - j2\pi f_0 D - j\phi_2} + n_2(t) \end{aligned} \tag{6.2}$$

which is depicted in Figure 6.3.

The complex-valued noise terms in (6.1) and (6.2) are assumed zero-mean with flat spectral density, that is,

$$P_{n_p}(f) = \begin{cases} \sigma_p^2, & |f| \leq B \\ 0, & |f| > B \end{cases}, \quad p = 1, 2 \tag{6.3}$$

The corresponding autocorrelation functions are denoted by  $\rho_{n_1}(\tau)$  and  $\rho_{n_2}(\tau)$ , respectively.

In this chapter we will assume that the mixing and filtering that occurs while generating the baseband signal does not significantly affect the baseband signal and the statistical characteristics of the signals received at the two receivers are only perturbed by the corresponding noise sources and the time delay.

## 6.3 Cross-Correlation

We model the signals at the two sensors to be [4]

$$\begin{aligned} r_1(t) &= s(t) + n_1(t) \\ r_2(t) &= s(t - D) + n_2(t) \quad 0 \leq t \leq T \end{aligned} \quad (6.4)$$

where the signal  $s(t)$  and noises  $n_1(t)$ ,  $n_2(t)$  are real baseband signals.  $D$  is the unknown delay, and  $T$  is the observation time. We assume that the signal  $s(t)$  and noises  $n_1(t)$  and  $n_2(t)$  are stationary, band-limited, zero mean stochastic signals mutually uncorrelated. We further assume that the correlation durations of the signals  $s(t)$ ,  $n_1(t)$ , and  $n_2(t)$ :  $|D| + \tau_s, \tau_{n_1}, \tau_{n_2}$ , respectively, are very small compared to the observation time  $T$ . In addition,  $s(t)$  is assumed to be ergodic.

The delay  $D$  can be estimated by  $\tau = \hat{D}$  for which the *cross-correlation function* (CCF), which we denote by  $\rho_{r_1 r_2}(\tau)$ , is maximized; that is

$$\rho_{r_1 r_2}(\hat{D}) = \max_{\tau} \rho_{r_1 r_2}(\tau) = \max_{\tau} \int_0^T r_1(t) r_2(t + \tau) dt \quad (6.5)$$

We will show that the estimate  $\hat{D}$  of  $D$  is unbiased. In addition, we will establish an expression for its MSE as a function of the observation time  $T$  and the autospectra of the signal  $s(t)$  and the noise  $n_1(t)$  and  $n_2(t)$ . In the derivations it is assumed that TDOA estimation error is very small compared to the correlation duration  $\tau_s$  of  $s(t)$ , that is,  $|\hat{D} - D| \ll \tau_s$ .

### 6.3.1 Error Analysis of the Cross-Correlation Method

The CCF  $\rho_{r_1 r_2}(\tau)$  consists of two terms: a signal term  $\rho_{ss}(\tau)$  and a noise term  $\rho_{nn}(\tau)$ , that is,

$$\rho_{r_1 r_2}(\tau) = \rho_{ss}(\tau) + \rho_{nn}(\tau) \quad (6.6)$$

where the signal term  $\rho_{ss}(\tau)$  is defined by

$$\rho_{ss}(\tau) \triangleq \int_0^T s(t) s(t + \tau - D) dt \quad (6.7)$$

The noise term  $\rho_{nn}(t)$  is defined by

$$\rho_{nn}(\tau) \triangleq \int_0^T s(t)n_2(t+\tau)dt + \int_0^T n_1(t)s(t+\tau-D)dt + \int_0^T n_1(t)n_2(t+\tau)dt \quad (6.8)$$

Expanding the signal term  $\rho_{ss}(\tau)$  in the neighborhood of  $\tau = D$  into a Taylor series up to order two, we get

$$\rho_{ss}(\tau) \approx \rho_{ss}(D) + \frac{1}{2}\rho_{ss}''(D)(\tau-D)^2 \quad (6.9)$$

[Note that  $d\rho_{ss}(\tau)/d\tau|_{\tau=D} \approx 0$ , because the correlation times of the  $s(t)$ ,  $n_1(t)$ , and  $n_2(t)$  are very short compared to the observation time  $T$ .] Let  $\hat{D}$  be the estimate of  $D$  obtained by maximizing  $\rho(\tau)$  in (6.6). Differentiating (6.6), we get

$$\left. \frac{d\rho_{\eta_1\eta_2}(\tau)}{d\tau} \right|_{\tau=\hat{D}} = \left. \frac{d\rho_{ss}}{d\tau} \right|_{\tau=\hat{D}} + \left. \frac{d\rho_{nn}}{d\tau} \right|_{\tau=\hat{D}} = 0 \quad (6.10)$$

Using (6.9) with (6.10), we get

$$\rho_{ss}''(D)(\hat{D}-D) + \rho'_{nn}(\hat{D}) = 0$$

Therefore,

$$\hat{D}-D = -\frac{1}{\rho_{ss}''(D)}\rho'_{nn}(\hat{D}) \quad (6.11)$$

where

$$\rho_{ss}''(D) = \left[ \frac{d^2}{d\tau^2} \int_0^T s(t)s(t+\tau-D)dt \right]_{\tau=D} \quad (6.12)$$

and

$$\rho'_{nn}(\hat{D}) = \left[ \int_0^T s(t)n_2'(t+\tau)dt + \int_0^T n_1(t)s'(t+\tau-D)dt + \int_0^T n_1(t)n_2'(t+\tau)dt \right]_{\tau=\hat{D}} \quad (6.13)$$

Our goal is to compute  $\mathcal{E}\{\hat{D}-D\}$  and  $\mathcal{E}\{(\hat{D}-D)^2\}$  over the ensemble of the stochastic processes  $s(t)$ ,  $n_1(t)$ , and  $n_2(t)$ .

Note that  $\hat{D}$  is an r.v. depending upon  $s(t)$ ,  $n_1(t)$ , and  $n_2(t)$ .  $\hat{D}$  is an unbiased estimate of  $D$ . This is seen using the following property of conditional expected values.

**Property 6.1** It is widely known that given two uncorrelated (and therefore unrelated) r.v.s  $u$  and  $v$ , the expected value of the random value  $\mathcal{E}\{u|v\} = \mathcal{E}\{u\}$ . That is, the occurrence of some value of  $v$  has no impact on any occurrence of a value of  $u$ .

Thus, using (6.11), we get

$$\mathcal{E}\{\hat{D}-D\} = \mathcal{E}\left\{-\frac{1}{\rho''_{ss}(D)} \mathcal{E}\{\rho'_{nn} | s, D\}\right\}$$

and because  $s(t)$ ,  $n_1(t)$ , and  $n_2(t)$  are uncorrelated then  $\mathcal{E}\{\rho'_{nn} | s, \hat{D}\} = 0$ , and we get  $\mathcal{E}\{\hat{D}-D\} = 0$ . Therefore this estimator is unbiased.

Note that for  $T \gg \tau_s$  and since  $s(t)$  is ergodic, the fluctuation  $\delta\rho_{ss}(\tau-D)$  defined by

$$\delta\rho_{ss}(\tau-D) \triangleq \frac{1}{T}[\rho_{ss}(\tau) - T\rho_{ss}(\tau-D)]$$

is negligible compared to the autocorrelation function  $\rho_{ss}(\tau-D)$  of the signal  $s(t)$ , that is,

$$\frac{|\delta\rho_{ss}(\tau-D)|}{|\rho_{ss}(\tau-D)|} \ll 1$$

By neglecting the fluctuation  $\delta\rho_{ss}(\tau-D)$ , we can approximate (6.12) as

$$\rho''_{ss}(\tau)|_{\tau=D} \approx T\rho_{ss}(\tau-D)|_{\tau=D} = T\rho_{ss}(0)$$

Assuming  $\rho_{ss}(0)$  exists and is nonzero, (6.11) yields

$$\hat{D} - D \approx -\frac{1}{T\rho_{ss}(0)}\rho'_{nn}(\hat{D})$$

Therefore, we conclude that

$$\mathcal{E}\{(\hat{D} - D)^2\} \approx \frac{1}{T^2[\rho_{ss}(0)]^2} \mathcal{E}\{\rho_{nn}'^2(\hat{D})\} \quad (6.14)$$

**Property 6.2** [5] The variance of an integral of a stochastic process is given by

$$\text{var} \left[ \int_0^T u(t) dt \right] = T \int_0^{2T} \left( 1 - \frac{|v|}{T} \right) \rho_{uu}(v) dv \quad (6.15)$$

where  $u(t)$  is a zero-mean stationary process with autocorrelation function  $\rho_{uu}(\tau)$ .

Assuming that the correlation duration  $\tau_u$  of  $u(t)$  is very small compared to the observation time,  $T$ , so that  $|v|/T \sim 0$ , (6.15) can be approximated by

$$\int_0^{2T} \left( 1 - \frac{|v|}{T} \right) \rho_{uu}(v) dv \approx \int_0^{2T} \rho_{uu}(v) dv \approx \int_{-\infty}^{\infty} \rho_{uu}(v) dv \quad (6.16)$$

Using (6.13), (6.16), and Property 6.1, we get

$$\begin{aligned} \mathcal{E}\{\rho_n'^2\} &= \mathcal{E}\{\mathcal{E}\{\rho_n'^2 | \hat{D}\}\} \\ &= T \left[ \int_{-\infty}^{\infty} \rho_{ss}(v) \rho_{n_2' n_2'}(v) dv + \int_{-\infty}^{\infty} \rho_{s's'}(v) \rho_{n_1 n_1}(v) dv + \int_{-\infty}^{\infty} \rho_{n_1 n_1}(v) \rho_{n_2' n_2'}(v) dv \right] \end{aligned} \quad (6.17)$$

where, because of uncorrelated signal and noise, the integrals of the cross products in (6.13) are zero.

**Property 6.3** [6] If  $u(t)$  is zero mean stationary process, the autocorrelation of its first derivative is given by

$$\rho_{u'u'}(\tau) = -\frac{d^2 \rho_{uu}(\tau)}{d\tau^2} \quad (6.18)$$

Using (6.14), (6.17), and (6.18), and integrating the first and third terms in (6.17) by parts, we get

$$\mathcal{E}\{(\hat{D}-D)^2\} = \frac{1}{T[\rho_{ss}''(0)]^2} \left[ \begin{aligned} & -\int_{-\infty}^{\infty} \rho_{n_1 n_1}(v) \rho_{ss}''(v) dv - \int_{-\infty}^{\infty} \rho_{n_2 n_2}(v) \rho_{ss}''(v) dv \\ & + \int_{-\infty}^{\infty} \rho'_{n_1 n_1}(v) \rho'_{n_2 n_2}(v) dv \end{aligned} \right] \quad (6.19)$$

The equivalent of (6.19) in the frequency domain is

$$\mathcal{E}\{(\hat{D}-D)^2\} = \frac{1}{T} \frac{1}{\left[ \int_{-\infty}^{\infty} (2\pi f)^2 G_{ss}(f) df \right]^2} \left[ \begin{aligned} & \int_{-\infty}^{\infty} (2\pi f)^2 G_{n_1 n_1}(f) G_{ss}(f) df \\ & + \int_{-\infty}^{\infty} (2\pi f)^2 G_{n_2 n_2}(f) G_{ss}(f) df \\ & \int_{-\infty}^{\infty} (2\pi f)^2 G_{n_1 n_1}(f) G_{n_2 n_2}(f) df \end{aligned} \right] \quad (6.20)$$

where we used the inverse Weiner-Kinchen Fourier relationship

$$\rho_{uu}(\tau) = \int_{-\infty}^{\infty} e^{j2\pi f \tau} G_{uu}(f) df$$

with  $\rho_{uu}(\tau)$  and  $G_{uu}(f)$  the autocorrelation function and the PSD of  $u(t)$ , respectively. Equations (6.19) and (6.20) are general expressions for the MSE of TDOA estimation by the cross-correlation method.

### 6.3.2 Flat Noise Spectra: Arbitrary Signal Spectrum

To express the MSE (6.20), as a function of SNRs and signal and noise bandwidths, first we will assume that the noise spectra are flat across a defined spectra. That is we assume that the spectra of the noise  $n_1(t)$  and  $n_2(t)$  are flat with double sided bandwidths  $2W_{n_1}$  and  $2W_{n_2}$ , and PSDs  $\bar{G}_{n_1 n_1}$  and  $\bar{G}_{n_2 n_2}$ , respectively.

For now we assume that the signal spectrum is arbitrary but with the bandwidth of  $2W_s$ . The “spillover” of the autospectra of the signal  $s(t)$ , and noise  $n_1(t)$  and  $n_2(t)$  outside their bandwidths is assumed to be negligible, and

$$W_{n_2} = W_{n_1} = W_n > W_s$$

The SNRs are given by

$$\gamma_1 = \frac{\rho_{ss}(0)}{2W_{n_1} \overline{G_{n_1 n_1}}} \quad (6.21)$$

$$\gamma_2 = \frac{\rho_{ss}(0)}{2W_{n_2} \overline{G_{n_2 n_2}}} \quad (6.22)$$

and effective bandwidths  $W$  (in rad/s) by

$$W_s^2 = \frac{\int_{-\infty}^{\infty} (2\pi f)^2 G_{ss}(f) df}{\int_{-\infty}^{\infty} G_{ss}(f) df} \quad (6.23)$$

$$W_{n_p}^2 = \frac{\int_{-\infty}^{\infty} (2\pi f)^2 G_{n_p n_p}(f) df}{\int_{-\infty}^{\infty} G_{ss}(f) df}, \quad p = 1, 2 \quad (6.24)$$

Substituting (6.22) and (6.23) into (6.20), and because the noise spectra are flat, we get

$$\mathcal{E}\{(\hat{D} - D)^2\} = \frac{1}{T} \frac{1}{W_s^2} \frac{1}{W_n} \frac{1}{2} \left[ \frac{1}{\gamma_1} + \frac{1}{\gamma_2} + \frac{W_n^2}{W_s^2} \frac{1}{\gamma_1} \frac{1}{\gamma_2} \right] \quad (6.25)$$

The factor  $W_n^2 / W_s^2$  is of significance for cases of narrowband signals embedded in broadband noise ( $W_n \gg W_s$ ) at low SNRs.

Now we establish conditions for which the validity of neglecting the fourth term in the Taylor expansion of the signal term applies.

### 6.3.3 Flat Noise Spectra: Flat Signal Spectrum

Now we assume that the spectra of the signal  $s(t)$ , and noise  $n_1(t)$ ,  $n_2(t)$  are flat with the same bandwidth, that is,

$$W_{n_1} = W_{n_2} = W_s$$

The bandwidth assumption is reasonable since in most EW applications the target environment is, in general, fairly well known or estimated. The assumption of flat

spectrums applies fairly well to the noise, but can be unreasonable for the signal spectrum, depending on the type of modulation employed by the targets. Ironically, the lower the SNR the more applicable this assumption becomes.

In the Taylor expansion of the signal term, (6.9), the third term vanishes because  $R_{ss}(\tau)$  is symmetric with respect to  $\tau = 0$ , and the fourth term (and therefore higher as well) in the Taylor expansion can be neglected compared to the second term provided that

$$\left| \frac{\text{Fourth term}}{\text{Second term}} \right| = \left| \frac{\frac{1}{24} \rho_{ss}^{(4)}(D)(\hat{D} - D)^4}{\frac{1}{2} \rho_{ss}^{(2)}(D)(\hat{D} - D)^2} \right| \ll 1 \quad (6.26)$$

Because the correlation times of  $s(t)$ ,  $n_1(t)$ , and  $n_2(t)$  are short compared to  $T$ , (6.26) can be expressed as

$$(\hat{D} - D)^2 \ll \frac{5}{\pi^2} \frac{1}{W_s^2} \quad (6.27)$$

Moreover, because the spectra are flat from (6.25) we get

$$\mathcal{E}\{(\hat{D} - D)^2\} = \frac{3}{4\pi^2} \frac{1}{T} \frac{1}{W_s^3} \frac{1}{\gamma} \quad (6.28)$$

where a combined SNR,  $\gamma$ , is defined by

$$\frac{1}{\gamma} = \frac{1}{2} \left[ \frac{1}{\gamma_1} + \frac{1}{\gamma_2} + \frac{W_n^2}{W_s^2} \frac{1}{\gamma_1} \frac{1}{\gamma_2} \right] \quad (6.29)$$

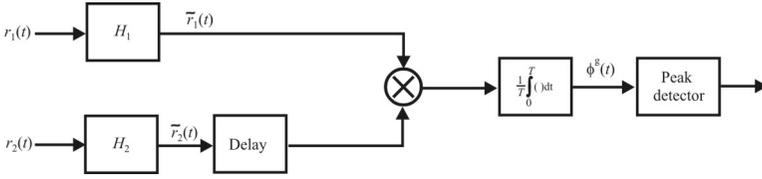
Combining (6.27) and (6.28), we get

$$W_s T \gamma \gg 0.15 \quad (6.30)$$

This is the required condition for the validity of the approximate results.

## 6.4 Generalized Cross-Correlation

The *generalized cross-correlation* (GCC) method of determining the TDOA between two signals [1–3] consists of prefiltering the two signals,  $r_1(t)$ ,  $r_2(t)$  by



**Figure 6.4** GCC flow diagram.

two linear filters  $h_1(t)$ , and  $h_2(t)$ , respectively, resulting in two filtered signals  $\tilde{r}_1(t)$  and  $\tilde{r}_2(t)$ :

$$\begin{aligned}\tilde{r}_1(t) &= h_1(t) * r_1(t) = \tilde{s}_1(t) + \tilde{n}_1(t) \\ \tilde{r}_2(t) &= h_2(t) * r_2(t) = \tilde{s}_2(t) + \tilde{n}_2(t)\end{aligned}\quad (6.31)$$

where

$$\begin{aligned}\tilde{s}_1(t) &= h_1(t) * s(t) \\ \tilde{s}_2(t - D) &= h_2(t) * s(t - D) \\ \tilde{n}_1(t) &= h_1(t) * n_1(t) \\ \tilde{n}_2(t) &= h_2(t) * n_2(t)\end{aligned}\quad (6.32)$$

and  $*$  denotes convolution (see Figure 6.4).<sup>1</sup> We will establish the optimum filters to minimize the MSE in this section.

As before, the delay  $D$  is estimated as  $\tau = \hat{D}$  for which the GCC  $\rho_{\tilde{r}_1\tilde{r}_2}^g(\tau)$  is maximized, i.e.

$$\rho_{\tilde{r}_1\tilde{r}_2}^g(\hat{D}) = \max_{\tau} \rho_{\tilde{r}_1\tilde{r}_2}^g(\tau) = \max_{\tau} \int_0^T \tilde{r}_1(t) \tilde{r}_2(t + \tau) dt \quad (6.33)$$

In the frequency domain, the GCC method is equivalent to multiplying the cross spectra of  $r_1(t)$  and  $r_2(t)$  by a weight function  $\Psi(f)$  given by

$$\Psi(f) = H_1(f) H_2^*(f) \quad (6.34)$$

where  $H_1(f)$  and  $H_2(f)$  are the Fourier transforms of  $h_1(t)$  and  $h_2(t)$ , respectively.

As in the discussion in Section 6.2, we assume that

<sup>1</sup> We use  $*$  to denote both convolution and conjugate transpose. The one that is intended should always be clear from the context in which  $*$  is used.

$$\begin{aligned}\rho_{\tilde{s}_1\tilde{s}_2}(\tau) &\approx T\rho_{\tilde{s}_1\tilde{s}_2}(D-\tau) \\ &= T\int_{-\infty}^{\infty}\Psi(f)G_{ss}(f)e^{j2\pi f(D-\tau)}df\end{aligned}$$

then

$$\rho'_{\tilde{s}_1\tilde{s}_2}(D) \approx T\int_{-\infty}^{\infty}j2\pi f\Psi(f)G_{ss}(f)df = 0$$

for  $\Psi(f)$  a real symmetric function.

As in Section 6.2, we can show that the estimate  $\hat{D}$  is unbiased and the MSE of the TDOA estimate is given by

$$\begin{aligned}\mathcal{E}\{(\hat{D}-D)^2\} &= \frac{1}{T} \frac{1}{\left[\int_{-\infty}^{\infty}(2\pi f)^2 G_{\tilde{s}_1\tilde{s}_2}(f)\right]^2} \\ &\times \int_{-\infty}^{\infty} \begin{bmatrix} (2\pi f)^2 [G_{\tilde{n}_1\tilde{n}_2}(f)G_{\tilde{s}_2\tilde{s}_2}(f) \\ + G_{\tilde{n}_1\tilde{n}_2}(f)G_{\tilde{s}_1\tilde{s}_1}(f) \\ + G_{\tilde{n}_1\tilde{n}_1}(f)G_{\tilde{n}_2\tilde{n}_2}(f)] \end{bmatrix} df\end{aligned}\quad (6.35)$$

Applying (6.32) to (6.35), we get

$$\begin{aligned}\mathcal{E}\{(\hat{D}-D)^2\} &= \frac{1}{T} \frac{1}{\left[\int_{-\infty}^{\infty}(2\pi f)^2 \Psi(f)G_{s_1s_2}(f)\right]^2} \\ &\times \int_{-\infty}^{\infty} \begin{bmatrix} (2\pi f)^2 \Psi^2(f) [G_{n_1n_1}(f)G_{ss}(f) \\ + G_{n_2n_2}(f)G_{ss}(f) \\ + G_{n_1n_1}(f)G_{n_2n_2}(f)] \end{bmatrix} df\end{aligned}\quad (6.36)$$

Hence, the MSE of TDOA estimation by the GCC method depends on the spectra of the signal and the noise, and on the weight function.

We can rewrite Equation (6.36) in a shorter notation as

$$\mathcal{E}\{(\hat{D} - D)^2\} = \frac{1}{T} \frac{\int_{-\infty}^{\infty} \Psi^2(f)A(f)df}{\left[ \int_{-\infty}^{\infty} \Psi(f)B(f)df \right]^2} \quad (6.37)$$

where

$$A(f) = (2\pi f)^2 [G_{n_1 n_1}(f)G_{ss}(f) + G_{n_2 n_2}(f)G_{ss}(f) + G_{n_1 n_2}(f)G_{n_2 n_1}(f)] \quad (6.38)$$

$$B(f) = (2\pi f)^2 G_{ss}(f) \quad (6.39)$$

The optimal weight function  $\Psi(f)$ , for which the MSE given by (6.37) is minimal is obtained by applying the Schwartz inequality by extending results from [7].

The Schwartz inequality [8] that pertains to two real functions  $f$  and  $g$  states

$$\left[ \int_{-\infty}^{\infty} f(v)g(v)dv \right]^2 \leq \int_{-\infty}^{\infty} f^2(v)dv \int_{-\infty}^{\infty} g^2(v)dv \quad (6.40)$$

Equality in (6.40) holds for  $f = Kg$ , where  $K$  is an arbitrary real constant.

By applying (6.40) to the denominator of (6.37), we get

$$\left[ \int_{-\infty}^{\infty} \Psi(f)\sqrt{A(f)} \frac{B(f)}{\sqrt{A(f)}} df \right]^2 \leq \int_{-\infty}^{\infty} [\Psi(f)\sqrt{A(f)}]^2 df \int_{-\infty}^{\infty} \left[ \frac{B(f)}{\sqrt{A(f)}} \right]^2 df \quad (6.41)$$

We assume that  $A(f)$  is a positive valued function.<sup>2</sup> From (6.41)

$$\frac{1}{\left[ \int_{-\infty}^{\infty} \Psi(f)B(f)df \right]^2} \geq \frac{1}{\int_{-\infty}^{\infty} A(f)\Psi^2(f)df \int_{-\infty}^{\infty} \frac{B^2(f)}{A(f)} df} \quad (6.42)$$

Multiplying (6.41) by the numerator of (6.37), we get

---

<sup>2</sup> From (6.38) we see that  $A(f)$  is the sum of products of PSDs. The PSD of a signal is certainly never negative since it is an indication of the power per Hz in a signal which cannot be negative. In some cases, however, a PSD can be zero and  $A(f)$  could be zero at some frequencies. We assume here that not all of the product terms are simultaneously zero, without much loss of generality.

$$\mathcal{E}\{(\hat{D} - D)^2\} \geq \frac{1}{T} \frac{1}{\int_{-\infty}^{\infty} \frac{B^2(f)}{A(f)} df} \quad (6.43)$$

Equality holds provided that

$$\Psi(f)\sqrt{A(f)} = K \frac{B(f)}{\sqrt{A(f)}}$$

or

$$\Psi(f) = K \frac{B(f)}{A(f)} \quad (6.44)$$

where  $K$  is an arbitrary constant.

## 6.5 Estimating the Time Delay with the Generalized Correlation Method

The weighting function  $\Psi(f)$  discussed in Section 6.4, given by (6.34), primarily serves to sharpen the peaks in the CCF thereby facilitating more accurate estimation of the time delay between the two signals  $r_1(t)$  and  $r_2(t)$ . Under ideal conditions where

$$\hat{G}_{\hat{r}_1\hat{r}_2}(f) \approx G_{r_1r_2}(f) \quad \forall f \quad (6.45)$$

$\Psi(f)$  should be chosen to ensure a sharp peak in  $\rho_{\hat{r}_1\hat{r}_2}(\tau)$  rather than a broad one in order to ensure good time-delay resolution. However, sharp peaks are more sensitive to errors introduced by finite observation time, particularly in cases of low SNR. Thus, as with other spectral estimation problems, the choice of  $\Psi(f)$  is a compromise between good resolution and stability.

For the model described by (6.31) and (6.32), the cross-correlation of  $r_1(t)$  and  $r_2(t)$  is

$$\rho_{\hat{r}_1\hat{r}_2}(\tau) = \alpha\rho_{ss}(\tau - D) + \rho_{n_1n_2}(\tau) \quad (6.46)$$

The Fourier transform of (6.46) gives the CSD

**Table 6.1** Candidate Processes

Process	Weight $\Psi(f) = H_1(f)H_2^*(f)$
Cross Correlation	1
Roth Impulse Response	$1 / G_{r_1 r_2}(f)$
SCOT	$1 / \sqrt{G_{r_1 r_1}(f)G_{r_2 r_2}(f)}$
PHAT	$1 /  G_{r_1 r_2}(f) $
Eckart	$G_{ss}(f) / [G_{n_1 n_1}(f)G_{n_2 n_2}(f)]$
Maximum Likelihood	$\frac{ \gamma_{12}(f) ^2}{ G_{r_1 r_2}(f) [1 -  \gamma_{12}(f) ^2]}$

Source: [9].

$$G_{r_1 r_2}(f) = \alpha G_{ss}(f)e^{-j2\pi fD} + G_{n_1 n_2}(f) \quad (6.47)$$

If  $n_1(t)$  and  $n_2(t)$  are uncorrelated, then  $G_{n_1 n_2}(f) = 0$  and the CSD between  $r_1(t)$  and  $r_2(t)$  is a scaled signal power spectrum times a complex exponential. Since multiplication in one domain is convolution in the transformed domain, it follows for  $G_{n_1 n_2}(f) = 0$  that

$$\rho_{r_1 r_2}(\tau) = \alpha \rho_{ss}(\tau) * \delta(t - D) \quad (6.48)$$

Equation (6.48) indicates that the delta function has been spread or “smeared” by the Fourier transform of the signal spectrum. If  $s(t)$  is a white noise source, then its Fourier transform is a delta function and no spreading takes place. For all other types of time waveforms some spreading will occur. In all cases, however, the autocorrelation function peaks at delay  $D$ .

In this section we examine five generalizations to  $\Psi(f) = 1$ ; in particular, we analyze the weighting functions indicated in Table 6.1 [9]. The purpose is to minimize the smearing.

### 6.5.1 Roth Process

The Roth weighting is given by [10]

$$\Psi_R(f) = \frac{1}{G_{r_1 r_1}(f)} \quad (6.49)$$

which yields

$$\hat{\rho}_{\hat{n}_1\hat{n}_2}^{(R)}(\tau) = \int_{-\infty}^{\infty} \frac{\hat{G}_{\hat{n}_1\hat{n}_2}(f)}{G_{\hat{n}_1\hat{n}_1}(f)} e^{j2\pi f\tau} df \quad (6.50)$$

When  $n_1(t) \neq 0$ , as is almost always the case in any real scenario [ $n_1(t)$  can pass through zero occasionally in most cases, but even then  $G_{n_1n_1}(f) \neq 0$ ], then

$$G_{\hat{n}_1\hat{n}_1}(f) = G_{ss}(f) + G_{n_1n_1}(f) \quad (6.51)$$

and

$$\rho_{\hat{n}_1\hat{n}_2}^{(R)}(\tau) = \delta(\tau - D) * \int_{-\infty}^{\infty} \frac{\alpha G_{ss}(f)}{G_{ss}(f) + G_{n_1n_1}(f)} e^{j2\pi f\tau} df \quad (6.52)$$

One advantage of this process is that it tends to suppress the frequency regions where  $G_{n_1n_1}(f)$  is large and  $\hat{G}_{\hat{n}_1\hat{n}_2}(f)$  is more likely to be in error. The Roth process spreads the peak being sought.

### 6.5.2 Smoothed Coherence Transform (SCOT)

Errors in  $\hat{G}_{\hat{n}_1\hat{n}_2}(f)$  can be caused by large  $G_{n_1n_1}(f)$ , large  $G_{n_2n_2}(f)$ , or both. Therefore, we don't know for sure whether to make  $\Psi_R(f) = 1/G_{\hat{n}_1\hat{n}_1}(f)$  or  $\Psi_R(f) = 1/G_{\hat{n}_2\hat{n}_2}(f)$ . The SCOT algorithm addresses this issue by making the weighting function

$$\Psi_S(f) = 1/\sqrt{G_{\hat{n}_1\hat{n}_1}(f)G_{\hat{n}_2\hat{n}_2}(f)} \quad (6.53)$$

which generates the CCF

$$\hat{\rho}_{\hat{n}_1\hat{n}_2}^{(S)}(\tau) = \int_{-\infty}^{\infty} \hat{\gamma}_{\hat{n}_1\hat{n}_1}(f) e^{j2\pi f\tau} df \quad (6.54)$$

where the coherence is approximated with

$$\hat{\gamma}_{n_1 n_1}(f) \triangleq \frac{\hat{G}_{n_1 n_2}(f)}{\sqrt{\hat{G}_{n_1 n_1}(f)\hat{G}_{n_2 n_2}(f)}} \quad (6.55)$$

In Figure 6.4,  $H_1(f) = 1/\sqrt{G_{n_1 n_1}(f)}$  and  $H_2(f) = 1/\sqrt{G_{n_2 n_2}(f)}$  in the SCOT process. When  $n_1(t) \neq 0$  and  $n_2(t) \neq 0$ , then the same spreading occurs as in the Roth process.

### 6.5.3 Phase Transform (PHAT)

The spreading of the peak in the last two processes is avoided in the ad hoc PHAT process. The weighting function in this case is

$$\Psi_p(f) = \frac{1}{|G_{n_1 n_2}(f)|} \quad (6.56)$$

which produces

$$\hat{\rho}_{n_1 n_2}^{(P)}(\tau) = \int_{-\infty}^{\infty} \frac{\hat{G}_{n_1 n_2}(f)}{|G_{n_1 n_2}(f)|} e^{j2\pi f\tau} df \quad (6.57)$$

with uncorrelated noise,  $G_{n_1 n_2}(f) = 0$ , and

$$|G_{n_1 n_2}(f)| = \alpha G_{ss}(f) \quad (6.58)$$

In the ideal case we have

$$\hat{G}_{n_1 n_2}(f) = G_{n_1 n_2}(f) \quad (6.59)$$

and

$$\frac{\hat{G}_{n_1 n_2}(f)}{|G_{n_1 n_2}(f)|} = e^{j\phi(f)} = e^{j2\pi fD} \quad (6.60)$$

which, of course, has unit amplitude and

$$\rho_{\hat{r}_1 \hat{r}_2}^{(P)}(\tau) = \delta(t - D) \quad (6.61)$$

In more realistic scenarios when (6.59) is not satisfied, the delta function is not produced, but is smeared somewhat but typically less than the previous two processes.

### 6.5.4 Eckart Filter

The Eckart filter maximizes the deflection criterion, which is the ratio of the change in mean correlator output due to the signal present to the standard deviation of the correlator output due to noise alone. For long averaging time,  $T$ , the deflection coefficient is given by

$$d^2 = \frac{L \left[ \int_{-\infty}^{\infty} H_1(f) H_2^*(f) G_{s_1 s_2}(f) df \right]^2}{\int_{-\infty}^{\infty} |H_1(f)|^2 |H_2(f)|^2 G_{n_1 n_1}(f) G_{n_2 n_2}(f) df} \quad (6.62)$$

where  $L$  is a constant proportional to  $T$ , and  $G_{s_1 s_2}(f)$  is the cross-power spectrum between  $s_1(t)$  and  $s_2(t)$ . For the model specified by (6.31) and (6.32),

$$G_{s_1 s_2}(f) = \alpha G_{ss}(f) e^{j2\pi f D} \quad (6.63)$$

From this we get

$$H_1(f) H_2^*(f) = \Psi_E(f) e^{j2\pi f D} \quad (6.64)$$

which maximizes  $d^2$  where

$$\Psi_E(f) = \frac{\alpha G_{ss}(f)}{G_{n_1 n_1}(f) G_{n_2 n_2}(f)} \quad (6.65)$$

### 6.5.5 Maximum Likelihood

The ML estimator selects as the estimate of delay the value of  $\tau$  at which

$$\rho_{\hat{r}_1 \hat{r}_2}^{(ML)} = \int_{-\infty}^{\infty} \hat{G}_{\hat{r}_1 \hat{r}_2}(f) \frac{1}{|G_{\hat{r}_1 \hat{r}_2}(f)|} \frac{|\gamma_{12}(f)|^2}{1 - |\gamma_{12}(f)|^2} e^{j2\pi f \tau} df \quad (6.66)$$

achieves a peak. The weighting is

$$\Psi_{\text{ML}}(f) = \frac{1}{|G_{\tau_2}(f)|} \frac{|\gamma_{12}(f)|^2}{1 - |\gamma_{12}(f)|^2} \quad (6.67)$$

where  $|\gamma_{12}(f)|^2 \neq 1$ , achieves the ML estimator. Under low SNR conditions, this ML process is equivalent to the Eckart prefiltering and cross correlation process.

### 6.5.6 Variance of the Delay Estimators

It can be shown [11] that the variance of these time delay estimators in the neighborhood of the true delay for weighting function  $\Psi(f)$  is given by

$$\text{var}\{\hat{D}\} = \frac{\int_{-\infty}^{\infty} |\Psi(f)|^2 (2\pi f)^2 G_{\tau_1}(f) G_{\tau_2}(f) [1 - |\gamma(f)|^2] df}{T \left[ \int_{-\infty}^{\infty} (2\pi f)^2 |G_{\tau_2}(f)| |\Psi(f)| df \right]^2} \quad (6.68)$$

The variance of the ML process is

$$\text{var}^{\text{ML}}\{\hat{D}\} = \frac{1}{2T \int_0^{\infty} (2\pi f)^2 \frac{|\gamma(f)|^2}{1 - |\gamma(f)|^2} df}$$

Numerical simulations of these processes are possible but are dependent on the particular signal and noise employed. Therefore, they do not shed much insight into the results.

## 6.6 Time Delay Estimation Using the Phase of the Cross-Spectral Density

### 6.6.1 Introduction

Two approaches to correlation-based TDOA estimation are available: time and frequency-domain-based estimators [12]. The more popular of these is the time-domain-based estimators, as described above, where the peak of the CCF is determined and used as the estimate of the TDOA. When using a frequency-domain estimator, frequency and direction filtering can be used to increase the

SNR by suppressing signals in all but one angular sector. The beamwidth can be quite narrow if the signals have a large bandwidth since the beamwidth is proportional to the reciprocal of the signal bandwidth.

In this section we present such a frequency-domain-based estimator where the phase of the CSD of the signal received at two separated intercept sites is used to determine the TDOA. Initially we assume perfect synchronization between the two receiving systems. We relax that assumption in this section and examine the effects of imperfect synchronization.

### 6.6.2 Data Model

The transmitted signal is received by two sensors whose outputs, after quadrature mixing, contain two noisy and differently delayed versions of the complex valued baseband signal  $s(t)$ . The transmitted signal is assumed to be unknown and accordingly  $s(t)$  is modeled as a zero-mean wide-sense stationary process, characterized by its autocorrelation function  $\rho_{ss}(\tau) = \mathcal{E}\{s(t+\tau)s^*(t)\}$ . Furthermore, we assume that  $s(t)$  is band-limited into the frequency range  $(-W, W)$  Hz and that its PSD is continuous in frequency  $f$ , that is the transmitted signal is assumed to be band-limited, but broadband. The two channels in the digital receivers are sampled with sampling frequency  $f_s$  Hz, such that  $f_s \geq 2W$ . Without loss of generality, we let  $f_s = 1$  Hz in the forthcoming.

Under an assumption of perfect receiver synchronization, the complex-valued output from two synchronous digital receivers are modeled as

$$\begin{aligned} r_1(n) &= s(t)|_{t=n} + n_1(n) \\ r_2(n) &= s(t)|_{t=n-D} + n_2(n) \end{aligned} \quad (6.69)$$

where  $D$  denotes the unknown normalized delay and the actual delay is given by  $D/f_s$ . The noise terms  $n_p(n)$ , ( $p = 1, 2$ ), are assumed zero-mean white (temporal and spatial) complex Gaussian with variance  $\sigma_p^2$ , respectively.

The CCF for the received signals is

$$\rho_{r_1 r_2}(m) = \mathcal{E}\{r_1(n+m)r_2^*(n)\} \quad (6.70)$$

where  $*$  denotes complex-conjugate. Then, since  $n_1(n)$  and  $n_2(n)$  are uncorrelated and  $s(t)$  is broadband,

$$\rho_{r_1 r_2}(m) = \rho_{ss}(\tau)|_{\tau=m+D} \quad (6.71)$$

One method to estimate the unknown delay  $D$  is to find the argument that maximizes the estimated CCF, say,  $\hat{\rho}_d(m)$ , followed by some interpolation to find the delay with sub-bin accuracy. In particular, triple parabolic interpolation for time-delay estimation is considered in [6]. Alternatively, the delay can be estimated from the spectral representation of the CCF. The CSD is defined by the DFT of the CCF

$$P_{r_1 r_2}(\omega) = \mathcal{F}_d \{ \rho_{r_1 r_2}(m) \} = \mathcal{F}_d \{ \rho_{ss}(\tau) \} \Big|_{\tau=m+D} \quad (6.72)$$

where  $\mathcal{F}_d \{ \cdot \}$  denotes the DFT. A translation in the time domain corresponds to a rotation in the frequency domain; that is,

$$P_{r_1 r_2}(\omega) = e^{j\omega D} \mathcal{F}_d \{ \rho_{ss}(\tau) \} \Big|_{\tau=m} = e^{j\omega D} P_s(\omega) \quad (6.73)$$

where  $P_s(\omega)$  is the PSD of the sampled version of the baseband signal  $s(t)$ .

The direct correlator estimator by peak picking the CCF followed by parabolic interpolation mentioned above is not statistically efficient at high SNR [13]. Accordingly, alternative methods for the sub-bin search are required. In this section, we rely on time delay estimation using phase data [2]. We note from (6.73) that

$$\Gamma_{r_1 r_2}(\omega) = \angle P_{r_1 r_2}(\omega) = \omega D \quad (6.74)$$

Now, estimating the unknown  $D$  in the time domain by maximization of the magnitude of the CCF has been transformed into fitting a straight line to the argument of the CSD. In Figure 6.5, the linear portion of the phase is used to estimate  $D$ .

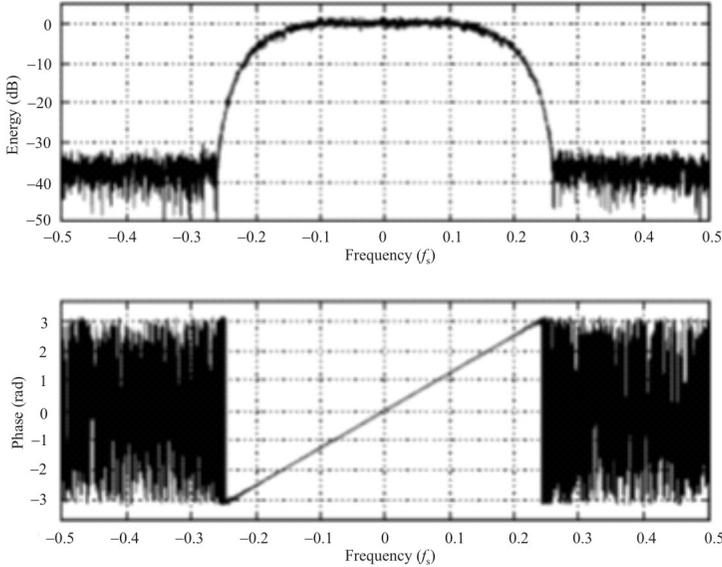
### 6.6.3 Properties of the Sample CSD

The finite length discrete signals  $r_1(n)$  and  $r_2(n)$  are given by

$$\{r_1(0), \dots, r_1(N-1)\} \quad (6.75)$$

and

$$\{r_2(0), \dots, r_2(N-1)\} \quad (6.76)$$



**Figure 6.5** Typical energy-phase representation of the CSD obtained in two EW systems. (Source: [12]. © Radiovetenskap och Kommunikation 2002. Reprinted with permission.)

where  $r_p(n) = 0$  for  $n < 0$  or  $n \geq N$  ( $p = 1, 2$ ). Replace the CCF in (6.70) with its common estimation

$$\hat{\rho}_{r_1 r_2}(m) = \begin{cases} \frac{1}{N} \sum_{n=0}^{N-1} r_1(n+m) r_2^*(n), & m = 1-N, \dots, N-1 \\ 0, & m > |N-1| \end{cases} \quad (6.77)$$

The estimated CSD is now given by the DFT of  $\hat{\rho}_{r_1 r_2}(m)$ . We have

$$\begin{aligned} \hat{P}_{r_1 r_2}(\omega) &= \mathcal{F}_d \{ \hat{\rho}_{r_1 r_2}(m) \} \\ &= \sum_{n=1-N}^{N-1} \hat{\rho}_{r_1 r_2}(n) e^{-j\omega n} \\ &= \frac{1}{N} \sum_{n=1-N}^{N-1} \sum_{m=1-N}^{N-1} r_1(n+m) r_2^*(n) e^{-j\omega n} \end{aligned} \quad (6.78)$$

where (6.77) with zero padding was used in the last equality. Let  $p = n + m$ ; then

$$\begin{aligned}
\hat{P}_{r_1 r_2}(\omega) &= \frac{1}{N} \left( \sum_{n=1-N}^{N-1} r_2^*(n) e^{j\omega n} \sum_{p=n+1-N}^{n+N-1} r_1(p) e^{-j\omega p} \right) \\
&= \frac{1}{N} \left( \sum_{n=1-N}^{N-1} r_2^*(n) e^{j\omega n} \right) \left( \sum_{p=n+1-N}^{n+N-1} r_1(p) e^{-j\omega p} \right) \quad (6.79)
\end{aligned}$$

Accordingly,

$$\begin{aligned}
\hat{P}_{r_1 r_2}(\omega) &= \frac{1}{N} \mathcal{F}_d \{r_1(m)\} \mathcal{F}_d^* \{r_2(m)\} \\
&= e^{j\omega D} P_s(\omega) + E(\omega) \quad (6.80)
\end{aligned}$$

where  $E(\omega)$  is an error term describing imperfections due to finite sample effects and noise. Note the similarity between (6.73) and (6.80).

Utilizing the DFT on a regular grid in place of the  $\mathcal{F}_d$  – operator, we obtain

$$\hat{\Gamma}_{r_1 r_2}(n) = \frac{2\pi n D}{B} + \epsilon(n) \quad (6.81)$$

where  $B \geq 2N - 1$  denotes the number of frequency bins, and  $\epsilon(n)$  is the error term describing imperfections due to finite samples effects and noise. With  $B = 2N - 1$ , the frequency index  $n$  spans the interval  $n = 1 - N, \dots, N - 1$ , where  $N$  is the length of the two sample records.

### 6.6.4 TDOA Estimation

In general, we assume to know very little about the SOIs, so we choose an estimator that does not require any probabilistic assumption on the signals. It is shown in [2, 4], that a *linear least-squares estimator* (LLSE) is statistically efficient for real-valued flat spectrum signals, but additional weighting is needed for non-flat spectrum signals. The proper frequency weighting function for signals with arbitrary spectrum is derived in [14].

From (6.81), we can constrain a first order polynomial model to have a zero bias term. The LLSE criterion becomes

$$J(D) = \sum_{n=1-N}^{N-1} \left( \hat{\Gamma}_{r_1 r_2}(n) - \frac{2\pi n D}{2N-1} \right)^2 \quad (6.82)$$

which is minimized by

$$D_{\text{LLS}} = \frac{3}{2\pi N(N-1)} \sum_{n=1-N}^{N-1} n \hat{\Gamma}_{\gamma_{12}}(n) \quad (6.83)$$

Of course the bandwidth of the signal,  $(-W, W)$  Hz, must be known and here,  $n = 1-N, \dots, N-1$  denotes the number of frequency-bins allocated by the signal. The stationary assumption on the signal yields phase values independent of one another [2, 15]. Hence the  $\hat{\Gamma}_{\gamma_{12}}(n)$ 's are independent of one another. The variance of  $\hat{\Gamma}_{\gamma_{12}}(n)$  is [16]

$$\sigma^2 \left[ \hat{\Gamma}_{\gamma_{12}}(n) \right] \approx \frac{1 - \gamma_{12}(n)}{\gamma_{12}(n)} \quad (6.84)$$

where  $\gamma_{12}(n)$  is the *discrete coherence function* [17] and the approximation comes from a small errors assumption. The variance of (6.83) follows as

$$\sigma^2 \left( \hat{D}_{\text{LLS}} \right) \approx \frac{9}{4\pi^2 N^2 (N-1)^2} \sum_{n=1-N}^{N-1} \frac{n^2 [1 - \gamma_{12}(n)]}{\gamma_{12}(n)} \quad (6.85)$$

For flat spectrum signals and equal channel noise powers  $\sigma_n^2$ , we obtain

$$\gamma_{12}(n) = \frac{\delta^2}{1 + 2\delta + \delta^2}$$

where  $\delta = \sigma_s^2 / \sigma_n^2$  is the SNR, and with  $\sigma_s^2$  denoting the signal power.

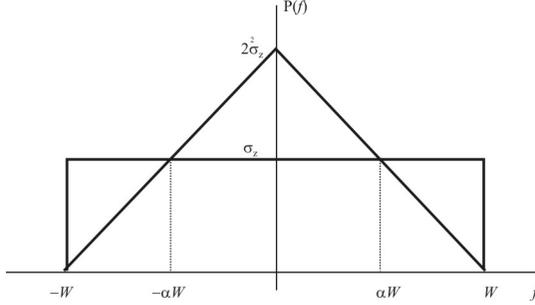
Evaluating (6.85) yields

$$\sigma^2 \left( \hat{D}_{\text{LLS}} \right) \approx \frac{3(2N-1)}{4\pi^2 N(N-1)} \frac{2\delta+1}{\delta^2} \quad (6.86)$$

### 6.6.5 Cramer-Rao Bound

The *Cramer-Rao bound* (CRB) is given by [18]

$$\text{CRB}(D / f_s) = \frac{1}{8\pi^2 T} \left[ \int_{-\infty}^{\infty} \frac{f^2 \gamma_{12}(f)}{1 - \gamma_{12}(f)} df \right]^{-1} \quad (6.87)$$



**Figure 6.6** Power spectral densities.

where  $T = N / f_s$  is the observation time, and  $\gamma_{12}(f)$  is the coherence function. From [18],

$$\gamma_{12}(f) = \frac{P_s^2(f)}{P_{r_1}(f)P_{r_2}(f)} \quad (6.88)$$

In (6.88),  $P_s(f)$ ,  $P_{r_1}(f)$ , and  $P_{r_2}(f)$  are the PSDs of the continuous time signals  $s(t)$ ,  $r_1(t)$ , and  $r_2(t)$ , respectively. We assume that  $s(t)$ ,  $n_1(t)$ , and  $n_2(t)$  are all strictly band-limited to the frequency range  $(-W, W)$  Hz. Without loss of generality, we assume Nyquist sampling, that is  $f_s = 2W$ .

### 6.6.5.1 CRB for Flat Spectrum Signals

Assuming flat spectrum signals (see Figure 6.6), that is

$$C_{12}(f) = \frac{\delta^2}{1 + 2\delta + \delta^2} \quad (6.89)$$

Accordingly,

$$\begin{aligned} \text{CRB}(D / f_s) &= \frac{f_s}{8\pi^2 N} \frac{2\delta + 1}{\delta^2} \left[ \int_{-W}^W f^2 df \right] \\ &= 16 \frac{3f_s}{8\pi^2 N W^3} \frac{2\delta + 1}{\delta^2} \\ &= \frac{3}{8\pi^2 N W^2} \frac{2\delta + 1}{\delta^2} \end{aligned} \quad (6.90)$$

where  $W = f_s / 2$  was used in the last equality. Proper scaling gives

$$\text{CRB}(D) = \frac{3}{2\pi^2 N} \frac{2\delta + 1}{\delta^2} \quad (6.91)$$

The result (6.91) forms a lower bound on the performance of any unbiased time delay estimator for signals with a fat spectrum. Note that for large  $N$ , (6.86) is equivalent to (6.91), and thus the LLSE is asymptotically efficient for flat spectrum signals.

### 6.6.5.2 CRB for a Triangular Spectrum

We approximate a signal that has been filtered with a signal that has a triangular spectrum as illustrated in Figure 6.6. Considering equal channel noise with a flat power spectral density  $\sigma_n^2$ , we define a frequency-dependent SNR as

$$\delta(f) = \frac{P_s(f)}{\sigma_n^2}, \quad |f| \leq W \quad (6.92)$$

Inserting (6.92) into (6.87) yields, for  $T = N / f_s$

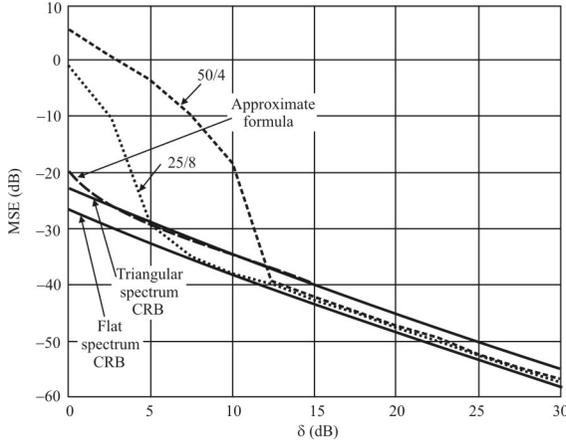
$$\text{CRB}(D / f_s) \approx \frac{f_s}{8\pi^2 N} \left[ \int_{-W}^W \frac{\delta^2(f)}{1 + 2\delta(f)} f^2 df \right]^{-1} \quad (6.93)$$

The major contribution to the CRB is given by the frequency regions with high SNR, that is for  $\delta(f) \gg 1$ . Thus, we approximate the CRB in (6.93) as

$$\text{CRB}(D / f_s) \approx \frac{f_s}{4\pi^2 N} \left[ \int_H \delta(f) f^2 df \right]^{-1} \quad (6.94)$$

where  $H$  denotes the high-SNR frequency regions. Consider a signal with triangular PSD according to Figure 6.6, where  $\delta = \sigma_s^2 / \sigma_n^2$  is the SNR within the full bandwidth  $W$ . Then

$$\delta(f) = 2\delta \left( 1 - \frac{|f|}{W} \right), \quad |f| \leq W \quad (6.95)$$



**Figure 6.7** Mean square error as function of SNR for  $p = 4$  and  $p = 8$ , respectively. The Cramer-Rao bound for flat spectrum signals and triangular spectrum signals are given as references.

The high-SNR frequency region is characterized by  $H \in (-\alpha W, \alpha W)$  for some  $\alpha$  in the interval  $0 < \alpha < 1$ . Inserting (6.95) into (6.94) gives

$$\begin{aligned} \text{CRB}(D / f_s) &\approx \frac{f_s}{16\pi^2 N \delta} \left[ \int_0^{\alpha W} f^2 \left( 1 - \frac{|f|}{W} \right) df \right]^{-1} \\ &= \frac{3}{2\pi^2 N \delta \alpha^3 W^2 (4 - 3\alpha)} \end{aligned} \tag{6.96}$$

where  $f_s = 2W$  was used. Proper scaling gives the CRB for the normalized delay  $D$

$$\text{CRB}(D) \approx \frac{6}{\pi^2 N \delta \alpha^3 (4 - 3\alpha)} \tag{6.97}$$

Comparing (6.91) with (6.97), we note that the latter result is at least twice as large as the former one. We may define the high-SNR region by  $\alpha$  given by the line crossing  $P_s(\alpha W) = \sigma_n^2$ , that is  $\alpha = 1 - 1/2\delta$ . The approximation (6.97) is quite accurate, as illustrated in Figure 6.7.

### 6.6.6 Other Considerations

For long delays, the phase unwrapping can be problematic and can lead to loss in performance due to erroneous estimates. One natural approach is to perform a first

initial estimate by peak-picking the magnitude of the CCF and precompensate the data according to this estimate. Then, a delay estimate with sub-bin accuracy is obtained by applying the method described above on the precompensated data. The final time delay estimate is obtained as the sum of the initial estimate from the CCF method and the correction obtained from the least-squares fit of the phase of the CSD.

Performance can be improved by averaging the CSD. Here, for simplicity, we average the CSD without overlap, that is based on  $N$  samples the CSD is calculated in  $p$  disjoint intervals based on  $N/p$  measurements. Increasing  $p$  lowers the SNR threshold. On the other hand, an increased  $p$ -value reduces the operating range of the estimator, that is, doubling  $p$  implies that the operating range of the estimator is reduced by half.

### Example

Flat spectrum signals are simulated in this example [12]. Based on 500 independent simulation runs, the performance of the least-squares fit of the phase of the CSD is investigated. Gaussian data were generated with an integer delay of  $D = 3$  samples. A block length of  $N = 200$  samples was considered. The mean-square error (MSE) as function of SNR is depicted in Figure 6.7. An initial integer-delay estimate was obtained by peak-picking the CCF. The correction was obtained by the least-squares fit (6.83). A sub-block length of 50 and 25 samples were used, that is  $p = 4$  and  $p = 8$ , respectively. The CSD in each sub-block was calculated using the discrete Fourier transform of the estimated CCF in (6.77), with some additional zero-padding.

The results are displayed in Figure 6.7. From the depicted curves we observe that 25/8 (i.e. a sub-block length of 25 samples and eight times averaging of the CSD) has a lower SNR threshold compared to 50/4. The performance above the threshold is similar for both methods, that is, they produce estimates with an MSE close to the CRB. The CRB is also included, given by (6.91) as is the CRB for the signal with a triangular spectrum.

We can also observe that the MSE is quite low ( $< -10$  dB) for rather low SNRs, a very desirable attribute for practical EW systems where the target SOIs can be significant ranges from the intercept sites.

### 6.6.7 Summary

Direction-finding in an EW scenario is considered and correlation-based TDOA estimators, using the phase of the CSD, analyzed. The results using full bandwidth flat-spectrum signals show that the considered estimator is asymptotically efficient. An approximate CRB for a triangular spectrum signal was derived. The triangular spectrum serves as a rough model of an SOI waveform using a roll-off filter. At least a 3 dB reduction in performance is encountered compared to a full-bandwidth flat-spectrum signal. This method shows promising results for use in an EW system with DF capabilities.

## 6.7 Effects of Frequency and Phase Errors in EW TDOA Direction-Finding Systems

### 6.7.1 Introduction

The time delay estimation method considered in this section is a frequency-domain-based method to estimate the TDOA based on correlation and the phase of the CSD [15, 19]. In order to digitize the signal of interest, a down conversion in frequency is often required using a superheterodyne receiver where an oscillator is used to mix the signal of interest to baseband (see Figure 6.3). In a two-channel TDOA based DF system the performance may be degraded by mismatches of the oscillators of the two intercept receivers. Due to the relatively short acquisition time the mismatch in the oscillators are assumed time-invariant during the acquisition time.

### 6.7.2 Perfect Synchronization

For a perfectly synchronized TDOA DF system the transmitter and the intercept receivers are tuned to exactly the same frequency ( $f_0 = f_1 = f_2$ ) and are phase-locked ( $\varphi_0 = \varphi_1 = \varphi_2 = 0$ ), this yields the model described in [4]. The resulting received signals follow from (6.1) and (6.2)

$$r_1(t) = s(t) + n_1(t) \quad (6.98)$$

$$r_2(t) = s(t - D)e^{-j2\pi f_0 D} + n_2(t) \quad (6.99)$$

The CCF of these signals is  $\rho_{r_1 r_2}(\tau) = \mathcal{E}\{r_1(t + \tau)r_2^*(t)\}$ , and it follows from (6.98) and (6.99) that

$$\rho_{r_1 r_2}(\tau) = e^{j2\pi f_0 D} \rho_{ss}(\tau + D) \quad (6.100)$$

where the autocorrelation function is given by  $\rho_{ss}(\tau) = \mathcal{E}\{s(t+\tau)s^*(t)\}$ . The CSD follows from the Fourier transform of (6.100), that is

$$\begin{aligned} P_{r_1 r_2}(f) &= e^{j2\pi f_0 D} \mathcal{F}\{\rho_{ss}(\tau + D)\} \\ &= e^{j2\pi(f_0 + f)D} P_{ss}(f) \end{aligned} \quad (6.101)$$

where  $\mathcal{F}\{\cdot\}$  denotes the Fourier transform. In the second equality  $P_{ss}(f) = \mathcal{F}\{\rho_{ss}(\tau)\}$  and the time-shift property of the Fourier transform was used. The phase of the CSD is

$$\Gamma_{r_1 r_2}(f) = \angle P_{r_1 r_2}(f) = 2\pi(f_0 + f)D \quad (6.102)$$

which is linear in  $f$  with slope  $2\pi D$  and can be used to estimate the TDOA  $D$ . We see that the unknown carrier frequency of the transmitter,  $f_0$ , only affects the bias of the phase and not the slope, upon which the estimate of  $D$  is based. This is the estimator that was described in Section 6.5.

### 6.7.3 Errors in Synchronization

The intercept receivers in Figure 6.1 are typically passive and do not communicate with the transmitter, that is,  $f_0$  and  $\varphi_0$  are unknown. The intercept receivers may be separated up to several tens of kilometers which most likely will result in frequency and phase offsets between the two oscillators LO<sub>1</sub> and LO<sub>2</sub>. We model this as  $f_1 = f_2 + \epsilon$  and  $\varphi_1 = \varphi_2 + \kappa$  where  $\epsilon$  and  $\kappa$  are unknown constants. For simplicity and without loss of generality, let  $f_0 = f_1 \neq f_2$  and  $\varphi_0 = \varphi_1 \neq \varphi_2$ . The received signal from receiver 1 is then given by (6.98) and the output from the second receiver is

$$\begin{aligned} r_2(t) &= z_2(t) e^{-j2\pi f_2 t - j\varphi_2} \\ &= s(t - D) e^{-j2\pi f_0 D} q(t) + n_2(t) \end{aligned} \quad (6.103)$$

where

$$q(t) = e^{j2\pi\epsilon t + j\kappa} \quad (6.104)$$

is the resulting quantity that describes the receiver imperfections. Since  $s(t)$ ,  $n_1(t)$ , and  $n_2(t)$  are assumed zero-mean, wide-sense stationary the resulting  $r_1(t)$  and  $r_2(t)$  are also zero-mean wss, and their autocorrelations are given by

$$\mathcal{E}\{r_1(t+\tau)r_1^*(t)\} = \rho_{ss}(\tau) + \rho_{n_1n_1}(\tau) \quad (6.105)$$

$$\mathcal{E}\{r_2(t+\tau)r_2^*(t)\} = \rho_{ss}(\tau) + \rho_{n_2n_2}(\tau) \quad (6.106)$$

The CCF between the complex-valued, analog, and wide-sense stationary outputs of the intercept receivers is

$$\begin{aligned} \rho_{r_1r_2}(t+\tau, t) &= \mathcal{E}\{r_1(t+\tau)r_2^*(t)\} \\ &= e^{j2\pi f_0 D} \rho_s(\tau+D)q^*(t) \end{aligned} \quad (6.107)$$

The result follows using (6.98) and (6.103). We note that the CCF depends on time, and in particular that

$$\rho_{r_1r_2}(t+\tau, t) = \rho_{r_1r_2}\left(t+\tau+\frac{k}{\epsilon}, t+\frac{k}{\epsilon}\right) \quad \forall k \quad (6.108)$$

To obtain a time-independent CCF, we time-average to eliminate the dependence on  $t$

$$\begin{aligned} \bar{\rho}_{r_1r_2}(\tau) &= \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} \rho_{r_1r_2}(t+\tau, t) dt \\ &= \lim_{T \rightarrow \infty} \frac{1}{T} e^{j2\pi f_0 D} \rho_{ss}(\tau+D) \int_{-T/2}^{T/2} q^*(t) dt \end{aligned} \quad (6.109)$$

which evaluates to  $\bar{\rho}_{r_1r_2}(\tau) \equiv 0$  due to the infinite observation interval. That is, when there are errors in frequency and phase of the receiver oscillators no information about the TDOA,  $D$ , can be found in the CCF. This is to be compared with (6.100) where finding the maxima of the CCF yields an estimate.

#### 6.7.4 Effects of Finite Sample Time

In practical applications the data acquisition is made over a finite time interval  $0 < T < T_{\max}$ . The acquisition interval is  $(-T/2, T/2)$  for any fixed  $T < \infty$  and  $D$  is assumed to be within  $(-T, T)$ . The signal in the first receiver is then given by

$$r_1(t) = s(t)p(t) + n_1(t)p(t) \quad (6.110)$$

where  $p(t) = u(t + T/2) - u(t - T/2)$  and  $u(t)$  is the step function, that is  $p(t) = 1$  for  $-T/2 \leq t \leq T/2$  and zero otherwise. The delayed signal from the second receiver is given by

$$r_2(t) = s(t - D)e^{-j2\pi f_0 D} q(t)p(t) + n_2(t)p(t) \quad (6.111)$$

where  $q(t)$  was introduced in (6.104). The CCF between the complex-valued analog receiver signals with finite observations is then

$$\rho_{r_1 r_2}(t + \tau, t) = \rho_{ss}(t + D)e^{j2\pi f_0 D} q^*(t)p(t)p(t + \tau) \quad (6.112)$$

Again, we remove the time dependency by time-averaging,

$$\bar{\rho}_{r_1 r_2}(\tau) = \frac{1}{T_U} \int_{T_U} \rho_{r_1 r_2}(t + \tau, t) dt \quad (6.113)$$

where  $\mathcal{U}$  is the nonzero region of  $\rho_{r_1 r_2}(t + \tau, t)$  and  $T_U$  is the time spanned by  $\mathcal{U}$ . From (6.110)–(6.112) it follows that  $\bar{\rho}_{r_1 r_2}(\tau) = 0$  for  $|\tau| > T$ . For  $|\tau| \leq T$

$$\bar{\rho}_{r_1 r_2}(\tau) = \begin{cases} \bar{\rho}_-(\tau), & -T < \tau < 0 \\ \bar{\rho}_+(\tau), & 0 \leq \tau < T \end{cases} \quad (6.114)$$

For  $0 \leq \tau < T$  inserting (6.112) into (6.113) yields

$$\bar{\rho}_+(\tau) = \frac{1}{T_U} \rho_{ss}(\tau + D) e^{j2\pi f_0 D} \int_{-T/2}^{T/2 - \tau} q^*(t) dt \quad (6.115)$$

where  $T_U = T - |\tau|$ . The limits of the integral follows directly from (6.112). A straightforward calculation gives

$$\bar{\rho}_+(\tau) = \rho_{ss}(\tau + D) e^{j2\pi f_0 D} \frac{\sin[\pi\epsilon(T - \tau)]}{\pi\epsilon(T - |\tau|)} q^*(-\tau/2) \quad (6.116)$$

$\bar{\rho}_-(\tau)$  follows from symmetry as

$$\begin{aligned}\bar{\rho}_-(\tau) &= \frac{1}{T\epsilon} \rho_{ss}(\tau+D) e^{j2\pi f_0 D} \int_{-T/2-\tau}^{T/2} q^*(t) dt \\ &= \rho_{ss}(\tau+D) e^{j2\pi f_0 D} \frac{\sin[\pi\epsilon(T+\tau)]}{\pi\epsilon(T-|\tau|)} q^*(-\tau/2)\end{aligned}\quad (6.117)$$

We see that for  $|\tau| < T$

$$\bar{\rho}_{\tau_1\tau_2}(\tau) = \rho_{ss}(\tau+D) e^{j2\pi f_0 D} \frac{\sin[\pi\epsilon(T-|\tau|)]}{\pi\epsilon(T-|\tau|)} q^*(-\tau/2)\quad (6.118)$$

and  $\bar{\rho}_{\tau_1\tau_2}(\tau) = 0$  for  $|\tau| > T$ . Now the CSD is given by the Fourier transform of (6.118)

$$P_{\tau_1\tau_2}(f) = e^{j2\pi f_0 D - j\kappa} \int_{-T}^T \rho_{ss}(\tau+D) e^{j2\pi f_0 D} \frac{\sin[\pi\epsilon(T-|\tau|)]}{\pi\epsilon(T-|\tau|)} e^{-j2\pi(f-\epsilon/2)\tau} d\tau\quad (6.119)$$

The phase of the CSD is used to estimate the TDOA,  $D$ , as described in Section 6.5.

### 6.7.5 White Noise Signal

The expression describing the CSD in (6.119) is quite complicated. However, by approximating the band-limited signal by a zero-mean white noise, that is,

$$\rho_{ss}(\tau) = \delta(\tau)\quad (6.120)$$

then the CSD (6.119) evaluates to

$$P_{\tau_1\tau_2}^{(w)}(f) = e^{j2\pi(f+f_0-\epsilon/2)D-j\kappa} \frac{\sin[\pi\epsilon(T-|D|)]}{\pi\epsilon(T-|D|)}\quad (6.121)$$

for  $|D| < T$ . We can see the familiar  $\text{sinc}(x)$  function that introduces oscillations into the CSD. Note that for some combinations of  $T$ ,  $\epsilon$ , and  $D$ , (6.121) is zero. This occurs specifically when

$$\epsilon T = n + \epsilon |D|, \quad n = 0, 1, 2, \dots\quad (6.122)$$

If  $n = 0$ ,  $T = |D|$ , or when the pulse length is equal to the TDOA,  $P_{r_1 r_2}^{(w)}(f) = 0$ , and the CSD contains no information about the TDOA.

The phase of (6.121) is

$$\angle P_{r_1 r_2}^{(w)}(f) = \begin{cases} 2\pi f D + 2\pi(f_0 - \epsilon/2)D - \kappa, & |D| < T \\ \text{undefined,} & \text{otherwise} \end{cases} \quad (6.123)$$

and we note that in the white noise signal case the frequency and phase errors only affect the bias of the phase-slope using the Fourier transform. The white-noise approximation is only valid for true wideband signals, that is, when the bandwidth of the signal equals or exceeds the receiver bandwidth ( $W_s \geq W$ ) and its spectrum is flat.

### 6.7.6 Simulation Results

The results of simulations based on sample data are run to evaluate the performance in the presence of frequency and phase errors are presented in this section. Considering a crystal oscillator and a GPS-caliber rubidium oscillator, then the relative frequency errors are typically  $\epsilon_r \sim 10^{-5}$  and  $\epsilon_r \sim 10^{-11}$ , respectively. For receiver mixer frequency  $f_0 = 10^9$  Hz and sample rate  $f_s = 10^7$  Hz, this results in  $\epsilon = 10^{-3} f_s$  Hz and  $\epsilon \sim 10^{-9} f_s$  Hz, hence the numerical simulations are based on  $10^{-9} f_s < \epsilon < 10^{-3} f_s$ . The parameter  $\kappa$  is drawn from the uniform distribution  $[-\pi, \pi]$ . Since the FFT is used to calculate the CSD, its discrete frequency bins combined with the frequency error  $\epsilon$  results in leakage into adjacent bins and a noise-like degradation. This effect is not described by the model in (6.119). In addition, not sampling at precisely the correct time causes leakage into adjacent frequency bins as well [20], which also is not included in (6.119).

To generate a wideband flat-spectrum signal a white Gaussian sequence is used. The estimate of the CCF differs from a delayed version of (6.120) due to the finite-length data. This estimation error is suppressed, but not eliminated, by windowing the received data and by averaging the estimate of the CCF. The additive noise used in the simulations are white Gaussian sequences. The CRB for  $D$  in an ideal receiver system with a white noise signal and equal power white Gaussian noise is given in Section 6.5 as (6.91)

$$\text{CRB}(\Delta) = \frac{3}{2\pi^2 N} \frac{2\delta + 1}{\delta^2} \quad (6.124)$$

where  $N = Tf_s$  is the number of samples used,  $\delta$  is the SNR, and  $\Delta = D / f_s$  samples. We define the SNR as

$$\delta = \frac{\rho_{ss}(0)}{\sigma_n^2} \quad (6.125)$$

where  $\sigma_n^2 = \sigma_{n_1}^2 = \sigma_{n_2}^2$ .

### 6.7.7 Estimator Performance

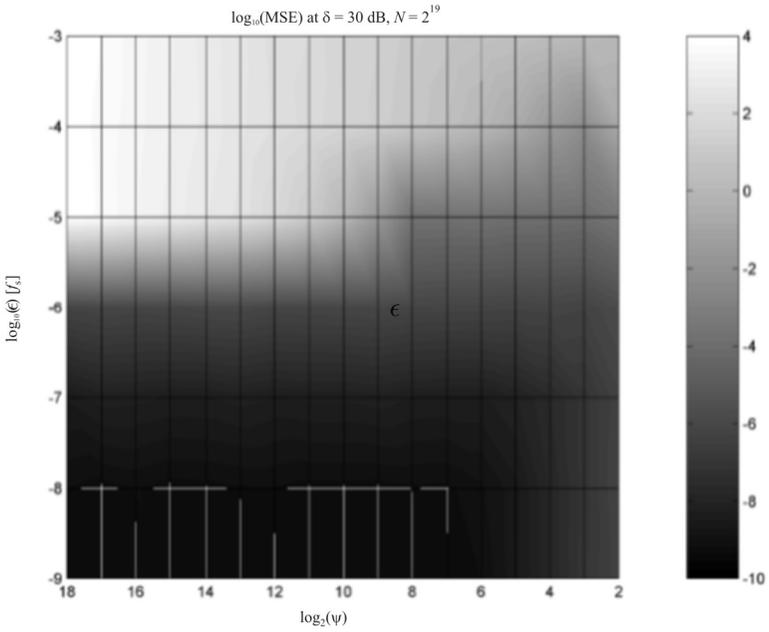
A frequency-based estimator using the phase of the CSD is used. This estimator was investigated in Section 6.5 for the error-free case, where the MSE was used as the performance criterion. Two cases are considered, high (30 dB) and low (-10 dB) SNR using  $N = 2^{19}$  (=524,288) complex samples. The MSE is numerically found through simulation for each case using different combinations of frequency errors  $\epsilon$  and block lengths,  $\psi$ , in the averaging. That is, the data of length  $N$  is divided into  $\Lambda$  overlapping blocks of length  $\psi$ . The results are presented in Figures 6.8 and 6.9 where the gray-scale defines the MSE for the scales on the right in the figures. For  $\delta = 30$  dB and  $N = 2^{19}$  complex data samples we note that the CRB is attained for block lengths larger than  $2^8$  samples and frequency errors  $\epsilon < 10^{-7} f_s$  Hz. The CRB for 30 dB is  $\sim 5 \times 10^{-10}$ .

The low SNR case (Figure 6.9) indicates a higher tolerance against frequency errors and the CRB is attained for low and moderate frequency errors using short blocks (high degree of averaging). In both cases the estimator performs badly in the presence of large frequency errors. This degradation is due to the bin-leakage in the FFT due to the frequency error  $\epsilon$ .

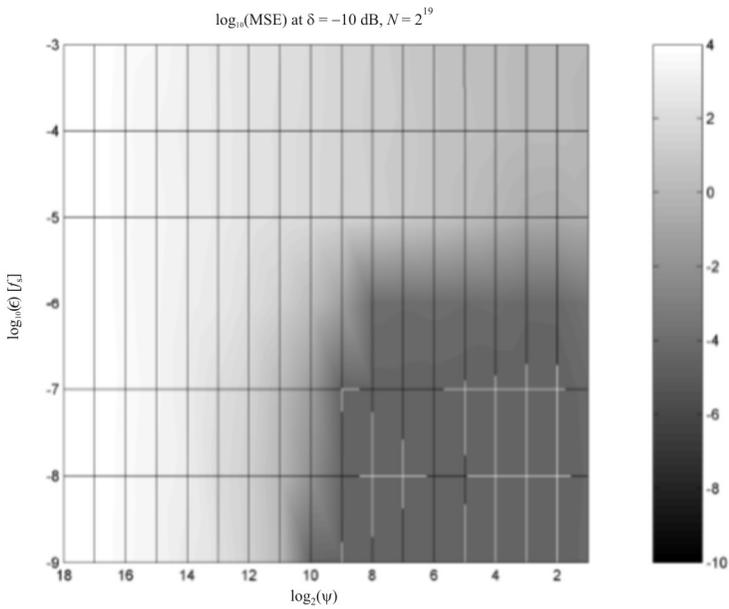
### 6.7.8 Ramifications

In Figures 6.8 and 6.9 we see that for both high (30 dB) and low (-10 dB) SNRs, a block length of  $\psi \sim 2^6 - 2^8$  complex samples gives low MSE using  $N = 2^{19}$  complex data. However, we have shown that this is valid only for data length  $N = 2^{19}$ . What block length should be chosen for a specific data length and a specific frequency error?

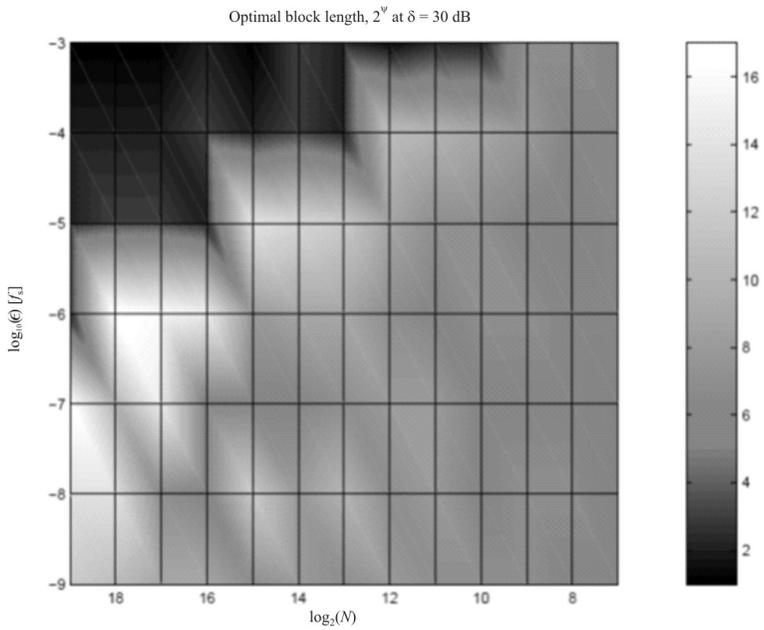
In Figures 6.10 and 6.11, the optimal block length is shown for combinations of data length and frequency errors. The high SNR simulation (Figure 6.10) shows that large block lengths should be chosen at all data lengths, with an exception for large frequency errors where the estimator performs badly and the size of the MSE is in parity with the block length squared. Choosing the block length to  $\psi \sim 2^6 - 2^{10}$  gives good results at all data lengths and frequency errors. Note that Figure 6.10 implies that for really large data lengths the optimal block length becomes



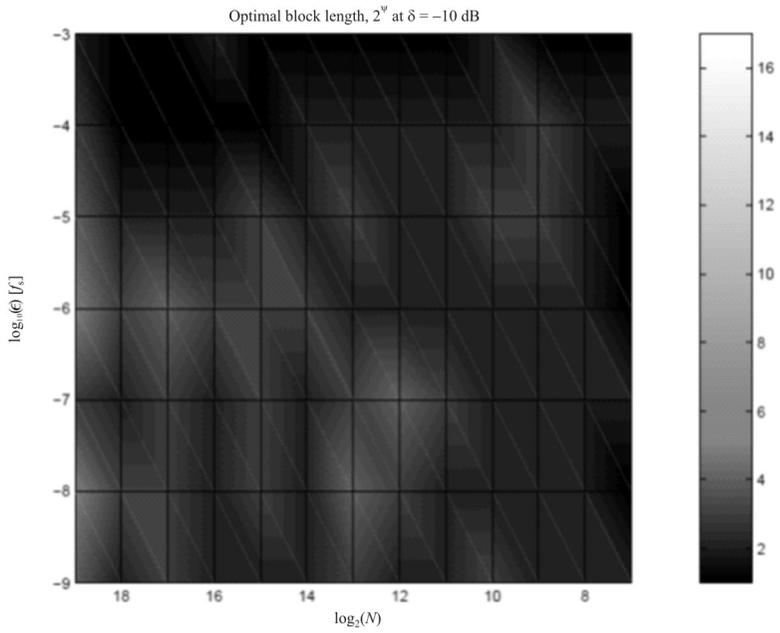
**Figure 6.8** For high SNR, here  $\gamma = 30$  dB, and small  $\epsilon$ ,  $\psi \sim 2^8 - 2^{18}$  gives the smallest MSE. However in presence of large frequency errors, shorter blocks of length  $\psi \sim 2^2 - 2^6$  give the lowest MSE. The white grid shows the level of the CRB, that is where the CRB is attained. (Source: [19]. © 2003 IEEE. Reprinted with permission.)



**Figure 6.9** For low SNR, here  $-10$  dB, block lengths of  $\psi \sim 2^4 - 2^6$  give the lowest MSE regardless of the frequency error. The white grid shows the level of the CRB, that is where the CRB is attained. (Source: [19]. © 2003 IEEE. Reprinted with permission.)



**Figure 6.10** For high SNR the conclusion is to choose the longest block length possible. However, for large frequency errors shorter blocks should be chosen. The sawtooth characteristic is due to limited steps in the simulation. (Source: [19]. © 2003 IEEE. Reprinted with permission.)



**Figure 6.11** For low SNR, shorter blocks (here  $\psi \sim 2^2$ – $2^4$  samples) give the lowest MSE. (Source: [19]. © 2003 IEEE. Reprinted with permission.)

small (for all  $\epsilon$ ) and the MSE becomes large. This is also seen in (6.109) and (6.118) since large  $N$  equals large  $T$ . The low SNR simulation (Figure 6.13) shows that short blocks gives the best result regardless of the frequency error. Here the degradation from the frequency error  $\epsilon$  (bin-leakage) is small compared to the degradation from the channel noise (SNR). The simulations also verify that the phase error  $\kappa$  is of no importance to the performance of the estimator as seen in (6.123).

### 6.7.9 Summary

A model describing a two-channel EW receiver system with frequency and phase imperfections in the receiver oscillators was established for a TDOA DF system. Numerical simulations verify that the oscillator phase is of no importance to the performance of the estimator. The simulations also show that the large frequency errors in an ordinary crystal oscillator are too large to be useful. However, a rubidium oscillator gives satisfactory results and the CRB for an ideal system (with no frequency error) is attained in some cases.

## 6.8 Concluding Remarks

We investigated some techniques for determining the TDOA of a signal arriving at two dispersed intercept sites in this chapter. This information can then be used with the methods discussed in Chapter 3 for estimating the geolocation of the target emitter from which the signal emanates. This is a critical function of EW systems as it is used to ascertain the electronic order of battle, as well as potentially determine target locations accurate enough for targeting of artillery and other kinetic weapons.

We established that the errors inherent in the synchronization of the receivers at the two intercept sites, that is frequency and phase offsets, have no effect on the accuracy of the TDOA estimate when the phase of the CSD is used to determine the TDOA.

### References

- [1] Knapp, C. Y., and G. C. Carter, "The Generalized Correlation Method for Estimation of Time Delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-24, No. 8, August 1976, pp. 320–326.
- [2] Carter, G. C., "Time Delay Estimation," Ph.D. Dissertation, University of Connecticut, Storrs, CT, 1976, pp. 67–70.
- [3] Meyr, H., "Delay-Lock Tracking of Stochastic Signals," *IEEE Transactions on Communications*, Vol. COM-24, No. 3, March 1976, pp. 331–339.

- [4] Azaria, M., and D. Hertz, "Time Delay Estimation by Generalized Cross Correlation Methods," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-32, No. 2, April 1984, pp. 280–285.
- [5] Papoulis, A., *Probability, Random Variables, and Stochastic Processes*, New York: McGraw-Hill, 1965, p. 325.
- [6] Papoulis, A., *Probability, Random Variables, and Stochastic Processes*, New York: McGraw-Hill, 1965, p. 316.
- [7] Hasab, J. C., and R. E. Boucher, "Optimum Estimation of Time Delay by a Generalized Correlator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-27, No. 8, August 1979, pp. 373–380.
- [8] Abramowitz, M., and I. A. Stegun (eds.), *Handbook of Mathematical Functions*, New York: Dover, 1972, p. 11.
- [9] Knapp, C. Y., and G. C. Carter, "The Generalized Correlation Method for Estimation of Time Delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-24, No. 8, August 1976, pp. 320–326.
- [10] Roth, P. R., "Effective Measurements Using Digital Signal Analysis," *IEEE Spectrum*, Vol. 8, April 1971, pp. 62–70.
- [11] MacDonald, V. H., and P. M. Schultheiss, "Optimum Passive Bearing Estimation," *Journal of the Acoustic Society of America*, Vol. 46, 1969, pp. 37–43.
- [12] Falk J., P. Handel, and M. Jansson, "Direction Finding for Electronic Warfare Systems Using the Phase of the Cross Spectral Density," *Proceedings Radiovetenskap och Kommunikation 2002*, Stockholm, Sweden, June, 2002, [www.s3.kth.se/signal/reports/02/IR-S3-SB-0215.pdf](http://www.s3.kth.se/signal/reports/02/IR-S3-SB-0215.pdf).
- [13] Jacovitti G., G. Scarano, "Discrete Time Techniques for Time Delay Estimation," *IEEE Transactions on Signal Processing*, Vol. 41, No. 2, February 1993, pp. 525–533.
- [14] Zhao Z., Z.-Q. Hou, "The Generalized Phase Spectrum Method for Time Delay Estimation," *Proceedings ICASSP 1984*, Vol. 3, March 1984, pp. 46.2/1–4.
- [15] Bendat, J. S., "Statistical Errors in Measurement of Coherence Functions and Input/Output Quantities," *Journal Sound and Vibration*, Vol. 59, No. 3, 1978, pp. 405–421.
- [16] Piersol, A. G., "Time Delay Estimation Using Phase Data," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 29, No. 3, Part 2, June 1981, pp. 471–476.
- [17] Carter, G. C., C. H. Knapp, and A. H. Nuttall, "Estimation of the Magnitude-Squared Coherence Function via Overlapped Fast Fourier Transform Processing," *IEEE Transactions Audio and Electroacoustics*, Vol. 21, No. 4, August 1983, pp. 337–344.
- [18] Carter, G. C., "Coherence and Time Delay Estimation," *Proceedings of the IEEE*, Vol. 75, No. 2, February 1987, pp. 236–255.
- [19] Falk, J., P. Handel, and M. Jansson, "Effects of Frequency and Phase Errors in Electronic Warfare TDOA Direction-Finding Systems," *Proceedings IEEE MILCOM*, 2003.
- [20] Poisel, R. A., *Introduction to Communication Electronic Warfare Systems*, Norwood, MA: Artech House, 2002, Section 9.3.2.



# Chapter 7

## Single-Site Location Techniques

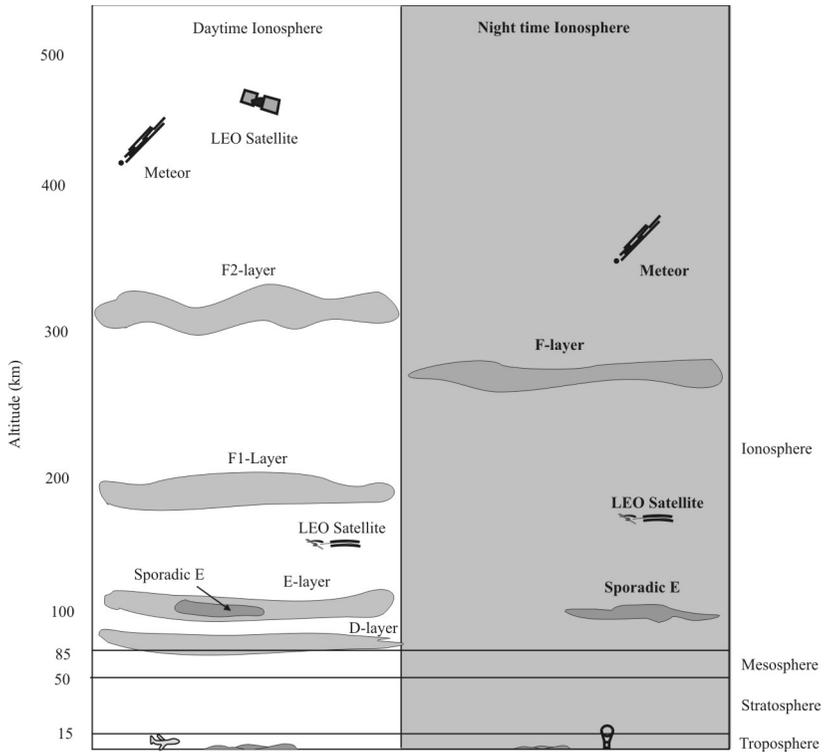
### 7.1 Introduction

Skywave HF signals can be geolocated by triangulation with two or more DF sites, just as signals in other frequency bands can be. In this chapter we will discuss techniques for geolocating HF signals using only a single sensor site. As will be seen, this is possible because long-range HF signals are reflected (refracted actually) in the ionosphere back to the Earth, and measuring the AOAs in three dimensions allows calculation of the PF of the transmitter when the effective height of the point of reflection is known. This height information is not required, however, when there are two or more ray paths arriving at the sensor sight and the angles of arrival of these rays can be measured and discerned.

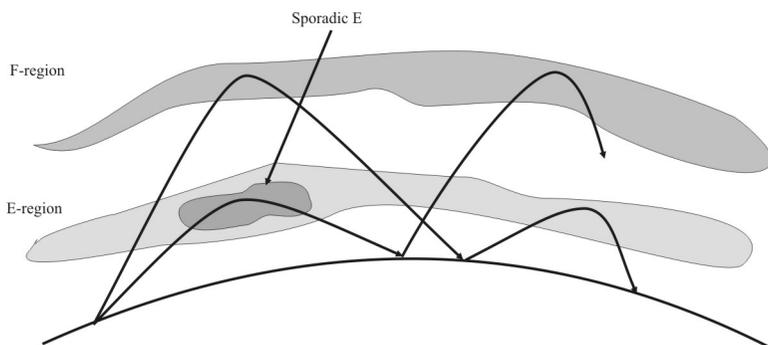
### 7.2 HF Signal Propagation

Under some circumstances, HF signals can propagate for considerable distances. This long-range communication capability of such signals was used by navies around the world for ship-to-shore and ship-to-ship communications before satellite systems were available. In fact, it is still used for that purpose.

The ionospheric layers that surround the Earth are responsible for the phenomenon. During the daytime there are four layers involved in the process, the lowest being the D-layer, the next the E-layer, and then two F-layers: the F1 and F2. At night the two F-layers combine into one and the D- and E-layers disappear, as shown in Figure 7.1. These layers are regions of ions that have been charged by bombardment of atoms with photons from the sun and the resulting free electrons. It is widely believed that it is the free electrons that are responsible for the refraction phenomenon; the positively charged ions have a minor role. They



**Figure 7.1** Structure of the atmosphere close to Earth.



**Figure 7.2** Ionospheric propagation modes vary and can consist of several hops.

interact with HF signals and the Earth's magnetic field in such a way that EM wave refraction is possible. Normally this is below 10 MHz or so, but can occur higher depending on the density of the free electrons.

The D-layer does not actually refract HF signals, but can attenuate them such that penetration is impossible. The E-layer refracts HF signals, as do the F-layers. Occasionally, a highly ionized small region will traverse rapidly through the E-layer. This is called sporadic E, and it significantly changes the characteristics of the reflecting process. The frequency characteristics change as well as the amplitude of the reflection.

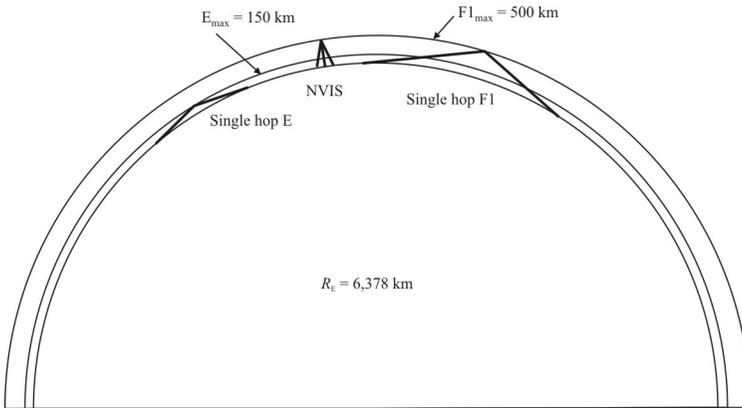
Propagation via the ionosphere is also possible under some circumstances in the low VHF range (up to 100 MHz) as well. The mechanism is the same as for the HF. Distances of up to 2,000 km are possible.

More than one reflection is possible by the EM wave reflecting off the ionosphere and returning to the ground, as shown in Figure 7.2. If the ground is suitably reflective (this is a reflection, not refraction), the signal will be reflected again upward. This can continue for several hops.

Short-term disturbances of the composition of the ionosphere in a region sometimes occur which impact HF radio propagation. They are super-ionized regions that last only a short time and are called *traveling ionospheric disturbances* (TID).

Short-range propagation of HF signals is also possible, of course. The modes of propagation involved with such signaling follow the same rules that VHF and higher signals follow. The direct wave travels straight from the transmit antenna to the receive antenna. This can only occur, however, if the two antennas are within radio line of sight of one another.

There is also a phenomenon known as surface wave propagation that occurs much more predominantly in the HF range than in higher ranges. This is caused by the interaction of the Earth's characteristics (mainly the dielectric constant along



**Figure 7.3** The Earth and ionosphere in correct perspective.

the propagation route) with that of the air immediately above it. Such waves follow the curvature of the Earth and can propagate several hundreds of kilometers.

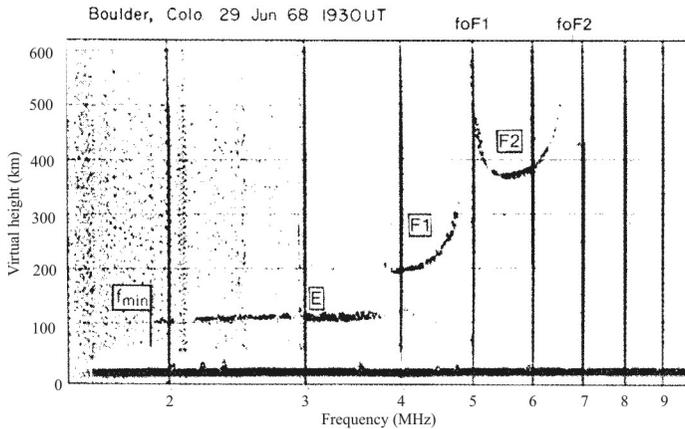
The geolocation of HF signals with these propagation modes is calculated by the same techniques as for VHF+ signals described elsewhere herein. The mode of propagation of interest in this chapter is that due to ionospheric reflections that, most of the time, are used for long-distance communications [*near vertical incidence sky wave* (NVIS) is an exception to this].

Figures 7.1 and 7.2 can be misleading because their dimensions have been intentionally exaggerated to show detail. The actual HF signal propagation paths would be more like those shown in Figure 7.3.

### 7.2.1 Ionograms

The heights of the various ionospheric layers are measured with *sounders*. These are devices that send a signal upward and measure the time delay when a response is received. The frequency is varied (usually swept) so that the ionospheric heights are measured versus frequency. The resultant graph of virtual height versus frequency is called an *ionogram*.

*Vertical sounding* is when the height is measured directly overhead of the receiving site. This is the only method typically available to SSL PF systems; since the targets to be located can normally be anywhere in azimuth and measuring the height of the ionosphere at other points would require many such sounders. When the ionospheric height is measured at some point other than directly overhead, it is referred to as *oblique sounding*. On HF communication paths such



**Figure 7.4** Typical ionogram display. This was obtained at Boulder, Colorado, on June 29, 1968 at 1930 hours UT. The critical frequencies for the layers can be ascertained, as can the LUF. The LUF is just below 2 MHz, while  $f_0E$  is just below 4 MHz,  $f_0F1$  is approximately 5 MHz, and  $f_0F2$  is about 6.6 MHz. (Source: [1]. © Peter Peregrinus 1989. Reprinted with permission.)

measurements are feasible, but, as mentioned, are not practical for calculating a PF.

The devices used to measure the height of the ionosphere versus frequency are called *ionospheric sounders*. Typically, a swept signal is radiated straight up for *vertical sounders* or at an angle for *oblique sounders*. The time of the return reflection is compared to when the signal was transmitted to ascertain the ionospheric height. The resultant displays of the sounder results are the ionograms. A typical vertical ionogram is illustrated in Figure 7.4 taken at Boulder, Colorado, on June 29, 1968, at 1930 hours UT.<sup>1</sup>

Vertical sounders, of course, measure the ionospheric height directly overhead. It is frequently assumed that the ionosphere is homogeneous, and the heights are the same throughout the region. The height is usually measured at the site of the PF system, and it is assumed to be the same height at the reflection point. This is rarely true, so such techniques for measuring the location of transmitters are not that accurate. EEPs with axes that are 10% of the range to the target are typical. Thus, if the target is 100 km from the PF system, the major axis of the EEP can be 10 km or more. It should also be noted that, although the reflecting surface discussed here was the ionosphere, any reflecting surface could

<sup>1</sup> *Universal time* (UT) is time at the Greenwich Observatory outside of London, United Kingdom, where the prime meridian ( $0^\circ$  longitude) is located. UT is also known as *Greenwich time* (GT), *Greenwich mean time* (GMT), and Zulu time.

also be used in some circumstances as long as the distance between that surface and the PF system is known. The effects of range between the emitter and receiver on the ionospheric characteristics are typified by the ionograms shown in Figure 7.5. The maximum frequency that can be used increases with increasing distance between the transmitter and receiver, even though the critical frequency, which is always measured directly overhead, remains the same. In fact, the *maximum usable frequency* (MUF) and critical frequency are related by *Snell's law*:

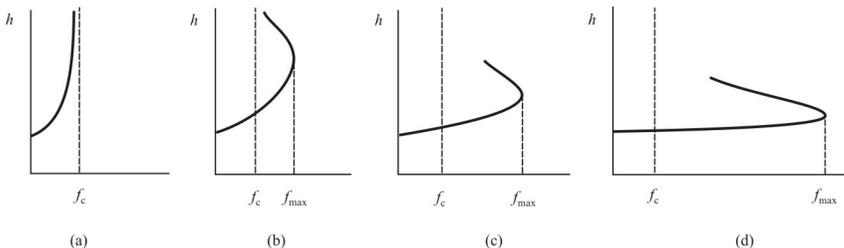
$$f_{\max} = f_c \sec \theta_0 \quad (7.1)$$

where  $\theta_0$  is the zenith angle (measured from the virtual point of reflection). The critical frequency for the E-layer is denoted by  $f_0E$ , while that for F1 is denoted by  $f_0F1$ , and for F2 by  $f_0F2$ .

### 7.2.2 Magnetic Field Effects

The principal effect of the Earth's magnetic field on ionospheric waves is to split the wave into two: one being left-handed (with respect to the magnetic field), the so-called *ordinary wave*, and the other being right-handed, the so-called *extraordinary wave* [2]. The directions here refer to the rotation direction of the polarization vector. These two components traverse the ionosphere via totally different paths. They recombine when they leave the ionosphere, and the net effect is to change the polarization of the wave headed downward toward the receiver.

The frequencies of these two waves are different. Furthermore, the relationship between these frequencies depends on the orientation of the wave propagation relative to the orientation of the magnetic dipole of the Earth. Thus, waves propagating north-south or south-north will have different characteristics than those propagating east-west or west-east.



**Figure 7.5** Oblique ionograms determined at different distances between the transmitter and receiver: (a) vertical, (b) short distance, (c) medium distance, and (d) long distance. Note that the trace is asymptotic to the critical frequency and that the maximum frequency increases with increasing distance.

### 7.3 Single-Site Location

The source of HF signals propagating via sky-wave paths can be located by triangulation using any of the techniques discussed previously. However, it is also possible to use a single PF system for this purpose under some circumstances. If the elevation of the ionospheric layer that the HF signal is refracting through or, more accurately, the equivalent height of the layer that is reflecting the signal is known, as shown in Figure 7.6, then the emitter can be located since we know the range to the target and its angle of arrival. The wave is assumed to be reflected at the midway point between the transmitter and the PF system. The elevation angle is related to the range and ionospheric height by

$$\tan \theta = \frac{B/2}{h} \quad (7.2)$$

so

$$B = 2h \tan \theta \quad (7.3)$$

Thus, by measuring the elevation AOA, the range to the target can be estimated. Although this derivation was given using planes for the ionosphere and the Earth, it can also be derived using spherical surfaces, which is more accurate.

For the SSL technique to function properly, the number of times the signal has been reflected off the ionosphere and/or the ground must be known. Otherwise, the range to the target cannot be determined. Signals that arrive from a longer range will arrive at a lower elevation angle and this may assist in determining the number of hops. Otherwise, it is necessary to assume some number of reflections (usually one).

### 7.4 Passive SSL

When the signal arrives at the sensor from two or more directions, multipath propagation modes are present. In such cases it is possible to calculate the target location without sounding the height of the ionosphere. This is called *passive SSL*.

Consider the situation depicted in Figure 7.7 that shows the signal being reflected off two different ionospheric layers. According to the Briet and Tuve Theorem [3, 4], even though the signals are in actuality refracted in the ionospheric layers, there are equivalent planes from which the signals can be

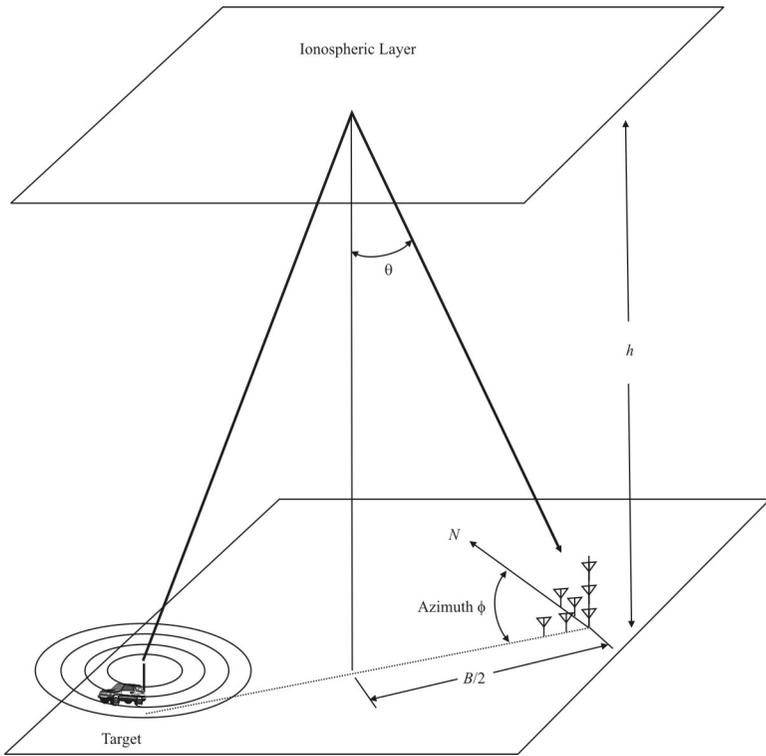


Figure 7.6 Single-site location of HF emitter.

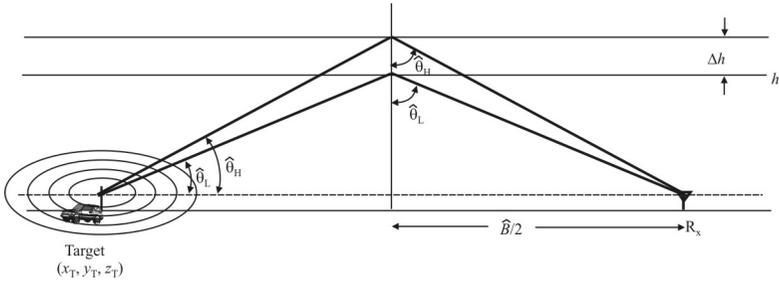


Figure 7.7 Multipath reflection.

assumed to be reflected. It is assumed that the Earth is flat, as are the equivalent reflecting planes, and the range can be determined as

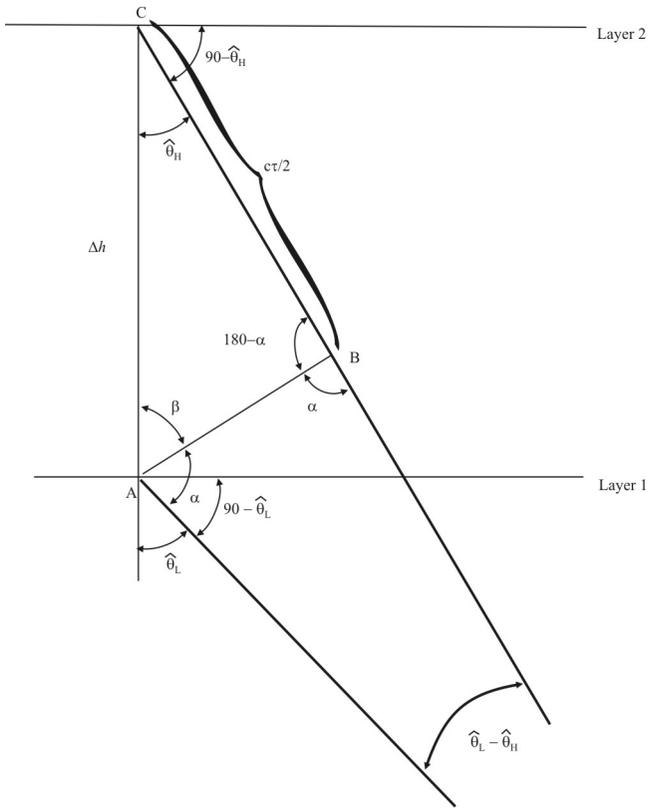
$$\begin{aligned} \tan \hat{\theta}_L &= \frac{\hat{B}/2}{h} & \tan \hat{\theta}_H &= \frac{\hat{B}/2}{h + \Delta h} \\ h &= \frac{\hat{B}/2}{\tan \hat{\theta}_L} = \frac{\hat{B}/2}{\tan \hat{\theta}_H} - \Delta h \\ \hat{B} &= \frac{2\Delta h}{\frac{1}{\tan \hat{\theta}_H} - \frac{1}{\tan \hat{\theta}_L}} \end{aligned} \tag{7.4}$$

Now consider the detail of the signal and ionosphere interaction shown in Figure 7.8. The TDOA of the two signals at the receiver site is denoted by  $\tau$  and the speed of propagation is denoted by  $c$ . Point A is the reflection point of the signal in the lower layer, while point C is that for the upper layer. Denoting the receive point as  $R_x$  (the transmit point could also be used with a mirror reflection of Figure 7.7), triangle  $ABR_x$  forms an isosceles triangle because the time to travel from A to  $R_x$  is the same as that from B to  $R_x$ , by construction of line AB. Summing the angles around the triangle defined by ABC then

$$\beta = \alpha - \hat{\theta}_H \tag{7.5}$$

From triangle  $ABR_x$ ,

$$2\alpha = 180 - (\hat{\theta}_L - \hat{\theta}_H)$$



**Figure 7.8** Multipath reflection details.

$$\alpha = 90 - \frac{\hat{\theta}_L - \hat{\theta}_H}{2} \quad (7.6)$$

and

$$\begin{aligned} \angle ABC &= 180 - \alpha \\ &= 90 + \frac{\hat{\theta}_L - \hat{\theta}_H}{2} \end{aligned} \quad (7.7)$$

From the law of sines,

$$\frac{c \tau / 2}{\sin \left( 90 - \frac{\hat{\theta}_H + \hat{\theta}_L}{2} \right)} = \frac{\Delta h}{\sin \left( 90 + \frac{\hat{\theta}_L - \hat{\theta}_H}{2} \right)} \quad (7.8)$$

Solving (7.8) for  $\Delta h$  and substituting into (7.4) yields

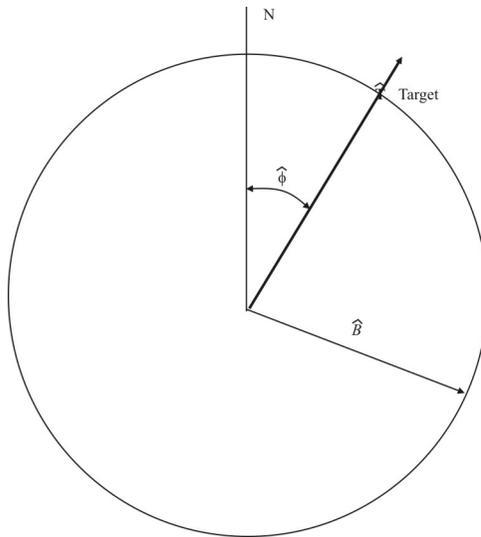
$$\hat{B} = \frac{c \tau \sin \left( 90 + \frac{\hat{\theta}_L - \hat{\theta}_H}{2} \right)}{\frac{1}{\tan \hat{\theta}_H} - \frac{1}{\tan \hat{\theta}_L} \sin \left( 90 - \frac{\hat{\theta}_L + \hat{\theta}_H}{2} \right)} \quad (7.9)$$

Therefore, the range can be determined by measuring the two elevation AOAs [if there are more than two, they can be combined two at a time in (7.9) and the result averaged].

As above, the target location is found by combining the range information with the azimuth angle of arrival, as illustrated in Figure 7.9.

## 7.5 Determining the Reflection Delay with the Cepstrum

The *cepstrum* is one processing technique for determining the time delay between two versions of the same signal or the echoes of a signal that has been reflected. Knowledge of the time delay is necessary for measuring both the azimuth AOA and the elevation AOA for SSL PF estimation. Other methods for measuring



**Figure 7.9** SSL target location.

differential time delay have also been proposed, including the *generalized correlation method* (GCM) [5] and the *adaptive eigenvalues decomposition algorithm* [6].

Its genesis lies in the technology of speech processing, but it has been widely applied outside of that area. The cepstrum can be employed to determine the time delays of the reflections described in the previous section.

Generally speaking, the complex cepstrum is found by computing the inverse *Fourier transform* (FT) of the logarithm of the FT of the signal [7]. The power cepstrum is also sometimes used. The complex cepstrum retains the phase information and therefore requires phase unwrapping, while the power cepstrum does not retain the phase.

If  $c(t)$  represents the baseband impulse response of the propagation path and  $g(t)$  represents the baseband impulse response of the transmitter and receiver, then the received demodulated signal is given by

$$r(t) = T(t) * h(t) * g(t) \quad (7.10)$$

where  $T(t)$  is the source signal and  $*$  denotes convolution. The source signal represented at the receiver output is therefore given by

$$s(t) = g(t) * T(t) \quad (7.11)$$

Therefore,

$$r(t) = s(t) * c(t) \quad (7.12)$$

The goal is to determine  $c(t)$  from (7.12). The power spectrum of  $r(t)$  is given by

$$|R(f)| = |S(f)||C(f)| \quad (7.13)$$

Calculating the natural logarithm,

$$\ln |R(f)| = \ln |S(f)| + \ln |C(f)| \quad (7.14)$$

The inverse FT of (7.14) (denoted by *IFT*) yields the power cepstrum of  $r(t)$

$$\begin{aligned} \hat{r}(t) &= IFT[\ln |R(f)|] \\ &= \hat{s}(t) + \hat{c}(t) \end{aligned} \quad (7.15)$$

As long as  $\hat{s}(t)$  and  $\hat{c}(t)$  do not overlap in the cepstrum domain,  $\hat{c}(t)$  can be found by filtering.

Note that

$$\begin{aligned} FT\{IFT[\ln|R(f)|]\} &= \ln|R(f)| \\ \exp\{\ln|R(f)|\} &= |R(f)| \\ IFT|R(f)| &= \gamma(t) \end{aligned}$$

where  $\gamma(t)$  is the autocorrelation function of  $r(t)$ .

To illustrate using the cepstrum for the purpose of determining the TDOA of the two multipath components, assume that the signal consists of impulse functions [8] that are by definition short, which are transmitted at intervals that are long enough so that the channel and all receiving equipment has settled by the time the next impulse occurs. In that case, the channel impulse response can be written as

$$r(t) = \delta(0) + A\delta(\tau) \quad (7.16)$$

where  $A$  is the normalized signal strength of the second multipath component and  $\tau$  is the TDOA to be determined. The power spectrum of  $r(t)$  is of the form

$$|R(f)| = 1 + A^2 + 2A \cos(2\pi f \tau) \quad (7.17)$$

which can be written in the form

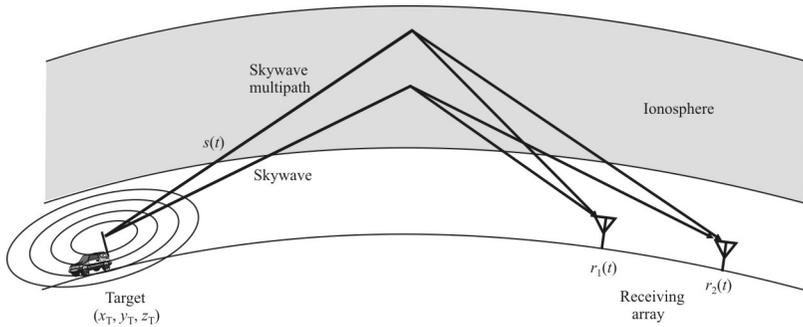
$$|R(f)| = (1 + A^2) \left[ 1 + \frac{2A}{1 + A^2} \cos(2\pi f \tau) \right] \quad (7.18)$$

Now using the series expansion of  $\ln(1 + x)$  as

$$\ln(1 + x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots \quad x < 1 \quad (7.19)$$

and ignoring all but the first term,

$$\ln(1 + x) \approx x \quad x \ll 1 \quad (7.20)$$



**Figure 7.10** Interfering multipath signals for passive ranging.

Using this approximation in  $\ln|R(f)|$  yields

$$\ln|R(f)| \approx \ln(1 + A^2) + 2A \cos(2\pi f \tau) \quad (7.21)$$

which is the cepstrum of  $r(t)$ . This cepstrum contains an impulse at the time delay equal to the TDOA.

The cepstrum is an example of an area of signal processing referred to as *homomorphic deconvolution* [9, 10]. Systems that obey the homomorphic rules are characterized by a generalization of the rules of superposition that are inherent to linear time invariant systems. Homomorphic deconvolution is a form of generalized filtering.

## 7.6 MUSIC Cepstrum SSL

Using (7.9) to compute the range to the target requires knowing the TDOA,  $\tau$ , between the arrival times of the signals at two antennas at the receiver (typically, more than two antennas are used and the results combined). Johnson, Black, and Sonstebly developed a method of computing this TDOA when interference is present using an adaptation of the MUSIC algorithm [11].

The geometry considered is illustrated in Figure 7.10. The transmitted signal,  $s(t)$ , is reflected off a lower layer and a multipath version of the same signal is reflected off a higher layer. The signals received at the receiving array, denoted  $r_1(t)$  and  $r_2(t)$  corresponding to antenna 1 and antenna 2, respectively, are a combination of the lower signal and the multipath component. The received signals can be represented as

$$r_1(t) = A_g s(t) + A_s s(t - \tau) \quad (7.22)$$

$$r_2(t) = A_g s(t - \tau_g) + A_s s(t - \tau_s - \tau) \quad (7.23)$$

where  $\tau_g$  is the propagation delay of the interfering signal between the two antennas at the receiver,  $\tau_s$  is the propagation delay for the lower signal, and  $A_g$  is the path attenuation of the interfering signal, assumed the same for both receive antennas, since it is assumed that the antennas are much closer together than the height of the reflecting ionosphere, likewise for  $A_s$ , which is the path attenuation of the lower signal. Calculating the Fourier transforms of (7.22) and (7.23) yields

$$\mathcal{F}[r_1(t)] = R_1(f) = (A_g + A_s e^{-j2\pi f \tau}) S(f) \quad (7.24)$$

$$\mathcal{F}[r_2(t)] = R_2(f) = [A_g e^{-j2\pi f \tau_g} + A_s e^{-j2\pi f (\tau_g + \tau)}] S(f) \quad (7.25)$$

where  $S(f)$  is the Fourier transform of  $s(t)$ .

The normalized cross-power spectrum of  $r_1(t)$  and  $r_2(t)$  is

$$P_{r_1 r_2}(f) = \frac{R_1(f) R_2^*(f)}{\sqrt{|R_1(f)|^2 |R_2(f)|^2}} \quad (7.26)$$

which, in polar form, becomes

$$P_{r_1 r_2}(f) = \frac{|A_1 A_2^* \arg(A_1 A_2^*) |S(f)|^2}{\sqrt{|A_1 A_1^* |S(f)|^2 |A_2 A_2^* |S(f)|^2}} \quad (7.27)$$

where

$$A_1 = A_g + A_s e^{-j2\pi f \tau} = |A_1| \angle A_1 \quad (7.28)$$

and

$$A_2 = A_g e^{-j2\pi f \tau_g} + A_s e^{-j2\pi f (\tau_g + \tau)} = |A_2| \angle A_2 \quad (7.29)$$

Expression (7.27), after trigonometric manipulation and with the addition of noise, reduces to

$$P_{\tau_1 \tau_2}(f) = e^{j\varphi(f)} + n(f) \quad (7.30)$$

where

$$\varphi(f) = \tan^{-1} \left\{ \frac{A_g^2 \sin 2\pi f \tau_g + A_s^2 \sin 2\pi f \tau_s + 2A_g A_s \cos[\pi f (2\tau + \tau_s - \tau_g)] \sin[\pi f (\tau_g + \tau_s)]}{A_g^2 \cos 2\pi f \tau_g + A_s^2 \cos 2\pi f \tau_s + 2A_g A_s \cos[\pi f (2\tau + \tau_s - \tau_g)] \cos[\pi f (\tau_g + \tau_s)]} \right\} \quad (7.31)$$

Assume that the signal is distributed over  $F$  frequencies so that

$$\vec{P}_{\tau_1 \tau_2}(f) = \begin{bmatrix} e^{j\varphi(f_1)} \\ e^{j\varphi(f_2)} \\ \vdots \\ e^{j\varphi(f_F)} \end{bmatrix} + \begin{bmatrix} n(f_1) \\ n(f_2) \\ \vdots \\ n(f_F) \end{bmatrix} \quad (7.32)$$

The spectral cross-correlation matrix is given by

$$\mathbf{R}_{\tau_1 \tau_2}(f) = \mathcal{E}\{\vec{P}_{\tau_1 \tau_2}(f) \vec{P}_{\tau_1 \tau_2}^\dagger(f)\} \quad (7.33)$$

The  $ij$ th element of  $\mathbf{R}_{\tau_1 \tau_2}(f)$  is given by

$$r_{ij}(f) = \mathcal{E}\{[e^{j\varphi(f_i)} + n(f_i)][e^{j\varphi(f_j)} + n(f_j)]\} \quad (7.34)$$

$$= e^{j[\varphi(f_i) - \varphi(f_j)]} + \sigma^2 \delta(f_i - f_j) \quad (7.35)$$

It is the spectral cross-correlation matrix  $\mathbf{R}_{\tau_1 \tau_2}(f)$  upon which the MUSIC processing is performed.

MUSIC determines the eigendecomposition of  $\mathbf{R}$ . This finds the eigenvalues and their associated eigenvectors. The eigenvalues  $\{\lambda_i\}$  are ordered such that  $\lambda_F \geq \lambda_{F-1} \geq \dots \geq \lambda_1$  with corresponding eigenvector matrix

$$\mathbf{E}_e = [\vec{e}_F \quad \vec{e}_{F-1} \quad \dots \quad \vec{e}_1] \quad (7.36)$$

For a single pair of coherent signals, as is the case here,  $\mathbf{E}_c$  will have rank 1 with a single eigenvector ( $\vec{e}_F$ ) and the nullspace matrix is given by the remaining eigenvectors as

$$\mathbf{E}_N = [\vec{e}_{F-1} \quad \vec{e}_{F-2} \quad \cdots \quad \vec{e}_1] \quad (7.37)$$

The MUSIC spectrum is given as

$$P(f) = \frac{1}{\vec{b}^H \mathbf{E}_N \mathbf{E}_N^H \vec{b}} \quad (7.38)$$

where

$$\vec{b} = [e^{j\phi(f_1)} \quad e^{j\phi(f_2)} \quad \cdots \quad e^{j\phi(f_F)}]^T \quad (7.39)$$

To find the TDOA to use in (7.9) for calculation of the range, the spectrum given by (7.38) is searched for the peak value.

When there are  $p$  paths, as opposed to the two just discussed, then

$$\varphi(f) = \tan^{-1} \left\{ \frac{\sum_{j=1}^p \sum_{i=1}^p k(i, j) V_i V_j \sin[\pi f(\tau_i + \tau_j)] \cos[\pi f(2\tau_{ij} - \tau_i + \tau_j)]}{\sum_{j=1}^p \sum_{i=1}^p k(i, j) V_i V_j \cos[\pi f(\tau_i + \tau_j)] \cos[\pi f(2\tau_{ij} - \tau_i + \tau_j)]} \right\} \quad (7.40)$$

$$k(i, j) = \begin{cases} 1, & i = j \\ 2, & i \neq j \end{cases} \quad (7.41)$$

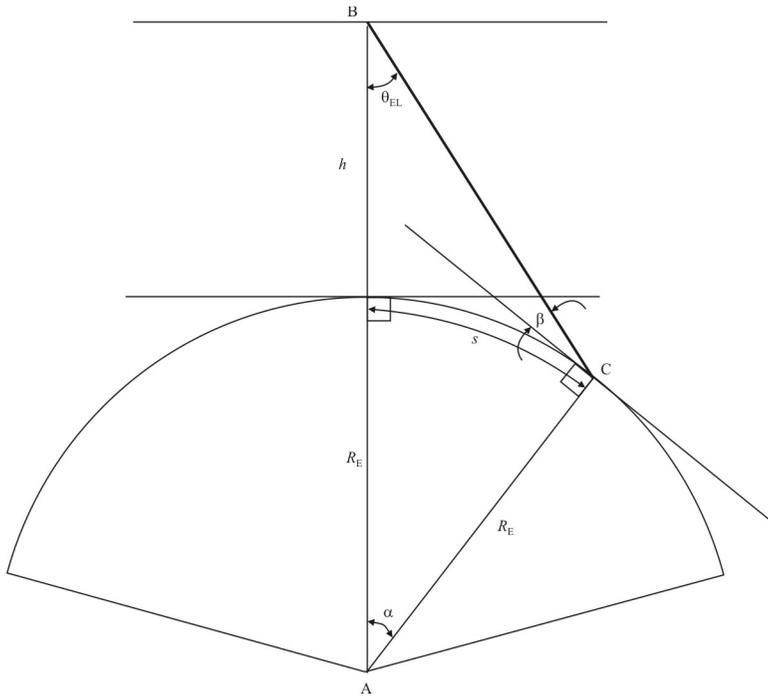
$$\tau_{ii} = 0 \quad \forall i \quad (7.42)$$

and the paths are characterized by

$$A_1 = \sum_{i=1}^p V_i e^{-j2\pi f \tau_{i1}} \quad (7.43)$$

and

$$A_2 = \sum_{i=1}^p V_i e^{-j2\pi f(\tau_{i1} + \tau_{i2})} \quad (7.44)$$



**Figure 7.11** Effects of the curvature of the Earth's surface.

The amplitude of the  $i$ th path is  $V_i$ , the delay between antennas for path  $i$  is  $\tau_i$ , and the delay between antenna  $i$  and the reference antenna (1) is  $\tau_{i1}$ .

## 7.7 Earth Curvature

The development above assumed that the surface of the Earth was a plane. For those cases where the range is 500 km or more, the curvature of the Earth cannot be ignored for purposes of determining the PF of the target. The geometry is illustrated in Figure 7.11. The distance from the receive sensor to the halfway point in the path is given by the arc  $s$  and

$$s = R_E \alpha \quad (7.45)$$

when  $\alpha$  is measured in radians and  $R_E$  is the radius of the Earth (assumed to be spherical). The totality of the angles around triangle ABC is

$$\alpha + \beta + \theta_{EL} = \frac{\pi}{2} \quad (7.46)$$

expressed in radians. Finally, by the law of sines,

$$\frac{R_E}{\sin \theta_{EL}} = \frac{R_E + h}{\sin(\pi/2 + \beta)} \quad (7.47)$$

Manipulating (7.45) through (7.47) yields

$$s = R_E \left[ \frac{\pi}{2} - \theta_{EL} - \cos^{-1} \frac{(R_E + h) \sin \theta_{EL}}{R_E} \right] \quad (7.48)$$

and  $B = 2s$ .

One of the effects of including the Earth's curvature is to decrease the MUF. This is because the effective distance between the transmitter and receiver is decreased.

The maximum one-hop range is determined by including the curvature. This range is given by

$$d_m = \sqrt{8R_E h_0} \quad (7.49)$$

and the MUF is given by

$$f_{\max} \approx f_c \sqrt{\left( \frac{R_E}{2h_0} \right)} \quad (7.50)$$

where  $h_0$  is the reflection height.

## 7.8 Skywave DF Errors

### 7.8.1 Introduction

Other than instrumental inaccuracies, errors in DF of skywave signals are caused primarily by *lateral deviations* of the estimated equivalent reflection point. Causes

of lateral deviations of the equivalent reflection point are due to ionospheric disturbances such as ionospheric tilts, TIDs, polarization rotation, and the lens-effects such as focusing and defocusing.

Sounding of the ionosphere for skywave DF, especially SSL, is usually accomplished obliquely. That is, directly overhead of the intercept site. Yet the skywave signals are usually propagating from considerable distances from the intercept site. The validity of oblique sounding results depends largely on the state of the ionosphere at the time of the sounding versus when estimating the DF. If the ionosphere is relatively undisturbed then oblique sounding is a useful tool. If it is disturbed that is not likely the case.

### **7.8.2 Magnetic Field Effects**

As mentioned, when the effects of the Earth's magnetic field are considered, the ray path can be significantly different from what is expected. The magnetic field causes the ordinary wave and extraordinary wave to take different paths and that can force the point of virtual reflection to be considerably different from the vertical (oblique) sounding results.

### **7.8.3 Ross Curve**

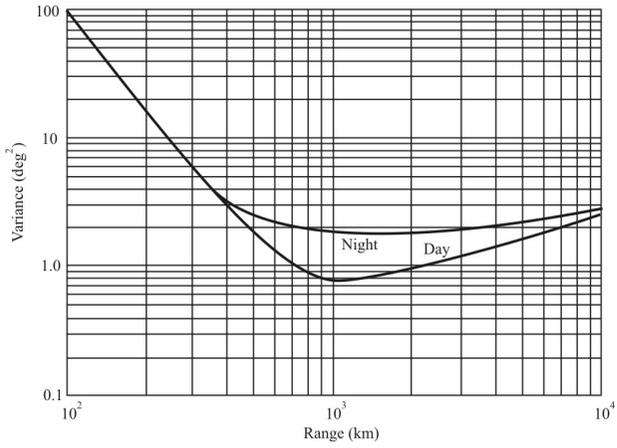
The Ross curve [12] shows the variance in DF bearing as a function of target range, as derived from a large amount of quality data mostly collected during World War II. The variance represents the ionospheric effect on the bearing error due to disturbances such as TIDs. There is a general decrease of the variance with range out to about 1,000 km, followed by an increase attributed to the presence of several propagation modes. A Ross curve is shown in Figure 7.12.

### **7.8.4 Bailey Curve**

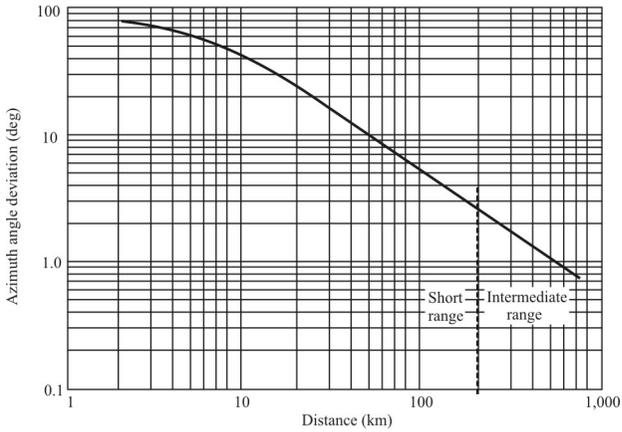
A Bailey curve of azimuth angle estimate deviation is shown in Figure 7.13 [13]. Like the Ross curve, we can see significant errors, well in excess of  $10^\circ$ , especially at short ranges. These lateral deviations change relatively slowly compared to other sources of error such as polarization.

## **7.9 Ray Tracing**

Ionospheric tilt can cause significant errors in PF computation with SSL techniques. If the ionosphere is not accurately modeled as a horizontal plane, the



**Figure 7.12** Ross curve.



**Figure 7.13** Bailey curve for a 5 km lateral displacement of the equivalent reflection point.

measured azimuth will be incorrect, as the effective tilt will cause the signal to be reflected at an incorrect angle.

The ionospheric tilts are caused by horizontal gradients of the refractive index in the ionosphere. This, in turn, is caused by nonuniform ionic densities. The effect is particularly problematic for SSL PF calculations when TIDs occur.

Mathematical models of the ionosphere have been developed in which it is not represented as a flat plane. *Ray tracing* is the appellation applied to utilization of these models, and, technically, the term also applies when the ionosphere is modeled as a plane. Theoretically, if details of the ionospheric structure are known, the ray paths taken by the signal between the transmitter and receiver can be retraced and the PF of the emitter can be determined by the intersection of the multiple paths. Measurement errors and noise, of course, prevent this from happening, and the characteristics of the ionosphere can only be determined to a certain degree of accuracy. Furthermore, in the case of PF on target emitters, the ionospheric height is normally measured at the location of the receiver, and a flat ionospheric plane is assumed. Measuring more details about the ionosphere is usually precluded when the targets in question are hostile. Therefore, modeling is a useful tool for increasing the accuracy of the measurement of the azimuth AOA.

The effects of refraction in a nonanisotropic medium can be accounted for using Fermat's principle of least path time.<sup>2</sup> Fermat's principle dictates that the path traversal time is minimized, which accounts for the apparent bending of a light ray at the interface of two media of different refractive indices. Using the calculus of variations, this can be expressed as

$$\delta \int_{E_1}^{E_2} \mu ds = 0 \quad (7.51)$$

where  $E_1$  and  $E_2$  are the two end points of the path and  $\mu$  is the refractive index of the ionosphere. Converting this to polar coordinates,

$$ds = \sqrt{dR^2 + R^2 d\alpha^2 + dz^2} \quad (7.52)$$

where  $R$  is the distance from the center of the Earth,  $\alpha$  is the great circle distance angle, and  $z$  is the lateral distance. With

$$R' = \frac{dR}{d\alpha} \quad z' = \frac{dz}{d\alpha} \quad L = \mu \sqrt{R'^2 + R^2 + z'^2}$$

<sup>2</sup> An anisotropic medium is one that accounts for the magnetic-ionic interaction effects. The effects of the Earth's magnetic field can be taken into account but are ignored here.

then, from (7.51),

$$\frac{d}{d\alpha} \left( \frac{\partial L}{\partial R'} \right) - \frac{\partial L}{\partial R} = 0 \quad (7.53)$$

and

$$\frac{d}{d\alpha} \left( \frac{\partial L}{\partial z'} \right) - \frac{\partial L}{\partial z} = 0 \quad (7.54)$$

Let

$$Q = \mu R' (R^2 + R'^2 + z'^2)^{-1/2} \quad (7.55)$$

and

$$W = \mu z' (R^2 + R'^2 + z'^2)^{-1/2} \quad (7.56)$$

then, because for path  $P$ ,

$$dP = \mu^{-1} ds \quad (7.57)$$

we get the following

$$\frac{dR}{dP} = Q \quad (7.58)$$

$$\frac{dQ}{dP} = \frac{1}{2} \frac{\partial \mu^2}{\partial R} + \frac{\mu^2 - Q^2 - W^2}{R} \quad (7.59)$$

$$\frac{d\phi}{dP} = \frac{(\mu^2 - Q^2 - W^2)^{1/2}}{R} \quad (7.60)$$

$$\frac{dW}{dP} = \frac{1}{2} \frac{\partial \mu^2}{\partial z} \quad (7.61)$$

$$\frac{dz}{dP} = W \quad (7.62)$$

Equations (7.58) through (7.62) specify the coordinates of a ray given in terms of  $R$ ,  $z$ , and  $\phi$  at every point along  $P$  given that the initial conditions and the refractive index are specified.

The refractive index is a function of the coordinates in the ionosphere and is approximated by

$$\mu(R, z, \alpha) = \left[ 1 - \frac{\beta N(R, z, \alpha)}{f^2} \right]^{1/2} \quad (7.63)$$

where  $N(R, z, \alpha)$  is the electron density (electrons per cubic centimeter),  $f$  is the frequency in megahertz, and  $\beta = 8.05 \times 10^{-5} \text{ MHz}^2 \text{ cm}^3$ .

Assuming that there are three layers in the ionosphere, the E, F1, and F2, then the electron density is a function of the height above the Earth,  $h$ , and is spherically symmetric. Using a cylindrical coordinate system, then,  $N(r, z, \alpha)$  is a function of  $R$  but not of  $z$  nor  $\alpha$ , (7.61) and (7.62) are not needed, and (7.58) to (7.60) reduce to a two-dimensional model.

The electron density is given by [14, 15]

$$N(r_1) = N_0 e^{[1 - r_1 - \sec \chi \exp(-r_1)]/2} \quad (7.64)$$

where  $\chi$  is the zenith angle of the sun relative to the ionosphere, where it has been tacitly assumed that  $dN(R)/dt$  is small. In addition,

$$R_1 = \frac{R - R_0}{H_p} \quad (7.65)$$

where  $R_0$  is the reference height where the production of ions is maximum when the sun is directly overhead.  $H_p$  is the *pressure scale height* given by

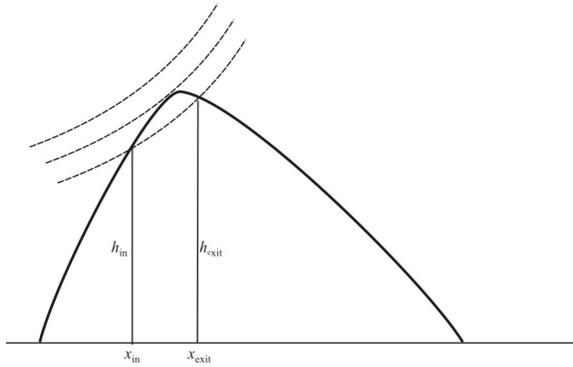
$$H_p = \frac{KT}{\bar{m}g} \quad (7.66)$$

with

$\bar{m}$  = mean molecular mass

$g$  = acceleration of gravity

$T$  = temperature (Kelvins)



**Figure 7.14** A horizontal tilt in the parabolic layer model causes an asymmetric ray path.

The irregular variations of the magnetic field disturbances are given by  $K$ .

When there are lateral gradients in the electron density in the ionosphere, then there is a dependence on altitude  $z$  and the electron density is denoted as  $N(R, z)$ . From (7.63)

$$\frac{\partial \mu^2(R, z)}{\partial R} = -\frac{\beta}{f^2} \frac{\partial N(R, z)}{\partial R} \quad (7.67)$$

and

$$\frac{\partial \mu^2(R, z)}{\partial z} = -\frac{\beta}{f^2} \frac{\partial N(R, z)}{\partial z} \quad (7.68)$$

For a ray that originates at sea level, then the initial conditions are that  $R$  is the approximated radius of the Earth,  $Q = \sin \theta_{EL}$ , and  $W = z = 0$ . This is sufficient information to solve (7.58) through (7.62) numerically.

The electron density in the ionospheric layers can be modeled as parabolas. In that case, several interesting characteristics emerge. An example of this is illustrated in Figure 7.14. In general, the entry and exit angles are not equal. They are, however, related by

$$\sin^2 \phi_e - \sin^2 \phi_i = A(x_e - x_i) \quad (7.69)$$

and

$$\cos^2 \phi_e - \cos^2 \phi_i = -C(z_e^2 - z_i^2) \quad (7.70)$$

where

$$A = -\frac{A_0}{f^2}$$

$$C = \left( \frac{f_c}{fy_m} \right)^2$$

$A_0$  is the gradient of  $(f_0F2)^2$  in the horizontal direction

$$z = h_m - h$$

$y_m$  is the semithickness of the layer

$h_m$  is the height of the maximum electron density

### 7.9.1 Parabolic Modeling

Parabolic mathematical models of the ionospheric layers have proven to be fairly reliable and they can be fit well to the layers with parameters determined from ionograms. Baker and Lambert [16] discuss multiquasiparabolic ionospheric models, initially developed by them [17] and Dyson and Bennett [18]. In that model, the electron density profile is given by

$$N(h) = N_m \left[ 1 \mp \left( \frac{h - h_m}{y_m} \right)^2 \right] \quad (7.71)$$

where  $N$  is the electron density at height  $h$ ,  $N_m$  is the maximum electron density at height  $h_m$ , and  $y_m$ , as above, is the layer semithickness. The minus sign corresponds to a parabolic layer, while the plus sign applies to an inverted parabolic layer. An ionosphere with an E-layer and an F-layer is thus modeled with two normal parabolic layers with an inverted parabolic transition segment between them. The height  $h_{12}$  at which the transition segment 2 blends with layer 3 and the semithickness of that transition segment  $y_{m2}$  are given by

$$h_{12} = -\frac{\left[ y_{m3}^2 \left( \frac{N_{m1}}{N_{m3}} - 1 \right) + h_{m3}^2 - h_{m1} h_{m3} \right]}{h_{m1} - h_{m3}} \quad (7.72)$$

and

$$y_{m2}^2 = \left( \frac{N_{m1}}{N_{m3}} \right) \left( -\frac{h_{t2} - h_{m1}}{h_{t2} - h_{m3}} \right) y_{m3}^2 \quad (7.73)$$

This model is illustrated in Figure 7.15, which compares two ionospheres with the Dudeney model. The two ionospheres are the same except the second one contains a slab of electrons of constant density above  $h_{mE}$ .

## 7.10 Accuracy Comparison of SSL and Triangulation for Ionospherically Propagated Signals

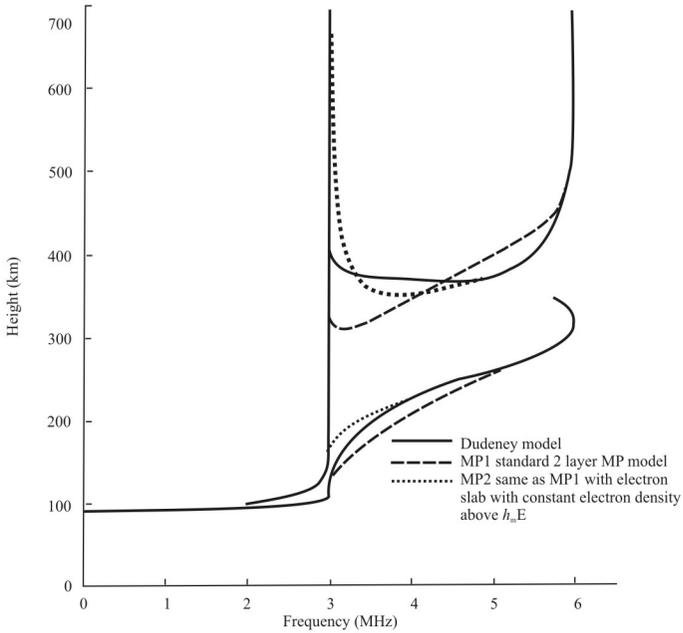
### 7.10.1 Introduction

Horing undertook an investigation of the comparison of the SSL technique to triangulation for long-range propagated HF signals [19]. The model he used for his analysis is shown in Figure 7.16. Since the analysis was about SSL, it was assumed that there was only one hop, so that the skywave signal is reflected only once at the ionosphere and that the projection of the propagating path onto the Earth's surface is a great circle. However, multipath transmissions (reflections at various heights), multiple hops, and deviations from a great circle path are all encountered. At the DF site, we therefore have to expect the multiple incidences of coherent waveforms with a variety of azimuths, elevations, phases, and amplitudes. We have already discussed the deleterious effects of lateral deviations on geolocation of HF emitters, for example. These factors cause azimuth and elevation measurement errors. In addition to the usual sources of fixing errors (equipment error, man-made and natural noise, emitter modulation, adjacent channel interference, and intermodulation), the SSL is also prone to special errors due to anomalies in the ionosphere and to virtual height estimation errors.

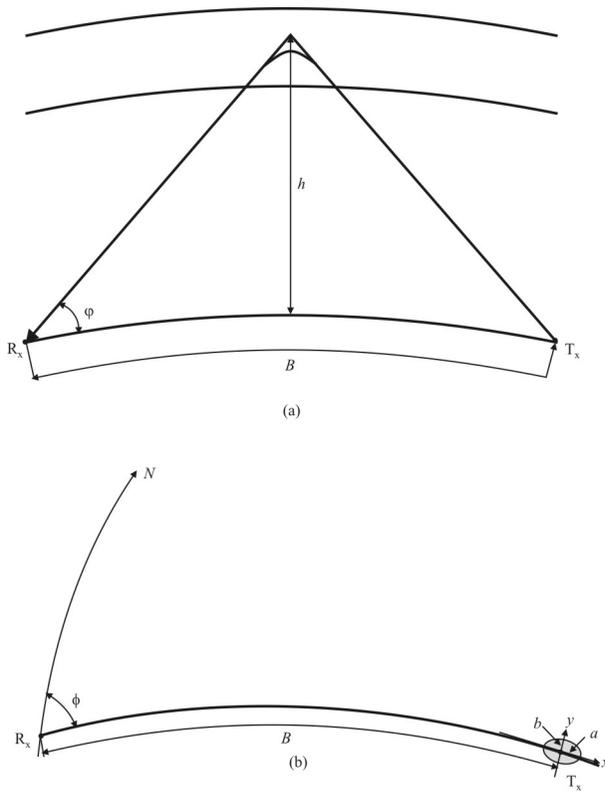
We will derive an approximation for the fixing errors of the SSL and to compare these errors with the fixing errors of a two-site DF triangulation system.

We assume that all the factors that affect the azimuth, elevation, and reflection height give rise to errors that have a mean of zero, a normal distribution, are mutually independent and small. Ideal wave propagation (single hop, great circle LOP) with errors of the type described is assumed. In the neighborhood of the target emitter, the Earth's surface is approximated by a tangential plane.

Under these assumptions, the fixing errors have a two-dimensional, normal distribution with ellipses as curves of constant probability. For the SSL, one of the ellipse's axes lies on the LOP as illustrated in Figure 7.16. These error ellipses,



**Figure 7.15** Plasma frequency profiles (lower curves) and corresponding ionograms (upper curves). In this case  $f_0E = 3$  MHz,  $h_mE = 110$  km,  $y_mE = 20$  km,  $f_0F = 6$  MHz,  $h_mF = 320$  km, and  $y_mF = 100$  km.



**Figure 7.16** SSL geometry.

and the error circle radii that can be derived from them, are used as measures of fix accuracy.

### 7.10.2 Spherical Model

Given the measured elevation angle  $\varphi^3$  and an equivalent reflection height,  $h$ , the estimated fix lies on a great circle. The azimuth is  $\phi$  as measured from north, and the great circle distance between the DF site and the target,  $B$ , is given by

$$B = 2R_E \left[ \cos^{-1} \left( \frac{\cos \varphi}{1 + h/R_E} \right) - \varphi \right] \quad (7.74)$$

where  $R_E = 6,370$  km, the radius of the Earth, which is assumed to be a perfect sphere. All angles are in radians.

#### 7.10.2.1 Effect of Azimuth Errors on the Fix

If the azimuth error is small ( $\Delta\phi \ll 1$ ), at the Earth's surface we can make the following approximation

$$\Delta y \approx \Delta\phi \sin \frac{B}{R_E} \triangleq \Delta\phi R_E A \quad (7.75)$$

The effect of azimuth error is illustrated in Figure 7.17. The effect of azimuth error is to widen the miss distance,  $\Delta y$ , as shown.

#### 7.10.2.2 Effect of Elevation and Height Errors on the Fix

We approximate the nonlinear expression for the arc  $B$ , which is a function of the elevation  $\varphi$  and the height of reflection,  $h$ , using the first term of its Taylor expansion. Doing so we get the following expressions for the fixing errors  $\Delta B_\varphi$ , and  $\Delta B_h$ , where we assume that the elevation and height errors  $\Delta\varphi$  and  $\Delta h$  are small:

---

<sup>3</sup> Notice that this is not the normal definition of elevation angle, which is usually specified relative to zenith, as in Figure 7.6.

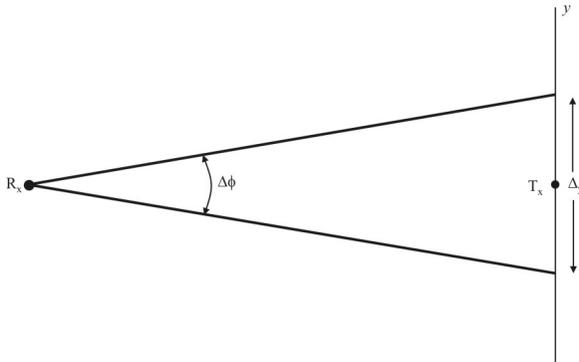


Figure 7.17 Effects of azimuth error.

$$\Delta B_{\phi} \approx \Delta \phi R_E \left\{ 1 - \sin \phi \left[ \left( 1 + \frac{h}{R_E} \right)^2 - \cos^2 \phi \right]^{-1/2} \right\}$$

$$\triangleq \Delta \phi R_E E \quad (7.76)$$

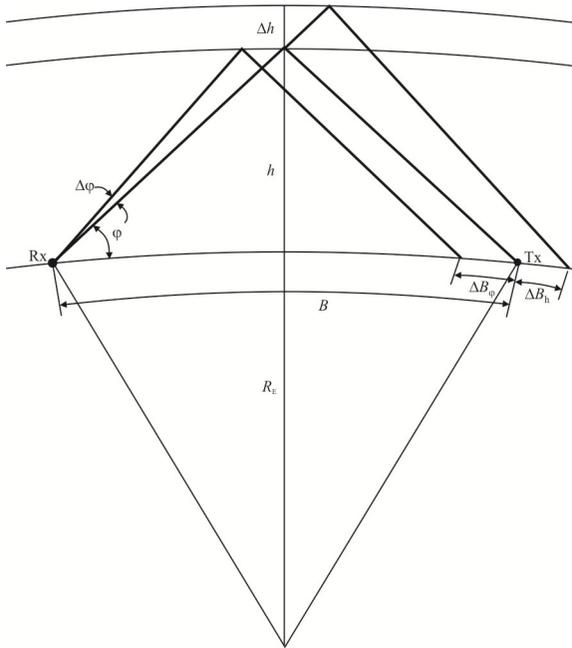
$$\Delta B_h \approx \frac{2\Delta h \cos \phi}{\left( 1 + \frac{h}{R_E} \right) \left[ \left( 1 + \frac{h}{R_E} \right)^2 - \cos^2 \phi \right]^{1/2}}$$

$$\triangleq \frac{\Delta h}{h} R_E H \quad (7.77)$$

The impacts of these measurement errors are illustrated in Figure 7.18. The net effect of both error sources is to influence the measurement of the range to the target.

### 7.10.2.3 Error Ellipse and Error Circle

The standard deviations for the  $x$  and  $y$  directions at the target are obtained from the fix errors given by (7.75) through (7.77) using the formulas below. We assume that the errors have normal distributions with  $\sigma_{\phi}$ ,  $\sigma_{\phi}$ , and  $\sigma_{h,rel} = \sigma_h/h$  as the standard deviations for the azimuth, elevation, and height of reflection, respectively



**Figure 7.18** Effects of elevation.

$$\sigma_y = \sigma_\phi R_E A \quad (7.78)$$

$$\sigma_{x\phi} = \sigma_\phi R_E E \quad (7.79)$$

$$\sigma_{xh} = \sigma_{h,rel} R_E H \quad (7.80)$$

We then can calculate the semiaxes of the error ellipse as

$$a = (\sigma_{x\phi}^2 + \sigma_{xh}^2)^{1/2} \quad (7.81)$$

$$b = \sigma_y \quad (7.82)$$

The factors  $A$ ,  $E$ , and  $H$  in (7.78) – (7.80) are functions of  $\phi$  and *factor E* is considerably larger than  $A$  and  $H$ . The effects of elevation errors is, therefore, particularly heavily weighted and so produces narrow ellipses with a small axis ratio given by

$$\frac{b}{a} = \left[ \frac{A^2 \sigma_\phi^2}{E^2 \sigma_\phi^2 + H^2 \sigma_{h,rel}^2} \right]^{1/2} \quad (7.83)$$

The probability that a fix lies within an error ellipse having semiaxes  $a$  and  $b$  as given by (7.81) and (7.82) is 39.9%. To obtain the semiaxes for an ellipse for another probability,  $P$ , the semiaxes are multiplied by the factor  $[-2 \log_{10}(1-P)]^{1/2}$ .

An error circle can be calculated from the semiaxes of the ellipse producing what is referred to as the CEP. The RMS error  $R_{RMS}$  is

$$R_{RMS} = 0.75 [a^2 + b^2]^{1/2} \quad (7.84)$$

which is accurate to within 10%.

The probability that the fix lies within a circle of radius  $R_{RMS}$  is a function of the semiaxes ratio  $b/a$ . If  $b/a = 0$ , the probability is 68%; for  $b/a = 1$ , the probability is 63%.

For the spherical model, (7.78)–(7.80) yield

$$R_{RMS} = R_E \left( A^2 \sigma_\phi^2 + E^2 \sigma_\phi^2 + H^2 \sigma_{h,rel}^2 \right)^{1/2} \quad (7.85)$$

Figure 7.19 [19] illustrates the RMS error for the special case with  $\sigma_\phi = 1^\circ$ ,  $\sigma_\phi = 3^\circ$ ,  $\sigma_{h,rel} = 5\%$ , and the two heights,  $h = 100$  km (E-layer reflection, and  $h = 300$  km (F-layer reflection). Figure 7.20 [19] shows the axis ratio of the error ellipse for this case.

### 7.10.3 Plane Model

When we assume that the Earth is flat and that the LOPs are straight lines then (7.81) and (7.82) simplify to

$$b = d \sigma_\phi \quad (7.86)$$

$$a = d \left[ \sigma_{h,rel}^2 + 4 \sigma_\phi^2 \left( \frac{h}{d} + \frac{d}{4h} \right)^2 \right]^{1/2} \quad (7.87)$$

where  $d$  is the distance measured in the plane between the target and the receiver.

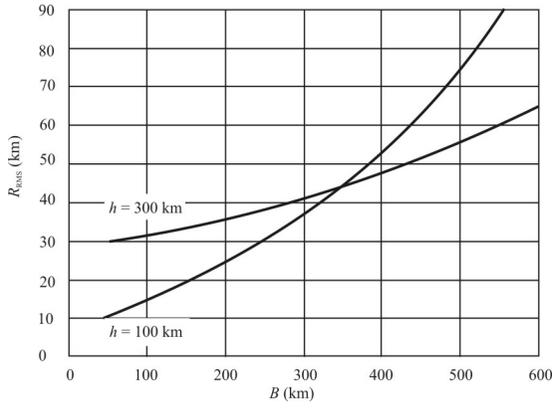


Figure 7.19 RMS error.

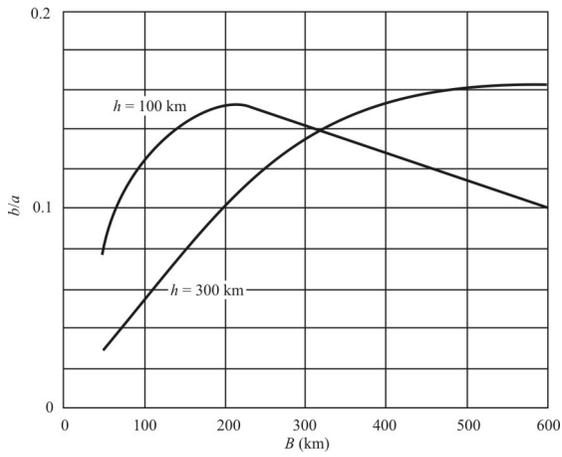
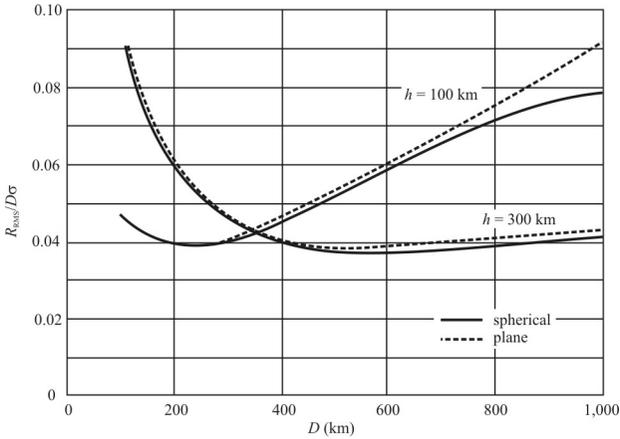


Figure 7.20  $b/a$  special case.



**Figure 7.21** Normalized RMS.

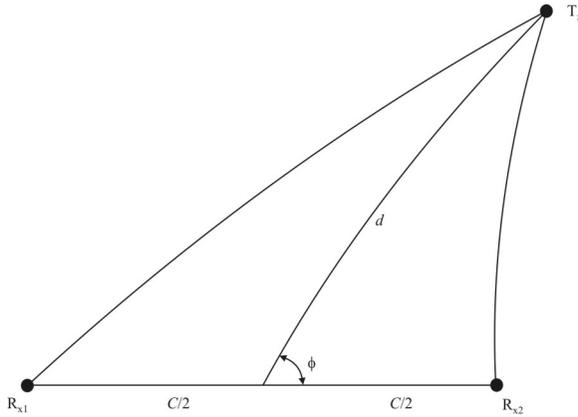
Figure 7.21 [19] shows the RMS error as a function of the plane model separation  $D = d$ , (7.86), (7.87), and (7.84), assuming the azimuth and the elevation errors are equal and that the height error is zero ( $\sigma_\phi = \sigma_\theta = \sigma, \sigma_{h,rel} = 0$ ). Figure 7.22 also shows the RMS error as a function of the spherical model separation  $D = B$ . We can see that, up to separations of about 800 km, the plane model is a satisfactory approximation.

#### 7.10.4 Comparison between SSL and Triangulation

For SSL, the radius of the error circle depends on the virtual height of reflection and its relative estimation error, the distance of the intercept site from the target, and the standard deviations of the azimuth error and the elevation error. For triangulation, the radius of the error circle depends on the standard deviations of the azimuth errors of the two DF sites, the triangulation baseline and the position of the emitter site relative to the baseline.

For comparison purposes we make the following simplifications:

- The standard deviations of all angular errors are the same (both azimuth angles for horizontal triangulation, and the azimuth and the elevation for SSL).
- For SSL, the virtual height error is negligible (this is a somewhat dubious assumption but necessary for comparison purposes).
- The plane model is used.



**Figure 7.22** Triangulation geometry.

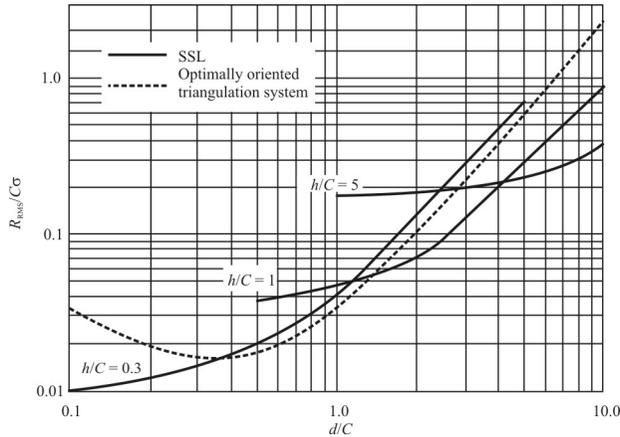
In the case of triangulation, the RMS error, which is normalized by dividing by the product of the baseline  $C$  and the standard deviation  $\sigma$  in degrees, depends on the polar coordinates constructed on the baseline,  $d/C$ , and  $\phi$  of the LOP intersection (see Figure 7.22).

$$\frac{R_{\text{RMS,tri}}}{C\sigma} = \frac{\pi}{180} \frac{\left\{ 2 \left[ \left( \frac{d}{C} \right)^2 + \frac{1}{4} \right] \left[ \left( \left( \frac{d}{C} \right)^2 - \frac{1}{4} \right)^2 + \left( \frac{d}{C} \right)^2 \sin^2 \phi \right] \right\}^{1/2}}{\frac{d}{C} \sin \phi} \quad (7.88)$$

So that a direct comparison of  $R_{\text{RMS,SSL}}$  can be made with  $R_{\text{RMS,tri}}$ , the SSL error circle radius will also be normalized by dividing by  $C$ , the baseline of the triangulation system which is to be used as a comparison. Equations (7.84), (7.86), and (7.87) yield

$$\frac{R_{\text{RMS,SSL}}}{C\sigma} = \frac{\pi}{180} \frac{d}{C} \left[ 1 + 4 \left( \frac{h/C}{d/C} + \frac{1}{4} \frac{d/C}{h/C} \right)^2 \right]^{1/2} \quad (7.89)$$

If the propagation conditions do not depend on the angle of incidence, the curves of constant  $R_{\text{RMS,SSL}}$  are, from (7.89), simply concentric circles about the receive site.



**Figure 7.23** Normalized RMS error.

When (7.88) is solved for  $d/C$  or  $\sin\phi$ , curves of constant  $R_{RMS,tri}$  are obtained. Errors are large in this case if  $|\sin\phi| \ll 1$ , which occurs when the target is very near the extended baseline defined by the two receivers. This is the familiar GDOP effect. As expected, the fix error is a minimum on the perpendicular bisector of the baseline ( $|\sin\phi| = 1$ ). In other words, the alignment of the triangulation system is optimal when the emitter is on this perpendicular.

Figure 7.23 illustrates comparison of the SSL RMS error with the triangulation error of an optimally aligned triangulation system. We note that the assumptions underlying the theoretical treatment of SSL errors at distances less than 50 km are not very realistic. At these distances, the ground wave and skywave can arrive with about the same amplitude and, because the elevation angle is so great, the effective aperture of the DF antenna system is greatly reduced. Errors due to propagation conditions then have an appreciably greater effect.

We can conclude from Figure 7.23:

- $h/C = 0.3$  ( $h = 100$  km,  $C = 333$  km): The SSL is comparable with a triangulation system having a very large baseline. At long ranges, error circle radii of the same order of magnitude can be obtained with both methods. At shorter ranges, a triangulation system is not very effective due to the obtuse angles of intersection.
- $h/C = 1$  ( $h = 100$  km,  $C = 100$  km): Triangulation is somewhat better for short ranges, whereas SSL is better over long ranges.

- $h/C = 5$  ( $h = 100$  km,  $C = 20$  km): Triangulation is superior at small distances. As we mentioned previously, SSLs cannot be used at distances less than 50 km. At long ranges, triangulation gives very acute angles of intersection so that the SSL, in this case, is far superior.

A field test was conducted at 7.5 and 7.8 MHz over a range of 138 km to compare actual, experimental results with the theoretical calculations. The results were quite comparable. The RMS error ranged from 12.5 km to 29 km, which is in line with approximately 10% of range, the expected result.

### 7.10.5 Summary

We can say from this analysis that the SSLs are better for long-range position fixing if a very large triangulation baseline cannot be established [if the simplifying assumption of equal angle errors is valid and the height of reflection error can be neglected (a bold assumption at best)]. If the elevation errors are considerably larger than the azimuth errors, and there is insufficient information about the height of the reflecting ionospheric layer, long-range triangulation carried out using even moderately sized baselines can be superior.

Empirical fixing-error data, obtained from an HF Doppler SSL, were in good agreement with the calculated values.

## 7.11 Concluding Remarks

Some methods for geolocating HF targets from a single sensor site have been discussed in this chapter. While triangulation and quadratic processing methods apply to HF targets as well as higher frequency targets, in some cases it may be advantageous to apply only a single sensor. Such techniques typically result in lower accuracy geolocations because of the necessity of normally requiring knowledge of the height of the ionosphere. Measuring this height with EW systems in a noncooperative fashion is inherently inaccurate because of the assumption of uniform height of the reflecting layer above the Earth's surface.

In addition to their utility in EW applications, SSL EW systems have been applied to ionospheric research projects. Such projects endeavor to characterize the ionosphere, allowing for scientific study of the atmosphere.

## References

- [1] Davies, K., *Ionospheric Radio*, London: Peter Peregrinus, 1989, p. 96.

- [2] Davies, K., *Ionospheric Radio*, London: Peter Peregrinus, 1989, pp. 164, 165.
- [3] Gething, P. J. D., *Radio Direction Finding and Superresolution*, 2nd ed., London: Peter Peregrinus Ltd., 1978, pp. 262–267.
- [4] Davies, K., *Ionospheric Radio*, London: Peter Peregrinus, 1989, p. 158.
- [5] Knapp, C. H., and G. C. Carter, “The Generalized Correlation Method for Estimation of Time Delay,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-24, No. 4, August 1976, pp. 320–327.
- [6] Benesty, J., “Adaptive Eigenvalue Decomposition Algorithm for Passive Acoustic Source Localization,” *Journal of the Acoustic Society of America*, Vol. 107, No. 1, January 2000, pp. 384–391.
- [7] Childers, D. G., D. P. Skinner, and R. C. Kemerait, “The Cepstrum: A Guide to Processing,” *Proceedings of the IEEE*, Vol. 65, No. 10, October 1977, pp. 1428–1443.
- [8] Eken, F., F. Atmaca, and E. Hepsaydir, “HF Multipath Dispersion Measurements Using Cepstral Signal Processing,” *Radio Science*, Vol. 26, No. 1, January–February 1991, pp. 15–21.
- [9] Roberts, L. R., *Signal Processing Techniques*, Anaheim, CA: Interstate Electronics Corporation, 1981, pp. 9-19–9-21.
- [10] Oppenheim, A. V., and R. W. Schaffer, *Digital Signal Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1975, Section 10.7.
- [11] Johnson, R. L., Q. R. Black, and A. G. Sonsteby, “HF Multipath Passive Single Site Radio Location,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 30, No. 2, April 1994, pp. 462–470.
- [12] Ross, W., “The Estimation of the Probable Accuracy of High-Frequency Direction-Finding Bearings,” *Journal of the Institute of Electrical Engineers*, Vol. 94, 1947, p. 722.
- [13] Bailey, A. D., “HF Direction-of-Arrival Studies over a Medium Range Path (452 km),” *Proceedings Conference on HF Radio Propagation*, Urbana, IL: University of Illinois, June 2–4, 1969, pp. 13–30.
- [14] Coleman, C. J., “On the Simulation of Backscatter Ionograms,” *Journal of Atmospheric and Solar-Terrestrial Physics*, Vol. 59, 1997, pp. 2089–2099.
- [15] Davies, K., *Ionospheric Radio*, London: Peter Peregrinus, 1989, p. 67.
- [16] Baker, D. C., and S. Lambert, “Range Estimation for SSL HFDF Systems by Means of a Multi-quasiparabolic Ionospheric Model,” *Proceedings of the IEE*, Vol. 136, Pt. H., No. 2, April 1989, pp. 120–127.
- [17] Baker, D. C., and S. Lambert, “Multi-Parabolic Ionospheric Model for SSL Application,” *Electronic Letters*, 1988, 24, pp. 425–426.
- [18] Dyson, P. L., and J. A. Bennett, “A Model of the Vertical Distribution of the Electron Concentration in the Ionosphere and Its Application to Oblique Propagation Studies,” *Journal of Atmospheric Terrestrial Physics*, Vol. 50, 1988, pp. 251–262.
- [19] Horing, H. C., “Comparison of the Fixing Accuracy of Single-Station Locators and Triangulation Systems Assuming Ideal Shortwave Propagation in the Ionosphere,” *Proceedings IEE*, Vol. 137, Part F, No. 3, June 1990, pp. 173–176.

# Appendix A

## Grassmann Algebra

### A.1 Background

Hermann Gunther Grassmann was born in 1809 near the border of Germany and Poland in a town called Stettin. He discovered an algebra of geometry that includes vectors. His algebra subsumes the common form of algebra that we all learn in high school. During his lifetime he received very little recognition for his discovery. William Kingdon discovered Grassmann's work about 1877, the year Grassmann died. Clifford's algebras are still used today for manipulation of vectors and linear mathematical systems.

A brief introduction to Grassmann algebra is presented in this appendix. The information included here is based on [1]. For those desiring more detailed information references [2–21] are recommended.

### A.2 Introduction

Grassmann's algebra is a theory of "extension," with the fundamental product operation being the exterior product. In this algebra, addition, negation, and multiplication by a scalar carry the same meaning as in the normal mathematics we are all familiar with. The product operation, however, is different.

The constructs of this algebra allow manipulation of geometric objects algebraically. That is, lines can be constructed from points, and lines can be manipulated. Planes can be constructed from lines and/or points and these planes can be manipulated algebraically. Higher forms of geometric shapes can be constructed and manipulated just as easily. Such manipulation includes functions such as finding the intersection of two or more geometric objects.

The common 3-D vectors are one form of geometric shapes that can be built in the algebra. Vectors such as these, of course, have a direction as well as

magnitude. However, that is not enough to completely specify a vector. It is also necessary to know where the vector is located in the 3-D space to determine the effects. For example, if the vector represents a force exerted on an object, it is necessary to know where on the object the force is exerted, yet there is no information inherent in the specification of the vector to convey this information. Grassmann algebra solves this problem as part of the specification of the vector. All the necessary information about the force is part of the vector.

The concept of direction associated with vectors is included as part of the specification of planes and higher-order objects. Thus, a plane has a “direction.” On the other hand, the concept of magnitude associated with vectors is not necessary for the manipulation of geometric objects, although it can be included if desired. This lack of magnitude is manifest in the lack of necessity of specifying a metric in the algebra, which is necessary in the familiar algebra of vectors.

### A.3 Exterior Product

Grassmann algebra is a mathematical system that describes and manipulates physical entities. It is a linear or vector algebra such that the product operation facilitates the notion of linear dependence. The product operation is normally called the *exterior product* or *wedge product* and is denoted by  $\wedge$ . Linear dependence is encapsulated in the notion that if vectors  $\vec{x}_1, \vec{x}_2, \dots$  are linearly dependent, then their exterior product is zero:

$$\vec{x}_1 \wedge \vec{x}_2 \wedge \dots \wedge \vec{x}_N = 0 \quad (\text{A.1})$$

On the other hand, if the vectors are linearly independent, then their exterior product is nonzero.

A vector  $\vec{x}_1$  can be interpreted as a direction in a linear space. If that space has a metric,<sup>1</sup> the  $|\vec{x}_1|$ , the magnitude of  $\vec{x}_1$ , is its length. The exterior product extends this vector to higher dimensions. The wedge product of two vectors,  $\vec{x}_1 \wedge \vec{x}_2$ , is called the *bivector* and can be thought of as the analog of direction to two-space,

---

<sup>1</sup> A metric is a generalization of distance. The *metric* for a space consisting of vectors  $\vec{x}_1, \vec{x}_2$ , and so on, is a function  $d$  such that:

- (1)  $d(\vec{x}_1, \vec{x}_2) = 0$  if and only if  $\vec{x}_1 = \vec{x}_2$ ;
- (2)  $d(\vec{x}_1, \vec{x}_2) = d(\vec{x}_2, \vec{x}_1)$ ;
- (3)  $d(\vec{x}_1, \vec{x}_2) \leq d(\vec{x}_1, \vec{x}_3) + d(\vec{x}_3, \vec{x}_2)$ .

that is, the *planar direction*. If this two-space has a metric, then the magnitude of  $\vec{x}_1 \wedge \vec{x}_2$  is its area. This can be extended to any number of dimensions.

Lines and planes are but examples of higher-order constructs that originate with points. A line is represented by the exterior product of any two points on it, or by any point on it with a vector parallel to it. A plane is the exterior product of any three points on it, any two points with a vector parallel to it, or any point on it with a trivector parallel to it.

The real algebra familiar to all of us as learned in high school is subsumed by Grassmann algebra. In that case the exterior product reduces to the normal product.

### A.3.1 Properties of the Exterior Product

Given the exterior product of two vectors  $\vec{x}_1$  and  $\vec{x}_2$ ,  $\vec{x}_1 \wedge \vec{x}_2$  is the bivector corresponding to them. This bivector is not a vector and therefore does not belong to the vector space in which  $\vec{x}_1$  and  $\vec{x}_2$  lie. The set of bivectors constructed with the vectors in the vector space in fact forms its own linear space.

The exterior product exhibits *antisymmetry* (or anticommutativity). That is,

$$\vec{x}_1 \wedge \vec{x}_2 = -\vec{x}_2 \wedge \vec{x}_1 \quad (\text{A.2})$$

Therefore, we deduce that

$$\begin{aligned} \vec{x}_1 \wedge \vec{x}_1 &= -\vec{x}_1 \wedge \vec{x}_1 \\ \vec{x}_1 \wedge \vec{x}_1 + \vec{x}_1 \wedge \vec{x}_1 &= 0 \\ 2(\vec{x}_1 \wedge \vec{x}_1) &= 0 \\ \vec{x}_1 \wedge \vec{x}_1 &= 0 \end{aligned} \quad (\text{A.3})$$

which indicates that  $\vec{x}_1$  is linearly dependent on itself (the coefficient is 1; that is,  $\vec{x}_1 = 1\vec{x}_1$ ).

#### Properties

1. The exterior product is associative,

$$(\vec{x}_1 \wedge \vec{x}_2) \wedge \vec{x}_3 = \vec{x}_1 \wedge (\vec{x}_2 \wedge \vec{x}_3)$$

2. The exterior product is distributive,

$$\vec{x}_1 \wedge (\vec{x}_2 + \vec{x}_3) = \vec{x}_1 \wedge \vec{x}_2 + \vec{x}_1 \wedge \vec{x}_3$$

3. If  $a$  and  $b$  are any scalars, then

$$(a + b) \wedge \vec{x}_1 = a\vec{x}_1 \wedge b\vec{x}_1$$

$$a\vec{x}_1 \wedge \vec{x}_2 = a(\vec{x}_1 \wedge \vec{x}_2)$$

4. Let  $\vec{e}_1, \vec{e}_2$ , and  $\vec{e}_3$  be three basis vectors in 3-D space. Then any vectors can be expressed in terms of these vectors as

$$\vec{x}_1 = a_1\vec{e}_1 + a_2\vec{e}_2 + a_3\vec{e}_3 \quad (\text{A.4})$$

$$\vec{x}_2 = b_1\vec{e}_1 + b_2\vec{e}_2 + b_3\vec{e}_3 \quad (\text{A.5})$$

where  $a_i$  and  $b_i$  are appropriate scalars. The bivector formed from  $\vec{x}_1$  and  $\vec{x}_2$  is given as

$$\begin{aligned} \vec{x}_1 \wedge \vec{x}_2 &= (a_1\vec{e}_1 + a_2\vec{e}_2 + a_3\vec{e}_3) \wedge (b_1\vec{e}_1 + b_2\vec{e}_2 + b_3\vec{e}_3) \\ &= a_1\vec{e}_1 \wedge b_1\vec{e}_1 + a_1\vec{e}_1 \wedge b_2\vec{e}_2 + a_1\vec{e}_1 \wedge b_3\vec{e}_3 \\ &\quad + a_2\vec{e}_2 \wedge b_1\vec{e}_1 + a_2\vec{e}_2 \wedge b_2\vec{e}_2 + a_2\vec{e}_2 \wedge b_3\vec{e}_3 \\ &\quad + a_3\vec{e}_3 \wedge b_1\vec{e}_1 + a_3\vec{e}_3 \wedge b_2\vec{e}_2 + a_3\vec{e}_3 \wedge b_3\vec{e}_3 \\ &= a_1b_1\vec{e}_1 \wedge \vec{e}_1 + a_1b_2\vec{e}_1 \wedge \vec{e}_2 + a_1b_3\vec{e}_1 \wedge \vec{e}_3 \\ &\quad + a_2b_1\vec{e}_2 \wedge \vec{e}_1 + a_2b_2\vec{e}_2 \wedge \vec{e}_2 + a_2b_3\vec{e}_2 \wedge \vec{e}_3 \\ &\quad + a_3b_1\vec{e}_3 \wedge \vec{e}_1 + a_3b_2\vec{e}_3 \wedge \vec{e}_2 + a_3b_3\vec{e}_3 \wedge \vec{e}_3 \\ &= (a_1b_2 - a_2b_1)\vec{e}_1 \wedge \vec{e}_2 \\ &\quad + (a_2b_3 - a_3b_2)\vec{e}_2 \wedge \vec{e}_3 + (a_3b_1 - a_1b_3)\vec{e}_3 \wedge \vec{e}_1 \end{aligned} \quad (\text{A.6})$$

The coefficients in (A.6) are those obtained by the normal cross product of  $\vec{x}_1$  and  $\vec{x}_2$ . However, the bivector space does not require a metric. The cross-product generates a vector orthogonal to  $\mathbf{x}_1$  and  $\mathbf{x}_2$  and therefore assumes the existence of a metric. In addition, the cross-product applies to two vectors in 3-D space. The exterior product formulation presented here applies to vector spaces of any, arbitrary dimension. For example, the trivector derived just as (A.6) was derived yields

$$\vec{x}_1 \wedge \vec{x}_2 \wedge \vec{x}_3 = (a_1 b_2 c_3 - a_3 b_2 c_1 + a_2 b_3 c_1 + a_3 b_1 c_2 - a_2 b_3 c_2 - a_2 b_1 c_3) \vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3 \tag{A.7}$$

5. A trivector such as (A.7) has a single component ( $\vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3$ ) whose coefficient is the determinant of the constituent vectors given by the term

$$(a_1 b_2 c_3 - a_3 b_2 c_1 + a_2 b_3 c_1 + a_3 b_1 c_2 - a_2 b_3 c_2 - a_2 b_1 c_3)$$

In any metric space, such as considered here, this coefficient is the volume of the 3-D parallelepiped formed from the constituent vectors. Note that if  $x_1$ ,  $x_2$ , and  $x_3$  are lying in a plane, then they are dependent and the trivector is zero, indicating that the volume is zero.

### A.3.2 *m*-Vectors

The exterior product of  $m$  1-vectors is called an *m-vector*. The number  $m$  is referred to as the *grade* of the *m*-vector. It is common practice to annotate an *m*-vector with its grade as an underscript. Thus,  $\vec{x}_1 \wedge \vec{x}_2 \wedge \vec{x}_3 \wedge \vec{x}_4$  is of grade 4, which is denoted  $\underset{4}{\gamma} = \vec{x}_1 \wedge \vec{x}_2 \wedge \vec{x}_3 \wedge \vec{x}_4$ . A 1-vector is usually not endowed with the underscript. The grade of a scalar is zero. With the dimension of the linear space denoted by  $N$ , the complementary grade of an *m*-vector is  $N - m$ .

**Property: Interchanging Order.** Interchanging the order of any factors in the exterior product, adjacent or otherwise, changes the sign of the product:

$$\dots \wedge \vec{x}_i \wedge \dots \wedge \vec{x}_j \dots = -(\dots \wedge \vec{x}_j \wedge \dots \wedge \vec{x}_i \dots) \tag{A.8}$$

**Property: Grade of *m*-Vectors.** The exterior product of an *m*-vector  $\underset{m}{\vec{\alpha}}$  and *k*-vector  $\underset{k}{\vec{\beta}}$  is an  $(m + k)$  vector:

$$\underset{m}{\vec{\alpha}} \wedge \underset{k}{\vec{\beta}} = \underset{m+k}{\vec{\gamma}} \tag{A.9}$$

**Property: Identity Element.** The *identity* for the exterior product is the unit scalar:

$$\vec{\alpha}_m = 1 \wedge \vec{\alpha}_m = \alpha_m \wedge 1 \quad (\text{A.10})$$

**Property: Scalar Factors.** Scalars factor out of products

$$\left( a \vec{\alpha}_m \right) \wedge \vec{\beta}_k = \vec{\alpha}_m \wedge \left( a \vec{\beta}_k \right) = a \left( \vec{\alpha}_m \wedge \vec{\beta}_k \right) \quad (\text{A.11})$$

**Property: Anticommutative.** An exterior product is anticommutative whenever the grades of both vectors is odd:

$$\vec{\alpha}_m \wedge \vec{\beta}_k = (-1)^{mk} \vec{\beta}_k \wedge \vec{\alpha}_m \quad (\text{A.12})$$

It is only vectors of an odd grade that are antisymmetric. If at least one of the vectors of an exterior product has an even grade, then the product commutes—that is, it is symmetric:

$$\left( \vec{x}_1 \wedge \vec{x}_2 \right) \wedge \vec{x}_3 = \vec{x}_3 \wedge \left( \vec{x}_1 \wedge \vec{x}_2 \right) \quad (\text{A.13})$$

**Property: Distributivity.** The exterior product is both left and right distributive under addition:

$$\left( \vec{\alpha}_m + \vec{\beta}_m \right) \wedge \vec{\gamma}_k = \vec{\alpha}_m \wedge \vec{\gamma}_k + \vec{\beta}_m \wedge \vec{\gamma}_k \quad \vec{\alpha}_m \wedge \left( \vec{\beta}_k + \vec{\gamma}_k \right) = \vec{\alpha}_m \wedge \vec{\beta}_k + \vec{\alpha}_m \wedge \vec{\gamma}_k \quad (\text{A.14})$$

**Property: Associativity.** The exterior product is *associative*:

$$\left( \vec{\alpha}_m \wedge \vec{\beta}_k \right) \wedge \vec{\gamma}_n = \vec{\alpha}_m \wedge \left( \vec{\beta}_k \wedge \vec{\gamma}_n \right) \quad (\text{A.15})$$

## A.4 Regressive Product

The regressive product can be thought of as a *dual product* to the exterior product. It is used to find the intersection of geometrical objects. The regressive product is denoted by  $\vee$ .

### A.4.1 Unions and Intersections of Spaces

Any given 1-element  $\vec{z}$  is in the subspace spanned by  $\vec{x}$  and  $\vec{y}$  if the exterior product of (nonzero)  $\vec{x} \wedge \vec{y}$  with  $\vec{z}$  is zero. Thus,  $\vec{x} \wedge \vec{y}$  is an element that can be used to *define* the subspace instead of the individual 1-elements  $\vec{x}$  and  $\vec{y}$ .

Thus, the *space* of  $\vec{x} \wedge \vec{y}$  as the set of all 1-vectors  $\vec{z}$  such that  $\vec{x} \wedge \vec{y} \wedge \vec{z} = \vec{0}$ . This can be extended to more general elements by defining the space of a simple element  $\vec{\alpha}_m$  as the set of all 1-elements  $\beta$  such that  $\vec{\alpha}_m \wedge \beta = \vec{0}$ .

Two elements are *congruent* if one is a scalar factor times the other. In a 3-D vector space, this is equivalent to saying that two vectors are congruent if they point in the same direction. The two vectors need not have the same magnitude nor the same point of origin. In fact, for general linear spaces, magnitude as a concept may not be necessary—it only becomes necessary when the space has a metric defined. The concepts of union and intersection only make sense up to congruence. Congruent elements define the same subspace.

A *union* of elements is an element defining the subspace they span *together*. It is similar in concept to the common union of sets. If  $A$  and  $B$  denote two not necessarily distinct sets of objects, then the union of  $A$  and  $B$ , usually denoted by  $A \cup B$ , is the set comprised of elements from  $A$  or elements from  $B$ . An *intersection* of elements is an element defining the subspace they span *in common*. Again, from the concepts of sets, the intersection of  $A$  and  $B$ , denoted by  $A \cap B$ , is defined as the set comprised of elements from  $A$  and  $B$ . See Figure A.1.

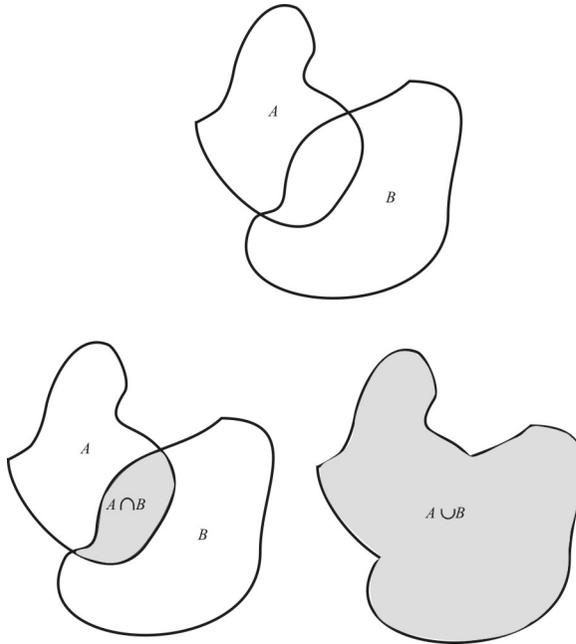
Given three independent 1-elements,  $\vec{x}$ ,  $\vec{y}$ , and  $\vec{z}$ , then a union of  $\vec{x} \wedge \vec{y}$  and  $\vec{y} \wedge \vec{z}$  is any element congruent to  $\vec{x} \wedge \vec{y} \wedge \vec{z}$ . An intersection of  $\vec{x} \wedge \vec{y}$  and  $\vec{y} \wedge \vec{z}$  is any element congruent to  $\vec{y}$ .

### A.4.2 Properties of the Regressive Product

The regressive product of an  $m$ -element and a  $k$ -element in an  $N$ -space is an  $(m + k - N)$ -element.

**Property: Associative.** The regressive product is associative:

$$\left( \vec{\alpha}_m \vee \vec{\beta}_k \right) \vee \vec{\gamma}_n = \vec{\alpha}_m \vee \left( \vec{\beta}_k \vee \vec{\gamma}_n \right) \quad (\text{A.16})$$



**Figure A.1** Union and intersection of sets.

**Property: Unit Element.** The unit  $n$ -element  $\vec{1}_N$  is the identity under the regressive product:

$$\vec{\alpha}_m = \vec{1}_N \vee_m \vec{\alpha}_m = \vec{\alpha}_m \vee_m \vec{1}_N \quad (\text{A.17})$$

**Property: Scalar Factors.** Scalars factor out of products.

$$\left( a \vec{\alpha}_m \right) \vee_k \vec{\beta}_k = \vec{\alpha}_m \vee_m \left( a \vec{\beta}_k \right) = a \left( \vec{\alpha}_m \vee_m \vec{\beta}_k \right) \quad (\text{A.18})$$

**Property: Anticommutative.** A regressive product is anticommutative whenever the complementary grades of the factors are both odd:

$$\vec{\alpha}_m \vee_k \vec{\beta}_k = (-1)^{(N-m)(N-k)} \vec{\beta}_k \vee_m \vec{\alpha}_m \quad (\text{A.19})$$

**Property: Distributive.** The regressive product is both left and right distributive under addition:

$$\left( \vec{\alpha}_m + \vec{\beta}_m \right) \vee_k \vec{\gamma}_k = \vec{\alpha}_m \vee_m \vec{\gamma}_k + \vec{\beta}_m \vee_m \vec{\gamma}_k \quad \vec{\alpha}_m \vee_m \left( \vec{\beta}_k + \vec{\gamma}_k \right) = \vec{\alpha}_m \vee_m \vec{\beta}_k + \vec{\alpha}_m \vee_m \vec{\gamma}_k \quad (\text{A.20})$$

### A.4.3 The Common Factor Axiom

Assume that  $\vec{x}$ ,  $\vec{y}$ , and  $\vec{z}$  are three independent vectors in a vector three-space. The *common factor axiom* states that the regressive product of the two bivectors  $\vec{x} \wedge \vec{z}$  and  $\vec{y} \wedge \vec{z}$  may also be expressed as the regressive product of a three-element  $\vec{x} \wedge \vec{y} \wedge \vec{z}$  with  $\vec{z}$ :

$$(\vec{x} \wedge \vec{z}) \vee (\vec{y} \wedge \vec{z}) = (\vec{x} \wedge \vec{y} \wedge \vec{z}) \vee \vec{z} \quad (\text{A.21})$$

Since the space is 3-D, we can write any three-element such as  $\vec{x} \wedge \vec{y} \wedge \vec{z}$  as a scalar factor times the unit three-element:

$$(\vec{x} \wedge \vec{y} \wedge \vec{z}) \vee \vec{z} = \left( a \vec{1}_3 \right) \vee \vec{z} = a \vec{z} \quad (\text{A.22})$$

Thus, the regressive product of two elements possessing an element in common is congruent to that element. That is, the regressive product of two elements defines their intersection:

$$(\vec{x} \wedge \vec{z}) \vee (\vec{y} \wedge \vec{z}) \equiv \vec{z} \quad (\text{A.23})$$

Let  $\vec{\alpha}_m, \vec{\beta}_k,$  and  $\vec{\gamma}_n$  be *simple* elements with  $m + k + n = N$ , where  $N$  is the dimension of the space. Then the common factor axiom states that

$$\left( \vec{\alpha}_m \wedge \vec{\gamma}_n \right) \vee \left( \vec{\beta}_k \wedge \vec{\gamma}_n \right) = \left( \vec{\alpha}_m \wedge \vec{\beta}_k \wedge \vec{\gamma}_n \right) \vee \vec{\gamma}_n \quad m + k + n = N \quad (\text{A.24})$$

When  $n$  is zero, the common factor axiom shows that the regressive product of an  $m$ -element with an  $(n - m)$ -element is a scalar that can be expressed as a regressive product with the unit  $\vec{1}$ .

$$\vec{\alpha}_m \vee \vec{\beta}_k = \left( \vec{\alpha}_m \wedge \vec{\beta}_k \right) \vee \vec{1}$$

#### A.4.4 The Common Factor Theorem

Suppose that  $\vec{x} \wedge \vec{y}$  and  $\vec{u} \wedge \vec{v}$  are noncongruent bivectors in a 3-D space. The regressive product of  $\vec{x} \wedge \vec{y}$  and  $\vec{u} \wedge \vec{v}$  is given by

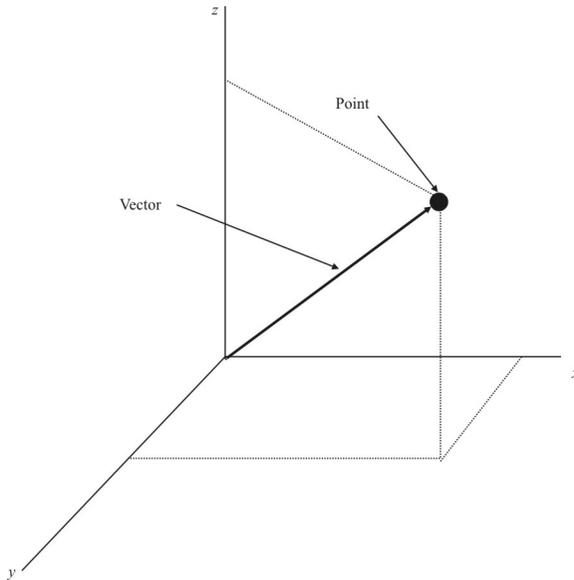
$$\vec{z} = (\vec{x} \wedge \vec{y}) \vee (\vec{u} \wedge \vec{v})$$

The *common factor theorem* states that  $\vec{z}$  can be expressed as

$$(\vec{x} \wedge \vec{y}) \vee (\vec{u} \wedge \vec{v}) = (\vec{x} \wedge \vec{y} \wedge \vec{v}) \vee \vec{u} - (\vec{x} \wedge \vec{y} \wedge \vec{u}) \vee \vec{v} \quad (\text{A.25})$$

#### A.4.5 The Intersection of Two Bivectors in a 3-D Space

As given by (A.7), in a three-space, the exterior product of three vectors will, in any given basis, give the basis trivector, multiplied by the determinant of the components of the vectors making up the trivector. The regressive product (intersection) of a vector with an element like the basis trivector completely



**Figure A.2** Vectors and points.

containing the vector gives an element congruent to itself. Thus, the intersection of the two bivectors is given by

$$\vec{z} = \det[\vec{x}, \vec{y}, \vec{v}]\vec{u} - \det[\vec{x}, \vec{y}, \vec{u}]\vec{v} \quad (\text{A.26})$$

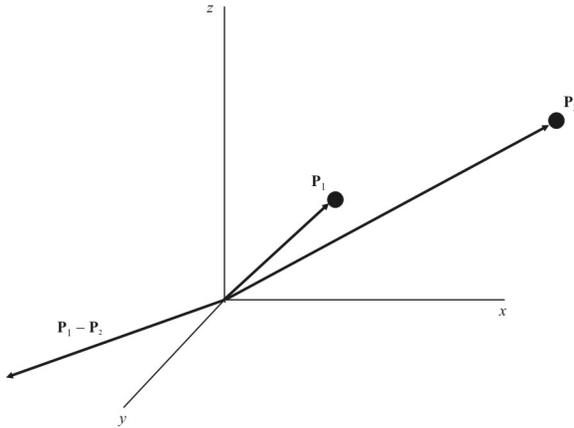
where  $\det[\vec{x}, \vec{y}, \vec{v}]$  is the determinant of the components of  $\vec{x}$ ,  $\vec{y}$ , and  $\vec{v}$ .

## A.5 Geometric Interpretations

All of the above discussion applies irrespective of any physical associations of, for example, elements and vectors, to physical objects. As noted, a metric is not necessary for the notions to apply, yielding a powerful algebra that can be used to manipulate higher-order geometric objects and shapes. In this section, associations to physical objects will be established and application of the algebra to their manipulation will be demonstrated.

### A.5.1 Points and Vectors

The element in a linear space is called a vector. Even though it need not be the case, the term *vector* is associated with the geometric concept of *direction*.



**Figure A.3** Difference of two points.

An element of a linear space may also be interpreted as a *point*. Of course, vectors may also be used to represent points, but only *relative to another given point* (the information for which is not in the vector). For a single vector, this other point is often the origin of the linear space. The origin of 3-D space, for example, is given by  $(x, y, z) = (0, 0, 0)$ , as illustrated in Figure A.2. These vectors are sometimes called *position vectors*. The reference point, however, need not be the origin and the concept of a vector can still be applied to define a point. We cannot interpret it physically in this case, however. When *both position and direction* are required in the same element, we must somehow include the information about the point of origin of the vector. A point can be represented in 3-D space by two (or more) vectors without the assumed reference to the origin, however, as illustrated in Figure A.3.

### A.5.2 Sums and Differences of Points

A *point* is defined as the sum of the *origin point* and a *vector*. If  $o$  is the origin and  $\vec{x}$  is a vector, then  $o + \vec{x}$  is a point:

$$\mathbf{P} \triangleq o + \vec{x} \quad (\text{A.27})$$

The vector  $\vec{x}$  is the position vector of the point  $\mathbf{P}$ . The *difference* of two points is a vector (not a point), since the origins cancel (remember that in Grassmann algebra, the operations of addition and subtraction are the familiar ones from high school linear algebra):

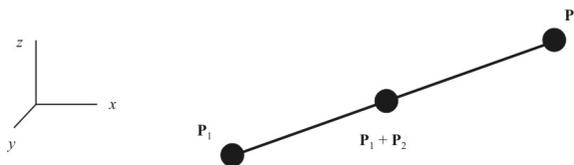


Figure A.4 Sum of two points.

$$\mathbf{P}_1 - \mathbf{P}_2 = (o + \vec{x}_1) - (o + \vec{x}_2) = \vec{x}_1 - \vec{x}_2$$

See Figure A.3.

A scalar multiple of a point is called a *weighted point*. For example, if  $m$  is a scalar,  $m\mathbf{P}$  is a weighted point with weight  $m$ . The *sum* of two points with unit weights gives the point halfway between them with a weight of 2. Such a sum is illustrated in Figure A.4.

$$\mathbf{P}_1 + \mathbf{P}_2 = (o + \vec{x}_1) + (o + \vec{x}_2) = 2\left(o + \frac{\vec{x}_1 + \vec{x}_2}{2}\right)$$

### A.5.3 Lines and Planes

The exterior product of a point and a vector gives a *line-bound vector*. A line is the space of a line-bound vector. That is, it consists of all the points on the line and all the vectors in the direction of the line. The reason for this is the lack of the point of origin of the vectors—all such origination points are included in the definition. Thus, all vectors in the direction of the line are included.

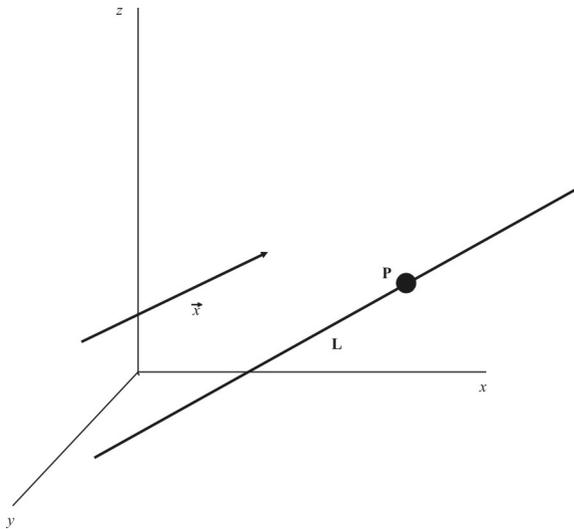
A line-bound vector can be represented by the line  $\mathbf{L}$  through the point  $\mathbf{P}$  in the direction of  $\vec{x}$  and any entity *congruent* to the exterior product of  $\mathbf{P}$  and  $\vec{x}$ .

$$\mathbf{L} \equiv \mathbf{P} \wedge \vec{x}$$

The concept of the line-bound vector is illustrated in Figure A.5.

A line may also be represented by the exterior product of *any two points* on it. Let  $\mathbf{Q}$  be any point on the line so that  $\mathbf{Q} = \mathbf{P} + a\vec{x}$  is the sum of  $\mathbf{P}$  and any scalar multiple of the vector. Then

$$\mathbf{L} \equiv \mathbf{P} \wedge \mathbf{Q} = \mathbf{P} \wedge (\mathbf{P} + a\vec{x}) = a\mathbf{P} \wedge \vec{x}$$



**Figure A.5** Line-bound vector; a line defined by a point and a vector is the line passing through the point in the direction of the vector and any entity congruent to the exterior product of  $\mathbf{P}$  and  $\vec{x}$ .

Therefore, as a general statement of these conditions:

$$\mathbf{L} \equiv \mathbf{P}_1 \wedge \mathbf{P}_2 \equiv \mathbf{P} \wedge \vec{x} \quad (\text{A.28})$$

These concepts extend to higher dimensions in the natural way. Thus, a plane  $\pi$  may be represented by the exterior product of any three points on it, any two points on it together with a vector in it (not parallel to the line joining the points), or a single point on it together with a bivector in the direction of the plane:

$$\pi \equiv \mathbf{P}_1 \wedge \mathbf{P}_2 \wedge \mathbf{P}_3 \equiv \mathbf{P}_1 \wedge \mathbf{P}_2 \wedge \vec{x} \equiv \mathbf{P}_1 \wedge \vec{x} \wedge \vec{y} \quad (\text{A.29})$$

Higher-dimensional geometric entities are constructed from lower-dimensional ones by simply computing their exterior product. A line can be obtained by taking the exterior product of a point with any point or vector *exterior* to it, as was just illustrated, or a plane can be constructed by taking the exterior product of a line with any point or vector exterior to it.

#### A.5.4 Intersection of Two Lines

The regressive product is used to find intersections of geometric entities. For example, to find the intersection of two lines in a plane, which will be a point in the plane, the regressive product of the two lines is computed. The lines can be represented as

$$\begin{aligned} \mathbf{L}_1 &\equiv \mathbf{P}_1 \wedge \vec{x}_1 = (o + \vec{v}_1) \wedge \vec{x}_1 = o \wedge \vec{x}_1 + \vec{v}_1 \wedge \vec{x}_1 \\ \mathbf{L}_2 &\equiv \mathbf{P}_2 \wedge \vec{x}_2 = (o + \vec{v}_2) \wedge \vec{x}_2 = o \wedge \vec{x}_2 + \vec{v}_2 \wedge \vec{x}_2 \end{aligned}$$

The point of intersection is found by

$$\mathbf{P} \equiv \mathbf{L}_1 \vee \mathbf{L}_2 = (o \wedge \vec{x}_1 + \vec{v}_1 \wedge \vec{x}_1) \vee (o \wedge \vec{x}_2 + \vec{v}_2 \wedge \vec{x}_2)$$

Four terms arise from this expression:

$$\begin{aligned} \mathbf{P} &\equiv (o \wedge \vec{x}_1) \vee (o \wedge \vec{x}_2) + (\vec{v}_1 \wedge \vec{x}_1) \vee (o \wedge \vec{x}_2) \\ &\quad + (o \wedge \vec{x}_1) \vee (\vec{v}_2 \wedge \vec{x}_2) + (\vec{v}_1 \wedge \vec{x}_1) \vee (\vec{v}_2 \wedge \vec{x}_2) \end{aligned}$$

The common factor theorem in (A.25) applied to these products generates:

$$(o \wedge \vec{x}_1) \vee (o \wedge \vec{x}_2) = (o \wedge \vec{x}_1 \wedge \vec{x}_2) \vee o - (o \wedge \vec{x}_1 \wedge o) \vee \vec{x}_2$$

$$\begin{aligned}
&= (o \wedge \vec{x}_1 \wedge \vec{x}_2) \vee o \\
(\vec{v}_1 \wedge \vec{x}_1) \vee (o \wedge \vec{x}_2) &= (\vec{v}_1 \wedge \vec{x}_1 \wedge \vec{x}_2) \vee o - (\vec{v}_1 \wedge \vec{x}_1 \wedge o) \vee \vec{x}_2 \\
&= -(\vec{v}_1 \wedge \vec{x}_1 \wedge o) \vee \vec{x}_2 \\
(o \wedge \vec{x}_1) \vee (\vec{v}_2 \wedge \vec{x}_2) &= -(\vec{v}_2 \wedge \vec{x}_2) \vee (o \wedge \vec{x}_1) \\
&= -(\vec{v}_2 \wedge \vec{x}_2 \wedge \vec{x}_1) \vee o + (\vec{v}_2 \wedge \vec{x}_2 \wedge o) \vee \vec{x}_1 \\
&= (\vec{v}_2 \wedge \vec{x}_2 \wedge o) \vee \vec{x}_1 \\
(\vec{v}_1 \wedge \vec{x}_1) \vee (\vec{v}_2 \wedge \vec{x}_2) &= (\vec{v}_1 \wedge \vec{x}_1 \wedge \vec{x}_2) \vee \vec{v}_2 - (\vec{v}_1 \wedge \vec{x}_1 \wedge \vec{v}_2) \vee \vec{x}_2 \\
&= 0
\end{aligned}$$

These simplifications arise because the following facts about vectors and exterior products were applied: (1) the term  $(o \wedge \vec{x}_1 \wedge o)$  is zero because of the exterior product of repeated factors; and (2) the four terms involving the exterior product of three vectors, for example,  $(\vec{v}_1 \wedge \vec{x}_1 \wedge \vec{x}_2)$ , are also zero, since any three vectors in a two-dimensional vector space must be dependent. Thus,

$$\mathbf{P} \triangleq (o \wedge \vec{x}_1 \wedge \vec{x}_2) \vee o + (o \wedge \vec{v}_2 \wedge \vec{x}_2) \vee \vec{x}_1 - (o \wedge \vec{v}_1 \wedge \vec{x}_1) \vee \vec{x}_2$$

Given a basis with basis vectors given by  $\vec{e}_1$  and  $\vec{e}_2$ , this expression can be represented as

$$\mathbf{P} = o + \frac{\text{Det}[\vec{v}_2, \vec{x}_2]}{\text{Det}[\vec{x}_1, \vec{x}_2]} \vec{x}_1 - \frac{\text{Det}[\vec{v}_1, \vec{x}_1]}{\text{Det}[\vec{x}_1, \vec{x}_2]} \vec{x}_2$$

where the coefficients of the vectors are the coefficients in the given basis.

To verify that  $\mathbf{P}$  does indeed lie on both the lines  $\mathbf{L}_1$  and  $\mathbf{L}_2$ , we only need to carry out the straightforward verification that the products  $\mathbf{P} \wedge \mathbf{L}_1$  and  $\mathbf{P} \wedge \mathbf{L}_2$  are both zero. Thus,

$$\mathbf{P} \wedge \mathbf{L}_1 = (o \wedge \vec{x}_1 + \vec{v}_1 \vec{x}_1) \wedge \left[ o + \frac{\text{Det}(\vec{v}_2, \vec{x}_2)}{\text{Det}(\vec{x}_1, \vec{x}_2)} \vec{x}_1 + \frac{\text{Det}(\vec{v}_1, \vec{x}_1)}{\text{Det}(\vec{x}_1, \vec{x}_2)} \vec{x}_2 \right]$$

$$\begin{aligned}
 &= o \wedge \vec{x}_1 \wedge o + \frac{\det(\vec{v}_2, \vec{x}_2)}{\det(\vec{x}_1, \vec{x}_2)} o \wedge \vec{x}_1 \wedge \vec{x}_1 \\
 &\quad + \frac{\det(\vec{v}_1, \vec{x}_1)}{\det(\vec{x}_1, \vec{x}_2)} o \wedge \vec{x}_1 \wedge \vec{x}_2 + \vec{v}_1 \wedge \vec{x}_1 \wedge o \\
 &\quad + \frac{\det(\vec{v}_1, \vec{x}_1)}{\det(\vec{x}_1, \vec{x}_2)} \vec{v}_1 \wedge \vec{x}_1 \wedge \vec{x}_1 + \frac{\det(\vec{v}_2, \vec{x}_2)}{\det(\vec{x}_1, \vec{x}_2)} \vec{v}_1 \wedge \vec{x}_1 \wedge \vec{x}_2
 \end{aligned}$$

Now

$$o \wedge \vec{x}_1 \wedge o = 0 \qquad \frac{\det(\vec{v}_2, \vec{x}_2)}{\det(\vec{x}_1, \vec{x}_2)} o \wedge \vec{x}_1 \wedge \vec{x}_1 = 0 \qquad \frac{\det(\vec{v}_2, \vec{x}_2)}{\det(\vec{x}_1, \vec{x}_2)} \vec{v}_1 \wedge \vec{x}_1 \wedge \vec{x}_1 = 0$$

because of the exterior product of like elements, leaving

$$\begin{aligned}
 &\frac{\det(\vec{v}_1, \vec{x}_1)}{\det(\vec{x}_1, \vec{x}_2)} o \wedge \vec{x}_1 \wedge \vec{x}_2 + \vec{v}_1 \wedge \vec{x}_1 \wedge o + \frac{\det(\vec{v}_1, \vec{x}_1)}{\det(\vec{x}_1, \vec{x}_2)} \vec{v}_1 \wedge \vec{x}_1 \wedge \vec{x}_2 \\
 &= \frac{\det(\vec{v}_1, \vec{x}_1)}{\det(\vec{x}_1, \vec{x}_2)} [o + \vec{v}_1] \wedge \vec{x}_1 \wedge \vec{x}_2 + \vec{v}_1 \wedge \vec{x}_1 \wedge o
 \end{aligned} \tag{A.30}$$

The first term on the right in (A.30) is zero because three elements in a two-space are, of necessity, dependent and therefore their exterior product is zero. Thus

$$\mathbf{P} \wedge \mathbf{L}_1 = \vec{v}_1 \wedge \vec{x}_1 \wedge o = 0 \tag{A.31}$$

Therefore,  $\mathbf{P}$  lies on  $\mathbf{L}_1$ . It can be similarly shown that  $\mathbf{P}$  lies on  $\mathbf{L}_2$ . Therefore,  $\mathbf{P}$  is the intersection of  $\mathbf{L}_1$  and  $\mathbf{L}_2$ .

This development is more complicated than for the simple linear algebra solution to the same problem. However, this can be generalized to any dimension and therefore any geometrical object, not just lines and points. The intersection of arbitrary geometrical shapes can be found.

## A.6 The Complement

### A.6.1 The Complement as a Correspondence Between Spaces

Given a linear space of dimension  $N$  with basis  $\vec{e}_1, \vec{e}_2, \dots, \vec{e}_N$ , the set of all the different  $m$ -element products of these basis elements forms the basis of another linear space of dimension  $\binom{N}{m}$ . The antisymmetric nature of the exterior product means that there are just as many basis elements in the linear space of  $(n - m)$ -elements as there are in the linear space of  $m$ -elements. There is a correspondence between the elements in the space of dimension  $N$  and that of dimension  $N - m$ . The  $(N - m)$ -element is called the *complement* of the  $m$ -element.

### A.6.2 The Euclidean Complement

In a 3-D linear space with basis  $\vec{e}_1, \vec{e}_2$ , and  $\vec{e}_3$ , the *Euclidean complement* of each of the basis elements is defined as the basis two-element whose exterior product with the basis element gives the basis three-element  $\vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3$ . Thus,

$$\vec{e}_1 \triangleq \vec{e}_2 \wedge \vec{e}_3 \Rightarrow \vec{e}_1 \wedge \vec{e}_1 = \vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3 \quad (\text{A.32})$$

$$\vec{e}_2 \triangleq \vec{e}_3 \wedge \vec{e}_1 \Rightarrow \vec{e}_2 \wedge \vec{e}_2 = \vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3 \quad (\text{A.33})$$

$$\vec{e}_3 \triangleq \vec{e}_1 \wedge \vec{e}_2 \Rightarrow \vec{e}_3 \wedge \vec{e}_3 = \vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3 \quad (\text{A.34})$$

The Euclidean complement of a general 1-element,  $\vec{x} = a\vec{e}_1 + b\vec{e}_2 + c\vec{e}_3$ , can now be defined by extending the definition on the basis elements by linearity,

$$\vec{\bar{x}} = \overline{a\vec{e}_1 + b\vec{e}_2 + c\vec{e}_3} \triangleq a\vec{e}_1 + b\vec{e}_2 + c\vec{e}_3 = a(\vec{e}_2 \wedge \vec{e}_3) + b(\vec{e}_3 \wedge \vec{e}_1) + c(\vec{e}_1 \wedge \vec{e}_2)$$

The product  $\vec{x} \wedge \vec{\bar{x}}$  expands to

$$\begin{aligned} \vec{x} \wedge \vec{\bar{x}} &= (a\vec{e}_1 + b\vec{e}_2 + c\vec{e}_3) \wedge (a\vec{e}_2 \wedge \vec{e}_3 + b\vec{e}_3 \wedge \vec{e}_1 + c\vec{e}_1 \wedge \vec{e}_2) \\ &= (a^2 + b^2 + c^2)\vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3 \end{aligned}$$

The complements of the basis two-elements are defined just as those for one-elements, such that the exterior product of a basis two-element with its complement is equal to the basis three-element. The complement of a two-element in three-space is therefore a one-element

$$\begin{aligned}\overline{\vec{e}_2 \wedge \vec{e}_3} = \vec{e}_1 &\Rightarrow \vec{e}_2 \wedge \vec{e}_3 \wedge \overline{\vec{e}_2 \wedge \vec{e}_3} = \vec{e}_2 \wedge \vec{e}_3 \wedge \vec{e}_1 = \vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3 \\ \overline{\vec{e}_3 \wedge \vec{e}_1} = \vec{e}_2 &\Rightarrow \vec{e}_3 \wedge \vec{e}_1 \wedge \overline{\vec{e}_3 \wedge \vec{e}_1} = \vec{e}_3 \wedge \vec{e}_1 \wedge \vec{e}_2 = \vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3 \\ \overline{\vec{e}_1 \wedge \vec{e}_2} = \vec{e}_3 &\Rightarrow \vec{e}_1 \wedge \vec{e}_2 \wedge \overline{\vec{e}_1 \wedge \vec{e}_2} = \vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3\end{aligned}$$

In addition,

$$\begin{aligned}\overline{\vec{1}} &= \vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3 \\ \overline{\vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3} &= \vec{1}\end{aligned}$$

### A.6.3 The Complement of a Complement

The complement of the complement of  $\mathbf{x}$  is just  $\mathbf{x}$  itself.

$$\begin{aligned}\vec{x} &= a\vec{e}_1 \wedge b\vec{e}_2 \wedge c\vec{e}_3 \\ \overline{\vec{x}} &= \overline{a\vec{e}_1 \wedge b\vec{e}_2 \wedge c\vec{e}_3} = a\overline{\vec{e}_1} \wedge b\overline{\vec{e}_2} \wedge c\overline{\vec{e}_3} = a\vec{e}_2 \wedge \vec{e}_3 + b\vec{e}_3 \wedge \vec{e}_1 + c\vec{e}_1 \wedge \vec{e}_2 \\ \overline{\overline{\vec{x}}} &= \overline{a\vec{e}_2 \wedge \vec{e}_3 + b\vec{e}_3 \wedge \vec{e}_1 + c\vec{e}_1 \wedge \vec{e}_2} \\ &= a\overline{\vec{e}_2 \wedge \vec{e}_3} + b\overline{\vec{e}_3 \wedge \vec{e}_1} + c\overline{\vec{e}_1 \wedge \vec{e}_2} = a\vec{e}_1 + b\vec{e}_2 + c\vec{e}_3\end{aligned}$$

The complement of *any* element is the element itself, apart from a possible sign:

$$\overline{\overline{\vec{\alpha}}} = (-1)^{m(N-m)} \vec{\alpha} \quad (\text{A.35})$$

### A.6.4 The Complement Axiom

The complement axiom states that the Euclidean complement of the exterior product of two elements is equal to the regressive product of their complements:

$$\overline{\vec{\alpha} \wedge \vec{\beta}} = \overline{\vec{\alpha}} \vee \overline{\vec{\beta}} \quad (\text{A.36})$$

We can immediately recognize the similarity of (A.36) to de Morgan's law in Boolean algebra.

## A.7 The Interior Product

The *interior product* of an element  $\vec{\alpha}_m$  with an element  $\vec{\beta}_k$  is denoted by  $\vec{\alpha}_m \odot \vec{\beta}_k$  and defined to be the *regressive product* of  $\vec{\alpha}_m$  with the complement of  $\vec{\beta}_k$ :

$$\vec{\alpha}_m \odot \vec{\beta}_k \triangleq \vec{\alpha}_m \vee \vec{\beta}_k \quad (\text{A.37})$$

The grade of an interior product  $\vec{\alpha}_m \odot \vec{\beta}_k$  is  $m + (N - k) - N = m - k$ .

The interior product is a generalization of the inner product of scalars and vectors.

### A.7.1 Inner Products and Scalar Products

The interior product of two elements  $\vec{\alpha}_m$  and  $\vec{\beta}_m$  of the same grade is called their *inner product*. Since the grade of an interior product is the difference of the grades of its factors, an inner product is always of grade zero, hence scalar.

When the two factors of the product are grade 1, the inner product is called a *scalar product*, which is the definition used in everyday life.

The inner product is symmetric, that is,

$$\vec{\alpha}_m \odot \vec{\beta}_m = \vec{\beta}_m \odot \vec{\alpha}_m \quad (\text{A.38})$$

### A.7.2 Calculating Interior Products

Let the linear space under consideration be a three-space with Euclidean metric and basis elements  $\vec{e}_1, \vec{e}_2$ , and  $\vec{e}_3$ . The scalar products  $\vec{e}_1 \odot \vec{e}_1$  and  $\vec{e}_1 \odot \vec{e}_2$  are

$$\begin{aligned} \vec{e}_1 \odot \vec{e}_1 &= \vec{e}_1 \vee \vec{e}_1 = \vec{e}_1 \vee (\vec{e}_2 \wedge \vec{e}_3) = (\vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3) \vee \vec{1} = \vec{1} \vee \vec{1} = 1 \\ \vec{e}_1 \odot \vec{e}_2 &= \vec{e}_1 \vee \vec{e}_2 = \vec{e}_1 \vee (\vec{e}_3 \wedge \vec{e}_1) = (\vec{e}_1 \wedge \vec{e}_3 \wedge \vec{e}_1) \vee \vec{1} = \vec{0} \vee \vec{1} = 0 \end{aligned}$$

That is, the scalar product of a basis element with itself is 1 while the scalar product of two different basis functions is 0.

Since identical basis two-elements are the same element, their inner product is one:

$$\begin{aligned} (\vec{e}_1 \wedge \vec{e}_2) \odot (\vec{e}_1 \wedge \vec{e}_2) &= (\vec{e}_1 \wedge \vec{e}_2) \vee \overline{(\vec{e}_1 \wedge \vec{e}_2)} \\ &= (\vec{e}_1 \wedge \vec{e}_2) \vee \vec{e}_3 = (\vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3) \vee \vec{1} = \vec{1} \vee \vec{1} = 1 \end{aligned}$$

Likewise, the inner product of nonidentical basis two-elements is zero:

$$\begin{aligned} (\vec{e}_1 \wedge \vec{e}_2) \odot (\vec{e}_2 \wedge \vec{e}_3) &= (\vec{e}_1 \wedge \vec{e}_2) \vee \overline{(\vec{e}_2 \wedge \vec{e}_3)} \\ &= (\vec{e}_1 \wedge \vec{e}_2) \vee \vec{e}_1 = (\vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_1) \vee \vec{1} = \vec{0} \vee \vec{1} = 0 \end{aligned}$$

If a basis two-element contains a given basis one-element, then their interior product is not zero:

$$\begin{aligned} (\vec{e}_1 \wedge \vec{e}_2) \odot \vec{e}_1 &= (\vec{e}_1 \wedge \vec{e}_2) \vee \vec{e}_1 \\ &= (\vec{e}_1 \wedge \vec{e}_2) \vee (\vec{e}_2 \wedge \vec{e}_3) = (\vec{e}_1 \wedge \vec{e}_2 \wedge \vec{e}_3) \vee \vec{e}_2 = \vec{1} \vee \vec{e}_2 = \vec{e}_2 \end{aligned}$$

If a basis two-element does *not* contain a given basis 1-element, then their interior product is zero:

$$\begin{aligned} (\vec{e}_1 \wedge \vec{e}_2) \odot \vec{e}_3 &= (\vec{e}_1 \wedge \vec{e}_2) \vee \vec{e}_3 \\ &= (\vec{e}_1 \wedge \vec{e}_2) \vee (\vec{e}_1 \wedge \vec{e}_2) = 0 \end{aligned}$$

### A.7.3 Expanding Interior Products

#### Property Interior Common Factor

If  $\vec{\alpha}_m$  is a simple element,

$$\begin{aligned} \vec{\alpha}_m \odot_k \vec{\beta}_k &= \sum_{i=1}^v \left( \vec{\alpha}_i \odot_k \vec{\beta}_i \right)_{m-k} \vec{\alpha}_i & k \leq m, \quad v = \binom{m}{k} \\ \vec{\alpha}_m &= \vec{\alpha}_1 \wedge_k \vec{\alpha}_1 = \vec{\alpha}_2 \wedge_k \vec{\alpha}_2 = \dots = \vec{\alpha}_v \wedge_k \vec{\alpha}_v \end{aligned}$$

In these expressions  $\vec{\beta}_k$  need not be simple. Thus, the interior product of a simple element  $\vec{\alpha}_m$  with another element of equal or

lower grade  $\vec{\beta}_m$  may be expressed as a linear combination of the  $v$  essentially different factors  $\vec{\alpha}_{m-k}$  (of grade  $m-k$ ) of the simple element of higher degree.

#### A.7.4 The Interior Product of a Bivector and a Vector

Given that  $\vec{x}$  is a vector and that  $\vec{x}_1 \wedge \vec{x}_2$  is a simple bivector, the *interior product* of  $\vec{x}_1 \wedge \vec{x}_2$  with  $\vec{x}$  is the vector  $(\vec{x}_1 \wedge \vec{x}_2) \odot \vec{x}$ . By the Interior Common Factor Theorem, or the formula above derived from it, this can be expanded to give

$$(\vec{x}_1 \wedge \vec{x}_2) \odot \vec{x} = (\vec{x} \odot \vec{x}_1) \vec{x}_2 - (\vec{x} \odot \vec{x}_2) \vec{x}_1$$

which is a linear combination of  $\vec{x}_1$  and  $\vec{x}_2$ . Thus, it is clearly contained in the bivector  $\vec{x}_1 \wedge \vec{x}_2$ . Taking the result's scalar product with  $\vec{x}$ ,

$$((\vec{x} \odot \vec{x}_1) \vec{x}_2 - (\vec{x} \odot \vec{x}_2) \vec{x}_1) \odot \vec{x} = (\vec{x} \odot \vec{x}_1)(\vec{x}_2 \odot \vec{x}) - (\vec{x} \odot \vec{x}_2)(\vec{x}_1 \odot \vec{x}) = \vec{0}$$

shows that the result is orthogonal to  $\vec{x}$ .

These concepts are not restricted to two dimensions and may readily be extended to geometric entities of arbitrary dimension.

## A.8 Concluding Remarks

The basics of an algebra discovered by Hermann Grassmann in the nineteenth century were presented in this appendix. The application of concern to EW target geolocation was presented in Section 3.3. That is by no means the only application of this algebra, however.

Grassmann algebra subsumes the linear algebra that most learn about in high school. By making the appropriate associations of exterior and interior products with everyday products, the mapping becomes clear.

## References

- [1] Browne, J., *Grassmann Algebra, Exploring Applications of Extended Vector Algebra with Mathematica*, incomplete draft, [www.ses.swin.edu.au/home/browne/grassmannalgebra/book/](http://www.ses.swin.edu.au/home/browne/grassmannalgebra/book/).
- [2] Armstrong, H. L., "On an Alternative Definition of the Vector Product in  $n$ -Dimensional Vector Analysis," *Matrix and Tensor Quarterly*, Vol. IX, No. 4, 1959, pp. 107–110.
- [3] Barton, H., "A Modern Presentation of Grassmann's Tensor Analysis," *American Journal of Mathematics*, Vol. XLIX, 1927, pp. 598–614.
- [4] Bowen, R. M., and C.-C. Wang, *Introduction to Vectors and Tensors*, New York: Plenum Press, 1976.
- [5] Burali-Forti, C., *Introduction à la Géométrie Différentielle suivant la Méthode de H. Grassmann*, Paris: Gauthier-Villars, 1897.
- [6] Burali-Forti, C., and R. Marcolongo, *Éléments de Calcul Vectoriel*, Paris: Hermann, 1910.
- [7] Carvallo, M. E., "La Méthode de Grassmann," *Nouvelles Annales de Mathématiques*, serie 3, Vol. XI, 1892, pp. 8–37.
- [8] Chevalley, C., *The Construction and Study of Certain Important Algebras*, Tokyo: Mathematical Society of Japan, 1955.
- [9] Clifford, W. K., "Applications of Grassmann's Extensive Algebra," *American Journal of Mathematics Pure and Applied*, Vol. I, 1878, pp. 350–358.
- [10] Clifford, W. K., *Mathematical Papers*, 1882, reprinted by Chelsea Publishing Co., New York, 1968.
- [11] Coffin, J. G., *Vector Analysis*, New York: Wiley, 1909.
- [12] Collins, J. V., "An Elementary Exposition of Grassmann's *Ausdehnungslehre* or Theory of Extension," *American Mathematical Monthly*, Vol. 6, 1899, pp. 193–198, 261–266, 297–301; Vol. 7, 1900, pp. 31–35, 163–166, 181–187, 207–214, 253–258.
- [13] Coolidge, J. L., "Grassmann's Calculus of Extension" in *A History of Geometrical Methods*, Oxford: Oxford University Press, 1940, pp. 252–257.
- [14] Crowe, M. J., *A History of Vector Analysis*, Notre Dame: Notre Dame Press, 1967.
- [15] Dibag, I., "Factorization in Exterior Algebras," *Journal of Algebra*, New York: Academic Press, Vol. 30, 1974, pp. 259–262.
- [16] Gibbs, J. W., *The Collected Works of J. Willard Gibbs Ph.D. L.L.D.*, New York: Longmans, 1928.
- [17] Grassmann, H. G., *Gesammelte mathematische und physikalische Werke*, F. Engel (ed.), Teubner: Leipzig, Vol. 1 (1896), Vol. 2 (1902, 1904), and Vol. 3 (1911).
- [18] Kannenberg, L. C., *A New Branch of Mathematics: The Ausdehnungslehre of 1844 and Other Works*, Chicago, IL: Open Court, 1995.
- [19] Whitehead, A. N., *A Treatise on Universal Algebra*, Cambridge, U.K.: Cambridge University Press, 1898.
- [20] Hyde, E. W., *The Directional Calculus*, Boston, MA: Ginn & Co., 1890.
- [21] Zaddach, A., *Grassmann's Algebra in der Geometrie*, Zurich: BI-Wissenschaftsverlag, 1994.



# Appendix B

## Nonlinear Programming Algorithms

### B.1 Introduction

In this appendix we describe some nonlinear programming algorithms. Specifically we discuss the method of steepest descent, the Gauss-Newton method, and the Levenberg-Marquardt algorithm. These are but a few of the many algorithms available for this function.

### B.2 Steepest Descent

#### B.2.1 Introduction

The *steepest descent method*, also known as the *gradient descent method*, was first proposed by Cauchy in 1847 [1]. Cauchy proposed the use of the gradient as a way of solving a nonlinear equation of the form

$$f(x_1, x_2, \dots, x_n) = 0 \tag{B.1}$$

where  $f$  is a real-valued continuous function that never becomes negative. The basis for the method is the simple observation that a continuous function should decrease, at least initially, if one takes a step along the direction of the negative gradient. The only difficulty then is deciding how to choose the size of the step. While this is easy to compute for special cases such as a convex quadratic function, the general case usually requires the minimization of the function in question along the negative gradient direction.

Despite its simplicity, the steepest descent method has played an important role in the development of the theory of optimization. Unfortunately, the method is known to be quite slow in most real-world problems and is therefore not widely used. Instead, more powerful methods such as the conjugate gradient method or quasi-Newton methods are frequently used.

### B.2.2 Method of Steepest Descent

Let  $f(\bar{x})$ ,  $\bar{x} \in \mathfrak{R}^n$ , and  $f : \mathfrak{R}^n \rightarrow \mathfrak{R}$ , be the function for which we want to find the minimum. We denote the gradient of  $f$  by  $\bar{g}_k = \bar{g}(\bar{x}_k) = \nabla f(\bar{x}_k)$ . The general idea behind most minimization methods is to compute a step along a given search direction,  $\bar{d}_k$ , for example,

$$\bar{x}_{k+1} = \bar{x}_k + \alpha_k \bar{d}_k, \quad k = 0, 1, \dots \quad (\text{B.2})$$

where the step length,  $\alpha_k$ , is chosen so that

$$\alpha_k = \arg \min_{\alpha} f(\bar{x}_k + \alpha \bar{d}_k) \quad (\text{B.3})$$

Here argmin refers to the argument of the minimum for the given function. For the steepest descent method, the search direction is given by  $\bar{d}_k = -\nabla f(\bar{x}_k)$ .

The steepest descent algorithm is as follows:

**Algorithm** *Steepest Descent*

```

Given an initial  $\bar{x}_0, \bar{d}_0 = -\bar{g}_0$ , and a convergence tolerance tol
for  $k = 0$  to  $k_{\max}$  do
  Set  $\alpha_k = \arg \min_{\alpha} f(\bar{x}_k - \alpha \bar{g}_k)$ 
   $\bar{x}_{k+1} = \bar{x}_k - \alpha \bar{g}_k$ 
  Compute
   $\bar{g}_{k+1} = \nabla f(\bar{x}_{k+1})$ 
  if  $\|\bar{g}_{k+1}\|_2 \leq \text{tol}$  then
    converged
  end if
end for

```

The two main computational advantages of the steepest descent algorithm is the ease with which a computer algorithm can be implemented and the low storage

requirements necessary,  $O(n)$ . The main work requirement is the line search required to compute the step length,  $\alpha_k$  and the computation of the gradient.

### B.2.3 Convergence

It is fairly easy to show that the steepest descent method has a linear rate of convergence, which is not too surprising given the simplicity of the method [2, 3]. Unfortunately, even for mildly nonlinear problems this will result in convergence that is too slow for any practical application.

Consider the case of minimizing the quadratic function

$$f(\vec{x}) = \frac{1}{2} \vec{x}^T \mathbf{Q} \vec{x} - \vec{b}^T \vec{x} \tag{B.4}$$

where  $\vec{b} \in \mathfrak{R}^n$ , and  $\mathbf{Q}$  is an  $n \times n$  symmetric positive definite matrix. Since  $\mathbf{Q}$  is symmetric and positive definite, all of the eigenvalues are real and positive. Arrange the  $n$  eigenvalues of the matrix  $\mathbf{Q}$  as  $\lambda_n \geq \dots \geq \lambda_2 \geq \lambda_1 > 0$ . The gradient of (B.4) is

$$\vec{g}(\vec{x}) = \mathbf{Q} \vec{x} - \vec{b} \tag{B.5}$$

so we can write one step of the method of steepest descent as

$$\vec{x}_{k+1} = \vec{x}_k - \alpha_k (\mathbf{Q} \vec{x}_k - \vec{b}) \tag{B.6}$$

where  $\alpha_k$  is chosen to minimize  $f(\vec{x})$  along the direction  $-\vec{g}_k$ . A simple calculation (for the quadratic case) yields the following equation for  $\alpha_k$

$$\bar{\alpha}_k = \frac{\vec{g}_k^T}{\vec{g}_k^T \mathbf{Q} \vec{g}_k} \tag{B.7}$$

To analyze the convergence, we consider the quantity  $f(\vec{x}_k) - f(\vec{x}^\diamond)$ , where  $\vec{x}^\diamond$  denotes the global minimizer of (B.4). We first notice that the unique minimizer of (B.4) is given by the solution to the linear system

$$\mathbf{Q} \vec{x}^\diamond = \vec{b} \tag{B.8}$$

Now consider

$$\begin{aligned}
 f(\bar{x}_k) - f(\bar{x}^\circ) &= \frac{1}{2}(\bar{x}_k^T \mathbf{Q} \bar{x}_k) - \frac{1}{2}[(\bar{x}^\circ)^T \mathbf{Q} \bar{x}^\circ - \bar{b}^T \bar{x}^\circ] \\
 &= \frac{1}{2}(\bar{x}_k^T \mathbf{Q} \bar{x}_k - (\mathbf{Q} \bar{x}^\circ)^T \bar{x}_k) - \frac{1}{2}[(\bar{x}^\circ)^T \mathbf{Q} \bar{x}^\circ - (\mathbf{Q} \bar{x}^\circ)^T \bar{x}^\circ] \\
 &= \frac{1}{2}(\bar{x}_k - \bar{x}^\circ)^T \mathbf{Q} (\bar{x}_k - \bar{x}^\circ)
 \end{aligned}$$

When the method of steepest descent with exact line searches is used on a strongly convex quadratic function then [2]

$$f(\bar{x}_{k+1}) - f(\bar{x}^\circ) \leq \left[ \frac{\kappa(\mathbf{Q}) - 1}{\kappa(\mathbf{Q}) + 1} \right]^2 f(\bar{x}_k) - f(\bar{x}^\circ) \quad (\text{B.9})$$

where  $\kappa(\mathbf{Q}) \triangleq \lambda_n / \lambda_1$  is the *condition number* of matrix  $\mathbf{Q}$ .

### Example

Consider the 3-D quadratic function given by

$$f(\bar{x}) = \frac{1}{2} \bar{x}^T \mathbf{Q} \bar{x} - \bar{b}^T \bar{x} \quad (\text{B.10})$$

where

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \tau & 0 \\ 0 & 0 & \tau^2 \end{bmatrix} \quad \bar{b} = - \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

The results of employing the steepest decent method are given in Table B.1. The convergence tolerance was set so that the algorithm would terminate when  $\|\bar{g}(\bar{x}_k)\|_2 \leq 10^{-6}$ . We can clearly see the effects of even a mildly large condition number as predicted by the error bound and as seen in the number of iterations required to achieve convergence.

**Table B.1** Steepest Descent.  $\kappa(\mathbf{Q}) = \lambda_n / \lambda_1$ 

$\tau$	Number of Iterations	$\kappa(\mathbf{Q})$	Bound
2	27	4	0.3600
5	161	25	0.8521
10	633	100	0.9801
20	2,511	400	0.9950
50	15,619	2,500	0.9984

### B.2.4 Scaling

One of the most important aspects in minimizing real problems is the issue of scaling. Because of the way that many scientific and engineering problems are initially formulated it is not uncommon to have variables with widely differing magnitudes. This can be due to many issues, but a common one is that variables have different physical units that can lead to the optimization variables having orders of magnitude differences. For example, one variable could be given in kilometers and another variable might be in milliseconds leading to a 6 order of magnitude difference. We would like to have all the variables in an optimization problem having roughly similar magnitudes, however. This leads to better decisions in which search direction to choose as well as in deciding when convergence is achieved. One fairly standard approach is to use a diagonal scaling based on what a “typical” value of a variable is expected to be. We then transform the variables by the scaling

$$\hat{\mathbf{x}} = \mathbf{D}\bar{\mathbf{x}} \quad (\text{B.11})$$

where  $\mathbf{D}$  is a diagonal scaling matrix. In the test problem given above, for example, one simple choice would be

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \tau & 0 \\ 0 & 0 & \tau^2 \end{bmatrix} \quad (\text{B.12})$$

### B.2.5 Extensions

Several modifications to the steepest descent method have been proposed. In 1988, Barzilai and Borwein [4] proposed two new step sizes for use with the negative gradient direction. Although their method did not guarantee descent in the objective function values, their numerical results indicated a substantial

improvement over the classical steepest descent method. One of their main observations was that the behavior of the steepest descent algorithm depended as much on the step size as on the search direction. They proposed instead the following procedure. First we write the new iterate as

$$\bar{x}_{k+1} = \bar{x}_k - \frac{1}{\alpha_k} \bar{g}_k \quad (\text{B.13})$$

Then, instead of computing the step size by doing a line search or using the formula for the quadratic case (B.7), we compute the step length,  $\alpha_k$ , through

$$\alpha_k = \frac{\bar{s}_{k-1}^T \bar{y}_{k-1}}{\bar{s}_{k-1}^T \bar{s}_{k-1}} \quad (\text{B.14})$$

where  $\bar{s}_{k-1} = \bar{x}_k - \bar{x}_{k-1}$  and  $\bar{y}_{k-1} = \bar{g}_k - \bar{g}_{k-1}$ . Using this new formula, Barzilai and Borwein were able to produce a substantial improvement in the performance of the steepest descent algorithm for certain test problems.

### B.3 Gauss-Newton Method

Functions used in a nonlinear regression have three types of variables. These three variables are the dependent variables, the independent variables, and the regression parameters.

The independent variables are usually the variables that are used as inputs while measuring the effect of them on other variables. The vector of independent variables is  $\bar{x} = [x_1 \ x_2 \ \cdots \ x_l]^T$ . The dependent variables are the observed outcome of varying the independent variables. The vector of dependent variables is  $\bar{y} = [y_1 \ y_2 \ \cdots \ y_m]^T$ . Both the independent variables and the dependent variables are model independent. They both can be determined experimentally without using a model.

A model is one or more equations that are used to correlate the dependent variables with respect to the independent variables. The model functions are denoted as  $f$ .

The regression parameters are parameters of the model used to correlate the dependent and independent variables and are model dependent. The regression parameters are the unknown variables to be determined in a nonlinear regression.

The vector of regression parameters is  $\bar{p} = [p_1 \ p_2 \ \cdots \ p_n]$ .

In a model, the dependent variables are functions of the independent variables and the model parameters. The difference between the values of the dependent variables predicted by the model and the values of the dependent variables actually observed is the error of the model, which is designated as  $\epsilon$ . That is,

$$y = f(x, p) + \epsilon \tag{B.15}$$

In vector form

$$\bar{y} = f(\bar{x}, \bar{p}) + \bar{\epsilon} \tag{B.16}$$

so

$$\bar{\epsilon} = \bar{y} - f(\bar{x}, \bar{p}) \tag{B.17}$$

A Taylor series for  $\bar{\epsilon}$  written about a set of regression parameters while holding the dependent and independent variables constant and retaining only the linear terms yields

$$\mathbf{J}_\epsilon = \begin{bmatrix} \frac{\partial(y_0 - f_0)}{\partial p_0} & \frac{\partial(y_0 - f_0)}{\partial p_1} & \dots & \frac{\partial(y_0 - f_0)}{\partial p_n} \\ \frac{\partial(y_1 - f_1)}{\partial p_0} & \frac{\partial(y_1 - f_1)}{\partial p_1} & \dots & \frac{\partial(y_1 - f_1)}{\partial p_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial(y_m - f_m)}{\partial p_0} & \frac{\partial(y_m - f_m)}{\partial p_1} & \dots & \frac{\partial(y_m - f_m)}{\partial p_n} \end{bmatrix} \tag{B.18}$$

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_0}{\partial p_0} & \frac{\partial f_0}{\partial p_1} & \dots & \frac{\partial f_0}{\partial p_n} \\ \frac{\partial f_1}{\partial p_0} & \frac{\partial f_1}{\partial p_1} & \dots & \frac{\partial f_1}{\partial p_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial p_0} & \frac{\partial f_m}{\partial p_1} & \dots & \frac{\partial f_m}{\partial p_n} \end{bmatrix} \quad (\text{B.19})$$

$$\bar{\epsilon}_{p_1} = \bar{\epsilon}_{p_0} + \mathbf{J}_\epsilon (\bar{p}_1 - \bar{p}_0) \quad (\text{B.20})$$

The  $m \times n$  matrix  $\mathbf{J}_\epsilon$  is known as the Jacobian of the error functions with respect to the regression parameters and the  $m \times n$  matrix  $\mathbf{J}$  is the Jacobian of the model functions with respect to the regression parameters.

Using (B.17) and noting that  $\bar{y}$  is constant with respect to the parameters, so their partial derivatives relative to the parameters are zero, gives

$$\mathbf{J}_\epsilon = -\mathbf{J} \quad (\text{B.21})$$

Substituting (B.21) in (B.20) gives

$$\bar{\epsilon}_{p_1} = \bar{\epsilon}_{p_0} - \mathbf{J}(\bar{p}_1 - \bar{p}_0) \quad (\text{B.22})$$

The square residual error  $\bar{s}_R$  about  $\bar{p}_1$  can be written as

$$\begin{aligned} \bar{s}_R &= \bar{\epsilon}_{p_1}^2 = [\bar{\epsilon}_{p_0} - \mathbf{J}(\bar{p}_1 - \bar{p}_0)]^2 = [\bar{\epsilon}_{p_0} - \mathbf{J}(\bar{p}_1 - \bar{p}_0)]^T [\bar{\epsilon}_{p_0} - \mathbf{J}(\bar{p}_1 - \bar{p}_0)] \\ &= \bar{\epsilon}_{p_0}^T \bar{\epsilon}_{p_0} + (\bar{p}_1 - \bar{p}_0)^T \mathbf{J}^T \mathbf{J} (\bar{p}_1 - \bar{p}_0) - 2\bar{\epsilon}_{p_0}^T \mathbf{J} (\bar{p}_1 - \bar{p}_0) \end{aligned} \quad (\text{B.23})$$

At the minimum square residual error, the derivatives of  $\bar{s}_R$  with respect to  $\bar{p}_1 - \bar{p}_0$  are equal to zero, so we get

$$\frac{d\bar{s}_R}{d(\bar{p}_1 - \bar{p}_0)} = 2\mathbf{J}^T \mathbf{J} (\bar{p}_1 - \bar{p}_0) - 2\mathbf{J}^T \bar{\epsilon}_{p_0} = 0 \quad (\text{B.24})$$

which yields the Gauss-Newton equation

$$(\mathbf{J}^T \mathbf{J})(\bar{p}_1 - \bar{p}_0) = \mathbf{J}^T \bar{\epsilon}_{p_0} \quad (\text{B.25})$$

so

$$\bar{p}_1 = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \bar{\epsilon}_{p_0} + \bar{p}_0 \quad (\text{B.26})$$

The Gauss-Newton algorithm works by starting with an initial guess of the parameters. These are used as  $\bar{p}_0$ . The residual error and the Jacobian are then solved. Then (B.26) is used to solve for  $\bar{p}_1$ , which becomes the new  $\bar{p}_0$  and the algorithm is repeated until it either converges or goes unstable.

## B.4 Levenberg-Marquardt Algorithm

### B.4.1 Introduction

The Levenberg-Marquardt (LM) algorithm is an iterative technique that locates the minimum of a multivariate function that is expressed as the sum of squares of nonlinear real-valued functions [5, 6]. It has become a standard technique for nonlinear least-squares problems, widely adopted in a broad spectrum of disciplines. LM can be thought of as a combination of steepest descent and the Gauss-Newton method described in the last section. When the current solution is far from the correct one, the algorithm behaves like a steepest descent method: slow, but guaranteed to converge. When the current solution is close to the correct solution, it becomes a Gauss-Newton method. Next, a short description of the LM algorithm based on the material in [5] is supplied. Note, however, that a detailed analysis of the LM algorithm is beyond the scope of this appendix and the interested reader is referred to [7–9] for more comprehensive treatments.

### B.4.2 Nonlinear Least-Squares Minimization

The problem for which the LM algorithm provides a solution is called *nonlinear least-squares minimization*. This implies that the function to be minimized is of the form

$$f(\bar{x}) = \frac{1}{2} \sum_{j=1}^m r_j^2(\bar{x})$$

where  $\bar{x} = [x_1 \ x_2 \ \cdots \ x_n]^T$  is a vector, and each  $r_j : \mathfrak{R}^n \rightarrow \mathfrak{R}$ . The  $r_j$  are referred to as *residuals* and it is assumed that  $m \geq n$ .

$f$  can be represented with a *residual vector*  $\bar{r} : \mathfrak{R}^n \rightarrow \mathfrak{R}^m$  defined by

$$\bar{r}(\bar{x}) \triangleq [r_1(\bar{x}) \ r_2(\bar{x}) \ \cdots \ r_m(\bar{x})]^T$$

Now,  $f$  can be rewritten as

$$f(\bar{x}) = \frac{1}{2} \|\bar{r}(\bar{x})\|^2$$

The derivatives of  $f$  can be written using the Jacobian matrix  $\mathbf{J}$  of  $\bar{r}$  with respect to  $\bar{x}$  whose entries are

$$\mathbf{J}(\bar{x}) = \frac{\partial r_j}{\partial x_i}, \quad 1 \leq j \leq m, 1 \leq i \leq n$$

Let us first consider the case where every  $r_i$  function is linear. Here, the Jacobian is constant and we can represent  $\bar{r}$  as a hyperplane through space, so that  $f$  is given by the quadratic

$$f(\bar{x}) = \frac{1}{2} \|\mathbf{J}\bar{x} + \bar{r}(0)\|^2$$

We also get

$$\nabla f(\bar{x}) = \mathbf{J}^T (\mathbf{J}\bar{x} + \bar{r}) \quad (\text{B.27})$$

and

$$\nabla^2 f(\bar{x}) = \mathbf{J}^T \mathbf{J}$$

Solving for the minimum by setting  $\nabla f(\bar{x}) = 0$ , we obtain

$$\bar{x}_{\min} = -(\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \bar{r} \quad (\text{B.28})$$

which is the solution to the set of *normal equations* given by

$$\mathbf{J}^T \mathbf{J} \bar{\mathbf{x}}_{\min} = -\mathbf{J}^T \bar{\mathbf{r}} \tag{B.29}$$

obtained by setting (B.27) equal to zero.

If a covariance matrix  $\Sigma_x$  for the measured vector  $\bar{x}$  is available, it can be incorporated into the LM algorithm by minimizing the squared  $\Sigma_x^{-1}$ -norm instead of the Euclidean  $\bar{r}^T \bar{r}$ . Accordingly, the minimum is found by solving a weighted least-squares problem defined by the *weighted normal equations*

$$\mathbf{J}^T \Sigma_x^{-1} \mathbf{J} \bar{\mathbf{x}}_{\min} = -\mathbf{J}^T \Sigma_x^{-1} \bar{\mathbf{r}} \tag{B.30}$$

The rest of the algorithm remains unchanged.

Returning to the general, non-linear case, we have

$$\nabla f(\bar{x}) = \sum_{j=1}^m r_j(\bar{x}) \nabla r_j(\bar{x}) + \mathbf{J}^T(\bar{x}) \bar{r}(\bar{x}) \tag{B.31}$$

$$\nabla^2 f(\bar{x}) = \mathbf{J}^T(\bar{x}) \mathbf{J}(\bar{x}) + \sum_{j=1}^m r_j(\bar{x}) \nabla^2 r_j(\bar{x}) \tag{B.32}$$

A useful property of least-squares problems is that given the Jacobian matrix  $\mathbf{J}$ , we can essentially get the Hessian  $[\nabla^2 f(\bar{x})]$  for free if it is possible to approximate the  $r_j$ s by linear functions  $[\nabla^2 r_j(\bar{x})$  are small] or the residuals  $[r_j(\bar{x})]$  themselves are small. The Hessian in this case simply becomes

$$\nabla^2 f(x) = \mathbf{J}^T(\bar{x}) \mathbf{J}(\bar{x}) \tag{B.33}$$

which is the same as for the linear case.

The common approximation used here is one of near linearity of the  $r_j$ s near the solution so that  $\nabla^2 r_j(\bar{x})$  are small. It is also important to note that (B.33) is only valid if the residuals are small. Large residual problems cannot be solved using the quadratic approximation, and consequently, the performance of the algorithms here is poor in such cases.

### B.4.3 LM as a Blend of Gradient Descent and Gauss-Newton Iteration

Vanilla gradient descent is the simplest, most intuitive technique to find minima in a function. Parameter update is performed by adding the negative of the scaled gradient at each step, that is,

$$\bar{x}_{i+1} = \bar{x}_i - \alpha \nabla f \quad (\text{B.34})$$

Simple gradient descent suffers from various convergence problems. Logically, we would like to take large steps down the gradient at locations where the gradient is small (the slope is gentle) and, conversely, take small steps when the gradient is large, so as not to rattle out of the minima. With the above update rule, we do just the opposite of this. Another issue is that the curvature of the error surface may not be the same in all directions. For example, if there is a long and narrow valley in the error surface, the component of the gradient in the direction that points along the base of the valley is very small while the component along the valley walls is quite large. This results in motion more in the direction of the walls even though we have to move a long distance along the base and a small distance along the walls.

This situation can be improved by using curvature as well as gradient information, namely second derivatives. One way to do this is to use Newton's method to solve the equation  $\nabla f(\bar{x}) = 0$ . Expanding the gradient of  $f$  using a Taylor series around the current state  $\bar{x}_0$ , we get

$$\nabla f(\bar{x}) = \nabla f(\bar{x}_0) + (\bar{x} - \bar{x}_0)^T \nabla^2 f(\bar{x}_0) + \text{higher order terms of } (\bar{x} - \bar{x}_0) \quad (\text{B.35})$$

If we neglect the higher-order terms (assuming  $f$  to be quadratic around  $\bar{x}_0$ ), and solve for the minimum  $x$  by setting the left hand side of (B.35) to 0, we get the update rule for Newton's method

$$\bar{x}_{i+1} = \bar{x}_i - [\nabla^2 f(\bar{x}_i)]^{-1} \nabla f(\bar{x}_i) \quad (\text{B.36})$$

where  $\bar{x}_0$  has been replaced by  $\bar{x}_i$  and  $\bar{x}$  by  $\bar{x}_{i+1}$ .

Since Newton's method implicitly uses a quadratic assumption on  $f$  (arising from neglecting the higher-order terms in the Taylor series expansion of  $f$ ), the Hessian need not be evaluated exactly. Rather the approximation of (B.33) can usually be used. The main advantage of this technique is rapid convergence. However, the rate of convergence is sensitive to linearity around the starting location.

We see that simple gradient descent and Gauss-Newton iteration are complementary in the advantages they provide. The LM algorithm is based on this observation, whose update rule is a blend of the above-mentioned algorithms and is given as

$$\bar{x}_{i+1} = \bar{x}_i - (\mathbf{H} + \alpha \mathbf{I})^{-1} f(\bar{x}_i) \quad (\text{B.37})$$

where  $\mathbf{H}$  is the Hessian matrix evaluated at  $\bar{x}_i$ . This update rule is used as follows. If the error goes down following an update, it implies that our quadratic assumption on  $f(\bar{x})$  is working and we reduce  $\alpha$  (usually by a factor of 10) to reduce the influence of gradient descent. On the other hand, if the error goes up, we would like to follow the gradient more and so  $\alpha$  is increased by the same factor. Thus:

### LM Algorithm

1. Do an update as according to (B.37).
2. Evaluate the error at the new parameter vector.
3. If the error has increased as a result of the update, then retract the step (i.e., reset the weights to their previous values) and increase  $\alpha$  by a factor of 10 or some such significant factor. Then go to (1) and try an update again.
4. If the error has decreased as a result of the update, then accept the step (i.e., keep the weights at their new values) and decrease  $\alpha$  by a factor of 10 or so.

The above algorithm has the disadvantage that if the value of  $\alpha$  is large, the calculated Hessian matrix is not used at all. We can derive some advantage out of the second derivative even in such cases by scaling each component of the gradient according to the curvature. This should result in *larger* movement along the directions where the gradient is *smaller* so that the classic “error valley” problem does not occur. This crucial insight was provided by Marquardt [6]. He replaced the identity matrix in (B.37) with the diagonal of the Hessian resulting in the LM update rule

$$\bar{x}_{i+1} = \bar{x}_i - [\mathbf{H} + \alpha \text{diag}(\mathbf{H})]^{-1} \nabla f(\bar{x}_i) \quad (\text{B.38})$$

Since the Hessian is proportional to the curvature of  $f$ , (B.38) implies a large step in the direction with low curvature (i.e., an almost flat terrain) and a small step in the direction with high curvature (i.e., a steep incline). We note that while the LM method is a heuristic, it works extremely well in practice. The major flaw is the need for a matrix inversion as part of the update. Even though the inverse is usually implemented using clever pseudoinverse methods such as SVD, the cost of the update becomes prohibitive after the model size increases to a few thousand parameters. For moderately sized models (of a few hundred parameters) however, this method is *much* faster than, say, vanilla gradient descent.

## B.5 Concluding Remarks

We discussed some nonlinear programming techniques in this appendix. Specifically we discussed the method of steepest descent, the Gauss-Newton method, and the Levenberg-Marquardt algorithm. These algorithms are used to find the minimum values of functions. They are certainly not the only algorithms for performing this function. They are provided as examples.

These techniques are included here because many of the approaches for geolocating target emitters rely on finding the minimum errors in functions, for example, least squares error and minimum mean-square error. Typically these functions are nonlinear and the algorithms discussed in this appendix can be used to find minima of nonlinear functions.

### References

- [1] Cauchy, A., "Methodes Generales Pour la Resolution des Systemes Dequations Simultanees," Academy of Science, Paris, Vol. 25, 1847, pp. 536–538.
- [2] Luenberger, D. G., and Y. Ye, *Linear and Nonlinear Programming*, New York: Springer, 2008.
- [3] Nash, S. G., and A. Sofer, *Linear and Nonlinear Programming*, New York: McGraw-Hill, 1996.
- [4] Barzilai, J., and J. Borwein, "Two-Point Step Size Gradient Methods," *IMA Journal of Numerical Analysis*, Vol. 8, 1988, pp. 141–148.
- [5] Levenberg, K., "A Method for the Solution of Certain Non-Linear Problems in Least Squares," *Quarterly of Applied Mathematics*, Vol. 2, No. 2, July 1944, pp. 164–168.
- [6] Marquardt, D. W., "An Algorithm for the Least-Squares Estimation of Nonlinear Parameters," *SIAM Journal of Applied Mathematics*, Vol. 11, No. 2, June 1963, pp. 431–441.
- [7] Nocedal, J., and S. J. Wright. *Numerical Optimization*, New York: Springer, 1999.
- [8] Kelley, C. T., *Iterative Methods for Optimization*. Philadelphia: SIAM Press, 1999.
- [9] Press, W. H., et al., *Numerical Recipes in C: The Art of Scientific Computing*, New York: Cambridge University Press, 1992.

# Acronyms

2-D	two-dimensional
3-D	three-dimensional
AAR	adaptive angular response
AM	amplitude modulation
AOA	angle of arrival
AOI	area of interest
ASE	aircraft survivability equipment
AWGN	additive white Gaussian noise
BLWN	band-limited white noise
BW	beamwidth
CAF	Cross-ambiguity function
CCF	cross-correlation function
CEP	circular error probable
CRB	Cramer-Rao bound
CSD	Cross-spectral density
DD	differential Doppler
DFT	discrete Fourier transform
DFT	direction finding
DME	distance measuring equipment
DOP	dilution of precision
DPD	discrete probability density

EEP	elliptical error probable
EKF	extended Kalman filter
EM	electromagnetic
EW	electronic warfare
FCC	Federal Communication Commission
FM	frequency modulation
FT	Fourier transform
GCC	generalized cross-correlaiton
GDOP	geometric dilution of precision
GPS	global positioning system
HF	high frequency
i.i.d.	independent and identically distributed
LCOA	location on the conic axis
LLSE	linear least-squares estimator
LMMSE	linear minimum mean-squares error
LMS	least-mean square
LO	local oscillator
LOB	line of bearing
LOP	line of position
LPI	low probability of intercept
LRT	likelihood ratio test
LSE	least-square error
LTl	linear time invariant
ME	maximum entropy
MEM	maximum entropy method
ML	maximum likelihood
MLE	maximum likelihood estimation

MLM	maximum likelihood method
MMSE	minimum mean-square error
MSC	multiple sample correlation
MSE	mean-square error
MUF	maximum usable frequency
MUSIC	multiple signal classification
NVI	near-vertical incidence
NVIS	near-vertical incidence skywave
PDF	probability density function
PEA	phase error amplitude
PF	position fix
PHAT	phase transform
PM	phase modulation
PX	plane intersection
RD	range difference
RGEA	relative gain error amplitude
RMS	root mean square
RWR	radar warning receiver
SAW	surface acoustic wave
SCOT	smoothed coherence transform
SDOA	scale difference of arrival
SI	spherical interpolation
SNR	Signal-to-noise ratio
SSL	single site location
SVD	singular value decomposition
SX	spherical intersection
TDOA	time difference of arrival
TID	traveling ionospheric disturbances

TLSE	total least-square error
TMA	target motion analysis
TNA	thermal noise algorithm
TOA	time of arrival
UAS	unattended aerial system
UAV	unattended aerial vehicle
UCA	uniform circular array
UKF	unscented Kalman filter
ULA	uniform linear array
UT	universal time
VHF	very high frequency
VSWR	voltage standing wave ratio
WLSE	weighted least-square error
wss	wide-sense stationary

# Symbols

<b>A</b>	Array manifold matrix
<b>C</b>	Covariance matrix
$C(t)$	Coherence function ( $t = k$ if discrete)
$\mathcal{C}^{m \times n}$	Array of $m \times n$ complex numbers
$\mathcal{C}$	Set of complex numbers
$D$	Time difference of arrival/time delay
$D$	Size of (discrete) array manifold
$\vec{e}$	Eigenvector
$f$	Frequency
<b>J</b>	Jacobian matrix
$L$	Number of signals
$M$	Number of sensors/antennas/receivers
$n(t)$	Noise waveform ( $t = k$ if discrete)
$N$	Number of samples
$\mathfrak{N}$	The set of natural numbers (integers)
$P$	Cross-correlation density
$P$	Power spectral density
$r(t)$	Received time waveform ( $t = k$ if discrete)
$R(\tau)$	Autocorrelation function
<b>R</b>	Data covariance matrix
$\mathfrak{R}^{m \times n}$	Array of $m \times n$ real numbers
$\mathfrak{R}$	The set of real numbers
$s(t)$	Transmitted time waveform ( $t = k$ if discrete)
$T$	Observation time
$V$	Variance
<b>V</b>	Variance matrix
$W$	Bandwidth
<b>W</b>	Weight matrix
$W(f)$	Weight function (of $f$ )
$\vec{\beta}$	Wavenumber
$\gamma$	Signal-to-noise ratio

$\phi$	Azimuth angle
$\lambda$	Wavelength
$\rho(t)$	Cross-correlation function ( $t = k$ if discrete)
$\sigma^2$	Variance
$\theta$	Elevation angle (usually measured from zenith)
$\vec{\theta}$	Unknown parameter vector

## About the Author

Richard A. Poisel received a B.S. in electrical engineering from the Milwaukee School of Engineering in 1969 and an M.S. in the same discipline from Purdue University in 1971. He spent three years in the military from 1971 to 1973. After his service he attended the University of Wisconsin, where he received a Ph.D. in electrical and computer engineering in 1977. From 1977 to 2004 he was with the same government organization, which has had several different names and is currently known as the U.S. Army Research, Development, and Engineering Command, Intelligence and Information Warfare Laboratory. During the 1993–1994 academic year, Dr. Poisel attended the MIT Sloan School of Management as a Sloan Fellow, receiving an M.B.A.

Initially a research engineer, Dr. Poisel eventually rose to the role of the director of the laboratory on an acting basis from 1997 to 1999. He was appointed chief scientist in 1999 and was relocated to the Army's Intelligence Center at Ft. Huachuca, Arizona, where he served as a technical advisor to the command group. Retiring from government service in 2004, he served as a Senior Engineering Fellow at Raytheon Missile Systems in Tucson, Arizona, from 2004 to 2011. He is currently a consultant on engineering for electronic warfare applications.

Dr. Poisel holds several patents in the communication electronic warfare subject area and is the author of the following, all published by Artech House: *Introduction to Communications Electronic Warfare Systems*, Second Edition, 2008; *Modern Communications Jamming Principles and Techniques*, Second Edition; *Target Acquisition in Communications Electronic Warfare Systems*; *Foundations of Communications Electronic Warfare*; *Electronic Warfare Target Location Methods*, 2005; and *Antenna Systems and Electronic Warfare Applications*, 2012.



# Index

- adaptive angular response 215
- adaptive eigenvalues decomposition 347
- Adcock 140
- additive white Gaussian noise (AWGN) 17, 112
- aircraft survivability equipment (ASE) 3
- angle of arrival (AOA) 1, 139, 141
- antisymmetry 377
- area of interest (AOI) 15
- array covariance 144
  
- Bailey curve 355
- beamforming 149, 169
- Butler matrix 168, 173, 175, 176, 178
  
- cardioid 160
- cepstrum 345
- circular error probable (CEP) 9
- common factor axiom 383
- complement axiom 393
- concentration 76
- controllability 108
- Cramer-Rao bound 316
- critical sum 288
- cross-correlation 296
- cross-correlation function 296
- cross-ambiguity function (CAF) 226, 267
  
- distance measuring equipment (DME) 122
- differential Doppler (DD) 2, 225, 255
- differential frequency 2
- dilution of precision 226, 244
- direction of arrival (DOA) 146
- discrete probability density (DPD) 15, 75
- Doppler 159
- dynamical systems 47
  
- Earth curvature 353
- Eckart filter 310
- eigenvalue decomposition 147
- eigenvalues 144
- electronic warfare (EW) 1
- elliptical error probable 8
- error covariance matrix 5
- Euclidean complement 392
- extended Kalman filter 63, 69
- exterior product 376
  
- extraordinary wave 340
  
- feasible bivectors 281, 285
- Fermat's principle 357
- finite sample time 323
- frequency modulation (FM) 271
- Frobenius norm 29
  
- Gauss-Newton 155, 399, 404
- generalized bearing 80, 135
- generalized correlation 347
- generalized cross-correlation 303
- generalized eigenvalue 144
- generalized inverse 286
- geometric dilution of precision (GDOP) 110, 247
- GPS 1
- gradient decent 3
- Gram-Schmidt orthogonalization 153
- Grassmann algebra 375
- Greenwich Mean Time (GMT) 339
  
- homomorphic deconvolution 349
  
- importance sampling 242
- independent and identically distributed (i.i.d.) 5
- inner product 394
- ionograms 338
- ionospheric propagation 337
- ionospheric sounders 339
- isochrone 235
  
- Jacobian 156
  
- Kalman filter 62
  
- lateral deviations 354
- least-square error (LSE) 15, 20, 30, 131, 151
- Least-squares 41, 274
- Levenberg-Marquardt 155, 399, 407
- likelihood ratio test (LRT) 272
- line of bearing (LOB) 1, 76
- line of position (LOP) 2, 139, 225
- linear least-squares error (LLSE) 315
- linear minimum mean-square error 52
- location on the conic axis (LOCA) 289

- maximum entropy method (MEM) 215
- maximum likelihood estimation (MLE) 15, 84, 92, 151, 215, 262, 310
- maximum usable frequency (MUF) 340
- mean-square error 52
- metric 376
- minimum mean-squares error 47, 159
- multipath reflection 344
- multiple sample correlation 99
- MUSIC 1, 139, 189
- MUSIC cepstrum 349
- mutual coupling 200
- m*-vectors 379
  
- navigation error 122
- near vertical incidence (NVI) 203
- near vertical incidence skywave (NVIS) 338
- noise eigenvalue 145
- nonlinear programming 154, 399
- nonlinear least-squares 233, 407
- normal equation 24
  
- oblique sounders 339
- oblique sounding 338
- observability 108
- ordinary wave 340
- orthogonality principle 53
  
- parabolic modeling 361
- parabolic interpolation 185
- particles 242
- passive SSL 341
- phase interferometry 156
- phase modulation (PM) 271
- phase transform 309
- plane intersection 277
- plane model 368
- plasma frequency profile 363
- position fix (PF) 1
- pressure scale height 359
- probability density function (PDF) 50
- pseudoinverse 28
  
- quadratic position-fixing 225
- quadrature hybrid 169
  
- radar warning receiver (RWR) 3
- range difference (RD) 2, 273
- ray tracing 355
- regressive product 380
- root mean-square (RMS) 78, 167
  
- Ross curve 355
- Roth process 307
- scalar product 394
- scale difference of arrival (SDOA) 252
- Schwartz inequality 305
- signal eigenvalue 145
- signal of interest (SOI) 199
- signal subspace 149, 190
- signal-to-noise ratio (SNR) 34, 218
- single site location (SSL) 335
- singular value decomposition (SVD) 28, 46
- smoothed coherence transform (SCOT) 308
- Snell's law 340
- spherical interpolation 276
- spherical intersection 277
- spherical model 365
- Stansfield 92
- static particle filter 240
- steepest descent 399
- steering vector 190
- stochastic gradient 237
- superresolution 213
- surface acoustic wave (SAW) 140, 180
- synchronization 321
  
- target motion analysis (TMA) 104
- Taylor series 6, 249, 297, 302
- thermal noise algorithm 215
- Tikhonov PDF 76
- time delay estimation 293
- time difference of arrival (TDOA) 1, 141, 167, 225, 227
- time of arrival (TOA) 1, 225
- total least-square error 15, 27, 45, 72
- triangulation 15
- trivector 287
  
- uniform circular array (UCA) 211
- uniform linear array (ULA) 199
- universal time 339
- unscented Kalman filter 74
  
- vertical sounders 339
- vertical sounding 338
- voltage standing wave ratio (VSWR) 170
  
- Watson-Watt 157
- wedge product 286, 376
- weighted least-square error (WLSE) 22
- wide-sense stationary (wss) 144