Analysis and Development of Blind Adaptive Beamforming Algorithms

Thomas E. Biedka

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Jeffrey H. Reed, Chair
Robert J. Boyle
Warren L. Stutzman
William R. Saunders
William H. Tranter

October 17, 2001
Blacksburg, Virginia

Keywords: Adaptive array, CMA, decision-directed, convergence analysis

Analysis and Development of Blind Adaptive Beamforming Algorithms

Thomas E. Biedka

(ABSTRACT)

This dissertation presents a new framework for the development and analysis of blind adaptive algorithms. An adaptive algorithm is said to be 'blind' if it does not require a known training sequence. The main focus is on application of these algorithms to adaptive antenna arrays in mobile radio communications. Adaptive antenna arrays can reduce the effects of cochannel interference, multipath fading, and background noise as compared to more conventional antenna systems. For these reasons, the use of adaptive antennas in wireless communication has received a great deal of attention in the literature.

There are several reasons why the study of blind adaptive algorithms is important. First, it is common practice to switch to a blind mode once the training sequence has been processed in order to track a changing environment. Furthermore, the use of a blind algorithm can completely eliminate the need for a training sequence. This is desirable since the use of a training sequence reduces the number of bits available for transmitting information.

The analysis framework introduced here is shown to include the well-known Constant Modulus Algorithm (CMA) and decision directed algorithm (DDA). New results on the behavior of the CMA and DDA are presented here, including analytic results on the convergence rate. Previous results have relied on Monte Carlo simulation. This framework is also used to propose a new class of blind adaptive algorithms that offer the potential for improved convergence rate.

Dedication

This dissertation is dedicated with love to Sarah.

Acknowledgments

I would like to take this opportunity to thank a few of the many people who have made the completion of this dissertation possible. This is a difficult task, since words alone can not express my gratitude to the people who have given both personal and technical guidance throughout this venture.

I would like to thank my colleagues at Raytheon Systems Company for their support and patience while I pursued my degree. In particular I would like to thank Tom Tillotson. I will always be grateful for his constant support and guidance. I have also learned a great deal from my close collaboration with Mark Kahn. He has contributed greatly to my growth and development both as an engineer and a person. I would also like to thank James Miller, John Payne, Steve Thornton, Darrell Young, and all the excellent engineers at Raytheon from whom I have learned so much. Special thanks for the extra effort required to take care of an offsite employee to Lisa Smith. I would also like to thank Emily Krzysiak of the US Air Force Rome Labs for her support and interest in our work. Through my work at E-Systems, and later Raytheon, I became friends with Dr. Brian Agee, whose work has formed the foundation for this dissertation. His insights and suggestions have been generous and helpful. I also made the acquaintance of Dr. Stephen Schell through my work at Raytheon. I value his friendship and advice greatly, and am thankful for his support and encouragement.

I would like to thank my colleagues and friends at Virginia Tech. First and foremost my thanks go to my committee chair Dr. Jeffrey Reed. His faith in my abilities, even when I had my own doubts, was instrumental in reaching my objectives. A heartfelt thank you to Dr. William Tranter. He gave his time and effort so generously, and showed me the true spirit of academic collaboration. Thanks also go to the rest of my committee members; Dr. Robert Boyle, Dr. William Saunders, and Dr. Warren Stutzman. I truly appreciate the time and effort they gave to serve on my PhD committee. My fellow students also contributed to the fullness of my experience at Virginia Tech, especially through the technical meetings held at PK's and the Cellar. Thank you to Audrey Ald, Carl Dietrich, Francis Dominique, Rich Ertel, Joe Liberti, James Hicks, Kevin and Donna Krizman, Raqib Mostofa, Bill Newhall, Paul Petrus, Max Roberts, Janet Wojcik, and others too numerous to include here. As with

any organization like MPRG, the real work is done by the administrative staff. Thank you for all the help and friendship – Jenny Frank, Rennie Givens, Cyndy Graham, Lori Hughes, Prab Koushik, Hilda Reynolds, and Aurelia Scharnhorst. An extra special thank you to Shelby Smith for taking care of so many details for my final defense.

I also want to acknowledge members of the Physical Education Teacher Education (PETE) family at Virginia Tech, including Dr. George Graham, Judy Graham, Mark Manross, Suzy Parker, Sonja Stone, and many others. I have been touched by the friendship and support you have given to both my wife Sarah and myself throughout our time at Virginia Tech.

I would like to thank the people at Tropian, Inc., my current employer, for their support while I completed my dissertation. Dr. Earl McCune, Dr. Gary Do, Wayne Lee, Dr. Paul Liang, Dr. Matt Mow, and Nancy Taylor – thank you.

Most importantly, I would like to thank my family for their love and support. The value that my father Frank and my mother Virginia placed on education and hard work gave me the ability to pursue my dreams. The love and support of my wife Sarah, to whom this dissertation is dedicated, has made this all worthwhile. Finally, I would be remiss not to thank my son, Paul Frank, for the extra bit of motivation to finish my dissertation in a timely manner.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The principal objective of this dissertation is to establish a new general framework for the development and analysis of blind adaptive algorithms for antenna arrays. In order to describe the contributions of this dissertation, the concepts of antenna array processing and blind adaptive algorithms must be introduced. Array processing is a broad area of study that includes any application where multiple antennas (or other sensors) are used to receive or transmit electromagnetic waves. Using multiple antennas in a receiver can reduce the effects of cochannel interference, multipath fading, and background noise. For these reasons, the use of antenna arrays in wireless communication has received a great deal of attention in the literature (e.g., see [2, 3, 4, 5, 6] and references therein). The potential benefits include improved geographical coverage of the service area, extended range, increased capacity, and improved quality of service. An array forms an improved estimate of the desired signal by weighting and summing the signals received at multiple spatially separated antennas, as illustrated in Figure 1.1. By appropriately selecting the weights, high gain can be placed in the direction of a desired signal, and low gain can be placed in the direction of interfering signals. This process is often referred to as beamforming, or spatial filtering [1]. The weighting applied to the signal received at each antenna may be fixed, or may be continuously adjusted, i.e., adapted, to track changes in the signal environment. The adaptive algorithm used to adjust the weights for each antenna is the focus of this dissertation.

The performance of the adaptive algorithm determines how well the array will perform relative to the optimal. The optimal approach generally requires the inclusion of a known training sequence in the desired signal. If the signal environment is changing, as is typically the case in mobile radio applications, a training sequence must be transmitted regularly. This reduces the amount of information that can be transmitted, which clearly is not de-

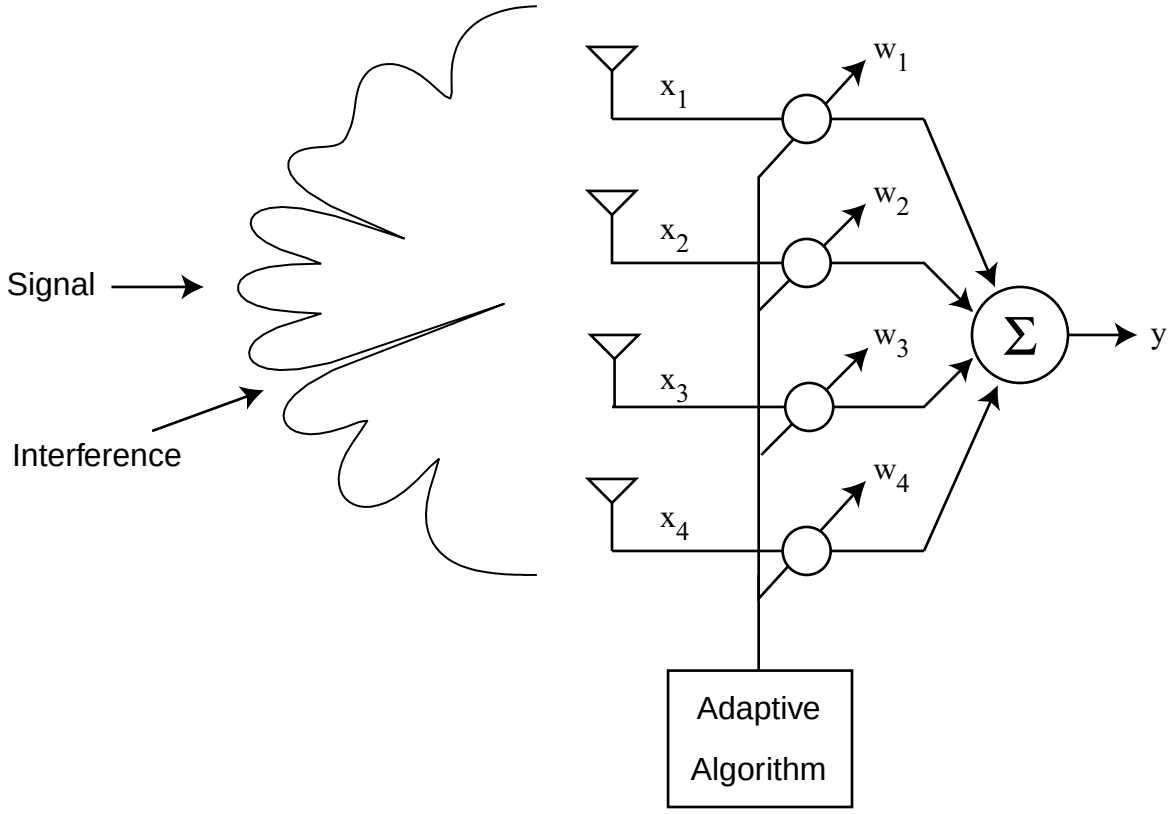

Figure 1.1: Conceptual block diagram of adaptive beamforming for narrowband signals.

sirable. An adaptive algorithm that does not require a training sequence, or other special knowledge of the environment, is known as a *blind* adaptive algorithm. The use of a blind algorithm can potentially eliminate the need for training sequences, thereby increasing the available data rate. However, blind algorithms have some drawbacks relative to conventional training sequence-based algorithms. First, blind algorithms cannot in general be guaranteed to converge to the desired solution, unlike the case when a known training sequence is used. Furthermore blind adaptive algorithms generally converge more slowly. For these reasons it is important to understand the behavior of existing blind algorithms, and develop new algorithms that offer faster and, perhaps more importantly, more reliable convergence. Towards this goal, this dissertation presents a new framework for the analysis and development of blind adaptive algorithms. The focus is on adaptive antenna array applications, but these algorithms can also be used in temporal (equalization) applications.

One commonly used blind adaptive algorithm is known as the Constant Modulus Algorithm (CMA). The CMA can be applied to any signal that has a constant (or nearly constant) envelope. Examples of such signals include FM, FSK, and PSK. Despite the wide use of CMA

in practical applications, its convergence behavior is poorly understood. The behavior of the CMA has principally been studied through computer simulation. Chapter 3 introduces an analysis framework that leads to important new *analytic* results on the convergence behavior of CMA. These analytic results can be used to determine the CMA convergence rate, and also lead to new insight regarding general algorithm behavior. The results presented in this chapter have been published in [7].

Another commonly used class of blind adaptive algorithms are known as Decision Directed Algorithms (DDA). A DDA can be applied to any digitally modulated signal. This algorithm is implemented by demodulating the beamformer (or equalizer) output, then treating the resulting estimated symbols in the same way that a known training sequence would be treated. This algorithm is used extensively in conjuction with a training sequence approach to track a changing environment during time intervals where a training sequence is not present. The analysis framework introduced in Chapter 3 can also be applied to a DDA, as shown in Chapter 4 and Chapter 5. Chapter 4 examines the behavior of a decision directed algorithm in Gaussian noise and interference, while Chapter 5 examines the behavior in constant modulus interference. Several new results have been obtained, including

1. a determination of the DDA convergence rate;
2. a comparison of CMA and DDA convergence rate;
3. a modified DDA that incorporates carrier phase estimation for greatly improved convergence rate;

Some of these results were presented in [8]. The algorithm studied in Chapter 4 has also been studied in [9], which has been widely cited in the literature, including the following peer-reviewed journal publications: [10, 11, 12, 13, 14, 15, 16]. However, the analysis in [9] leads to distinctly different conclusions than those presented in Chapter 4. It is believed that the results in Chapter 4 are more accurate. This conclusion is supported by extensive simulation results, and by a detailed examination of the analysis in [9], which is presented in Section 4.4.

The final chapter in this document, Chapter 6, outlines a number of potential directions for future research. Some of the topics discussed are nearly mature and include preliminary results. Other topics are less mature and are only briefly discussed. One important area for future research is the use of the framework introduced here to develop new blind algorithms. A new algorithm called the Least Squares General Modulus Algorithm (LSGMA) is introduced. This algorithm is very similar to a CMA, except that a general magnitude non-linearity is used in place of the hard-limit non-linearity used in CMA. The LSGMA

offers faster convergence relative to the LSCMA in some situations. Other potential areas for future research are outlined as well, including the behavior of CMA and DDA with a finite data sequence, the extension of the general analysis framework to include extraction of multiple signals, and exploitation of multiple non-linearities for blind adaptation.

As with any research endeavor, publication is important both for dissemination and for the opportunity to interact with other researchers in the field. The following journal articles and conference papers have been published as a result of the work outlined in this dissertation, or describe previous work that has been used in the preparation of this document:

- T.E. Biedka, J.H. Reed, and W.H. Tranter, "Statistics of Blind Spatial Signature Estimators", Proc. of the Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, November 2000.
- T.E. Biedka, W.H. Tranter, and J.H. Reed, "Convergence Analysis of the Least Squares Constant Modulus Algorithm in Interference Cancellation Applications", *IEEE Trans. on Communications*, March 2000, pp. 491-501.
- T.E. Biedka, J.H. Reed, and W.H. Tranter, "Mean Convergence Rate of a Decision Directed Adaptive Beamformer with Gaussian Interference", *First IEEE Sensor Array and Multichannel Signal Processing Workshop*, March 2000.
- T. Biedka, B. Holden, S. Thornton, W. Ferguson, R. Hammons, B. Johnson, S. Kailas, V. Liau, A. Paulraj, and S. Sandhu, "Implementation of a Prototype Smart Antenna for Low Tier PCS", *Proc. of the IEEE Vehicular Technology Conference*, Houston, TX, May 1999
- T.E. Biedka, "A Comparison of Initialization Schemes for Blind Adaptive Beamforming", *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1998, pp. 1665-1668.
- K.J. Krizman, T.E. Biedka, and T.S. Rappaport, "Wireless Position Location: Fundamentals, Implementation Strategies, and Sources of Error" , *Proc. of IEEE Vehicular Technology Conference*, May 1997. (Also included in T.S. Rappaport, ed., *Smart Antennas: Adaptive Arrays, Algorithms, and Wireless Position Location*, IEEE Press, 1998)
- T.E. Biedka, W.H. Tranter, and J.H. Reed, "Convergence Analysis of the Least Squares Constant Modulus Algorithm" , *Proc. of the Thirtieth Asilomar Conference on Signals, Systems, and Computers*, Nov 1996.

- T.E. Biedka, J.H. Reed, and B.D. Woerner, "Direction Finding Methods for CDMA Systems", *Proc of the Thirtieth Asilomar Conference on Signals, Systems, and Computers*, Nov 1996.

- T.E. Biedka, "A Method for Reducing Computations in Cyclostationarity-Exploiting Beamforming", *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1995.

- T.E. Biedka and M.F. Kahn, "Methods for Constraining a CMA Beamformer to Extract a Cyclostationary Signal" , *Proc. of the Second Workshop on Cyclostationary Signals*, Monterey, CA, August 1994.

- M.F. Kahn, M.A. Mow, W.A. Gardner, and T.E. Biedka, "A Recursive Programmable Canonical Correlation Analyzer", *Proc. of the Second Workshop on Cyclostationary Signals*, Monterey, CA, August 1994.

- T.E. Biedka, "Subspace-Constrained SCORE Algorithms", *Proc. of the Twenty Seventh Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, November, 1993.

Chapter 2

Adaptive Antenna Arrays in Mobile Wireless Applications

This chapter presents a brief overview of adaptive beamforming in mobile wireless communication applications. The motivation for using adaptive beamforming is that such systems can reduce the effects of cochannel interference, multipath fading, and background noise as compared to more conventional antenna systems. The potential benefits include improved geographical coverage of the service area, extended range, increased capacity, and improved quality of service. For these reasons, the use of adaptive array systems in wireless communication has received a great deal of attention in the literature (e.g., see [2, 3, 4, 5, 6] and references therein). These systems exploit the spatial separation of co-channel signals in order to extract a desired signal and reject co-channel interference. This is accomplished by weighting and summing the signals received at multiple antennas in such a way that the desired signal combines coherently, while the undesired noise and interference combines incoherently. This process is illustrated conceptually in Figure 1.1.

When applied in mobile communications, an adaptive antenna array is often referred to as a form of *smart antenna*. This term is generally meant to include switched beam systems and steered beam systems as well. The distinguishing feature of smart antennas, as opposed to more conventional antennas, is the ability to change the effective antenna pattern. The various versions of smart antennas differ principally in the degree to which the antenna pattern can be controlled. A switched beam or steered beam system attempts to place maximum gain towards the bearing of the desired user. The system is only able to control the bearing where the antenna pattern is maximum. The performance of an adaptive array will always be better than that of a steered or switched beam system and is therefore the focus of this tutorial. However, an adaptive array is generally more complicated, and hence more

expensive, than a switched/steered beam system. In some applications the added expense of a fully adaptive array may not warrant the performance improvement over a simpler switched or steered beam system. The relative performance of a steered beam system versus a fully adaptive system can depend greatly on the propagation environment (e.g., the angle spread) and the multiple access scheme (e.g., TDMA versus CDMA).

This chapter presents an overview of some of the concepts and methods used in adaptive antenna array systems. The tutorial material presented here is only intended as a brief review of adaptive array techniques and their application in mobile radio communications. The reader is directed to more detailed information found in the cited references as appropriate. This overview is organized into the following general topics:

1. mathematical background;
2. mobile radio propagation environment; and
3. adaptive algorithm review;

2.1 Mathematical Background

In this section some of the common concepts and terminology associated with beamforming are defined and discussed. Important topics include:

1. the narrowband model for array data;
2. the array response vector;
3. the antenna array beampattern;
4. ambiguities in the array response vector;
5. array aperture; and
6. the optimal weight vector.

2.1.1 Narrowband Model

A fundamental concept in adaptive beamforming is the *narrowband model* for the signals received by the array of antennas. In this model, a small time delay (relative to the inverse signal bandwidth) is replaced with a simple carrier phase shift. The individual antennas in

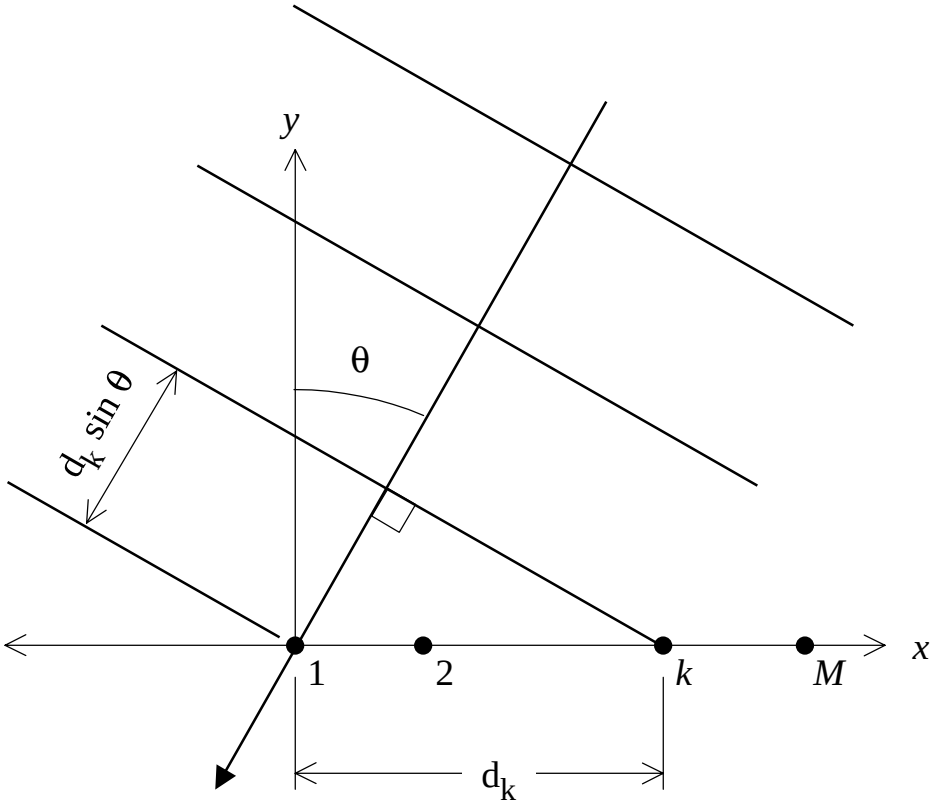

Figure 2.1: Time difference of arrival for a plane wave signal received by an array of M antennas, illustrating the dependence on signal angle of arrival (AOA) θ .

an array are generally closely spaced (on the order of centimeters, or meters in the extreme). The time delay seen by a signal as it propagates across an antenna array will therefore be on the order of nanoseconds. It is important to recognize that the narrowband model is used to describe the difference in the signal received at one antenna relative to another antenna. The use of the narrowband model for the array data does not preclude the study of frequency selective channels.

Consider a situation where a plane wave signal is received by an array of M antennas, as illustrated in Figure 2.1. The signal observed at one antenna will be a delayed version of the signal observed at some other antenna. Let antenna #1 be the reference antenna. Denote the signal observed at this antenna as

$$x_1(t) = s(t) + q_1(t) \tag{2.1}$$

where $s(t)$ is the incident signal and $q_1(t)$ is the background environmental noise and receiver

noise. Then the signal observed at the kth antenna will be

$$x_k(t) = s(t + \tau_k) + q_k(t) \tag{2.2}$$

where τ_k is the time difference of arrival (TDOA). By examination of Figure 2.1 it can be seen that

$$\tau_k = \frac{d_k \sin(\theta)}{c}. \tag{2.3}$$

where d_k is the separation of the first and kth antenna, c is the propagation speed, and θ is the AOA of the signal. For an electromagnetic wave propagating in free space, $c \approx 3.0 \times 10^8$ meters/sec. Note that if the signal arrives from 0° (broadside to the array) the TDOA is equal to zero. We will now show that if τ_k is small, the signals observed at each antenna differ only by a phase shift.

Since we are dealing with communication signals, where the carrier frequency can be many orders of magnitude greater than the bandwidth, the *analytic* representation of the received signals is useful to this discussion. Assume that the signal is bandpass, so that its spectrum is limited to a certain band of frequencies centered about the carrier frequency ω_c. The signal can be expressed as

$$s(t) = s_I(t) \cos(\omega_c t) - s_Q(t) \sin(\omega_c t) \tag{2.4}$$

where $s_I(t)$ and $s_Q(t)$ are the In-phase and Quadrature components, respectively, of the signal. Because the original signal $s(t)$ is bandpass, the signals $s_I(t)$ and $s_Q(t)$ are *low-pass*. The *complex envelope* of the signal $s(t)$ is defined as

$$\tilde{s}(t) = s_I(t) + j s_Q(t) \tag{2.5}$$

This is sometimes referred to as the *complex baseband* representation of the signal. The signal can also be represented as

$$s(t) = r(t) \cos(\omega_c t + \phi(t)) \tag{2.6}$$

where

$$r(t) = \sqrt{s_I^2(t) + s_Q^2(t)} \tag{2.7}$$

is the magnitude and

$$\phi(t) = \tan^{-1} \frac{s_Q(t)}{s_I(t)} \tag{2.8}$$

is the phase of the information bearing portion of the signal. The original signal can be recovered from the complex envelope via

$$s(t) = \text{Real}\left\{\tilde{s}(t) \exp(j\omega_c t)\right\} \tag{2.9}$$

Now consider the effect of adding a time shift τ_k to the signal. Using the complex baseband representation of the signal, we have

$$s(t+\tau_k) = \text{Real}\left\{\tilde{s}(t+\tau_k)\exp(j\omega_c t)\exp(j\phi_k)\right\} \tag{2.10}$$

where

$$\phi_k = \omega_c \tau_k. \tag{2.11}$$

Because the signal is bandpass, the complex envelope $\tilde{s}(t)$ is slowly varying relative to the carrier period. Therefore

$$\tilde{s}(t+\tau_k) \approx \tilde{s}(t) \tag{2.12}$$

and

$$\begin{aligned} s(t+\tau_k) &\approx \text{Real}\left\{\tilde{s}(t)\exp(j\omega_c t)\exp(j\phi_k)\right\} &(2.13)\\ &\approx s_I(t)\cos(\omega_c t+\phi_k) - s_Q(t)\sin(\omega_c t+\phi_k) &(2.14)\end{aligned}$$

This shows that the effect of a small time delay τ_k can be accurately modeled as a simple phase shift ϕ_k *if* the delay is small relative to the inverse bandwidth of the signal. Note that if we express the phase shift ϕ_k in terms of the signal carrier wavelength λ_c,

$$\phi_k = \omega_c \tau_k = \left(\frac{2\pi c}{\lambda_c}\right)\left(\frac{d_k \sin\theta}{c}\right) = \frac{2\pi d_k \sin\theta}{\lambda_c} \tag{2.15}$$

In the remainder of this document we will assume that the received signals are in a complex baseband representation, or are 'nearly' at baseband, which would correspond to some relatively small error in carrier frequency knowledge at the receiver. Some of the practical issues involved in converting a signal to a complex baseband format are addressed in Section 2.6.

Assuming that the received signal is in a complex baseband representation, we have

$$s(t+\tau_k) \approx a_k(\theta)s(t) \tag{2.16}$$

where the complex scalar

$$a_k(\theta) = e^{j\phi_k} \tag{2.17}$$

and ϕ_k is given by (2.15). In some cases, the phase shift ϕ_k may not be a unitary function of AOA. If

$$a_k(\theta_1) = a_k(\theta_2)\ \forall\ k \text{ with } \theta_1 \neq \theta_2, \tag{2.18}$$

then the angles θ_1 and θ_2 are said to correspond to an *array ambiguity*. As an example, the array geometry shown in Figure 2.1 exhibits an ambiguity at $\theta_2 = \pi - \theta_1$, since $\sin\theta = \sin(\pi-\theta)$. This ambiguity will exist for any linear array, i.e., an array whose antennas are distributed along a straight line. Other ambiguities will exist if the separation between

sensors is larger than $d_k = \lambda_c/2$. This can be seen by the following. Ignore the unavoidable ambiguity between θ and $\pi - \theta$ by considering only $-\pi/2 < \theta < +\pi/2$. If θ is restricted to this range, $-1 \leq \sin(\theta) \leq 1$. To avoid ambiguities in this interval, the phase ϕ must be strictly between $-\pi$ and π. That is, the phase must not wrap. This implies

$$\pi > \frac{2\pi d \sin\theta}{\lambda_c} \tag{2.19}$$

Noting that the maximum value of $\sin\theta$ is $+1$, we must have

$$\pi > \frac{2\pi d}{\lambda_c} \tag{2.20}$$

$$d < \frac{\lambda_c}{2} \tag{2.21}$$

In general, array ambiguities are to be avoided because the array cannot distinguish a desired signal and interfering signal at ambiguous AOAs. Thus, for example, the array would not be able to null the interference without also nulling the desired signal. It should be emphasized that this does *not* imply that the interelement spacing must always be less than $\lambda_c/2$ when the number of antennas is greater than 2. This issue is addressed later in Subsection 2.1.6.

Since we are dealing with multi-dimensional data, it is convenient to work with a vector representation of the data. To accomplish this, we introduce the array response vector. The array response vector describes the amplitude and phase changes seen by a signal as it propagates across the array of antennas. The array response vector for a signal incident from θ is the $M \times 1$ complex vector given by

$$\mathbf{a}(\theta) = \begin{bmatrix} 1 \\ a_2(\theta) \\ \vdots \\ a_M(\theta) \end{bmatrix} \tag{2.22}$$

where M is the number of antennas in the array. Note that the first entry in $\mathbf{a}(\theta)$ is unity, since this corresponds to the reference antenna. In practice, each antenna may have different gain, and each channel of the receiver may have different gain. Furthermore the effects of antenna coupling, uncertainty in antenna position, and near field scattering make it unlikely that the phase shift seen at each antenna will be exactly given by (2.15). The process of determining the array response $\mathbf{a}(\theta)$ as a function of θ is known as *array calibration*, and is accomplished by placing a transmitter at a known angle θ and estimating $\mathbf{a}(\theta)$ from the received data. This is a non-trivial task, particularly if accurate knowledge of the array response is required over a range of carrier frequencies. For these reasons, the array response vector should be treated in practice as an unknown complex vector whose value is, at best, only approximately known.

To complete the introduction of the narrowband model, we wil use it to describe the data received by an antenna array. The $M \times 1$ complex vector

$$\mathbf{x}(n) = \begin{bmatrix} x_1(n) \\ x_2(n) \\ \vdots \\ x_M(n) \end{bmatrix} \tag{2.23}$$

will be referred to as the observed data vector at discrete time instant n. $\mathbf{x}(n)$ is also sometimes referred to as a data snapshot. When only one signal is incident on the array,

$$\mathbf{x}(n) = \mathbf{a}(\theta)s(n) + \mathbf{q}(n) \tag{2.24}$$

where the $M \times 1$ complex vector $\mathbf{q}(n)$ contains background noise and receiver noise. When there are L incident signals,

$$\mathbf{x}(n) = \sum_{i=1}^{L} \mathbf{a}(\theta_i)s_i(n) + \mathbf{q}(n) \tag{2.25}$$

or alternatively

$$\mathbf{x}(n) = \mathbf{A}\mathbf{s}(n) + \mathbf{q}(n) \tag{2.26}$$

where the ith column of the $M \times L$ matrix $\mathbf{A}$ is the array response vector for the ith signal, and the $L \times 1$ matrix $\mathbf{s}(n)$ contains the source signals.

The narrowband model is exact only for sinusoidal signals. However, the model is generally a good approximation in many situations. The most obvious source of deviation from the narrowband model is the actual bandwidth of the incident signal. As a rule of thumb, the ratio of the signal bandwidth B to the carrier frequency f_c should be small. This ratio is often referred to as the relative bandwidth. Values less than 10%, i.e. $B/f_c < 0.1$, are generally considered small. However, the phase difference between sensors depends on signal bandwidth *and* the time delay between sensors. The phase difference can change significantly over the bandwidth of the signal if the signal bandwidth is large *or* if the array is large. For example, a signal impinging on a linear array from near broadside results in small relative delays from sensor to sensor, and so the effects of signal bandwidth will be small. However, the same signal incident on the same array from an angle well off broadside may result in large relative delays, making the narrowband model inappropriate. As a counter example, the signal may have a small relative bandwidth, but if the sensors are very far apart, the signal observed at one sensor will not be simply a phase-shifted version of the signal observed at another. An extreme example would be a TDOA geolocation system where multiple widely spaced sensors are used.

In practice, any deviation from the narrowband model is detrimental to the performance of a narrowband beamformer. This is usually manifested as a limit in the ability to null interference, i.e., a limit in the null depth. This means that if very deep nulls are desired, the narrowband model must be truly appropriate, or a *wideband* beamformer must be used. A wideband beamformer uses tapped delay lines or a frequency channelized architecture to accommodate the frequency-dependent nature of the array response. Conversely, if very deep nulls are not required, more deviation from the narrowband model can be tolerated.

2.1.2 Antenna Weighting and Antenna Array Beampatterns

The signals received at an array of antennas can be weighted and summed to improve the quality of the desired signal, as illustrated in Figure 1.1. This process is described mathematically by

$$y(n) = \sum_{i=1}^{M} w_i^* \, x_i(n) \tag{2.27}$$

where w_i^* is the weight applied to the data received at the ith sensor. It is a common convention to apply the conjugate of the weight because it simplifies later math. Using matrix notation, the antenna array output can be written as

$$y(n) = \mathbf{w}^H \mathbf{x}(n) \tag{2.28}$$

where $\mathbf{w}$ is the $M \times 1$ vector of complex antenna weights and $(\cdot)^H$ denotes conjugate transpose. $\mathbf{w}$ is commonly referred to as a beamformer weight vector. Substituting the model for the received data (2.25) into the expression for the array output, we have

$$y(n) = \sum_{i=1}^{L} \mathbf{w}^H \mathbf{a}(\theta_i) s_i(n) + \mathbf{w}^H \mathbf{q}(n) \tag{2.29}$$

It can be seen that the inner product of the weight vector $\mathbf{w}$ and the array response vector $\mathbf{a}(\theta)$ determines whether a signal received from angle θ is nulled or passed. The beampattern, defined as

$$S(\theta) = \left| \mathbf{w}^H \mathbf{a}(\theta) \right|^2 \tag{2.30}$$

describes the gain versus AOA for a particular weight vector. The beampattern is in many ways analogous to the magnitude of the frequency response of an FIR filter. Care should be taken when using the beampattern to determine the behavior of a beamformer, since only the magnitude response is shown. In particular, if the environment contains multipath, the phase response may be equally important.

An example of an array beampattern is shown in Figure 2.2. The array contains 4 antennas, and these antennas are arranged along a straight line, with the inter-element spacing equal

to $\lambda/2$, with λ being the carrier wavelength. This would be described succinctly as a Uniform Linear Array (ULA) with $\lambda/2$ interelement spacing. The antenna weights are all equal to unity, that is, $\mathbf{w} = [\ 1\ 1\ 1\ 1\ ]^T$. The Half Power Beamwidth (HPBW) is an important parameter of any antenna. This is the width of the main beam of the antenna gain pattern 3 dB below the maximum gain. For a large ULA, the HPBW (in radians) is approximately

$$HPBW = 1/L_\lambda \tag{2.31}$$

where L_λ is the array *aperture* in wavelengths [19]. The array aperture is the length of the array. Clearly the HPBW decreases as the array aperture increases. This is demonstrated by Figure 2.3. This shows the beampattern for a 16-element ULA with the weights again equal to unity. Note that the main beam is much narrower than for the 4-element ULA. However, the sidelobe height is not much lower than for the 4-element array.

The sidelobe height can be reduced by applying a non-uniform weighting to the antennas. This is illustrated in Figure 2.4. Here the 16-element ULA is weighted with a Hamming window. The figure clearly shows that the sidelobe height has been greatly reduced. Other antenna weights can be chosen to trade mainlobe width for sidelobe height. This is very similar to the use of a data window in FIR filter design.

The weight vector $\mathbf{w}$ can also be used to steer the beam towards a desired direction. In order to have maximum gain in a certain direction, the weight vector must compensate for the phase shift (delay) corresponding to the signal's AOA. This allows each copy of the received signal to combine coherently, while signals arriving from other angles will combine non-coherently. This is accomplished by setting the weight vector equal to the array response vector for the desired angle. This is why an array response vector is also known as a *steering* vector. We show later in Subsection 2.1.3 that $\mathbf{w} = \mathbf{a}(\theta)$ is the optimal weighting for a single signal received from angle θ in spatially white noise. An example of beamsteering with a 16-element ULA is shown in Figure 2.5. The weight vector $\mathbf{w}$ is set equal to the array response vector $\mathbf{a}(\theta)$ for $\theta = 15^\circ$. The figure clearly shows that this results in maximum gain at $\theta = 15^\circ$. Note that the mainlobe width is also larger than was the case when the array was steered towards broadside (0°). This is due to the fact that the effective aperture is smaller for a signal received from off broadside. That is, from the perspective of the transmitter, the array aperture appears smaller when the signal AOA is off broadside. For a ULA, the aperture is maximum at broadside, and is minimum at endfire ($\theta = \pm 90^\circ$), where the effective aperture is zero. The effective aperture of a linear array is

$$L_{\text{eff}} = L_\lambda \sin\theta \tag{2.32}$$

In order to reject co-channel interference, it is clearly desirable to minimize mainlobe width and sidelobe height. In order to accomplish this, the number of antennas in the array

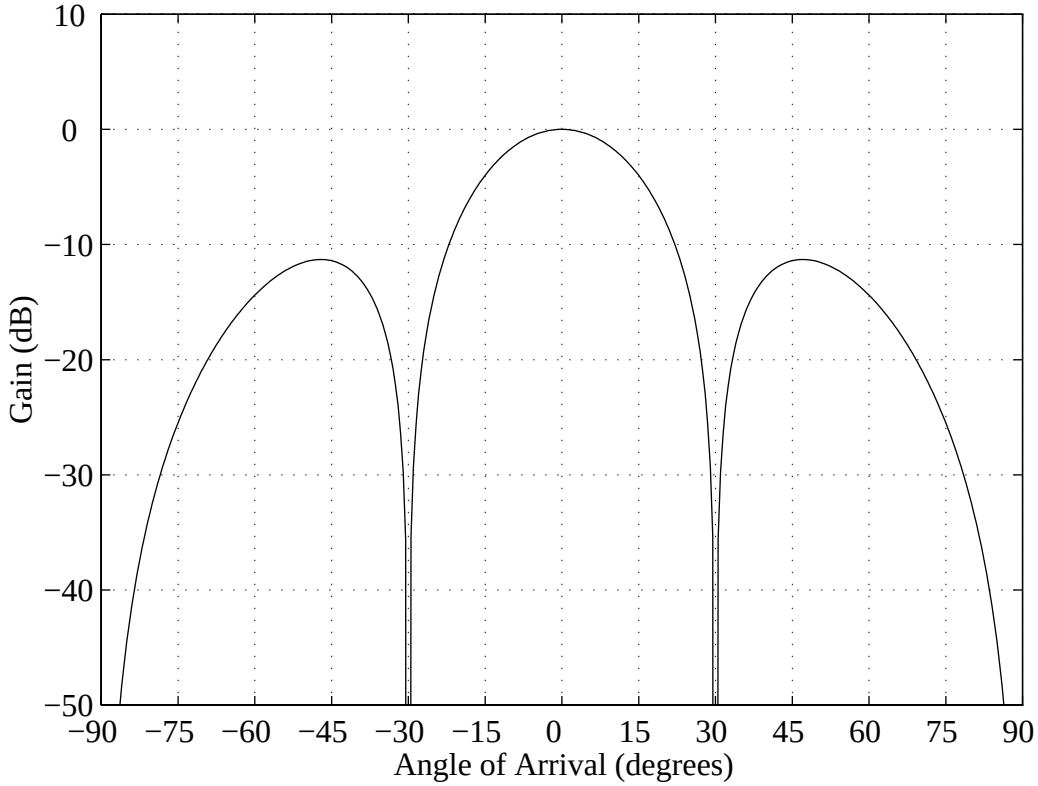

Figure 2.2: Beampattern for a uniformly weighted 4-element ULA with interelement spacing equal to $\lambda_c/2$.

must be fairly large, which is impractical in many applications. Furthermore, the antenna weighting methods discussed above are not able to reject co-channel interference that falls in the mainlobe of the antenna pattern. For this reason we turn to other antenna weighting methods which offer greatly improved performance. This is the topic of the next subsection.

2.1.3 Optimal Weight Vector

A reasonable approach to finding the optimal weight vector is to maximize the Signal to Interference and Noise Ratio (SINR) in the beamformer output. This will maximize the beamformer gain in the direction of the desired signal while simultaneously minimizing the gain towards the noise and interference. Assume that the array receives multiple signals. The beamformer output SINR for the kth signal can be written as

$$\rho_k = \frac{\sigma_k^2 \mathbf{w}^H \mathbf{a}_k \mathbf{a}_k^H \mathbf{w}}{\mathbf{w}^H \mathbf{R}_{ii} \mathbf{w}} \tag{2.33}$$

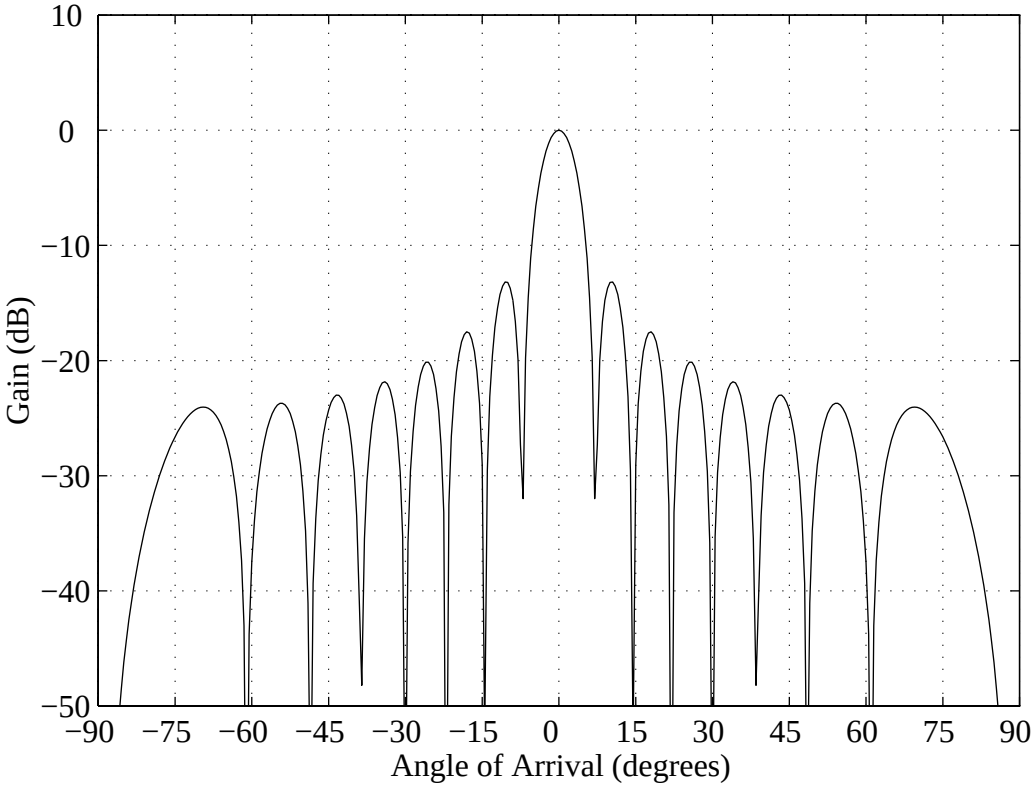

Figure 2.3: Beampattern for a uniformly weighted 16-element ULA with interelement spacing equal to $\lambda_c/2$.

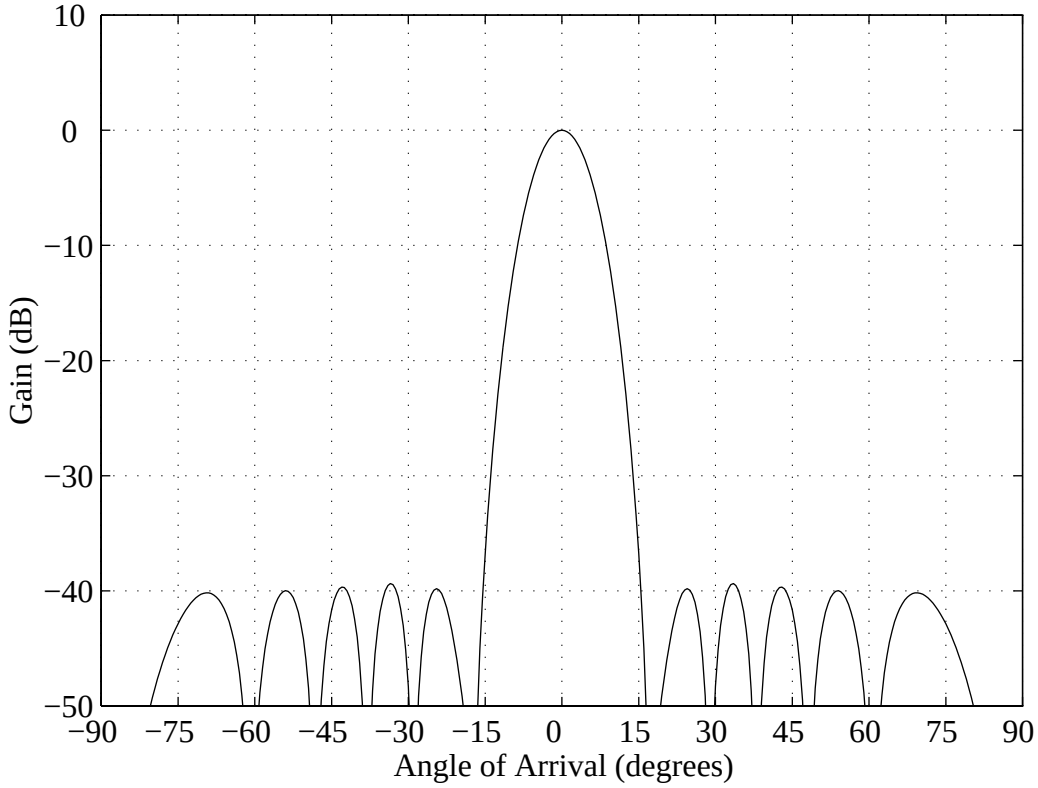

Figure 2.4: Beampattern for a Hamming weighted 16-element ULA with interelement spacing equal to $\lambda_c/2$.

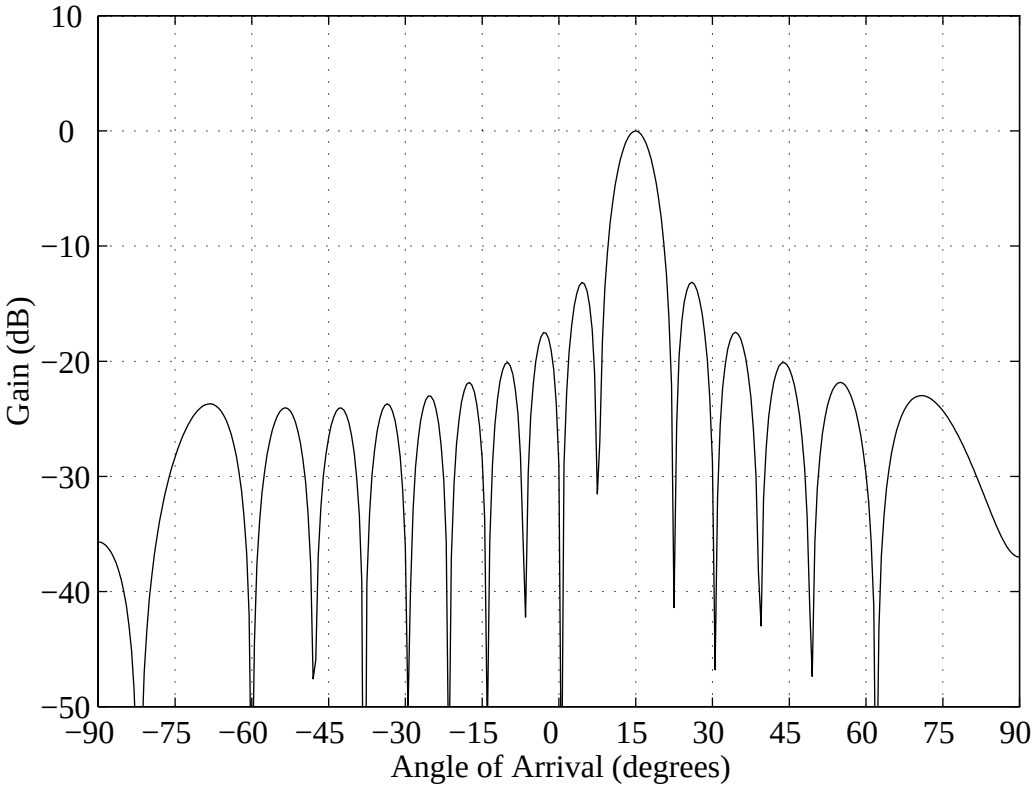

Figure 2.5: Beampattern for a uniformly weighted 16-element ULA that has been steered to 15°.

where σ_k^2 is the power of the kth signal,

$$\mathbf{R}_{ii} = \sum_{i=1, i\neq k}^{L} \sigma_i^2 \mathbf{a}_i \mathbf{a}_i^H + \mathbf{R}_{qq} \tag{2.34}$$

is the noise and interference covariance matrix, and

$$\mathbf{R}_{qq} = \mathcal{E}\Big\{ \mathbf{q}(n)\mathbf{q}^H(n) \Big\} \tag{2.35}$$

is the noise covariance matrix. To find the weight vector that maximizes the output SINR (2.33), we will make use of the Cauchy-Schwarz inequality for complex vectors, which states that for any two complex vectors $\mathbf{u}$ and $\mathbf{v}$,

$$\mathbf{u}^H \mathbf{v}\mathbf{v}^H \mathbf{u} \leq \mathbf{u}^H \mathbf{u}\mathbf{v}^H \mathbf{v} \tag{2.36}$$

with equality if and only if $\mathbf{u} \propto \mathbf{v}$, where $\propto$ is read as 'proportional to'. Define the vector

$$\mathbf{u} \triangleq \mathbf{R}_{ii}^{1/2} \mathbf{w} \tag{2.37}$$

Note that $\mathbf{R}_{ii}$ is Hermitian and full-rank. The only way that $\mathbf{R}_{ii}$ could not be full rank is if no thermal noise were present in the antennas and the receivers. Therefore $\mathbf{R}_{ii}^{-1/2}$ is defined in the natural way, and is a Hermitian matrix. The output SINR (2.33) can then be written as

$$\rho_k = \frac{\mathbf{u}^H \mathbf{R}_{ii}^{-1/2} \mathbf{a}\mathbf{a}^H \mathbf{R}_{ii}^{-1/2} \mathbf{u}}{\mathbf{u}^H \mathbf{u}} \tag{2.38}$$

Now define the vector

$$\mathbf{v} \triangleq \mathbf{R}_{ii}^{-1/2} \mathbf{a} \tag{2.39}$$

This allows the output SINR to be written as

$$\rho_k = \frac{\mathbf{u}^H \mathbf{v}\mathbf{v}^H \mathbf{u}}{\mathbf{u}^H \mathbf{u}} \tag{2.40}$$

Scaling the vector $\mathbf{u}$ has no effect on the SINR, so we need only to maximize the numerator in the above expression in order to maximize the output SINR. By the Cauchy-Schwarz inequality (2.36), this requires that $\mathbf{u} \propto \mathbf{v}$. Thus the weight vector that maximizes the output SINR is, by (2.37) and (2.39),

$$\begin{aligned} \mathbf{w}_{\text{opt}} &= \mathbf{R}_{ii}^{-1/2} \mathbf{u} && (2.41) \\ &= \mathbf{R}_{ii}^{-1/2} \mathbf{R}_{ii}^{-1/2} \mathbf{a} && (2.42) \\ &= \mathbf{R}_{ii}^{-1} \mathbf{a} && (2.43) \end{aligned}$$

The corresponding optimal output SINR is obtained by substituting (2.43) into (2.33), which yields

$$\rho_{\text{opt}} = \sigma_k^2 \mathbf{a}^H \mathbf{R}_{ii}^{-1} \mathbf{a} \tag{2.44}$$

It is important to note that multiplying the weight vector by any complex scalar has no effect on the output SINR. It can be seen by inspection of (2.43) that scaling the array response vector of the desired signal is equivalent to scaling the weight vector. This shows that the optimal weight vector is *not dependent* on the signal power σ_k^2. However, the optimal output SINR is clearly dependent on the received signal power.

An alternative expression for the optimal weight vector is given by

$$\mathbf{w}_{\text{opt}} = \mathbf{R}_{xx}^{-1}\mathbf{a} \tag{2.45}$$

where

$$\mathbf{R}_{xx} = \mathcal{E}\Big\{ \mathbf{x}(n)\mathbf{x}^H(n) \Big\} \tag{2.46}$$

is the covariance matrix of the observed data vector $\mathbf{x}(n)$. This weight vector can be obtained by minimizing the Mean Square Error (MSE) between the beamformer output and the transmitted signal $s(n)$. It can also be obtained as the solution to the following constrained optimization problem:

> Find the weight vector $\mathbf{w}$ that minimizes the total beamformer output power, subject to the constraint that the beamformer gain in the direction of the desired signal is unity.

Stated mathematically, the problem is

$$\min_{\mathbf{w}} \mathbf{w}^H\mathbf{R}_{xx}\mathbf{w} \quad \text{S.T.C.} \quad \mathbf{w}^H\mathbf{a} = 1. \tag{2.47}$$

The weight vector that solves this optimization problem is often referred to as the Minimum Variance Distortionless Response (MVDR) weight vector. We will now show that (2.45) is optimal in the maximum output SINR sense.

Using the matrix inversion lemma, which states that [20]

$$\left(\mathbf{C} + \mathbf{u}\mathbf{u}^H\right)^{-1} = \mathbf{C}^{-1} - \frac{\mathbf{C}^{-1}\mathbf{u}\mathbf{u}^H\mathbf{C}^{-1}}{1 + \mathbf{u}^H\mathbf{C}^{-1}\mathbf{u}} \tag{2.48}$$

we have

$$\mathbf{R}_{xx}^{-1} = \mathbf{R}_{ii}^{-1} - \frac{\sigma_s^2\mathbf{R}_{ii}^{-1}\mathbf{a}\mathbf{a}^H\mathbf{R}_{ii}^{-1}}{1 + \sigma_s^2\mathbf{a}^H\mathbf{R}_{ii}^{-1}\mathbf{a}} \tag{2.49}$$

This leads to

$$\mathbf{R}_{xx}^{-1}\mathbf{a} = \mathbf{R}_{ii}^{-1}\mathbf{a} - \frac{\sigma_s^2\mathbf{a}^H\mathbf{R}_{ii}^{-1}\mathbf{a}}{1 + \sigma_s^2\mathbf{a}^H\mathbf{R}_{ii}^{-1}\mathbf{a}}\mathbf{R}_{ii}^{-1}\mathbf{a} \tag{2.50}$$

$$= \left(1 - \frac{\sigma_s^2\mathbf{R}_{ii}^{-1}\mathbf{a}\mathbf{a}^H}{1 + \sigma_s^2\mathbf{a}^H\mathbf{R}_{ii}^{-1}\mathbf{a}}\right)\mathbf{R}_{ii}^{-1}\mathbf{a} \tag{2.51}$$

Since the quantity in parenthesis in the above expression is a scalar, the weight vector given by (2.45) is optimal in the maximum output SINR sense.

It is important to point out that simply substituting a finite time estimate of the covariance matrix, such as

$$\hat{\mathbf{R}}_{xx} = \sum_{n=0}^{N-1} \mathbf{x}(n)\mathbf{x}^H(n) \tag{2.52}$$

for $\mathbf{R}_{xx}$ in (2.45) does *not* in general yield good performance. The MVDR is known to be hypersensitive to error in the array response vector $\mathbf{a}$. That is, small errors in $\mathbf{a}$ will lead to large loss in output SINR. This issue is addressed is more detail in Subsection 2.4.1.

Optimal Weight Vector - Single Incident Signal It is instructive to examine the optimal weight vector for the special case of a single signal received in spatially white noise. The background noise $\mathbf{q}(n)$ is spatially white if it is uncorrelated from sensor to sensor. Assuming that the noise power in each receiver is equal to unity, the background noise will have covariance matrix

$$\mathbf{R}_{qq} = \mathcal{E}\Big\{ \mathbf{q}(n)\mathbf{q}^H(n) \Big\} = \mathbf{I} \tag{2.53}$$

where $\mathbf{I}$ is the $M \times M$ identity matrix. Modeling the background noise as unit variance is typically done so that the Signal to White Noise Ratio (SWNR) is equivalent to the received signal power. The optimal weight vector (2.43) for spatially white background noise then reduces to

$$\mathbf{w}_{\text{opt}} = \mathbf{a} \tag{2.54}$$

Thus the optimal weight vector for a single signal received in spatially white noise is simply the array response vector of the desired signal. Using the array response vector as a weight vector phase aligns the signals observed at each sensor and allows them to sum coherently. The background noise at each sensor will combine noncoherently because it is uncorrelated from sensor to sensor.

Now consider the optimal output SINR relative to the received SINR. The optimal output SINR (2.44) is equal to

$$\rho_{opt} = \sigma_s^2 \mathbf{a}^H \mathbf{R}_{ii}^{-1} \mathbf{a} = \sigma_s^2 \mathbf{a}^H \mathbf{a} \tag{2.55}$$

Without loss of generality, assume that each antenna is unity gain. This implies that

$$\mathbf{a}^H \mathbf{a} = M \tag{2.56}$$

and in turn

$$\rho_{opt} = M\sigma_s^2 \tag{2.57}$$

Thus the optimal beamformer will improve the SINR by a factor of M when only one signal is received and the background noise is spatially white. This is often termed the *white noise gain* of the array. If co-channel signals must be nulled, the optimal output SINR will be strictly less than $M\sigma_s^2$. This is therefore a useful bound on the output SINR that can be achieved with a beamformer in any co-channel environment.

These arguments have demonstrated that the beampatterns shown in Figures 2.2, 2.3, and 2.5 are optimal if only one signal is incident on the array. Note that the use of a tapering window to reduce sidelobe height, as in Figure 2.4, is *not optimal* for spatially white noise.

We now examine some optimal beampatterns for situations where multiple signals are received. These examples are limited to cases where the signals are uncorrelated, and each signal arrives from only one direction, i.e., there is no multipath. The optimal array response in multipath environments is discussed later in Subsection 2.2.1.

We first consider the 4-element ULA with $\lambda/2$ spacing. We place the desired signal at broadside (0°) and a co-channel interferer at 15°. We let the signals have the same power, which is 20 dB higher than the noise floor. We will refer to the signal strength relative to the noise floor as the Signal to White Noise Ratio (SWNR). Figure 2.6 shows the optimal beampattern for this particular scenario. Clearly there is a deep null steered towards the interferer. The optimal output SINR (2.44) for this scenario is 23.8 dB. If the interferer were not present, the optimal output SINR would be 20 dB + $10\log_{10}(4) = 26$ dB.

Figure 2.7 shows the optimal beampattern for a similar scenario, except that two uncorrelated 20 dB co-channel interfering signals are also incident from -60° and 45°, for a total of three interfering signals. This is the maximum number of interfering signals that can be nulled by this 4-element array. The optimal output SINR for this scenario is 21.2 dB, which is lower than the optimal output SINR for the previous example with a single interferer. In general, the maximum number of nulls that can be steered by an M-element adaptive array is $M-1$. This issue is addressed in more detail in Subsection 2.1.5.

2.1.4 Maximum Likelihood Weight Vectors for Multiple Signals

The weight vector derived in Subsection 2.1.3 is optimal in the maximum output SINR sense. It can also be derived as the Maximum Likelihood (ML) estimator for a single known signal in Gaussian noise and interference. If the noise is not Gaussian, the ML weight vector is no longer given by 2.43, since the MMSE solution is equivalent to the ML solution only when the noise is Gaussian. Furthermore, (2.43) is not the ML weight vector for multiple signals

in Gaussian noise.

We will now examine the ML estimator for the case where the array receives L signals and the background noise is spatially white and Gaussian. Assume that the array response vector for each of the L incident signals is known. The ML estimator of the signals is found by minimizing

$$\left\| \mathbf{X} - \mathbf{A}\hat{\mathbf{S}} \right\|_F^2 \tag{2.58}$$

where

$$\| \mathbf{A} \|_F = \sqrt{\sum_i \sum_j |a_{ij}|^2} \tag{2.59}$$

is the Frobenious matrix norm,

$$\hat{\mathbf{S}} = \begin{bmatrix} \hat{\mathbf{s}}_1 \\ \hat{\mathbf{s}}_2 \\ \vdots \\ \hat{\mathbf{s}}_L \end{bmatrix} \tag{2.60}$$

is the $L \times N$ matrix of signal estimates, and

$$\mathbf{A} = [\ \mathbf{a}_1\ \mathbf{a}_2\ \cdots \mathbf{a}_L\ ] \tag{2.61}$$

is the $M \times L$ corresponding set of known array response vectors. The resulting ML signal estimates are given by

$$\hat{\mathbf{S}} = \mathbf{W}^H \mathbf{X} \tag{2.62}$$

where the ith column of the weight matrix

$$\mathbf{W} = \mathbf{A}\left(\mathbf{A}^H\mathbf{A}\right)^{-1} \tag{2.63}$$

extracts the ith signal. It is straightforward to show that

$$\mathbf{W}^H\mathbf{A} = \mathbf{I} \tag{2.64}$$

so that each column of $\mathbf{W}$ steers zero gain (infinitely deep nulls) on the interfering signals. Thus, these weight vectors *do not* achieve the optimal output SINR. However, if the SWNR of each signal is reasonably high, the output SINR of (2.63) is very close to the optimal SINR.

It is unlikely that the array response vectors of any of the received signals will be known *a priori*. A much more likely scenario is one where multiple received signals are known, and the array response vectors are unknown. Such would be the case where the multiple received signals contained a known training sequence. By a similar argument used to find the ML estimator for the unknown signals given the known array response vectors, we can find the

ML estimator for the unknown array response vectors given the known received signals. The resulting ML estimator is

$$\hat{\mathbf{A}} = \mathbf{X}\mathbf{S}^H \left(\mathbf{S}\mathbf{S}^H\right)^{-1} \tag{2.65}$$

where it is again assumed that all the incident signals are known and the background noise is Gaussian and spatially white. The corresponding ML weight matrix for estimating the received signals is given by (2.63) with $\mathbf{A}$ replaced by its ML estimate $\hat{\mathbf{A}}$.

2.1.5 Degrees of Freedom and Overloaded Arrays

A key parameter of a beamformer is the number of interferers that can be nulled. In theory, a beamformer with M sensors can steer $M-1$ nulls. This can be seen by the following argument. Let the columns of the $M \times L$ matrix $\mathbf{A}$ correspond to the array response vectors of the L received signals. Assume for convenience that the first column of $\mathbf{A}$ corresponds to the desired signal. A weight vector that extracts the desired signal while completely rejecting the other received signals satisfies

$$\mathbf{w}^H\mathbf{A} = \mathbf{c} = [1\ 0\ 0\ \cdots 0] \tag{2.66}$$

If $L \leq M$, then the system of equations (2.66) is underdetermined, and a solution to the set of linear equations (2.66) exists. The solution is given by

$$\mathbf{w} = \mathbf{A}\left(\mathbf{A}^H\mathbf{A}\right)^{-1}\mathbf{c}^H \tag{2.67}$$

If $L > M$, then an exact solution to (2.66) does not exist. However, a vector that is as close as possible to the desired solution in a least squares sense *does* exist, and is given by [20]

$$\mathbf{w} = \left(\mathbf{A}\mathbf{A}^H\right)^{-1}\mathbf{A}\mathbf{c}^H \tag{2.68}$$

This shows that an adaptive array may not fail completely when the number of received signals exceeds the number of antennas, but the performance can be expected to degrade, perhaps dramatically, for $L > M$. When the number of incident signals L exceeds the number of antennas M, the array is said to be *overloaded*.

As an example, we consider the 4-element ULA, 3-interferer scenario corresponding to Figure 2.7, but now add an additional 20 dB interferer at $-30°$. Now the number of interferers exceeds the degrees of freedom for the array. The optimal beampattern for this scenario is shown in Figure 2.8. The corresponding optimal output SINR is 4.3 dB. This is much lower than the optimal output SINR of 21.2 dB that could be achieved with three interferers. However, the beampattern in Figure 2.8, while not steering nulls on the interfering signals, does place low gain in those directions. The input SINR is -6.1 dB, so in fact the array has

improved the SINR by approximately 10.4 dB. Thus the array is clearly capable of improving the SINR of the desired signal, even though the array is overloaded.

The array in the above example is only slightly overloaded. As another example, we will consider a scenario where the 4-element ULA is highly overloaded. We place 16 interferers in a nearly uniform angular distribution as illustrated in Figure 2.9. The total power of all 16 interferers is 20 dB, i.e., the SWNR of each interferer is approximately 8.0 dB. The desired signal is again incident from 0° with 20 dB SWNR. The received SINR at each antenna is approximately 0 dB. The optimal beampattern is shown in Figure 2.9. The corresponding optimal output SINR is 8.9 dB, which shows that the array can be effective in reducing the co-channel interference. Note that the beampattern in Figure 2.9 is very similar to the beampattern of Figure 2.2, which corresponds to the optimal weight vector for spatially white noise. This demonstrates that a large number of interferers, when equally distributed and with nearly equal power, is similar to spatially white noise. This has implications for CDMA applications. In CDMA, the users are intentionally co-channel, and the power of each user is typically closely controlled in order to maximize capacity [21].

To gain further insight into the performance of an adaptive array in overloaded environments, we perform the following experiment. We again use the 4-element ULA with a 20 dB desired signal at 0°. We then add L interfering signals at random angles θ, with θ uniformly distributed over $-\pi/2 < \theta < \pi/2$. The total power of the L interfering signals is equal to 20 dB, with the noise floor at 0 dB. All interfering signals have the same power. The optimal output is calculated, and the experiment is repeated for 10,000 trials, with the AOAs being randomly selected for each trial. The mean optimal output SINR over 10,000 independent trials is shown in Figure 2.10 as a function of the number of interfering signals.

To this point, we have only considered uniformly spaced arrays with interelement spacing equal to $\lambda/2$. In the next section we show that other array geometries can yield improved performance relative to a ULA.

2.1.6 Array Aperture, Ambiguities, and Resolution

If two signals have the same AOA a beamformer will not be able to separate them. A reasonable question then arises: What angular separation is required in order to extract the desired signal? Unfortunately, there is no straightforward answer to this. However, it will be helpful to consider the case of a single signal and a single interferer received in spatially white noise. In this case

$$\mathbf{R}_{\text{ii}} = \left(\sigma_2^2 \mathbf{a}_2 \mathbf{a}_2^H + \mathbf{I}\right) \tag{2.69}$$

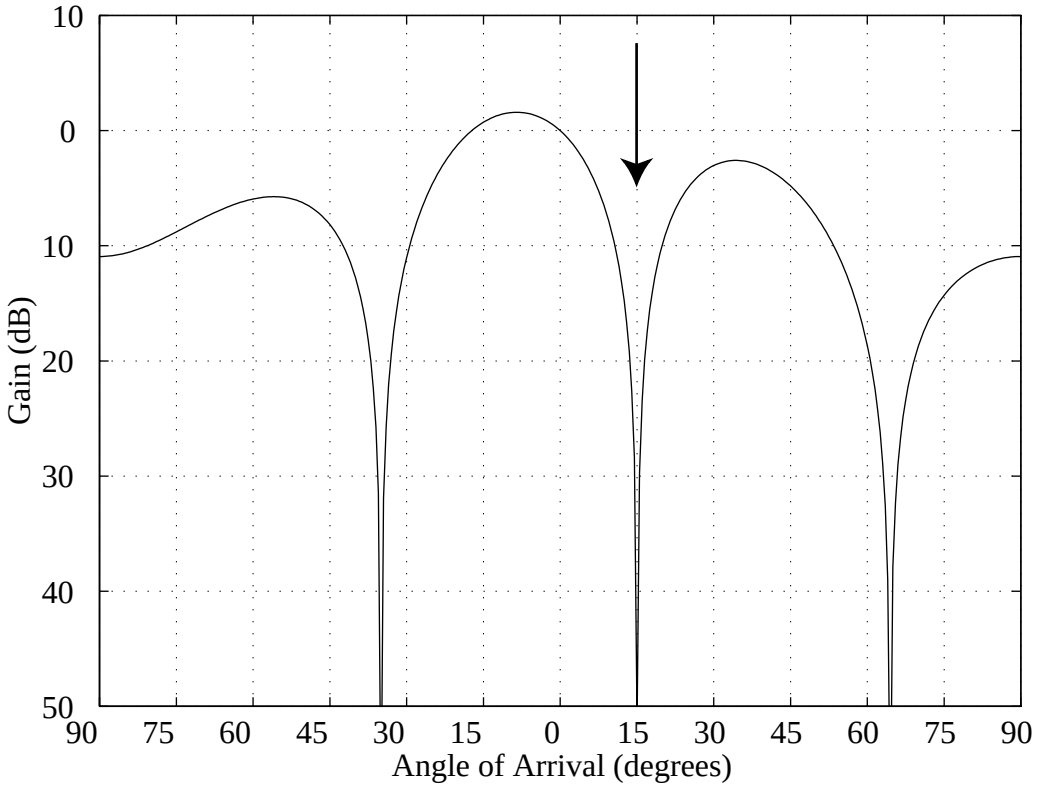

Figure 2.6: Optimal beampattern for a 4-element ULA with a desired signal incident from 0° and a single 20 dB co-channel interferer incident from 15°.

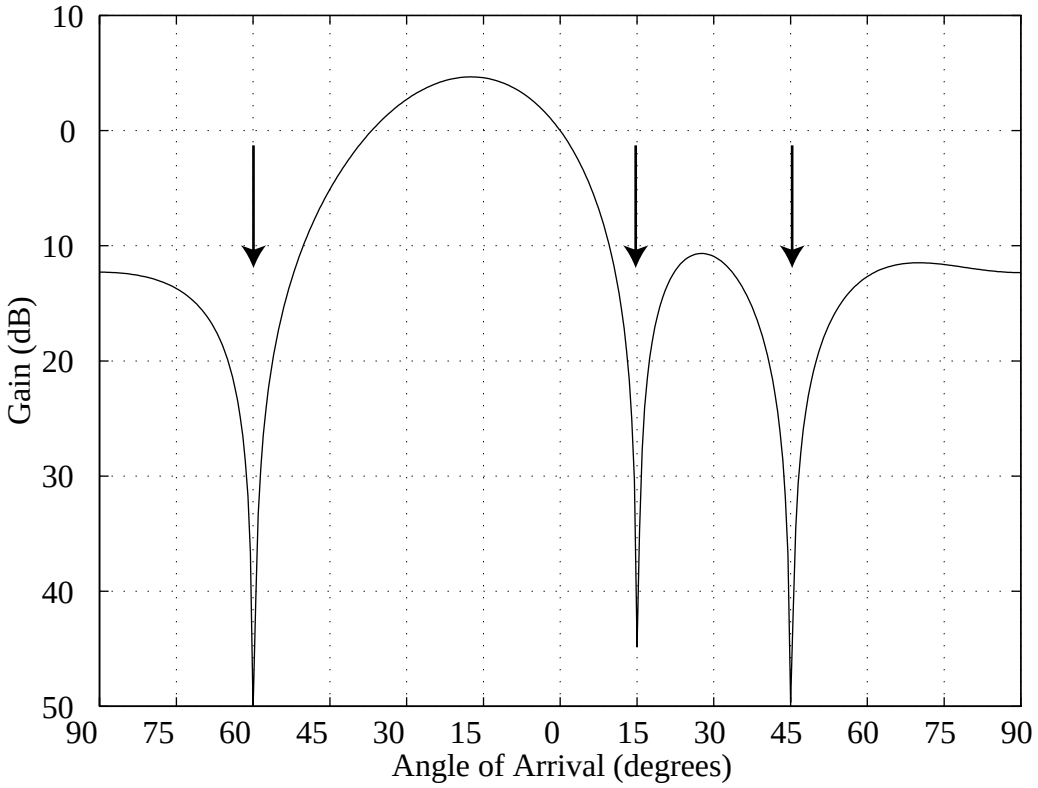

Figure 2.7: Optimal beampattern for a 4-element ULA with a desired signal incident from 0° and three 20 dB co-channel interfering signals incident from -60°, 15°, and 45°.

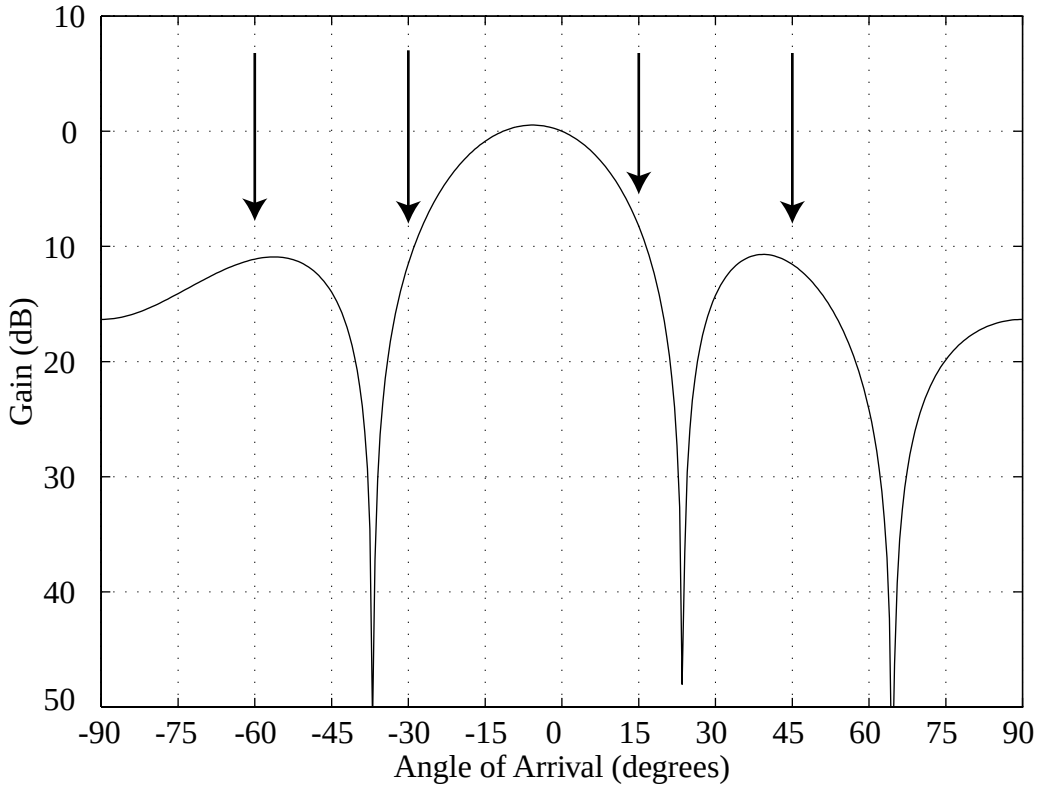

Figure 2.8: Optimal beampattern for a 4-element ULA with a desired signal incident from 0° and four 20 dB co-channel interfering signals incident from -60°, -30°, 15°, and 45°. The array is overloaded.

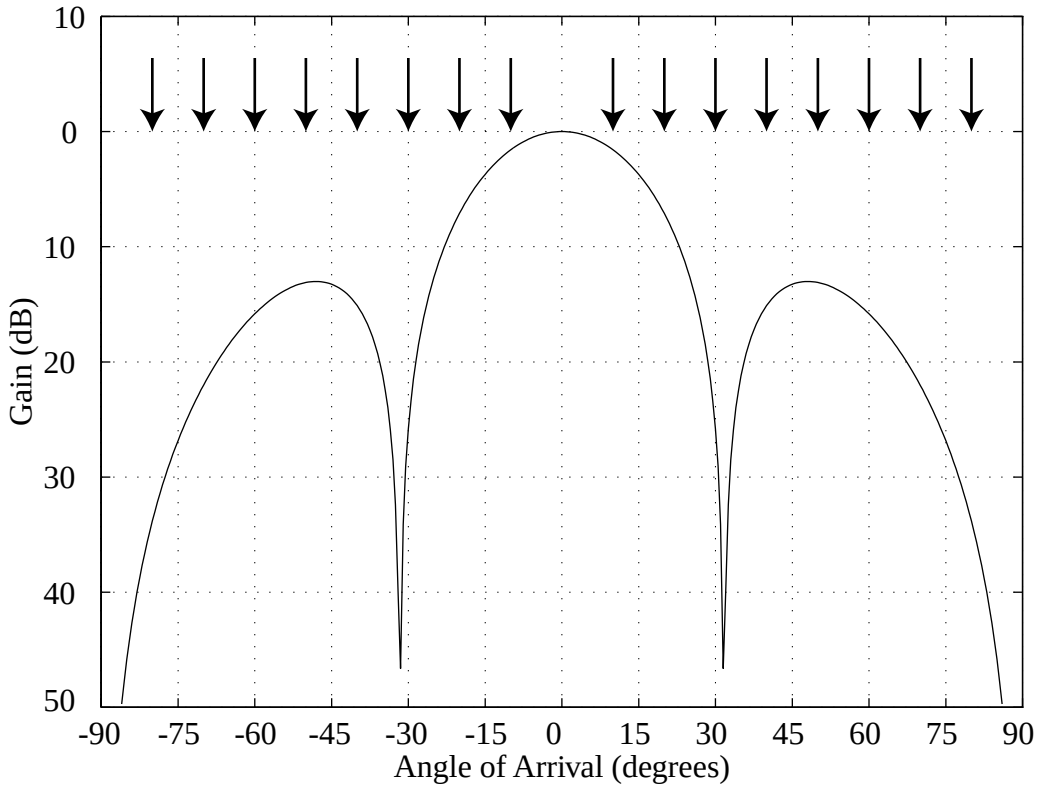

Figure 2.9: Optimal beampattern for a 4-element ULA with a desired signal incident from 0° and 16 co-channel interfering signals, each having 8 dB SWNR.

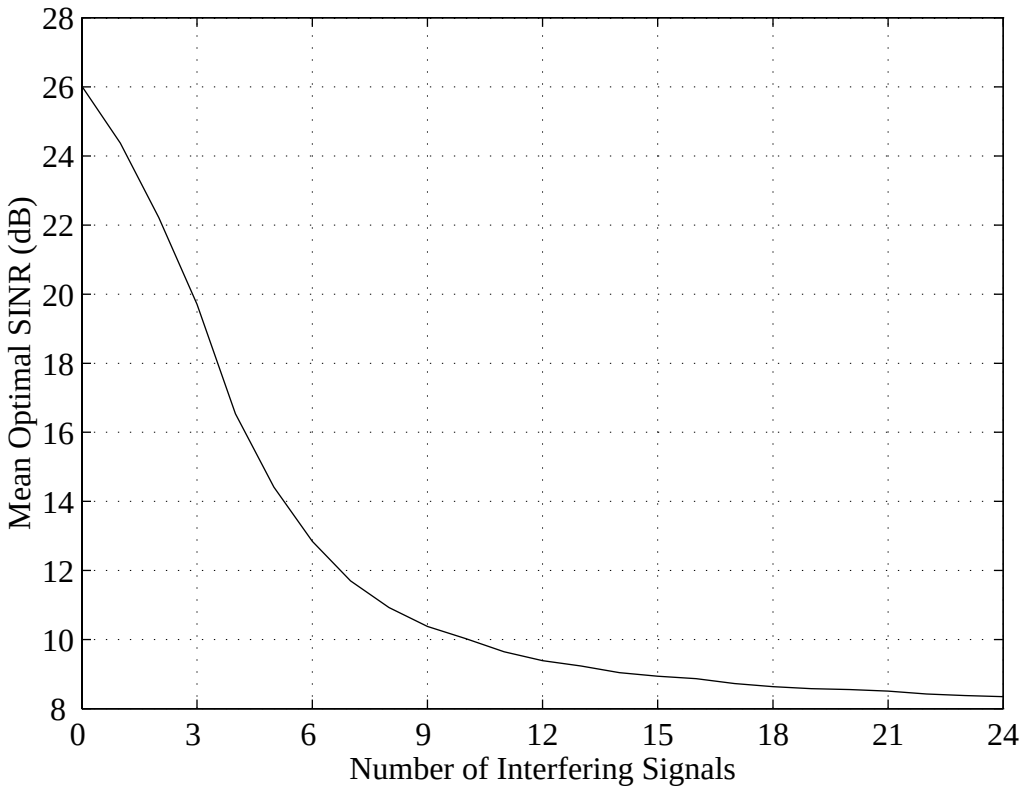

Figure 2.10: Mean optimal output SINR versus the number of interfering signals with a 4-element ULA. The AOA of the 20 dB desired signal is 0 dB, and the AOA of each interfering signal is randomly selected. The total interference power is held constant at 20 dB.

where $\mathbf{a}_2^H$ is the array response vector for the second (interfering) signal and $\mathbf{I}$ is the identity matrix. Using the matrix inversion lemma (2.48),

$$\mathbf{R}_{\text{ii}}^{-1} = \left(\mathbf{I} - \frac{\sigma_2^2 \mathbf{a}_2 \mathbf{a}_2^H}{1 + \sigma_2^2 \mathbf{a}_2^H \mathbf{a}_2} \right). \tag{2.70}$$

The optimal output SINR is given from (2.44) as

$$\rho_{\text{opt}} = \sigma_1^2 \mathbf{a}_1^H \mathbf{R}_{\text{ii}}^{-1} \mathbf{a}_1 \tag{2.71}$$

$$= \sigma_1^2 \mathbf{a}_1^H \left(\mathbf{I} - \frac{\sigma_2^2 \mathbf{a}_2 \mathbf{a}_2^H}{1 + \sigma_2^2 \mathbf{a}_2^H \mathbf{a}_2} \right) \mathbf{a}_1. \tag{2.72}$$

Assume that the sensors are unity gain so that $\mathbf{a}_2^H \mathbf{a}_2 = M$. Also assume that the received signals are much stronger than the background noise so that $\sigma_2^2 \gg 1$. Then

$$\rho_{\text{opt}} = \sigma_1^2 \left(M - \frac{\left| \mathbf{a}_1^H \mathbf{a}_2 \right|}{M} \right). \tag{2.73}$$

It can be seen that the output SINR is dependent on the number of sensors in the array and also on the inner product of the array response vectors. The output SINR depends for the most part on the similarity of the array response vectors for the incident signals. If the array response vectors are very similar, their inner product will be large and the output SINR will be small. Increasing the spacing between sensors will increase the relative phase difference between sensors. This in turn will make the array response vectors less similar and improve the ability to resolve incident signals. Based on these arguments, it would seem that the larger the array, the better. This is only true up to the point where the array response begins to exhibit *ambiguities* (See Subsection 2.1.1). Array ambiguities are to be avoided, in general, because if a desired signal and interferer lie at ambiguous AOAs, their array response vectors are identical, and the interferer cannot be nulled without also nulling the desired signal. Therefore the sensors should be as far apart as possible, i.e., the array aperture should be as large as possible, without introducing ambiguities.

Based on the observations made in Subsection 2.1.1, it might be concluded that for an M-element linear array the inter-element spacing should be no more than $\lambda_c/2$ in order to avoid ambiguities. However, the spacing between individual sensors can be much larger than this without incurring ambiguities so long as one pair of sensors are less than $\lambda_c/2$ apart. This will be illustrated through a simple example. Consider the problem of determining antenna placement for a three element linear array. The output SINR for an array of 3 sensors will be strictly greater than the output SINR that can be achieved with a two element subarray of the overall array. This can be proved by noting that if this were not true, the output SINR could be increased simply by placing zero weighting on some antenna elements. Thus

the third sensor can be spaced more than $\lambda/2$ away from the next closest sensor, and the two closely spaced elements will prevent array ambiguities.

The resolution of an array can be examined by plotting the optimal output SINR as a function of signal separation. Figure 2.11 shows such a plot for a 4-element ULA with $\lambda/2$ interelement spacing. The desired signal is assumed to have 9 dB SWNR and is incident from 0°. A single 9 dB SWNR interfering signal is also present, and the AOA of the interferer is varied. Clearly the array has difficulty extracting the desired signal when the angular separation is small.

Now consider the effect of increasing the array aperture. We keep three of the antennas at their original locations, with $\lambda/2$ spacing, but move the 4th antenna so that it is separated by 4 λ from its nearest neighbor. That is, the array elements are located at x-axis coordinates

$$x = [\ \ 0 \ \ 1/2 \ \ 1 \ \ 5 \ \ ] * \lambda \tag{2.74}$$

with the y-axis coordinate equal to zero, i.e., the array lies along the x-axis. According the argument made earlier, this array should not exhibit any ambiguities. This is supported by Figure 2.12, which shows the beampattern for a signal incident from 0°. Clearly the sidelobes are much higher than with the ULA, and the mainlobe is not well defined. However, the white noise gain is identical to that of the ULA, since this is only dependent on the number of antennas. Furthermore, the resolution of this array is much better than for the ULA. Figure 2.13 shows the optimal beampattern for this large aperture array with a desired signal at 0° and a 20 dB interfering signal at 15°. Comparing Figure 2.13 with the analogous beampattern for the ULA shown in Figure 2.13, we might conclude that the array with larger aperture has 'sharper' nulls. Thus we would expect the resolution to be higher. This is demonstrated conclusively in Figure 2.14, which shows the optimal output SINR as a function of signal separation. Comparing Figure 2.14 with 2.11, it is clear that the array with larger aperture has higher angular resolution.

Antenna placement can itself be a complicated design issue. Practical effects such as antenna coupling and near-field scattering affect the behavior of the antenna array pattern. Furthermore the placement of the antennas may be constrained by other issues, such as mechanical stability, or aesthetic issues.

2.2 Mobile Radio Propagation Environment

In this section we review some of the salient characteristics of the mobile radio channel as they affect the performance of an adaptive array. A more complete review of the mobile radio channel can be found in [22]. We are primarily concerned with cellular-type applications,

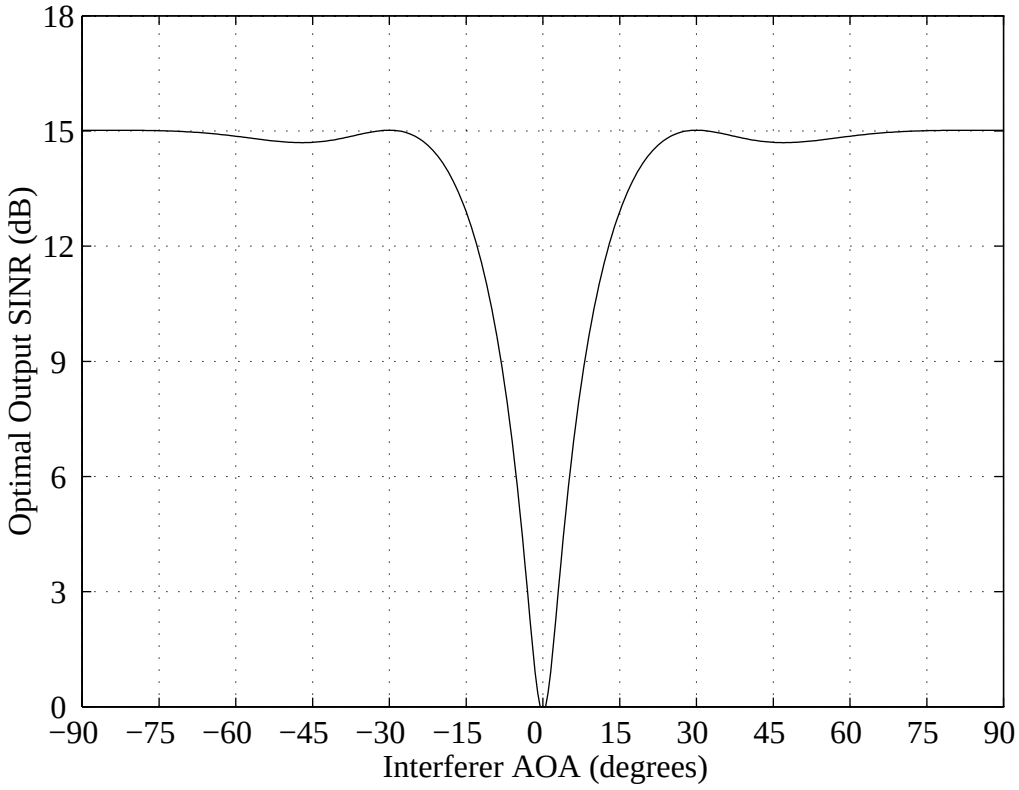

Figure 2.11: Optimal output SINR versus signal angular separation. The array is a 4-element uniform linear array with interelement spacing equal to $\lambda_c/2$. A 9 dB signal is received from 0° and the AOA of a 9 dB interferer is varied from -90° to +90°.

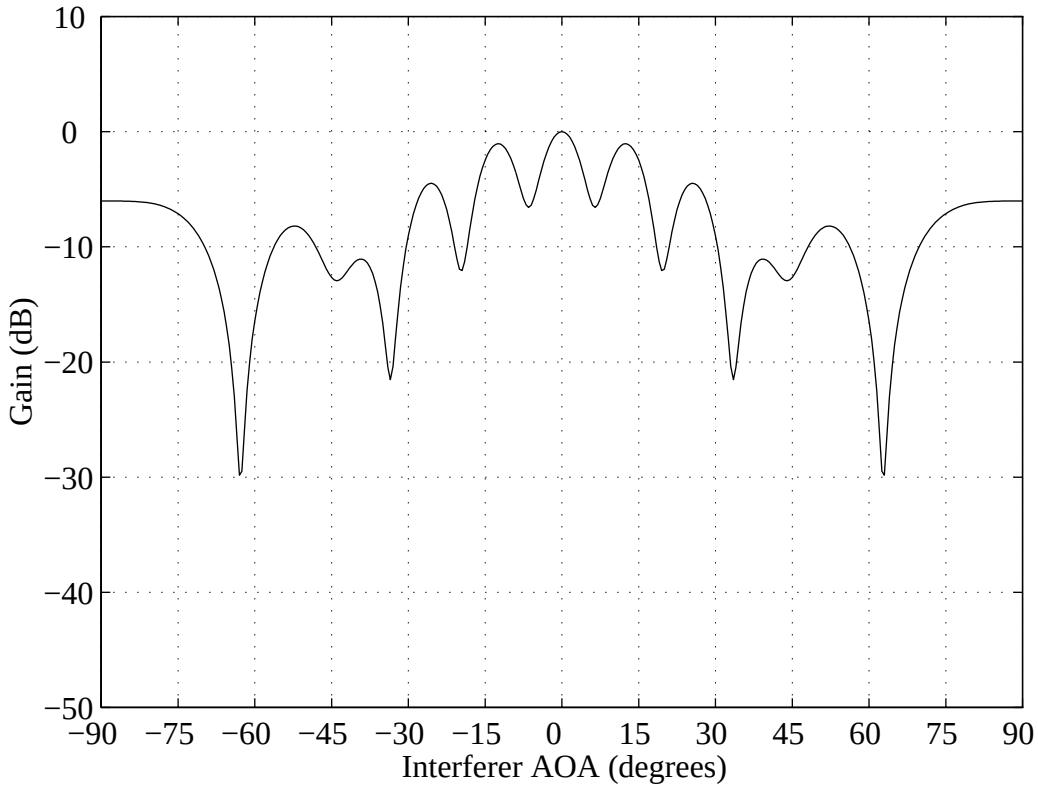

Figure 2.12: Optimal beampattern for a single signal incident from 0° received in spatially white noise. The array is a 4-element linear array with elements located at $[\ 0\ \ 1/2\ \ 1\ \ 5\ ] * \lambda$.

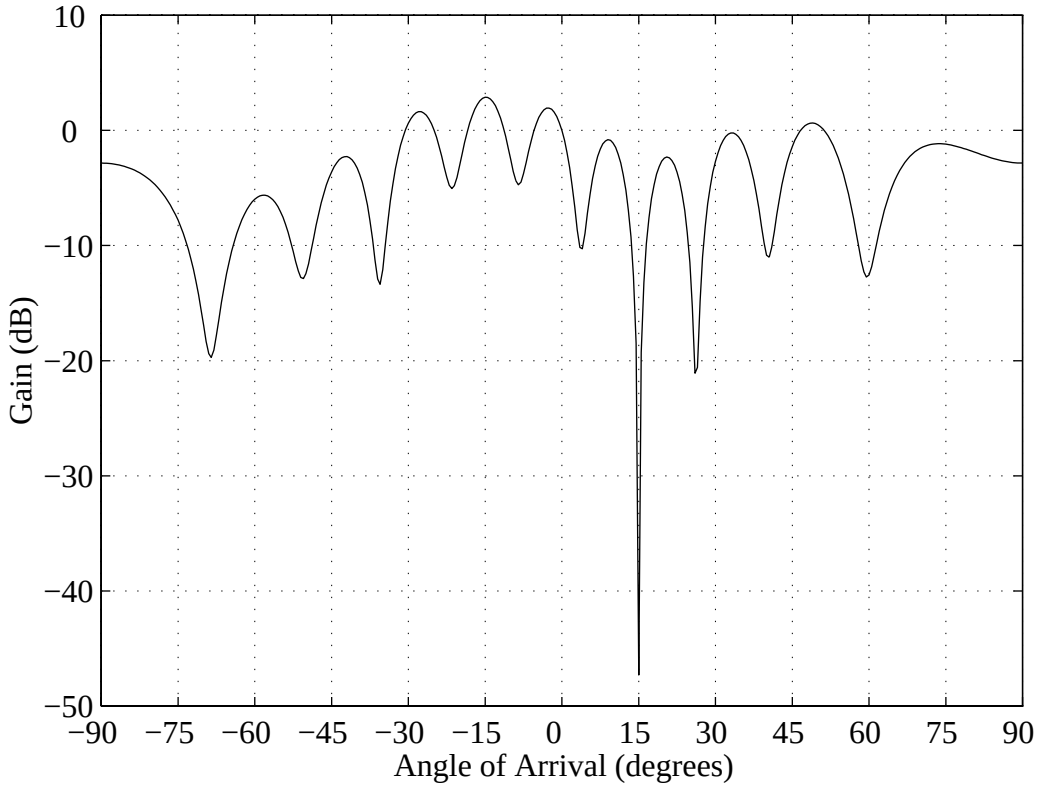

Figure 2.13: Optimal beampattern for a signal incident from 0° with a 20 dB interferer incident from 15°. The array is a 4-element linear array with elements located at $[\ 0\ \ 1/2\ \ 1\ \ 5\ ] * \lambda$.

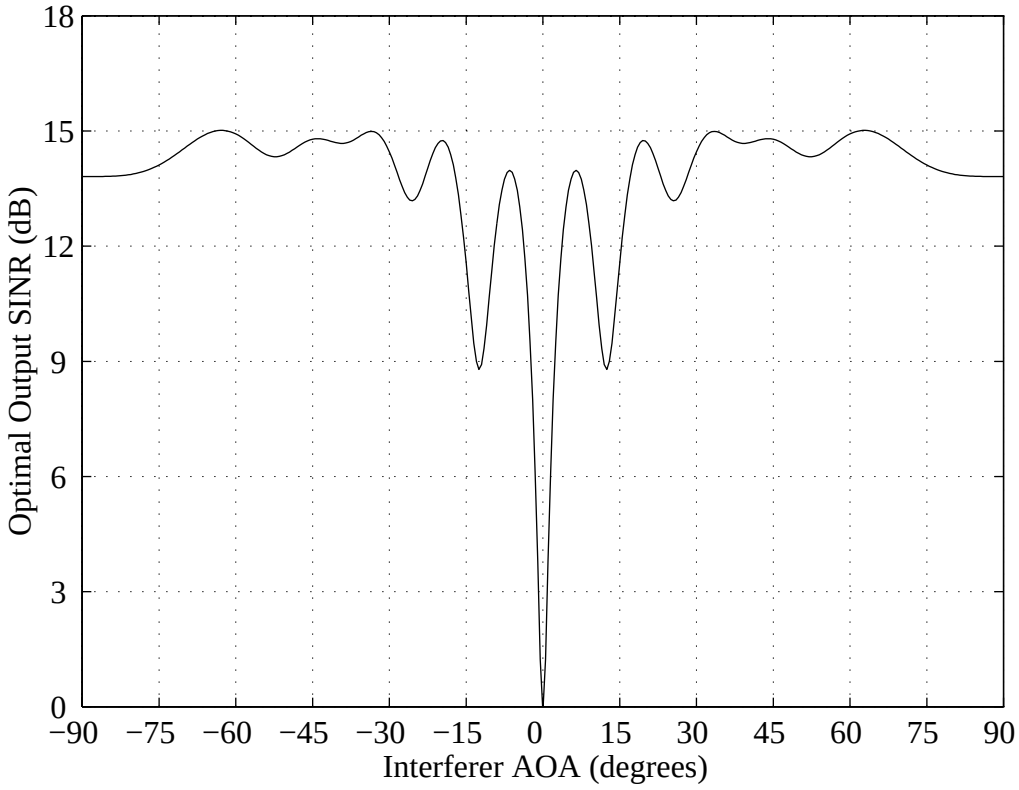

Figure 2.14: Optimal output SINR versus signal angular separation. The array is a 4-element linear array with elements located at $[\ 0\ \ 1/2\ \ 1\ \ 5\ ] * \lambda$. A 9 dB signal is received from 0° and the AOA of a 9 dB interferer is varied from -90° to +90°.

where frequency reuse is employed to maximize capacity, and the mobile users are linked to a fixed basestation. The generally complicated propagation characteristics in urban and suburban environments create a situation where the transmitted signal may arrive at the receiver via different paths. This is referred to as a multipath environment. If the paths have significantly different lengths (resulting in a delay spread that is large relative to the symbol period), the channel is said to be frequency selective. If the paths are nearly the same length, the channel is said to be flat. The mobility of the user, combined with multipath, create signal envelope *fading*. Fading occurs when multipath components arriving at an antenna combine destructively, so that the signal power becomes very small. This undesired effect limits the capacity of a mobile radio network. An adaptive array has the potential to reduce the effects of fading and time delay spread (when combined with MLSE, equalization, or a Rake). Several channel parameters affect the performance of an adaptive array, including

1. angle spread due to multipath,
2. time dispersion to due multipath,
3. Doppler spread due to multipath and the relative motion of the transmitter and receiver.

These parameters affect the ability to separate signals (beamformer output SINR) and the rate at which the algorithms must track the environment. The required update rate in turn has a large impact on computational requirements. The topics reviewed in this section include:

1. the spatial signature, which is used to describe the array data in a multipath environment;
2. signal resolution in a multipath environment;
3. Doppler spread and the required weight vector update rate.

2.2.1 Spatial Signature

The spatial signature is used to describe the signal received by an antenna array when the signal arrives from multiple directions with insignificant delay spread. In such situations, the array data can be modelled as

$$\mathbf{x}(n) = \mathbf{a}s(n) + \mathbf{q}(n) \tag{2.75}$$

where the spatial signature $\mathbf{a}$ is in general given by

$$\mathbf{a} = \int_0^{2\pi} g(\theta)\mathbf{a}(\theta)\, d\theta \tag{2.76}$$

where $g(\theta)$ is the complex angular distribution of the received signal. If the angular distribution is a continuous function, the multipath is said to be *diffuse*. If $g(\theta)$ is zero everywhere except for a finite set of discrete angles, then the multipath is said to be *specular*. Specular multipath can be modeled as

$$\mathbf{a} = \sum_{i=1}^{L} g_i \mathbf{a}(\theta_i) \tag{2.77}$$

where L is the number of received paths, g_i is the complex amplitude of each path, and θ_i is the AOA of each path. Note that the specular multipath model (2.77) is subsumed by the diffuse multipath model (2.76) simply by allowing $g(\theta)$ to contain Dirac delta functions. In the most general case the signal angular distribution would contain both discrete (specular) components and diffuse components. Many different models have been proposed for the angular distribution $g(\theta)$. A review of many of these models is presented in [23].

Several key properties of the spatial signature have a great impact on the behavior of adaptive arrays in mobile wireless applications. The following points should be emphasized:

1. When $L > 1$, the spatial signature can change rapidly as a function of user position, which is not the case when $L = 1$ (no multipath). Thus the presence of multipath can actually improve the ability to separate closely spaced signals.

2. The angular spread partially determines the rate at which the spatial signature varies with user position.

3. The array aperture also affects the rate at which the spatial signature varies.

4. If the g_i are mutually uncorrelated, and nearly equal in power, the spatial signature $\mathbf{a}$ approaches a complex Gaussian distribution as the number of paths L grows large.

These points will now be discussed in more detail.

We will first demonstrate that the presence of multipath can improve the ability of a beamformer to separate received signals. An important point is that the expressions for the optimal weight vector (2.43) and optimal output SINR (2.44) are *still valid* in a multipath environment – the array response vectors of the desired signal and interfering signal are simply replaced with the corresponding spatial signatures. Consider the following scenario. The array is a 4-element ULA with $\lambda/2$ interelement spacing. A 20 dB desired signal is incident from 0°. A 17 dB interferer is incident from 0°, which is the *same* AOA as the desired signal.

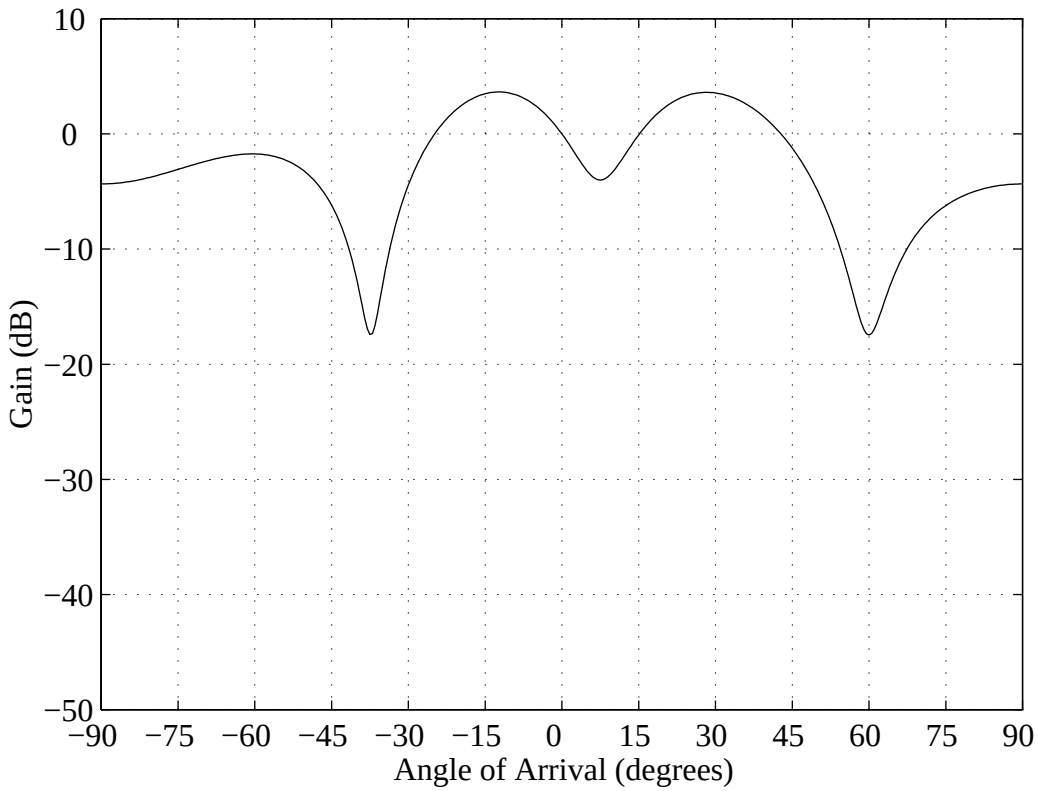

Figure 2.15: Optimal beampattern for a 4-element ULA with the desired signal incident from 0°. A 17 dB interferer is also incident from 0° and a second, perfectly correlated version of the same interferer is incident from 15°.

As would be expected, the optimal output SINR for this scenario is very low (approximately 3 dB) since the interferer cannot be nulled without also nulling the desired signal. Now a multipath reflection of the interferer is added to the environment. This multipath arrives from 15° with the same power as the path from 0°. We assume that the multipath components are perfectly correlated. The optimal output SINR is now 20 dB, which is a dramatic improvement over the case where no multipath was present. Figure 2.15 shows the optimal beampattern for this multipath scenario. Note that the beampattern does not steer nulls, and in fact has equal gain towards 0° and 15°, towards the AOA of the multipath. The array can cancel the interferer by controlling the phase of the array response so that the components from 0° and 15° combine destructively. Figure 2.16 shows the phase response of the antenna pattern. The phase at an AOA of 0° is +90°, while the phase at an AOA of 15° is -90°, so that the multipath components will cancel.

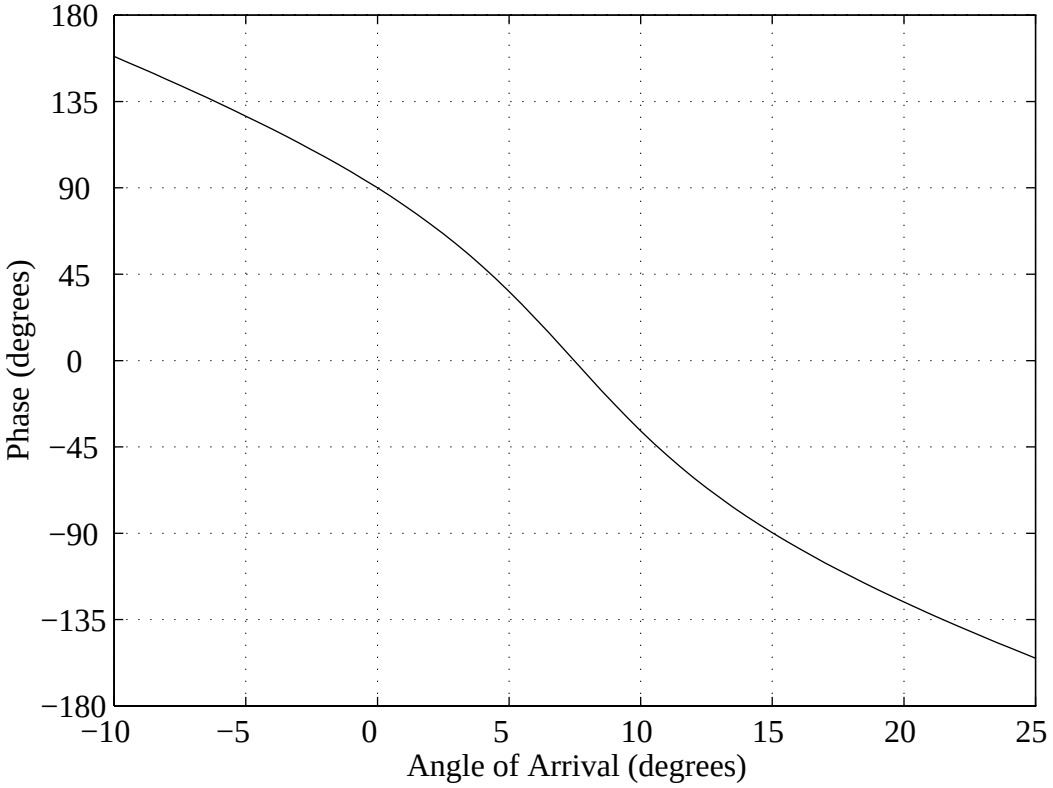

Figure 2.16: Phase of the optimal array response with a 17 dB interferer incident from 0° and a second, perfectly correlated version of the same interferer incident from 15°.

The next point to illustrate is that small changes in a user's position can correspond to large changes in the spatial signature. Consider a simple two-ray multipath model, where two copies of the same signal arrive at the receiver from different paths. Let the difference in length of these two paths be δ meters. Then the differential time delay between the two paths is $\tau = \delta/c$, where c is the speed of light in meters per second. If we assume that the time delay is small relative to the inverse bandwidth of the signal, i.e., $\tau << 1/\text{BW}$, the two copies of the signal are nearly perfectly correlated. This is referred to as a *flat fading* environment, since the channel frequency response is flat. Since the delay is small, it can be accurately modelled as a simple phase shift. This phase shift is

$$\phi = \frac{2\pi f_c \delta}{c} \tag{2.78}$$

where f_c is the carrier frequency. This can be expressed in terms of carrier wavelength λ_c as

$$\phi = \frac{2\pi\delta}{\lambda_c} \tag{2.79}$$

The spatial signature for this two-ray multipath scenario is proportional to

$$\mathbf{a} = \mathbf{a}(\theta_1) + e^{j\phi}\mathbf{a}(\theta_2) \tag{2.80}$$

where θ_1 and θ_2 are the AOA of the two paths. It can be seen that the spatial signature will change if ϕ changes, even if θ_1 and θ_2 remain constant. This is precisely what will occur in practice when the receiver (or transmitter) position changes, assuming that the transmitter-receiver separation is large relative to the wavelength. If the change in position is a significant fraction of one wavelength, the differential delay τ will change. For some perspective on what would constitute a significant change, the PCS band in the United States is near 2 Ghz; this corresponds to a wavelength of approximately 15 cm. The amount by which the spatial signature will change is dependent on several factors, including:

1. the rate at which δ changes with receiver or transmitter motion, which depends on the geometry of the receiver, transmitter, and scattering elements;
2. the similarity of $\mathbf{a}(\theta_1)$ and $\mathbf{a}(\theta_2)$, which depends on the array aperture and the angular spread $\theta_1 - \theta_2$.

To gain more perspective on this issue we will consider the simplified receiver-transmitter geometry shown in Figure 2.17. A transmitted signal arrives at the receiver from a direct path and from a reflected path. For simplicity, we assume that the reflector is a perfectly smooth plane of infinite extent, and that it lies parallel to the direct path. It is straightforward to show that

$$d_2 \cos\Delta = d_1 \tag{2.81}$$

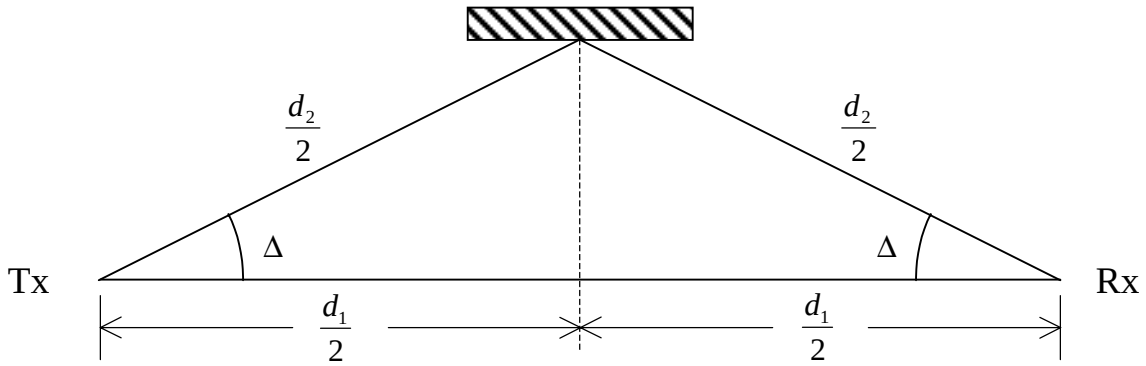

Figure 2.17: Simple two-path geometry used to illustrate the effect that small transmitter displacement can have on the spatial signature.

where d_1 and d_2 are the path length corresponding to the direct and reflected path, respectively, and Δ is the angle between the two paths. The angular difference Δ is the *angle spread* for this scenario. The difference in path length is

$$\delta = d_1 - d_2 = d_1 - \frac{d_1}{\cos\Delta} \tag{2.82}$$

and the relative phase shift between the two paths is

$$\phi_1 = \frac{2\pi}{\lambda}\left(d_1 - \frac{d_1}{\cos\Delta}\right) \tag{2.83}$$

Note that as Δ grows small, the relative phase shift between the two paths approaches zero. Now assume that the receiver moves directly towards the transmitter a distance x. Also assume that x is small relative to d_1, so that the angle between the two paths Δ remains nearly constant. The relative phase shift between the two paths is now

$$\phi_2 = \frac{2\pi}{\lambda}\left(d_1 - x - \frac{d_1 - x}{\cos\Delta}\right) \tag{2.84}$$

The phase difference has therefore changed by

$$\phi_1 - \phi_2 = \frac{2\pi x\,(\cos\Delta - 1)}{\lambda\cos\Delta} \tag{2.85}$$

If ϕ_1 is different from ϕ_2, the spatial signature (2.80) will be different even though the AOA of each multipath component remains essentially constant.

Figure 2.18 shows a plot of (2.85) as a function of receiver displacement x for several different values of angle spread Δ. Note that the angle spread has a large impact on the rate at which

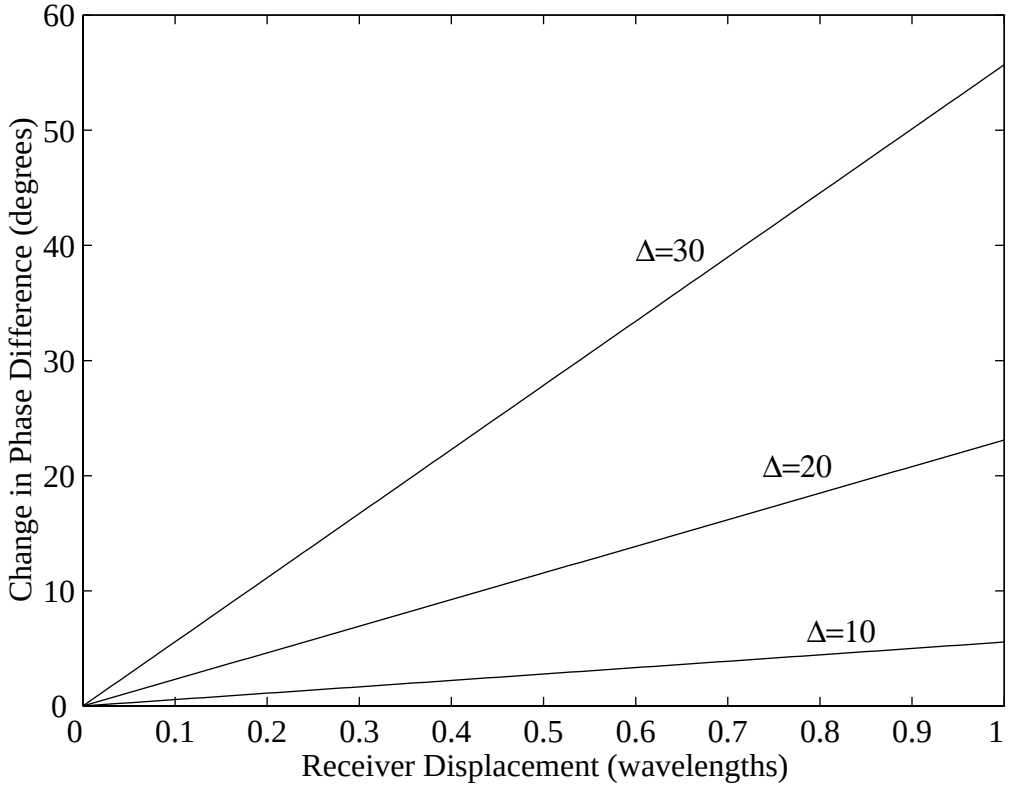

Figure 2.18: Phase difference between two received paths as a function of receiver displacement for several different values of angle spread Δ in a simple two-ray geometry.

the spatial signature will change. If the angle spread is small, the spatial signature will change slowly with changing receiver position. This is a very simple example, and is meant only to illustrate the basic concept of the effect of receiver (or transmitter) motion. However, the general conclusions drawn from this simple example hold in more realistic environments as well.

We finally examine the behavior of the spatial signature as the number of received components grows large. The central limit theorem states that a random variable Z that is formed as a sum of a large number of independent random variables will tend to have a Gaussian distribution in the limit. Stated formally[24],

> Assume that $g_1, g_2, \cdots g_n$ is a sequence of independent, identically distributed random variables with mean μ and variance σ^2. Let
>
> $$Z = \frac{1}{n} \sum_{i=1}^{n} \frac{g_i - \mu}{\sigma} \tag{2.86}$$
>
> Then as $n \to \infty$, Z converges, in distribution, to a standard Gaussian distribution with zero mean and unit variance. There are several variations in the central limit theorem, including
>
> 1. The random variables g_i are independent with the same mean and variance but are *not* identically distributed.
> 2. The g_i are independent and have different finite variances, and the variances are neither too small nor too large.

It is clear that (2.86) and (2.77) are directly analogous, which shows that the spatial signature tends to a Gaussian distribution *if* the g_i are independent. For example, the g_i would be independent if the g_i have the same amplitude but uncorrelated, uniformly distributed phase.

As the angle spread of the signals becomes large, each entry in the spatial signature will tend to an uncorrelated, complex Gaussian random variable. The envelope of a complex Gaussian random variable is Rayleigh distributed. This is one way to motivate the well-known Rayleigh model for multipath fading.

2.2.2 Doppler Spread and Fading

The spatial signature model in (2.76) assumes that the transmitter, receiver, and objects surrounding the transmitter and receiver are not moving. Relative motion will cause the path length of each multipath component to change with time, causing the amplitude and

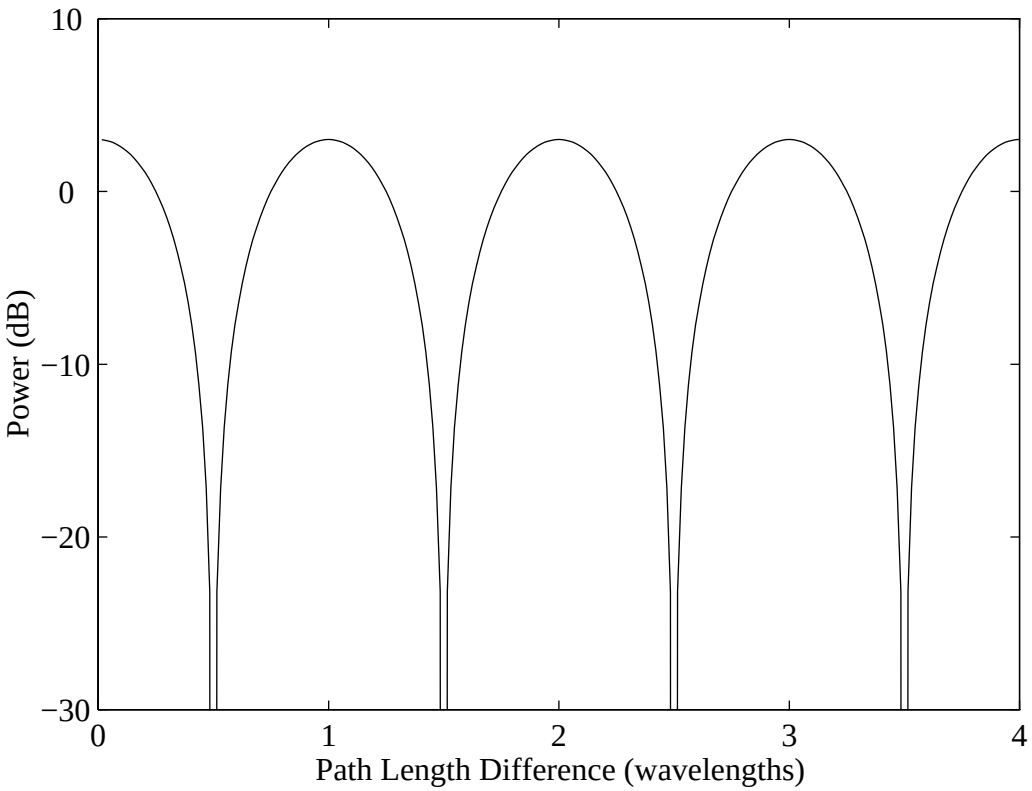

Figure 2.19: Typical fading envelope for a two-ray multipath environment, when each path has identical power.

phase at each antenna to vary with time. This is shown explicitly by modeling the spatial signature as

$$\mathbf{a}(n) = \int_0^{2\pi} g(\theta, n)\mathbf{a}(\theta)\, d\theta \tag{2.87}$$

where the angular distribution $g(\theta, n)$ is now time dependent. The rate of change of $g(\theta, n)$ is generally slow compared to the symbol rate. This implies that $g(\theta, n)$ is nearly constant over several symbol periods. This is referred to as *slow fading*. Most fading communication channels exhibit slow fading, with the exception of the HF band.

Consider a two-ray multipath environment where the amplitude of each path is equal. As the transmitter moves in this simple example, the received amplitude will peak when the signals combine in phase. When the signals are exactly 180° out of phase, the signals will combine destructively, and the amplitude will be zero, i.e., the signal will be in a *fade*. This occurs when path length difference is an integer multiple of the carrier wavelength. This is illustrated in Figure 2.19, which shows the fading envelope for this simple two-ray multipath scenario. Note that if one of the paths is stronger than the other, the signal power cannot

be completely canceled, and the fading will not be as deep.

The change of transmitter position with time induces a linearly time-varying phase shift – this is simply a frequency shift. This shift is referred to as *Doppler*. The Doppler induced by relative transmitter/receiver motion is given by

$$f_d = \frac{\nu}{\lambda} \cos\theta \tag{2.88}$$

where ν is the speed of the transmitter, λ is the carrier wavelength, and θ is the angle between the relative direction of travel and the bearing towards the receiver. That is, $\theta = 0$ when the transmitter moves directly towards the receiver. The Doppler shift is maximum in this case. If the multipath components arrive at the receiver with the *same* Doppler shift, there will be no fading since the signals all have the same carrier frequency. It is only when there is a difference, or *Doppler spread* , that fading will occur.

Now consider Figure 2.20, which shows a fading envelope with five equal power specular multipath components. The Doppler spread in this case is normalized to one. The frequency of each path, relative to the maximum Doppler shift, is [0 4.1 8.3 12.7 16]/16]. Note that the received envelope is not periodic. Figure 2.20 shows that the received envelope undergoes deep fades, and has a much more random appearance than in the two-ray case.

We argued earlier that the sum of a large number of multipath components arriving at an antenna tends to a complex Gaussian distribution if the amplitude and phase of each arriving component is independent. The distribution of the magnitude r of a complex Gaussian random variable follows a Rayleigh distribution, given by

$$p(r) = \begin{cases} \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) & 0 \geq r < \infty \\ 0 & r < 0 \end{cases} \tag{2.89}$$

In addition to the distribution of the envelope, the duration and frequency of occurrence of deep fades affects the performance of the communication system. These parameters are dependent on the Doppler spectrum. The Doppler spectrum describes the frequency spreading that occurs when a sinusoid is transmitted through a fading channel. For example, the Doppler spectrum in the five-ray example considered earlier would be zero everywhere except for five spectral lines at the five discrete Doppler shifts. One commonly used model for the Doppler spectrum is due to Clarke, and is given by

$$S(f) = \frac{1}{\pi f_m \sqrt{1 - \left(\frac{f - f_c}{f_m}\right)^2}} \tag{2.90}$$

where f_m is the maximum Doppler frequency, given by

$$f_m = \frac{\nu}{\lambda} \tag{2.91}$$

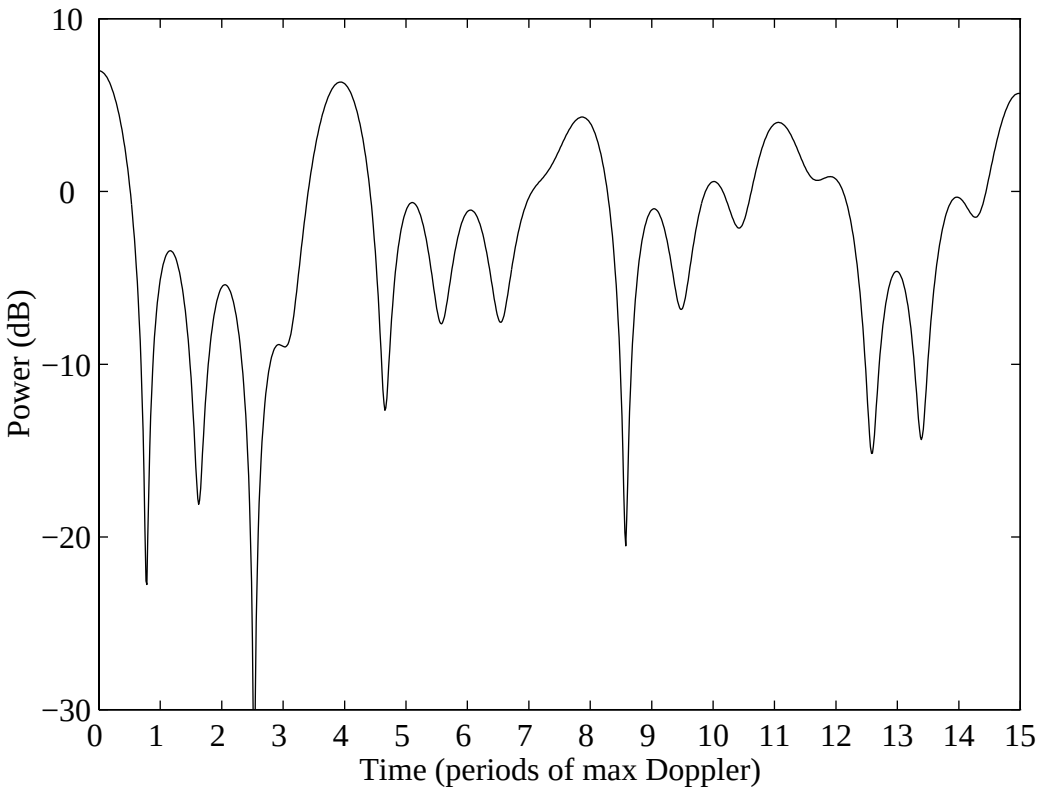

Figure 2.20: Fading envelope for a five path environment with each path having equal power and discrete, incommensurate Doppler shift.

where ν is the speed of the transmitter, λ is the carrier wavelength, and f_c is the carrier frequency. This model results from assuming that the multipath is received from all directions with equal power. This only requires that either the transmitter or receiver be surrounded by scatterers. That is, if the mobile subscriber is surrounded by many close reflectors, the signal arriving at the basestation may still have Rayleigh fading and Doppler spectrum given by (2.90), even though from the perspective of the basestation the signal arrives from a very narrow range of AOA. This is due to the reciprocity of transmission paths. Combining the Rayleigh envelope distribution with Clarke's Doppler spectrum model leads to fading envelopes such as the one illustrated in Figure 2.21. The estimated PSD of the complex envelope is shown in Figure 2.22. This PSD was estimated using the Welch method with a 4096 point Hamming window, 50% overlap, and 131072 temporal samples with the maximum Doppler frequency equal to 0.125. This PSD is included both to illustrate the shape of the Doppler spectrum given by Clarke's model (2.90) and to demonstrate the accuracy of the simulation.

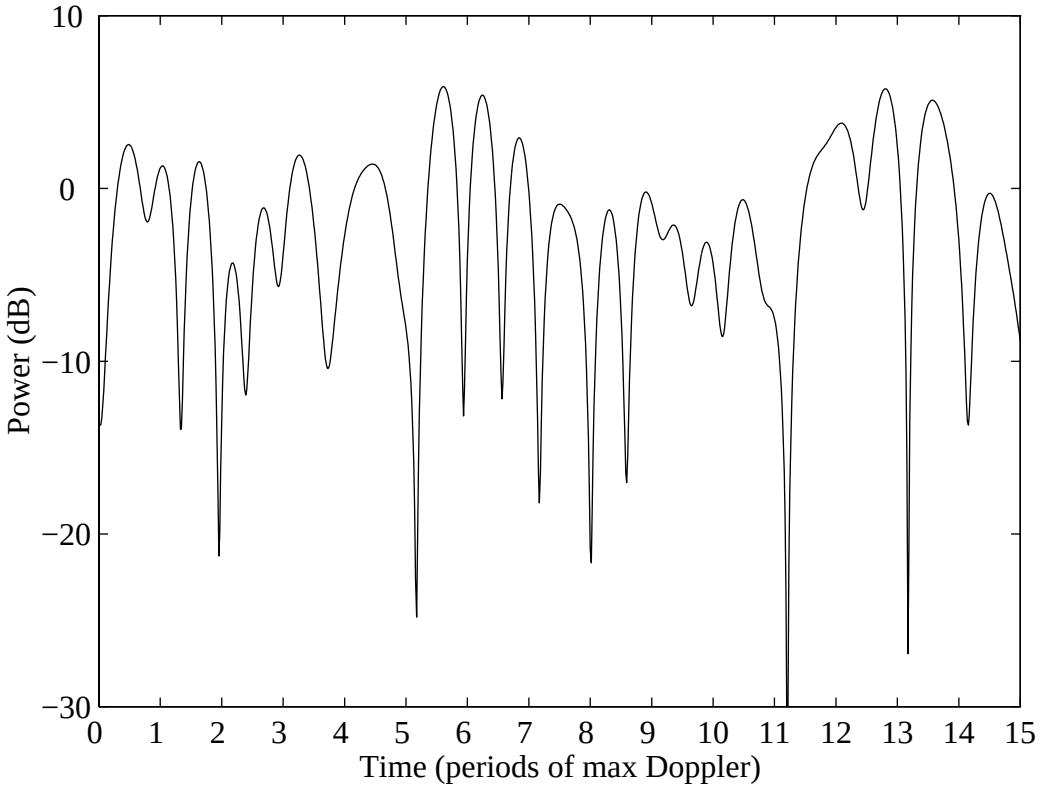

Figure 2.21: Typical Rayleigh fading envelope using Clarke's model for the Doppler spectrum.

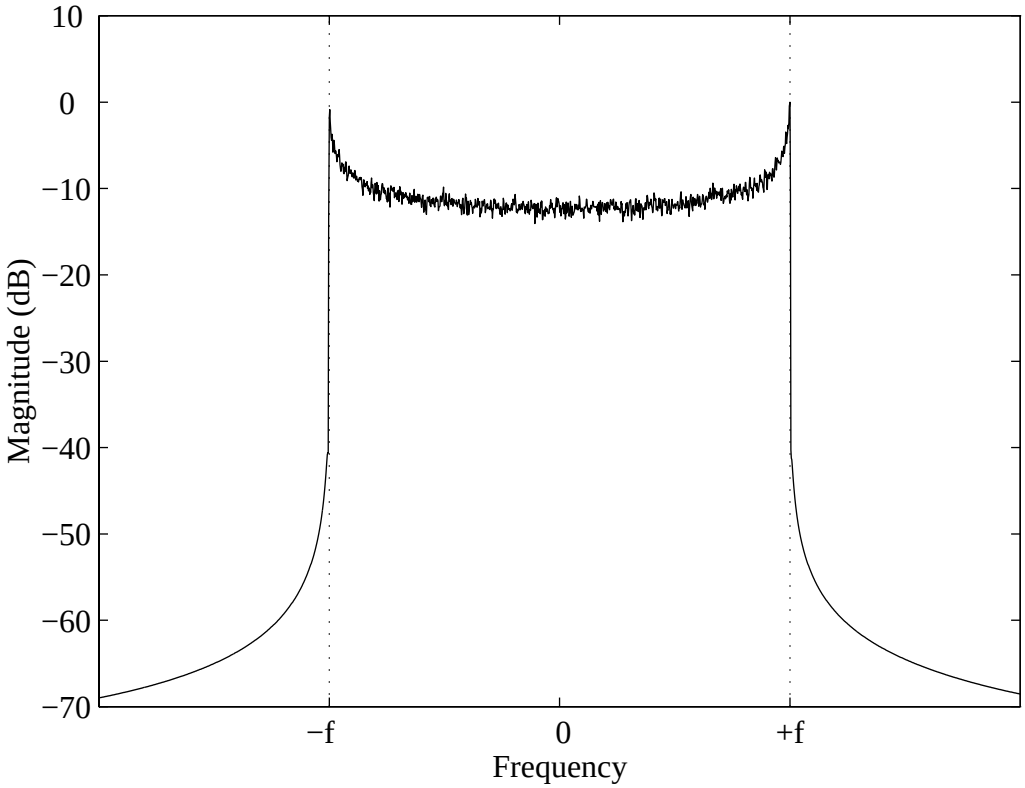

Figure 2.22: Estimated PSD of the complex envelope from a simulation of Clarke's fading model.

2.2.3 Diversity Reception

Diversity reception is used in many mobile wireless communication systems in order to combat fading [25]. The basic concept of all diversity techniques is to exploit multiple transmission channels that have uncorrelated fading. If the fading on each channel is uncorrelated, it is unlikely that all channels will be in a fade simultaneously. Different forms of diversity include:

1. spatial;
2. temporal;
3. frequency; and
4. polarization diversity.

These forms of diversity can be exploited through several different approaches, including, in order of increasing complexity:

1. switched diversity, where the 'best' channel is selected and the other channels are neglected;
2. equal gain combining, where all channels are phase aligned, then summed without regard to the signal strength in each channel;
3. maximum ratio combining (MRC), where the channels are weighted and summed to maximize the output power.

We could also add adaptive beamforming to this list; this could be termed 'optimal combining' [26]. The principal difference between MRC and adaptive beamforming is that MRC makes no attempt to differentiate between signal power and interference power – MRC only seeks to maximize the total output power.

The performance of a diversity receiver depends in large part on the correlation between the received envelopes, with uncorrelated envelopes providing the best performance. A method for generating two Rayleigh fading envelopes having a prescribed correlation is described in [27].

2.2.4 Excess Delay Spread

It was stated in Subsection 2.1.1 that the use of the narrowband model does not preclude the study of frequency selective channels. To illustrate this point, consider a situation where multipath is present, and the signal received at the reference antenna is given by

$$x_1(t) = s(t) + s(t+\tau) + q_1(t) \tag{2.92}$$

where τ is the excess delay spread and τ is large relative to the symbol period. In such a case the delay τ cannot be modelled by a simple phase shift. The signal received at the kth antenna is

$$x_k(t) = s(t-\Delta_{k,1}) + s(t+\tau-\Delta_{k,2}) + q_k(t) \tag{2.93}$$

where $\Delta_{k,i}$ is the delay seen by the ith multipath component as it propagates across the array. Note that, in general, $\Delta_{k,1} \neq \Delta_{k,2}$ since each path may arrive from a different direction. However, if the antennas are relatively closely spaced, each delay $\Delta_{k,i}$ is small relative to the inverse signal bandwidth, and

$$x_k(t) \approx m(t)\cos(\omega_c t - \phi_{k,1}) + m(t+\tau)\cos(\omega_c t - \phi_{k,2}) + q_k(t) \tag{2.94}$$

Thus the narrowband model is still appropriate, but each path having significant excess delay spread must be treated as a separate signal.

2.3 Classical Adaptive Algorithms

There are two main motivations for adaptive, as opposed to fixed, beamforming. First, adaptive processing provides a means for finding appropriate beamformer weights in an unknown environment. Since the signal environment will most likely be incompletely known, this is an important consideration. The second motivation for using adaptive processing is to track changes in the signal environment caused by, e.g, moving emitters. In this section we describe several commonly used adaptive architectures. We motivate these architectures under the assumption that a known training signal is available. However, many of these adaptive architectures can be modified to operate without the benefit of a training signal. Such algorithms are generally known as *blind* adaptive algorithms and will be discussed in more detail in Section 2.5.

2.3.1 Direct Least Squares

The Least Squares (LS) weight vector is given by

$$\mathbf{w}_{\mathrm{LS}} = \hat{\mathbf{R}}_{xx}^{-1}\hat{\mathbf{R}}_{xs} \tag{2.95}$$

where

$$\hat{\mathbf{R}}_{xx} = \sum_{n=0}^{N-1} \mathbf{x}(n)\mathbf{x}^H(n) \tag{2.96}$$

is the sample covariance matrix and

$$\hat{\mathbf{R}}_{xs} = \sum_{n=0}^{N-1} \mathbf{x}(n)s^*(n) \tag{2.97}$$

is the sample cross-correlation between the observed data $\mathbf{x}(n)$ and the known training signal $s(n)$. It is assumed that $s(n)$ is known only for $0 \leq n < N$. The LS weight vector minimizes the sample mean square error between the beamformer output and the finite duration training sequence. That is,

$$\mathbf{w}_{\text{LS}} = \arg\min_{\mathbf{w}} \sum_{n=0}^{N-1} \left| \mathbf{w}^H\mathbf{x}(n) - s(n) \right|^2 \tag{2.98}$$

where N is the length of the training signal. If the noise and interference has a Gaussian distribution, and if only a single incident signal is known, the LS method yields the maximum output SINR that can be obtained with a finite duration training sequence. If the noise and interference is not Gaussian, minimizing the mean of the squared error is not optimal. This is mainly an issue if the noise is impulsive in nature. If multiple signals are known, this can be exploited from a ML framework, as discussed earlier in Subsection 2.1.4.

The output SINR of the LS weight vector approaches the optimal output SINR as $N \to \infty$. As a rule of thumb, the number of independent samples in the training sequence $s(n)$ should at least twice the number of antennas in the array [28]. The main drawback of the LS method is the computational complexity. Generally speaking, the number of computations required is on the order of M^3, where M is the number of antennas [20]. Another potential drawback is latency. An LS weight vector cannot be calculated until all N samples have been collected. This introduces an unavoidable latency of N samples before any steps have been taken to calculate the weight vector. However, in beamforming applications, where the number of antennas M is usually small, solution of (2.95) may not be prohibitively expensive, and the rapid convergence rate may be worth the extra computational load.

It should be noted that in a practical, real-time implementation of (2.95), the matrix inverse is usually not explicitly calculated. Efficient techniques for solving (2.95) generally involve a Cholesky decomposition, or some form of triangularization of $\mathbf{R}_{xx}$. Furthermore, the LS weight vector can be found without explicitly calculating the covariance matrix. This is done by orthogonalizing the data instead of the covariance matrix.

2.3.2 Least Mean Square (LMS)

The Least Mean Square (LMS) algorithm is one of the most well-established adaptive algorithms, dating to early work by Widrow and others [29]. A detailed analysis can be found in [30]. It has been studied extensively, yet some important work remains to be done to completely understand its behavior [31]. The LMS is a type of gradient descent algorithm, and is perhaps more accurately referred to as a Stochastic Gradient Descent (SGD) algorithm. The name SGD arises from the fact that the gradient estimate typically used in a steepest descent algorithm is replaced with an instantaneous, and hence noisy (i.e., stochastic) estimate of the gradient. In general, a steepest descent algorithm iteratively finds the weight vector $\mathbf{w}$ that minimizes a cost function $J(\mathbf{w})$ by the recursion

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \nabla_w J(\mathbf{w}) \tag{2.99}$$

where μ is a positive step size and ∇_w is the gradient operator with respect to the weight vector. The selection of the step size μ is critical. If too large a value is selected, the algorithm will become unstable and the weight vector $\mathbf{w}$ will grow without bound. The SGD weight vector update replaces the gradient vector with an instantaneous estimate of the gradient. When the cost function is of the form

$$J(\mathbf{w}) = \mathcal{E}\left\{ \left| \mathbf{w}^H \mathbf{x}(n) - s(n) \right|^2 \right\} \tag{2.100}$$

the SGD weight vector update can be written as

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \mathbf{x}(n)\, e^*(n) \tag{2.101}$$

where the error signal $e(n)$ is given by

$$e(n) = y(n) - s(n) \tag{2.102}$$

The most attractive feature of the SGD approach is its low computational complexity. The main drawback of the SGD method is its very slow convergence rate. It is not uncommon for an SGD method to require hundreds or even thousands of iterations for convergence.

2.3.3 Normalized LMS

The Normalized LMS (NLMS) is very similar to the LMS method, but offers much faster convergence with only a slightly higher computational load. The NLMS update is given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \frac{\mathbf{x}(n)}{\mathbf{x}^H(n)\mathbf{x}(n)}\, e^*(n) \tag{2.103}$$

2.3.4 Recursive Least Squares (RLS)

The RLS method is not a steepest descent technique, but rather an iterative, approximate solution to a LS problem. Because it is an approximate LS method, it typically offers much faster convergence than SGD methods. The overview presented here closely follows the development presented in [29], and is mainly intended to illustrate the basic concept behind the RLS approach.

The key feature of the original RLS approach is an iterative method for estimating the inverse data covariance matrix $\mathbf{R}_{xx}^{-1}$. The covariance matrix is estimated using an exponentially decaying data window, so that the estimate of $\mathbf{R}_{xx}$ at time n is

$$\hat{\mathbf{R}}_n = \sum_{i=0}^{n} \alpha^{n-i} \mathbf{x}(i)\mathbf{x}^H(i) \tag{2.104}$$

where the scalar

$$0 < \alpha < 1 \tag{2.105}$$

provides the estimate a 'fading memory' that is needed to track non-stationary environments. The covariance matrix estimate can be updated recursively by noting that

$$\hat{\mathbf{R}}_n = \alpha \hat{\mathbf{R}}_{n-1} + \mathbf{x}(n)\mathbf{x}^H(n) \tag{2.106}$$

Now consider a recursive update for an estimate of the matrix inverse of $\hat{\mathbf{R}}_n$. Using Woodbury's identity (2.48), we have

$$\hat{\mathbf{R}}_n^{-1} = \alpha \hat{\mathbf{R}}_{n-1}^{-1} - \frac{\hat{\mathbf{R}}_{n-1}^{-1}\mathbf{x}(n)\mathbf{x}^H(n)\hat{\mathbf{R}}_{n-1}^{-1}}{1 + \frac{1}{\alpha}\mathbf{x}^H(n)\hat{\mathbf{R}}_{n-1}^{-1}\mathbf{x}(n)} \tag{2.107}$$

The cross-correlation vector $\mathbf{R}_{xs}$ can be updated recursively in the same manner as (2.106). This allows for a recursive implementation of the LS algorithm that has relatively low complexity. Note that the initial estimate used for $\hat{\mathbf{R}}_n^{-1}$ can have a great impact on RLS behavior. In the absence of *a priori* information, an initial value that is proportional to the identity matrix is often used in practice.

2.4 DF/Copy

In addition to beamforming methods that exploit a known training sequence, a great number of algorithms have been developed which are based on Direction Finding (DF). (An excellent overview of modern DF approaches can be found in [32]). DF-based beamforming methods first estimate the AOA of the desired signal (and perhaps other signals in the

environment), then use this information to calculate the beamformer weight vector. These methods are sometimes referred to as DF/Copy methods, since DF is performed before a 'copy' of the desired signal is obtained. These methods are probably of limited practicality in mobile wireless applications. There are several reasons for this. First, performing DF in an environment containing coherent multipath requires the use of complicated DF algorithms. Second, the use of the methods requires that the array response vectors be known. This typically requires array calibration, which is difficult and expensive to perform. Another issue is that the multipath may be diffuse, in which case DF is meaningless. Finally, any beamforming method that relies on DF is sensitive to some extent on DF and calibration error. Algorithms that exploit known training sequences or some property of the signal (e.g., FM and FSK signals exhibit the constant modulus property) are not dependent on array calibration, and generally perform much better in practice than DF-based methods. Even though DF/Copy methods are of limited practicality, a review of some of the more common DF/Copy approaches offers insight into the general behavior of beamforming algorithms.

2.4.1 MVDR

In subsection 2.1.3 the optimal weight vector was derived by explicitly maximizing the output SINR, using the assumption that the array response vector and the covariance matrix of the noise and interference are known. Given accurate array calibration data, the array response vector for the desired signal can be determined by DF. However, the noise and interference covariance matrix cannot be determined from the observed data because the desired signal is also present. An alternative to explicitly maximizing the output SINR is to minimize the total output power subject to the constraint that the gain in the desired look direction be unity. This approach yields the minimum variance distortionless response (MVDR) weight vector, given by

$$\mathbf{w}_{MVDR} = \mathbf{R}_{\text{xx}}^{-1}\mathbf{a}(\theta) \tag{2.108}$$

where $\mathbf{R}_{\text{xx}} = \left\langle \mathbf{x}\mathbf{x}^H \right\rangle$ is an estimate of the observed data covariance matrix. For infinite collect time, the MVDR weight vector differs only by a scale factor from the weight vector found by explicitly maximizing the output SINR. Thus the MVDR weight vector is optimal in the maximum output SINR sense. Unfortunately, the MVDR approach is known to be hypersensitive to errors in the array response vector $\mathbf{a}(\theta)$, which arise through array calibration error and DF error [33].

This is demonstrated in Figure 2.23. The scenario used in this demonstration is similar to that considered in Figure 2.6 – a four element uniform linear array, with a 20 dB SOI at 0°, and a 20 dB SNOI at 15°. Figure 2.23 shows the output SINR of the MVDR beamformer

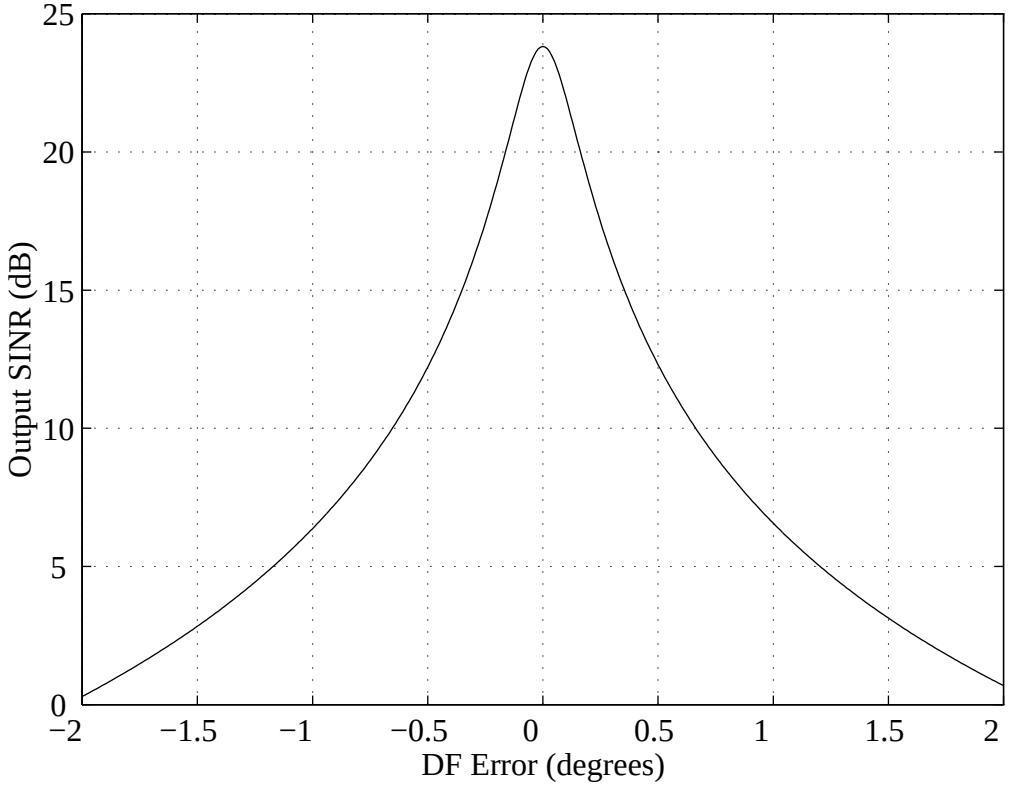

Figure 2.23: Output SINR of the MVDR beamformer as a function of DF error.

as a function of DF error. That is, the weight vector is calculated according to

$$\mathbf{w} = \mathbf{R}_{xx}^{-1}\mathbf{a}(\theta + \Delta) \tag{2.109}$$

where Δ is the DF error. Note that the output SINR drops dramatically even with very small DF error. The MVDR also has poor performance in multipath environments. For these reasons, the MVDR approach is rarely used in practice.

2.4.2 Principal Components

The hypersensitivity of the MVDR beamformer may be viewed as the result of the beamformer having excess degrees of freedom. When there is DF error, the beamformer is able to minimize the output power by nulling the desired signal as well as the interferer. One solution to this problem is to reduce the degrees of freedom by constraining the weight vector to be a linear combination of the signal subspace eigenvectors [34, 35, 36, 37]. This will be referred to here as the signal subspace constrained (SSC) approach. In the case where

the background noise is white, and there are fewer incident signals L than sensors M, the observed data correlation matrix may be expressed as

$$\mathbf{R}_{\text{xx}} = \mathbf{U}_{\text{S}}\mathbf{\Sigma}_{\text{S}}\mathbf{U}_{\text{S}}^{H} + \sigma_{\text{N}}\mathbf{U}_{\text{N}}\mathbf{U}_{\text{N}}^{H} \tag{2.110}$$

where $\mathbf{U}_{\text{S}}$ is an $M \times L$ matrix of signal subspace eigenvectors, $\mathbf{\Sigma}_{\text{S}}$ is an $L \times L$ diagonal matrix of the corresponding eigenvalues, and $\mathbf{U}_{\text{N}}$ is an $M \times (M-L)$ matrix of signal nullspace eigenvectors. Since the steering vector $\mathbf{a}$ lies in the signal subspace, $\mathbf{U}_{\text{N}}^{H}\mathbf{a} = 0$ which in turn implies

$$\mathbf{R}_{\text{xx}}^{-1}\mathbf{a} = \left(\mathbf{U}_{\text{S}}\mathbf{\Sigma}_{\text{S}}^{-1}\mathbf{U}_{\text{S}}^{H} + \frac{1}{\sigma_{\text{N}}}\mathbf{U}_{\text{N}}\mathbf{U}_{\text{N}}^{H}\right)\mathbf{a} \tag{2.111}$$

$$= \mathbf{U}_{\text{S}}\mathbf{\Sigma}_{\text{S}}^{-1}\mathbf{U}_{\text{S}}^{H}\mathbf{a} \tag{2.112}$$

This clearly shows that $\mathbf{w}_{opt}$ lies in the signal subspace. Using (2.112) to solve for the beamformer weight vector has been shown in practice to yield significantly better results than direct solution of (2.43).

Many other approaches for robust DF/Copy have been proposed in the literature. In particular the artificial noise injection (ANI) approach should be mentioned. This approach calls for finding the beamformer weight vector by solving

$$\mathbf{w}_{ANI} = \left(\mathbf{R}_{\text{xx}} + \sigma\mathbf{I}\right)^{-1}\mathbf{a}(\theta) \tag{2.113}$$

where σ is the power of the artificially injected noise. This results in a beamformer that is much more robust to DF and calibration error than MVDR, but at the cost of reduced output SINR. Another drawback of this approach is the difficulty in determining the correct value for σ.

2.4.3 Least Squares Beamformer

This approach assumes that the steering vectors of all L incident signals are known or have been estimated. Let $\mathbf{A}$ be the $M \times L$ matrix whose columns are the steering vectors of the incident signals. Then find the signal estimates $\hat{\mathbf{s}}(n)$ that minimizes

$$\text{E}\left\{\left|\mathbf{x}(n) - \mathbf{A}\hat{\mathbf{s}}(n)\right|^{2}\right\} \tag{2.114}$$

The resulting signal estimate is

$$\hat{\mathbf{s}}(n) = \left(\mathbf{A}^{H}\mathbf{A}\right)^{-1}\mathbf{A}^{H}\mathbf{x}(n) \tag{2.115}$$

and the corresponding beamformer weight vectors are

$$\mathbf{w}_{ls} = \mathbf{A}\left(\mathbf{A}^H\mathbf{A}\right)^{-1}. \tag{2.116}$$

Note that the weight vectors for extracting all the incident signals are computed. This beamformer places unity gain on the desired signal and places zero gain (an infinitely deep null) on the interferers. This can be seen by noting that

$$\begin{aligned} \mathbf{w}_{ls}^H\mathbf{A} &= \left(\mathbf{A}^H\mathbf{A}\right)^{-1}\mathbf{A}^H\mathbf{A} && (2.117)\\ &= \mathbf{I}_L && (2.118) \end{aligned}$$

where $\mathbf{I}_L$ is the $L \times L$ identity matrix. Because this beamformer steers infinitely deep nulls, and does not consider the effects of white noise, it does not achieve the optimal output SINR. However, if the input SWNR of the signals is reasonably high, the difference in output SINR of the Least Squares and optimal beamformer will be small. This method is more robust to DF error than the MVDR approach since it does not seek to minimize output power subject to some (possibly erroneous) weight vector constraint.

2.5 Blind Adaptive Beamforming

To this point we have limited our discussion to algorithms that exploit a known training sequence in the received data. In so doing, we have neglected some practical issues that must be addressed. In particular, the exploitation of a training sequence places several requirements on the receiver that can be difficult to meet in practice. First, the timing of the training sequence must be known. That is, the receiver may know that the transmitter periodically transmits a certain sequence of training symbols, but how does the receiver know when to expect that sequence? Second, the carrier frequency must be known, at least for classic approaches that seek to directly minimize MSE. Carrier and timing synchronization are very difficult to obtain in low SINR and multipath environments. However, these are the very environments where adaptive beamforming offers the most benefits. Therefore, the use of algorithms that do not require carrier or timing synchronization (at least initially) is highly desirable. One way to avoid the need for such synchronization is to exploit some property that the desired signal is known to exhibit. This generally leads to so-called 'blind' adaptive algorithms. An adaptive algorithm is described as blind if it does not require a known training sequence. There are additional reasons why blind adaptive algorithms are used in practice, in addition to eliminating (or reducing) synchronization requirements. First, it is common practice to switch to a blind mode after a training sequence has been processed in order to track a changing environment. Furthermore, the use of a blind algorithm can

completely eliminate the need for a training sequence. This is desirable since the use of a training sequence reduces the number of bits available for transmitting information.

These techniques in general exploit some property that the desired signal is known to exhibit so that an indirect measure of the output SINR can be obtained. By optimizing an appropriate cost function based on this property, a high quality estimate of the desired signal can be obtained. One such property is the constant modulus property exhibited by phase and frequency modulated signals. Another is the second order cyclostationarity exhibited by many communication signals, including PSK, QAM, and AM, at cycle frequencies such as the baud rate, double carrier frequency, and sums and differences of these. This section presents a brief overview of several important classes of blind adaptive algorithms, including:

1. decision directed algorithms (DDA);
2. constant modulus algorithms (CMA);
3. time-, frequency-, and code-gated algorithms;
4. spectral self-coherence restoral (SCORE);

An important blind signal separation approach not discussed here is Independent Component Analysis (ICA). An overview of ICA is presented in [38]. These methods operate on the assumption that the received signals have a non-Gaussian distribution, and use a discrimination function to distinguish between independent signals. Essentially they exploit the property that, for example, a mix of two random processes with a Laplacian distribution will not be Laplacian. Thus these methods can be applied to signals that have kurtosis greater than (or less than) the kurtosis of Gaussian noise. Note that a mix of two Gaussian random processes is itself Gaussian, and so the ICA methods cannot be used to separate Gaussian mixtures. These techniques have only recently been applied to communication signals, and it is felt that other blind algorithms offer the potential for better performance.

2.5.1 Decision Directed Algorithms

Perhaps the most natural approach to blind adaptive signal processing is the decision-directed approach, which can be applied to any digitally modulated signal. In this approach, the received data is passed through a linear combiner, then the output is demodulated. The estimated symbols are then treated in the same manner as a known training signal. A decision directed algorithm (DDA) is often used in combination with a training sequence. The training sequence is used to obtain an initial solution, so that the symbol error rate

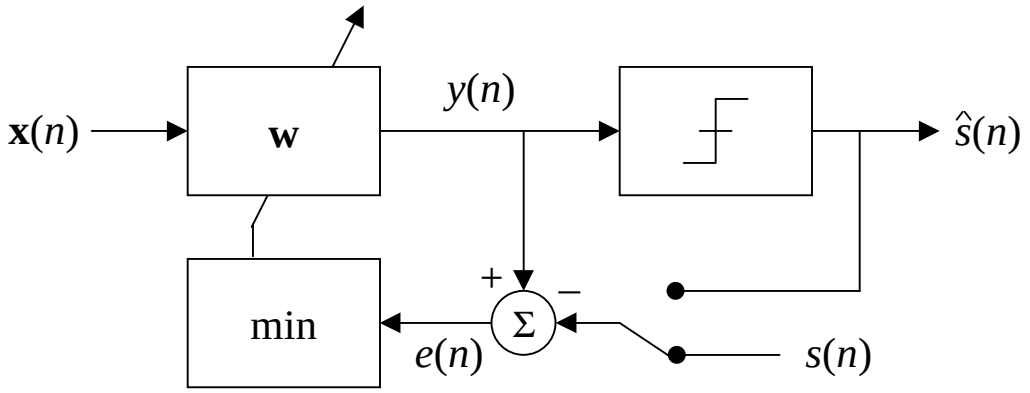

Figure 2.24: Simplified block diagram of an adaptive linear combining algorithm, with a switch between a known training signal mode and decision directed mode. This block diagram omits the timing synchronization and carrier synchronization that must be performed.

is low. Then the algorithm is switched to a decision-directed mode to further refine the solution, and to track changes in the environment. This is illustrated in Figure 2.24. This simplified block diagram omits the timing synchronization and carrier synchronization that must be performed. That is, the symbol timing, timing of the training sequence relative to the received signal, and the carrier frequency of the received signal must be estimated before the adaptive algorithm can operate properly. The need for synchronization is often a difficult problem to overcome in practice. Synchronization is very difficult to obtain in low SINR and highly dispersive environments. However, these are precisely the environments for which an adaptive antenna array offers the greatest benefits. This provides additional motivation for the use of blind algorithms such as CMA, which do not require carrier synchronization and are robust to symbol timing errors.

A DDA is sometimes referred to as exploiting the 'finite alphabet' property of digitally modulated signals. This simply refers to the fact that a digital message is known to take on one of a finite set of possible values, i.e, the signalling is drawn from a finite alphabet.

Many different versions of DDA can be derived, including LMS, Normalized LMS, RLS, and block LS. The block LS method is referred to in this document as the Least Squares DDA, or LSDDA, and is studied in detail in Chapter 4.

2.5.2 Constant Modulus Algorithms

In this subsection a brief overview of CMA is presented. The CMA is perhaps the most well-known blind algorithm, and is used in many practical applications because it does not

require carrier synchronization. In general, CMA seeks a beamformer weight vector that minimizes a cost function of the form

$$F_{pq} = \langle \, (|y(n)|^p - 1)^q \, \rangle \tag{2.119}$$

where $y(n) = \mathbf{w}^H \mathbf{x}(n)$ is the estimate of the desired signal, $\mathbf{w}$ is the beamformer weight vector, and $\mathbf{x}$ is the observed data. By selecting values of 1 or 2 for p and q, different versions of CMA may be obtained. It is common to refer to a particular CMA as a (p, q)-CMA. The cost function (2.119) makes sense from an intuitive perspective since noise and interference will destroy the constant modulus property of the desired signal. Passing the signal through a frequency selective channel will also destroy the constant modulus property. Thus the CMA cost function provides an indirect measure of the SINR of the received signal. It has been shown [39, 40] that minimizing the (2,2) CMA cost function is equivalent to minimizing the kurtosis κ_y of the beamformer (or equalizer) output, where

$$\kappa_y \triangleq \frac{\langle |y(n)|^4 \rangle}{\langle |y(n)|^2 \rangle^2}. \tag{2.120}$$

Using this definition, a constant modulus signal, such as FM or rectangular pulse shaped PSK, has a kurtosis of one. In comparison, complex Gaussian noise has a kurtosis of two. The SNR of a constant modulus signal received in Gaussian noise can be estimated from the kurtosis of the noisy data. It can be shown that, for infinite time average,

$$\text{SNR} = \frac{2 - \kappa_y + \sqrt{2 - \kappa_y}}{\kappa_y - 1} \tag{2.121}$$

As $\kappa_y \to 1$, the SNR $\to \infty$. Furthermore, as $\kappa_y \to 2$, the SNR $\to 0$. Therefore if the background noise is Gaussian, CMA is capable of extracting any signal that has a kurtosis less than two.

One of the attractive features of the CMA is that carrier synchronization is not required. Furthermore, it can be applied successfully to non-constant modulus signals if the kurtosis is less than two. This means that the CMA can be applied to, for example, PSK signals that have non-rectangular pulse shape. This is important because this implies that the CMA is also robust to symbol timing error when applied to pulse-shaped PSK signals. Pulse shaping typically is used to limit the occupied bandwidth of the transmitted signal. Pulse shaping a PSK signal has the effect of introducing variation in the signal envelope, i.e. the kurtosis of a pulse shaped PSK signal is larger than one. If the pulse-shaped PSK signal is filtered and sampled appropriately, (baud synchronous sampling), then the samples will have a constant envelope. If there is some error in symbol timing, the resulting samples will not have a constant envelope, and the kurtosis will be larger than one. However, the kurtosis will still be less than two, so that CMA can still be applied. It is well known that the more

stringent the pulse-shaping, i.e., the lower the excess bandwidth, the greater the envelope variation, and hence the larger the kurtosis. The modulation format can also affect the kurtosis. For example $\pi/4$-QPSK is commonly used in practice because it has lower kurtosis than conventional QPSK. This is important because signals with small envelope variation undergo less distortion when transmitted through a non-linear amplifier.

The technique first proposed for minimizing a CMA cost function uses a Stochastic Gradient Descent (SGD) approach [41]. The weight vector $\mathbf{w}$ is updated according to

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \, \hat{\nabla}_w F_{pq} \tag{2.122}$$

where μ is a positive step size and $\hat{\nabla}_w$ is an instantaneous estimate of the gradient of the cost function with respect to the weight vector. When the (2,2) CMA cost function is used, the resulting SGD algorithm is given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu(|y(k)|^2 - 1)\mathbf{x}(k)y^*(k) \tag{2.123}$$

where $y(k) = \mathbf{w}^H\mathbf{x}$ is the processor output. When the (1,2) CMA cost function is used, the resulting SGD algorithm is

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu\mathbf{x}(k)\left(y(k) - \frac{y(k)}{|y(k)|}\right)^* \tag{2.124}$$

Because of the slow convergence rate associated with all SGD algorithms, techniques with better convergence rates were sought.

Gooch and Lundell proposed the Orthogonalized CMA, or O-CMA, algorithm in [42]. This algorithm is similar in form to the recursive least squares (RLS) algorithm. It is based on the (1,2) CMA cost function and is given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\mathbf{R}^{-1}(k+1)\mathbf{x}(k)\epsilon^*(k) \tag{2.125}$$

where

$$\epsilon^*(k) = y(k)/\left|y(k)\right| - y(k) \tag{2.126}$$

and

$$\mathbf{R}^{-1}(k+1) = \frac{\mathbf{R}^{-1}(k)}{1-\alpha} - \frac{1}{1-\alpha}\left[\frac{\alpha\mathbf{R}^{-1}(k)\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{R}^{-1}(k)}{(1-\alpha) + \alpha\mathbf{x}^H(n)\mathbf{R}^{-1}(k)\mathbf{x}(n)}\right] \tag{2.127}$$

This algorithm exhibits improved convergence properties compared to the SGD version.

Another fast converging optimization technique is the Least Squares CMA [43, 44]. The LSCMA is briefly introduced here, and is studied in detail in Chapter 3. The LSCMA has been found to converge very rapidly and is also guaranteed to be stable. This is a block

update technique which uses an alternating projections approach to minimize the (1,2) CMA cost function. From a block of N samples of observed data, an initial estimate for the desired signal is formed. The signal is then divided by its magnitude to yield

$$d(n) = \frac{y(n)}{|y(n)|}. \tag{2.128}$$

The signal $d(n)$ is the hard limit of $y(n)$. This operation may be viewed as a projection onto the set of signals which have the desired constant modulus property. The updated weight vector is found by minimizing the average squared error between $d(n)$ and a new estimate of the desired signal by

$$\mathbf{w} = \mathbf{R}_{\mathrm{xx}}^{-1}\mathbf{r}_{xd} \tag{2.129}$$

where

$$\mathbf{r}_{xd} = \langle\, \mathbf{x}(n)\, d^{*}(n) \,\rangle\,. \tag{2.130}$$

The covariance matrix $\mathbf{R}_{\mathrm{xx}}$ is computed from the same block of data used to form the estimate of the desired signal. This process is then repeated using the new weight vector and is continued until the algorithm has converged. Thus the approach first projects the signal onto the set of signals which has the desired property, and then finds the closest approximation to this projected signal that can be obtained using a beamformer. The technique as described above uses the same data for all the iterations. However, the technique will also converge if non-overlapping blocks of data are used at each iteration. In this case the weight vector computed from a previous block of data is used to form the estimated signal for the next block. This latter approach might be more practical in a real-time system.

The only parameter that must be selected with LSCMA is the block size N. Smaller block sizes yield faster convergence at the cost of higher misadjustment error. Therefore the block size used in LSCMA is similar in some ways to the adaptation step parameter μ used in SGD. A major difference is that LSCMA is numerically stable for any value of N. Typical values for N range from 30 to 1000 samples of observed data. An interesting feature of LSCMA is that convergence as a function of the number of weight vector updates is nearly constant. Thus larger block sizes result in slower convergence only because the number of samples in each block is larger.

One potential drawback of the CMA is that, in an environment containing multiple CM signals, a CMA beamformer will typically extract the strongest signal. This may or may not be the desired signal. The solution proposed by many people is to extract *all* of the CM signals in the environment, and then determine which is the desired signal. Several different methods to extract multiple CM signals have been proposed. One approach is to apply a CMA beamformer from the data, cancel the CMA beamformer output from the array data using an adaptive canceller, and then apply a second CMA beamformer to this

modified array data. This process is then repeated for as many stages as there are signals in the environment. This is referred to as the Multi-Stage CMA [45]. This method generally performs sub-optimally since the cancellation stage limits the output SINR of subsequent stages. An alternative approach that offers much better performance is the Multi-Target CMA [46]. The MT-CMA applies multiple CMA beamformers independently and *in parallel* to the same set of array data, then performs additional processing to ensure that each beamformer extracts a different CM signal. This additional processing typically performs some sort of orthogonalization on the set of signals extracted by the multiple independent CMA beamformers.

The Iterative Least Squares with Projection [47] can be also be used to extract multiple CM signals. The ILSP can be viewed as a direct generalization of the LSCMA to multiple signals. This is sometimes referred to as an *alternating projections* approach. The ILSP is implemented with the following iterative procedure:

1. At the kth iteration, generate a set of L beamformer outputs $\mathbf{Y}_k$ with

$$\mathbf{Y}_k = \mathbf{W}_k^H \mathbf{X} \tag{2.131}$$

   where each column of $\mathbf{W}_k$ is a beamformer weight vector.

2. Hard-limit each of the beamformer outputs, so that

$$\mathbf{D}_k = \frac{\mathbf{Y}_k}{|\mathbf{Y}_k|} \tag{2.132}$$

   where the absolute value and division here are understood to correspond to element-by-element operations.

3. Estimate a new set of spatial signatures

$$\hat{\mathbf{A}}_{k+1} = \mathbf{X}\mathbf{D}_k^H \left(\mathbf{D}_k \mathbf{D}_k^H\right)^{-1} \tag{2.133}$$

4. Form a new set of weight vectors

$$\mathbf{W}_{k+1} = \hat{\mathbf{A}}_{k+1} \left(\hat{\mathbf{A}}_{k+1}^H \hat{\mathbf{A}}_{k+1}\right)^{-1} \tag{2.134}$$

The Analytic CMA (ACMA) algorithm presented by van der Veen in [48, 49] should also be noted. This algorithm solves directly for a set of beamformer weight vectors that spatially separate a set of CM signals. The ACMA, although effective in many situations, is fairly complex and its behavior with closely spaced and/or low SNR signals is not clear. For these reasons it is recommended in [48] that the ACMA be used to initialize the LSCMA, and that several iterations of the LSCMA be used to find the optimal solutions for the weight vectors.

2.5.3 Time-, Frequency-, and Code-Gated Algorithms

Many signals are gated in either time or frequency. A push-to-talk signal, for example, is gated in time, while a narrowband signal in wideband interference is gated in frequency. A CDMA signal becomes gated in frequency when it is despread. Several different algorithms have been developed to exploit these properties [50, 51, 52, 53]. The main idea of the property-gated approach is that some operation can be performed on the array data so that the desired signal is emphasized relative to the background noise and interference. For example, the background noise and interference covariance matrix can be measured in a frequency hop channel before or after the hop. This provides a great deal of information about the environment.

For illustrative purposes, the time-gated algorithm will be examined. While the desired signal is off, the covariance matrix of the background noise and interference $\mathbf{R}_{\text{off}}$ is computed. When the desired signal is on, the covariance matrix of the desired signal plus the background noise and interference, $\mathbf{R}_{\text{on}}$, is computed. Then the weight vector that maximizes the quantity

$$F = \frac{\mathbf{w}^H \mathbf{R}_{\text{on}} \mathbf{w}}{\mathbf{w}^H \mathbf{R}_{\text{off}} \mathbf{w}} \tag{2.135}$$

is computed. This weight vector maximizes the output SINR of the desired signal because

$$\mathbf{R}_{\text{on}} = \sigma_s^2 \mathbf{a}\mathbf{a}^H + \mathbf{R}_{\text{ii}} \tag{2.136}$$

$$\mathbf{R}_{\text{off}} = \mathbf{R}_{\text{ii}} \tag{2.137}$$

and therefore

$$F = SINR + 1. \tag{2.138}$$

The weight vector that maximizes (2.135) is found by solving the generalized eigenequation

$$\mathbf{R}_{\text{on}} \mathbf{w} = \lambda \mathbf{R}_{\text{off}} \mathbf{w} \tag{2.139}$$

for the dominant eigenvector. This can also be used for detecting the presence of a transient signal whose on-time is unknown. This could be done by computing $\mathbf{R}_{\text{on}}$ and $\mathbf{R}_{\text{off}}$ for adjacent (in time) blocks of data, solving the generalized eigenequation, and using the dominant eigenvalue as the detection statistic. When there is no change in the signal environment, the dominant eigenvalue will be approximately unity. When a new signal appears in $\mathbf{R}_{\text{on}}$, the dominant eigenvalue will be approximately equal to the optimal output SINR of the new signal.

Signal	kf_0 [a]	$2f_c$ [b]	$2f_c \pm kf_0$
BPSK	$\surd$	$\surd$	$\surd$
QPSK	$\surd$		
SQPSK	k even		k odd

[a] f_0 denotes baud rate, k a nonzero integer
[b] f_c denotes carrier frequency

Table 2.1: Cyclic features of some bauded signals.

2.5.4 Self Coherence Restoral

The class of Self Coherence Restoral (SCORE) algorithms are designed to differentiate between desired signals and interference by exploiting the second order cyclostationarity exhibited by the desired signal [54, 55, 56, 57]. A complex baseband signal $s(n)$ is said to exhibit second-order cyclostationarity if the lag-product waveform $s(n)s^*(n-\tau)$ contains a finite strength sine wave component, or *spectral line*. The same signal exhibits conjugate cyclostationarity if the lag-product waveform $s(n)s(n-\tau)$ contains a spectral line. Typical cyclic features of some commonly used digital communication signals are shown in Table 2.1. Features associated with the doubled carrier frequency correspond to conjugate cyclic correlation.

The magnitude and phase of the spectral line at cycle frequency α for lag τ is given by the (asymmetric) cyclic autocorrelation function defined by

$$\mathrm{R}_{ss}^{\alpha}(\tau) \triangleq \left\langle s(n)s^*(n-\tau)e^{-j2\pi\alpha n} \right\rangle_{\infty} \tag{2.140}$$

where $\langle\cdot\rangle_{\infty}$ denotes infinite time average. The relative strength of the spectral line is given by the cyclic correlation coefficient, defined as

$$\rho_{ss}^{\alpha}(\tau) \triangleq \frac{\mathrm{R}_{ss}^{\alpha}(\tau)}{\mathrm{R}_{ss}(0)}. \tag{2.141}$$

Techniques which exploit cyclostationarity in general rely on the property that noise and interference will not, in the limit as the collect time approaches infinity, contribute to a cyclic correlation or cyclic spectrum. For finite collect time, there will be some residual contribution from noise and interference. The weaker the cyclic feature, the longer the collect time must be in order to sufficiently decorrelate the noise. Therefore the cyclic correlation coefficient is an important parameter for predicting the performance of cyclostationarity exploiting algorithms. The strength of cyclic features associated with data rates (e.g. chip and baud features) is directly dependent on the excess bandwidth, with low excess bandwidth

signals exhibiting weak second-order cyclostationarity. An approximate expression for the magnitude of the cyclic correlation coefficient as a function of the excess bandwidth B is [57]

$$|\rho_{ss}^{\alpha}(\tau)| \cong \frac{B/8}{1-B/4} \tag{2.142}$$

Note that this function rolls off very quickly as the excess bandwidth decreases. Severe pulse shaping can completely eliminate second order cyclostationarity associated with chip or baud rates.

There are several different versions of SCORE, with each version behaving differently depending on the environment. The simplest version is Least Squares (LS) SCORE. The weight vector for this version of SCORE is typically computed as

$$\mathbf{w}_{ls} = \mathbf{R}_{\text{xx}}^{-1}\mathbf{r}^{\alpha} \tag{2.143}$$

where $\mathbf{r}^{\alpha}$ is an $M \times 1$ vector given by

$$\mathbf{r}^{\alpha} = \left\langle \mathbf{x}(n)\, x_i^*(n-\tau)e^{-j2\pi\alpha n} \right\rangle, \tag{2.144}$$

$\mathbf{x}(n)$ is the $M \times 1$ vector of observed data, α is the cycle frequency being exploited, τ is the lag parameter, and $\langle\cdot\rangle$ denotes a time averaging operation. If only the desired signal exhibits cyclostationarity at cycle frequency α, then $\mathbf{r}^{\alpha}$ approaches the array response vector of the desired signal as the collect time approaches infinity. Thus $\mathbf{w}_{ls}$ converges to the optimal weight vector. However, in low SIR environments the convergence is very slow. In such environments, the cross-SCORE method will outperform LS-SCORE. Cross-SCORE can be motivated from the Programmable Canonical Correlation Analyzer (PCCA) framework [55], as discussed below.

The cross-SCORE algorithm essentially yields a weight vector which maximizes the *magnitude* of the cyclic correlation of the beamformer output. In an environment where only one signal exhibits the cyclostationarity property being exploited, this yields a solution that will extract the desired signal with nearly the optimal output SINR. When more than one incident signal exhibits cyclostationarity at the cycle frequency being exploited, cross-SCORE will in general extract a linear combination of these signals. The exception occurs when the incident signals exhibit the same cyclic feature but with different strength. Thus, cross-SCORE would be unable to separate multiple signals that have the same modulation format.

The cross-SCORE weight vectors are solutions to the generalized eigenequation

$$\mathbf{R}_{\text{xx}}^{-1}\mathbf{R}_{\text{xx}}^{\alpha}\mathbf{R}_{\text{xx}}^{-1}\left(\mathbf{R}_{\text{xx}}^{\alpha}\right)^{H} \tag{2.145}$$

where

$$\mathbf{R}_{\text{xx}}^{\alpha} = \left\langle \mathbf{x}(n)\,\mathbf{x}^{H}(n-\tau)e^{-j2\pi\alpha n} \right\rangle \tag{2.146}$$

is a finite-time estimate of the cyclic correlation matrix of $\mathbf{x}(n)$ for cycle frequency α and lag τ. Note that the vector $\mathbf{r}^{\alpha}$ used in LS-SCORE is the ith column of $\mathbf{R}_{\text{xx}}^{\alpha}$. A key point is that cross-SCORE requires computation of the complete $M \times M$ observed data cyclic correlation matrix, while LS-SCORE requires only one column of this matrix.

In contrast with Least Squares SCORE and cross-SCORE, the phase-SCORE algorithm is capable of separating uncorrelated signals with the same modulation format if the phase of the cyclic features is different. For example, this would be the case if the incident signals had different baud timing. The phase-SCORE weight vectors are found by solving the eigenequation

$$\mathbf{R}_{\text{xx}}^{-1}\mathbf{R}_{\text{xx}}^{\alpha}(\tau)\mathbf{w} = \lambda\mathbf{w}. \tag{2.147}$$

Each of the L^{α} dominant eigenvectors of the phase-SCORE eigenequation will extract one of the desired signals, assuming that the signals are independent and the phase of the cyclic features is different. If the phase of the cyclic features is not different, phase-SCORE behaves like cross-SCORE in that it will extract a linear combination of the signals exhibiting the desired cyclic feature. In addition, if a desired signal is received from multiple angles (i.e., coherent multipath is present) phase-SCORE will extract a linear combination of the signals. Thus neither phase-SCORE nor cross-SCORE performs properly in the presence of coherent multipath.

The convergence rate of the SCORE algorithms can be improved in some cases by constraining the SCORE weight vectors to lie in the signal subspace of the observed data covariance matrix [58]. In the case where the background noise is white, and there are fewer incident signals L than sensors M, the observed data correlation matrix may be expressed as

$$\mathbf{R}_{\text{xx}} = \mathbf{U}_{\text{S}}\mathbf{\Sigma}_{\text{S}}\mathbf{U}_{\text{S}}^{H} + \sigma_{\text{N}}\mathbf{U}_{\text{N}}\mathbf{U}_{\text{N}}^{H} \tag{2.148}$$

where $\mathbf{U}_{\text{S}}$ is an $M \times L$ matrix of signal subspace eigenvectors, $\mathbf{\Sigma}_{\text{S}}$ is an $L \times L$ diagonal matrix of the corresponding eigenvalues, and $\mathbf{U}_{\text{N}}$ is an $M \times (M - L)$ matrix of signal nullspace eigenvectors. Define the rank-L, least squares approximation to $\mathbf{R}_{\text{xx}}^{-1}$ as

$$\mathbf{R}_{\text{xx}}^{\#} \triangleq \mathbf{U}_{\text{S}}\mathbf{\Sigma}_{\text{S}}^{-1}\mathbf{U}_{\text{S}}^{H}. \tag{2.149}$$

Then subspace-constrained versions of the cross-SCORE and phase-SCORE algorithms can be obtained simply by substituting $\mathbf{R}_{\text{xx}}^{\#}$ for $\mathbf{R}_{\text{xx}}^{-1}$ in (2.145) and (2.147), respectively. This improves the performance of SCORE by eliminating perturbations due to slowly converging signal nullspace eigenvectors. The incorporation of a signal subspace constraint is beneficial only when the underlying assumptions are valid. However, the subspace-constrained SCORE algorithms have been shown to be fairly robust to modeling errors.

2.5.5 Programmable Canonical Correlation Analysis

The PCCA framework yields a set of weight vectors that maximize the correlation between a linear combination of two sets of data $\mathbf{x}(n)$ and $\mathbf{z}(n)$. Typically $\mathbf{x}(n)$ is the observed data and $\mathbf{z}(n)$ is a training data set obtained through some transformation of the observed data. The transformation is designed so that it decorrelates the interference and noise but does not decorrelate the desired signal. Then the only component that causes correlation between $\mathbf{x}(n)$ and $\mathbf{z}(n)$ is due to the desired signal. Finding the linear combination of $\mathbf{x}(n)$ that maximizes the correlation between the two data sets then is equivalent to extracting the desired signal.

The PCCA weight vectors $\mathbf{W}_x$ for the observed data are given by the dominant eigenvectors of the matrix

$$\mathbf{T}_x = \mathbf{R}_{\mathrm{xx}}\mathbf{R}_{\mathrm{xz}}\mathbf{R}_{\mathrm{zz}}^{-1}\mathbf{R}_{\mathrm{zx}}. \tag{2.150}$$

The linear combiner weights $\mathbf{W}_z$ for the training data set $\mathbf{z}(n)$ are given by the dominant eigenvectors of the matrix

$$\mathbf{T}_z = \mathbf{R}_{\mathrm{zz}}\mathbf{R}_{\mathrm{zx}}\mathbf{R}_{\mathrm{xx}}^{-1}\mathbf{R}_{\mathrm{xz}}. \tag{2.151}$$

Note that the combiner weights for the training data set do not need to be computed. The vectors $\mathbf{W}_x$ are used as the beamformer weight vectors. The cross-SCORE algorithm can be derived using a frequency shifted (and possibly time delayed) version of the observed data for the training set. Thus for cross-SCORE

$$\mathbf{z}(n) = \mathbf{x}(n-\tau)e^{j2\pi\alpha n}. \tag{2.152}$$

Substituting this into (2.150) yields the matrix used in the cross-SCORE eigenequation.

The PCCA framework can also be used, for example, to develop SCORE methods which exploit multiple cycle frequencies. In this case the training set $\mathbf{z}(n)$ has a larger dimension than the observed data set. The training set can be obtained by a number of other transformations, such as filtering, as well. This makes the PCCA useful as a general framework for developing blind adaptive algorithms.

2.5.6 Recursive PCCA

The PCCA framework as described in Subsection 2.5.5 is not able to incorporate non-linear operators, such as the hard-limit employed in CMA or the hard-decision employed in decision-directed algorithms. An modification of the PCCA to include recursion, feedback, and training set constraints for improved blind adaptive spatial filtering is described in [59]. This is referred to as the Recursive PCCA (RPCCA). The RPCCA family of adaptive spatial filters

Ranking	Algorithm
1.	SCORE with a doubled-carrier feature (only present with AM and BPSK)
1. (tie)	Time- and Frequency-Gated Algorithms
2.	Decision-Directed Algorithms (DDA)
3.	Constant Modulus Algorithms (CMA)
4.	SCORE with a baud feature

Table 2.2: Relative performance of blind adaptive algorithms, according to output SINR that can be achieved with a finite amount of data and convergence speed.

is shown in [59] to include previously established beamforming techniques, including the CMA, as well as several new processors. A key difference between PCCA and RPCCA is that RPCCA is iterative, whereas PCCA is not. Each RPCCA iteration includes a property mapping step (where multiple signal properties can be exploited), and a PCCA step, to maximize the correlation between a linear combination of the training set and the beamformer output. The RPCCA property mapping could in principal include any or all of the following:

1. hard-limiting, to exploit CM properties;
2. frequency-shifting, to exploit cyclostationarity;
3. hard-decisions; to exploit finite-alphabet properties.

One area for future research identified in Chapter 6 is the use of the RPCCA framework to simultaneously exploit CM properties and finite-alphabet properties.

2.5.7 Comments on Relative Performance of Blind Adaptive Algorithms

To this point, we have only described the theory of operation of several blind algorithms, without regard to their relative performance. We will now examine relative performance, with particular emphasis on the convergence rate. It is enlightening to consider how well the cost function of a particular blind algorithm approximates the output SINR, especially for a small number of samples. For example, when a decision directed algorithm is used, an accurate estimate of the SINR can be obtained as the difference between the hard decisions and the soft decisions. This is also true for CMA – the SINR can be accurately determined from a relatively small number of samples. Conversely, the correlation of a signal with a frequency-shifted version of itself may not yield a reliable estimate of SINR when a baud

feature is exploited. This is due to the fact that the cyclic correlation may exhibit a relatively high degree of variance even for high SINR. The exception here is that the cyclic correlation *is* a reliable measure of SINR if the conjugate cyclic correlation of an DSB-AM or BPSK signal is exploited.

An AM signal is perfectly correlated with a frequency-shifted and conjugated version of itself. This feature is very strong, and the conjugate cross-SCORE algorithm converges nearly as quickly as a known training signal method. The Time- and Frequency-Gated algorithms also exploit a very strong feature, and again converge nearly as quickly as a training signal method. Decision-directed algorithms may take several iterations to converge, but if no decision errors are made, the output SINR upon convergence will clearly be the same as if a known training signal method is used. The main drawback of a decision-directed algorithm is that it cannot be guaranteed to converge. The constant modulus property is weaker than the finite alphabet property exploited by a decision-directed algorithm, and hence CMA converges more slowly than DDA. Finally, the cyclic correlation exhibited at the baud rate by some digital modulations is dependent on the excess bandwidth. Pulse shaping tends to limit this excess bandwidth. This is generally a weak feature (the cross-correlation between the signal and the frequency shifted signal is small), and so SCORE converges very slowly. Table 2.2 summarizes the relative performance of the blind adaptive algorithms discussed here.

2.6 Adaptive Array Hardware Requirements

Implementation of adaptive beamforming places certain unique requirements on receiver design. Similar requirements must be met in order to implement equal gain combining and maximum ratio combining. The objective is to convert the array data to a complex baseband representation, while preserving the narrowband model. The data observed at each antenna may be frequency shifted, filtered, and decimated without violating the narrowband model so long as each signal is processed in an *identical* manner. A simple gain and phase offset for each channel is all that is allowed. In practice, the narrowband model will hold only approximately. This section identifies some receiver-induced deviations from the narrowband model, the effects of these deviations on beamforming performance, and design considerations for combating these effects. The following issues are addressed:

1. coherent mixing and ADC synchronization;
2. receiver channel matching;
3. receiver dynamic range;

2.6.1 Coherent Mixing and ADC Synchronization

Figure 2.25 shows a simplified block diagram of a phase coherent receiver. The main difference between a coherent receiver and a conventional receiver is that the mixer LO's and Analog to Digital Converter (ADC) clocks are derived from the same sources. This preserves the correlation (i.e., the coherence) of each version of the signal. A related issue is the need for synchronizing the ADC clocks. The signal in each receiver channel must be sampled nearly simultaneously.

Note that the phase of each copy of the LO into the mixer will have different phase. This will introduce a different phase offset into each signal being mixed. However, this phase offset affects all the co-channel signals in the same way, so that the narrowband model is preserved.

This simplified receiver uses a single mixer stage to downconvert the received signals to a low IF. Multiple downconversion stages are often preferred in practice in order to control intermodulation terms and image frequencies.

There are essentially two alternative approaches to converting an RF signal to complex baseband. One approach is quadrature mix the signal to baseband, using a 90° phase splitter to obtain the in-phase and quadrature components at baseband. Then the I and Q legs are individually digitized. This approach has several drawbacks. The most important drawback is that the phase splitter will not provide exactly 90° of phase shift. This is often referred to as I-Q imbalance. Any I-Q imbalance will cause deviation from the narrowband model, and limit the null depth that can be achieved with an adaptive beamformer. Another drawback is that two ADCs are required.

For these reasons, the preferred approach is to mix the RF signal to a relatively low IF, digitize, and then convert to complex baseband digitally. This has the advantage of requiring only one ADC per channel. Furthermore, the digital conversion to complex baseband can be implemented with very high accuracy. The drawback of this approach is that a high speed digital filter is required when implementing the digital downconversion. If the ADC sample rate is very high, the digital filter can be difficult to implement. However, special hardware often makes this practical, especially if high decimation rate filters are used.

2.6.2 Receiver Channel Matching

One of the main sources, perhaps *the* main source, of deviation from the narrowband model is caused by *receiver mismatch*. This means that the filters used in each receiver channel are not identical. Typically, each filter in the receiver has a certain magnitude and/or phase

ripple, depending on the type of filter. Typical amplitude ripple may be on the order of 0.1-1.0 dB, and phase ripple may be on the order of a few to tens of degrees. The ripple amplitude will typically be similar from channel to channel, but the ripples will not occur at identical frequencies. This causes the frequency response of each channel to be slightly different. This in turn means that each signal can no longer be modelled as a simple phase shift and gain offset relative to the reference antenna. This is manifested in a limit to the null depth that can be achieved with the system. Channel matching can be a serious challenge if very deep nulls are required, since matching of analog filters is not trivial.

Mismatch between analog filters in the receiver can often be corrected by FIR filtering. The filter coefficients are determined by first injecting a boresight signal into all receiver channels to measure the difference in channel transfer functions, then solving for the coefficients that will make the transfer function through each channel the same. Note that this process can not correct any non-linear distortion.

When oversampling, the filter bandwidth is typically much larger than the signal bandwidth. With many filters, the number of ripples remains constant as a function of bandwidth, so increasing the filter bandwidth means that the signal is affected by fewer ripples in the response. This in turn means that the channels are better matched. This is one of the advantages of oversampling.

2.6.3 Receiver Testing

We now describe a simple test that can be performed in the laboratory to determine the accuracy of the narrowband model given the non-ideal nature of the receiver. The test is to generate a signal at a carrier frequency of interest, split the signal, and inject the signal into two or more channels of the receiver. The signals are then digitized and logged. The logged data is then processed to determine the ability to cancel the signal collected in one channel from the signal collected in another channel. Ideally, if the receiver channels are perfectly matched, and if the receiver has infinite dynamic range, and if no thermal noise is present, the signal in one channel may be perfectly cancelled the signal in any other channel. In practice, the channels will not be identical, and a residual signal will remain after the cancellation process. As the channels become better matched, the signals from each channel will become more similar, and the power of the residual signal will decrease. Thus the power of the residual signal after cancellation relative to the input power can be used as a figure of merit. The process of calculating this figure of merit is illustrated in Figure 2.26. We seek to minimize the power of the residual signal, given by

$$\sigma_e^2 = \langle\, e(n)e^*(n) \,\rangle \tag{2.153}$$

$$= \left\langle \left| x_1(n) - g^* x_2(n) \right|^2 \right\rangle \tag{2.154}$$

This is a standard Least Squares (LS) problem, with the well-known solution

$$g = R_{x_2 x_2}^{-1} R_{x_1 x_2} \tag{2.155}$$

$$= \frac{\langle\, x_1(n) x_2^*(n) \,\rangle}{\langle\, x_2(n) x_2^*(n) \,\rangle} \tag{2.156}$$

The ratio of input power to the power of the residual cancellation signal is an approximate measure of the maximum null depth that can be achieved with the receiver. Care should be taken when using this procedure that the full dynamic range of the receiver is used.

We would expect to get better results as the signal bandwidth decreases. The best possible performance will be achieved with a tone input signal, since this has (approximately) zero bandwidth, and the channel mismatch will be eliminated.

The above describes a method to test any two channels of the receiver. The overall nulling capabilities of the receiver are best tested by considering all the channels simultaneously. This can be accomplished by calculating the data covariance matrix, then calculating the ratio of the dominant eigenvalue to the smallest eigenvalue. The dominant eigenvalue is the maximum output power that can be achieved with a beamformer, subject to the constraint that the inner product of the weight vector be equal to unity. The smallest eigenvalue is the minimum output power that can be achieved with a beamformer. Thus this ratio gives a good approximation to the null depth that can be achieved with the receiver.

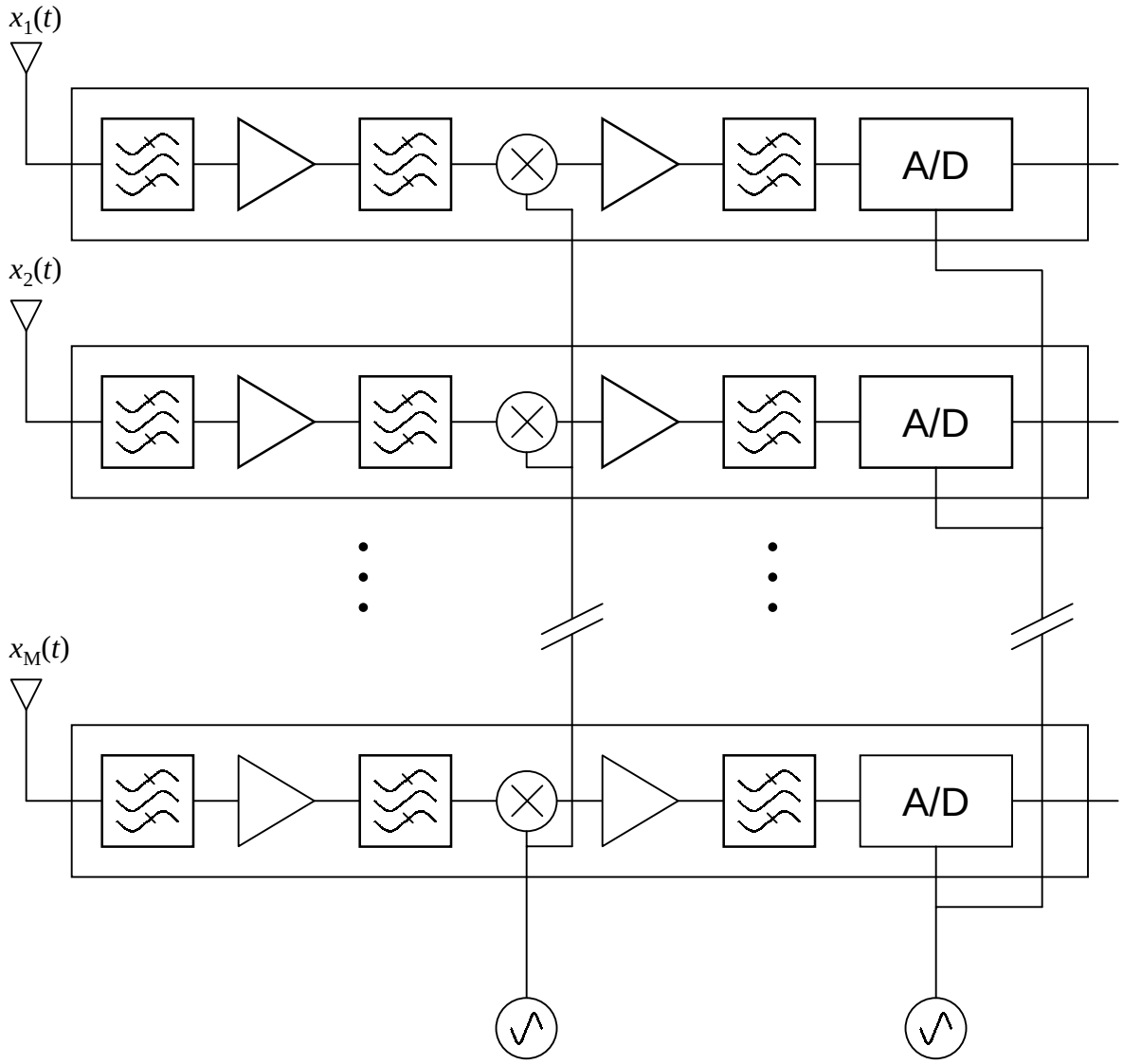

Figure 2.25: Simplified diagram of a phase coherent receiver for use in a digital adaptive antenna array or diversity combining system.

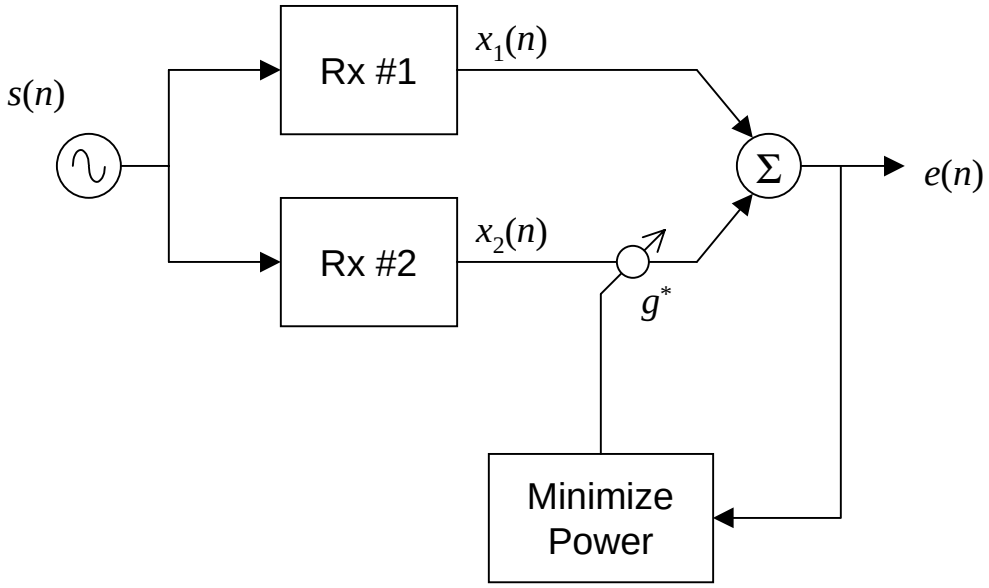

Figure 2.26: A method for measuring channel matching in a coherent receiver.

2.7 Network System Issues

Several system level issues arise when adaptive arrays are applied to wireless networks. These issues vary depending on the multiple access scheme, and on the duplexing scheme.

2.7.1 Downlink Beamforming

In order to take full advantage of the benefits of adaptive beamforming, the uplink and downlink must be of comparable quality. The ability to separate multiple co-channel users on the uplink is not advantageous unless the same feat can be performed on the downlink. Unfortunately, separation of users on the downlink is much more difficult to achieve. The main reason for this is that it is very unlikely that more than two antennas can be used at the mobile handset, thus limiting the ability to reject co-channel interference. As an alternative to adaptive combining at the handset, it is theoretically possible to have transmit nulls in the downlink antenna pattern. However, this is much more difficult to achieve than in the uplink antenna pattern. This is even more difficult in an urban or suburban mobile radio environment, where the spatial signature can change significantly if the user moves a fraction of a carrier wavelength.

If the array response on the forward (base to mobile) and reverse (mobile to base) links are identical, the optimal transmission weight vector is in principal equal to the conjugate of the optimal receive weight vector. However, most cellular systems use Frequency Division Duplexing (FDD) for the forward and reverse links. This frequency separation means that the spatial signatures on the reverse and forward links can be very different, even if the multipath AOA and delay spread are identical. Just as small changes in user position can correspond to large changes in the spatial signature, small changes in carrier frequency can correspond to large changes in the spatial signature.

Downlink beamforming is less of an issue for CDMA systems. In these systems the downlink signals are synchronous, and each signal arrives at the user with the same power and the same delay spread. Therefore, if the signals are orthogonal at the transmitter, they will be orthogonal at the receiver. The only distortion will be due to excess delay spread.

One alternative is to design the system so that the uplink is more heavily loaded than the downlink. For example, if multiple co-channel signals can be reliably separated at the base, but not at the mobile, it may make sense to consider a multiple access scheme where the uplink and downlink bandwidth are asymmetric. Another alternative is to use a Time Division Duplex (TDD) scheme, so that the same carrier frequency is used for the forward and reverse links. This allows the reciprocity of the uplink and downlink channels to be exploited.

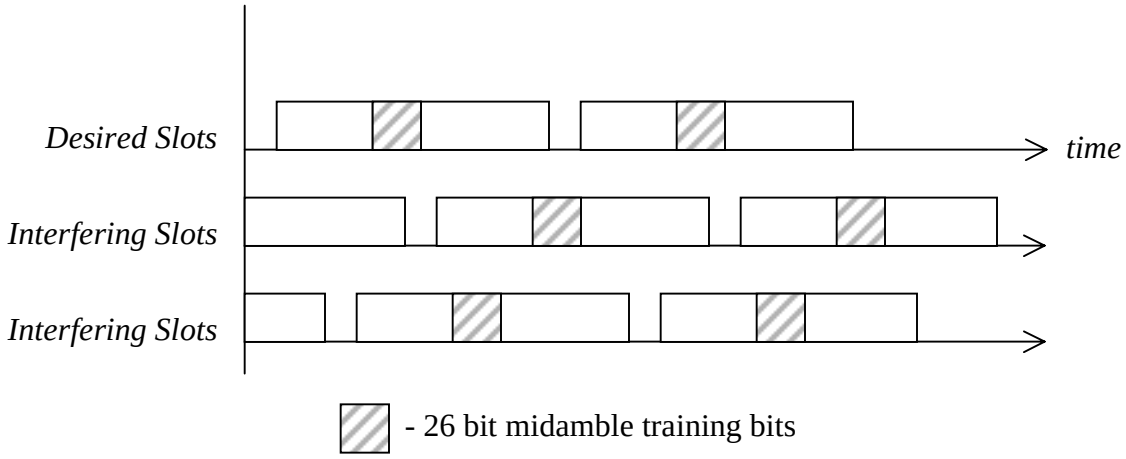

Figure 2.27: Illustration of the effect of asynchronous TDMA interference. The location of the midamble training sequence in GSM is shown as an example.

The optimal receive antenna weights are also the optimal transmit antenna weights in a TDD system.

2.7.2 Adaptive Arrays in TDMA Applications

One of the main issues when attempting to apply adaptive antenna arrays to TDMA signals is the typically asynchronous nature of the interference. This has two important effects. First, if a weight vector is calculated using a training sequence, the data used for training will most likely not be valid over the entire duration of the desired user's slot. A second effect is that the interference is highly non-stationary. Each desired slot may be subject to several different interference environments over its duration.

Consider the illustration in Figure 2.27, showing a desired TDMA frame and two interfering TDMA frames. The interfering frames will originate from other cells, and the frame timing of each basestation is not synchronized. Thus the start time of the interfering slots is essentially a random variable from the perspective of the receiver. Each interfering slot will have a unique location, and hence a unique spatial signature. Every transition of an interfering slot will cause the interference environment to change. The training sequence is only present during a particular time instant, and a weight vector calculated using this training sequence will not be valid over the entire slot.

This is a difficult problem to solve in practice, and limits the frequency-reuse that can be used in a TDMA system where adaptive beamforming is employed.

2.7.3 Adaptive Arrays in CDMA Applications

The use of adaptive arrays in CDMA might be more practical than in TDMA for the following reasons:

1. Unlike TDMA, the interference is not rapidly changing.
2. The capacity of a CDMA system is typically limited on the reverse link due to the asynchronous nature of the mobile subscribers. Because the users are asynchronous, the orthogonality of the spreading sequences cannot be maintained. On the forward link, the basestation can easily control the transmitted signals to ensure that they are orthogonal. Therefore, the use of an adaptive array at the basestation can increase capacity, even if downlink beamforming is not implemented.
3. A CDMA modulation is designed to operate in co-channel interference, and the capacity can increase with a small reduction in co-channel interference levels. In contrast, a TDMA waveform, which is typically not spread spectrum, must have relatively high SNR in order to have low BER, and so does not benefit as much from small improvements in SINR. In other words, a TDMA beamformer must completely eliminate co-channel interference for low BER. This is more difficult than simply trying to reduce the co-channel interference which is always present in a CDMA system.

One aspect of CDMA beamforming that is unique to this application is the option of beamforming before or after the despreading. To implement a post-despreading beamformer, the coherence of the signals from each antenna is maintained by multiplying each received signal by the same de-spreading code. This preserves the spatial characteristics (spatial signatures) of all received signals. Beamforming after the despreading would be preferred since this corresponds to a much lower data rate, and hence a much lower computational load. However, beamforming before despreading offers some performance benefits. The main benefit is that the changing interference environment can be better tracked.

Glossary

This Glossary contains some of the terminology and abbreviations commonly used in antenna array processing.

AOA - Angle of Arrival. Usually denoted by θ.

Angle Spread - In a multipath environment, the signal arrives from multiple AOAs. The angular spread describes the angular extent from which the signal arrives.

Aperture - The largest dimension of the antenna array, usually expressed in wavelengths.

Array Calibration - The process of measuring the steering vectors for all AOA's of interest. This usually involves placing a signal at a known AOA and then measuring the array response vector. The process is repeated for multiple AOAs and/or frequencies. This is a costly and in some cases impractical process.

Array Response Vector - The $M \times 1$ complex vector, usually denoted by $\mathbf{a}(\theta)$, that contains the gain and phase difference observed at each antenna (relative to a reference antenna) for a signal incident from AOA θ.

Beampattern - The gain (usually in dB) versus AOA for a particular array of antenna elements, antenna weights, and carrier frequency.

Blind Adaptive Algorithm - A class of adaptive algorithms that do not require a training signal or knowledge of the desired signal's AOA, but instead exploit some other property (e.g., constant modulus) of the desired signal.

Channel Mismatch - The *difference* in frequency response between each channel of the receiver used in the beamformer. Channel mismatch causes deviation from the narrowband model and hence limits the null depth that may be achieved.

Cochannel Interference - An interferer that shares the same frequency band as the desired signal, e.g., another signal with the same carrier frequency.

Copy - The process of obtaining an estimate (i.e., copy) of an incident signal.

Copy/DF - A DF method which uses the beamformer weight vector (presumably obtained with a blind adaptive algorithm) to estimate the AOA.

CMA - Constant Modulus Algorithm. A blind adaptive algorithm that assumes that minimizing the variance of the output signal's envelope, i.e., minimizing $J_{pq} = \mathrm{E}\{(|y(n)|^p - 1)^q\}$, is equivalent to maximizing the output signal SINR. Applicable to FM signals and certain digital signals if they are sampled appropriately.

Coherent Receiver - A multi-channel receiver whose mixers are all derived from the same oscillators, so that phase coherence is maintained across all channels.

Cyclostationary - A signal is said to exhibit second order cyclostationarity if it is correlated with a frequency shifted, time-delayed (and possibly conjugated) version of itself. Almost all communication signals exhibit this property.

Decision Directed Algorithm (DDA) - A blind adaptive algorithm that replaces the known symbols typically used in an adaptive algorithm with symbols that are estimated from the received data.

DF - Direction Finding. The process of determining the AOA of a signal.

Linear Array - An antenna array where all sensors lie on the same axis.

Maximum Attainable SINR - An alternative expression for the optimal output SINR.

MVDR - Minimum Variance Distortionless Response. The weight vector obtained by solving the following constrained optimization problem: Minimize the output power of the beamformer subject to the constraint that the gain at the AOA of the desired signal is unity.

Narrowband Model - A critical modelling assumption made in many if not most beamforming applications. Basically this assumes that the only difference between the signal observed at one sensor and the signal observed at any other sensor is a simple gain and phase difference. Usually requires that the sensors be relatively close together and that the incident signals be narrowband compared to the carrier frequency.

Overloaded Array - An environment where the number of incident signals L is greater than the number of antennas M, thus exceeding the number of nulls that the array can steer.

RAKE - A type of matched filter used in multipath environments, typically in CDMA applications. Multiple copies of the received signal are delayed (ideally matching the channel multipath profile), weighted, and summed.

Signal Subspace - In an environment with L incident signals, the space spanned by the L dominant eigenvectors of the observed data covariance matrix. This is the same space spanned by the steering vectors of the L incident signals.

SCORE - Self Coherence Restoral. A blind adaptive algorithm that is applicable to any signal that exhibits second order cyclostationarity.

SINR - Signal to Interference and Noise Ratio.

SNOI - Signal Not Of Interest, i.e., a co-channel interfering signal.

SOI - Signal Of Interest, i.e., the desired signal.

Spatial Filtering - An alternative, and perhaps more accurate, expression for beamforming.

Spatial Signature - This $M \times 1$ complex vector, usually denoted by $\mathbf{a}$, contains the gain and phase difference observed at each antenna (relative to a reference antenna) when a signal is received from multiple directions (multipath). The spatial signature is a linear combination of array response vectors.

Spurious Free Dynamic Range (SFDR) - A means of measuring the linearity and sensitivity of a receiver. This is the difference in power between full scale and the strongest intermodulation terms or other spurs that appear in the receiver output.

Steering Vector - An alternative name for an array response vector.

SWNR - Signal to White Noise Ratio. Usually used to describe the power of the incident signals relative to the (assumed spatially white) background noise.

Uniform Linear Array (ULA) - An array where the sensors are distributed along a straight line with equal antenna separation.

Weight Vector - The $M \times 1$ column vector of antenna weights, where M is the number of antennas in the array.

Chapter 3

Analysis of the Constant Modulus Algorithm

Summary

The convergence behavior of the Least Squares Constant Modulus Algorithm in an adaptive beamforming application is examined. It is assumed that the desired signal and the interference are uncorrelated. The improvement in output SIR with each iteration of the algorithm is predicted for several different signal environments. Deterministic results are presented for an environment containing two complex sinusoids. Probabilistic results are presented for a constant modulus desired signal with a constant modulus interferer and with a Gaussian interferer. The asymptotic improvement in output SIR as the output SIR becomes high is also derived. The results of Monte Carlo simulations using sinusoidal, FM, and QPSK signals are included to support the derivations.

3.1 Introduction

We will study a member of the class of adaptive algorithms generally known as Godard or Constant Modulus Algorithms (CMA) [60, 41]. These algorithms can be used for adaptive beamforming, equalization, and other applications when the desired signal has a constant envelope. Examples of such signals include FM, PSK, and FSK. The CMA can also be applied to many non-CM signals (e.g., pulse-shaped PSK, QAM) although the performance may be degraded relative to the case where the desired signal is CM [40, 39]. The main advantage of CMA is that it is a 'blind' adaptive algorithm, i.e., it does not require a training signal. Other

blind adaptive algorithms have been designed to exploit cyclostationarity [54, 61, 58], known signal constellation [9], known spreading code in CDMA [62], and time or frequency gated properties [50, 51, 52]. The first CMA to be proposed was based on a Stochastic Gradient Descent (SGD) form [41]. The main drawback of this method is its slow convergence. A faster converging CMA similar in form to the Recursive Least Squares method is the Orthogonalized CMA [42]. Another fast converging CMA is the Least Squares CMA (LSCMA) [43, 44], which is a block-update iterative algorithm. It is guaranteed to be stable and is easily implemented.

Despite the generally accepted use of the LSCMA, few analytical results on its convergence have appeared in the open literature. (Portions of this chapter have previously appeared in [7] and [63]). The performance of the algorithm has instead been demonstrated through Monte Carlo simulation. The lack of analytical results is due to the difficulty of analyzing the non-linear CMA cost function. Existing work on the convergence behavior of CMA mostly deals with finding minima of the CMA cost function and finding undesirable stable equilibria in equalization applications (e.g., see [64, 65] and references therein).

A notable exception is the work by Treichler and Larimore on convergence of SGD CMA in an environment containing two complex sinusoids [66]. Their work predicts the output power of each sinusoid in a temporal filtering application. The Analytic CMA (ACMA) algorithm presented by van der Veen in [48] should also be noted. This algorithm solves directly for a set of beamformer weight vectors that spatially separate a set of CM signals. The ACMA, although effective in many situations, is fairly complex and its behavior with closely spaced and/or low SNR signals is not clear. For these reasons it is recommended in [48] that the ACMA be used to initialize the LSCMA, and that several iterations of the LSCMA be used to find the optimal solutions for the weight vectors.

In this chapter we determine the convergence rate of the LSCMA in some simple environments, including: (1) high output SIR; (2) sinusoidal desired signal and sinusoidal interferer; (3) CM desired signal and CM interferer; (4) CM desired signal and Gaussian interferer. We assume that the interference is uncorrelated with the desired signal. The convergence rate is expressed in terms of the SIR improvement achieved with one iteration of the LSCMA. We first examine the situation where the LSCMA output SIR is high. We show that if the interference is perfectly removable, each LSCMA iteration will increase the output SIR by approximately 6 dB. This result is valid for any CM desired signal (arbitrary angle modulation), and any uncorrelated interference. We next examine an environment containing two complex sinusoids, and show that the LSCMA output SIR can be predicted for each iteration. The results are analogous to those presented in [66]. An environment containing two CM signals, each having random phase, is then considered. It is shown that the average behavior of the LSCMA in this environment is similar to the deterministic behavior in the two-sinusoid environment. Finally, an environment containing a CM desired signal and

Gaussian interference is examined.

3.2 Overview of LSCMA

The objective of the adaptive beamformer is to obtain a high-quality estimate of a desired signal in the presence of cochannel interference using an array of antennas. Letting the $M \times 1$ vector $\mathbf{x}(n)$ represent the signals and noise received at an array of M antennas at discrete time index n gives

$$\mathbf{x}(n) = \sum_{i=1}^{L} \mathbf{a}_i s_i(n) + \mathbf{q}(n) \tag{3.1}$$

where the $M \times 1$ vector $\mathbf{a}_i$ is the spatial signature corresponding to signal $s_i(n)$ and the $M \times 1$ vector $\mathbf{q}(n)$ contains environmental and receiver noise. The model defined by (3.1) is often referred to as the narrowband model, since it is assumed that the signals are narrowband relative to the carrier frequency. The signals are also assumed to have unit variance and to be temporally uncorrelated with each other and with the background noise. That is,

$$\mathcal{E}\Big\{ s_i(n)\, s_j^*(n) \Big\} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{3.2}$$

where $\mathcal{E}\{\cdot\}$ denotes expectation. The signal power is incorporated in the spatial signature, which describes the amplitude and phase difference between the signal received at a reference antenna and all other antennas. In the absence of multipath, the spatial signature is generally referred to as an array response vector. The array response vector is dependent on the angle of arrival (AOA) of the signal, the array geometry, the gain pattern of each antenna, the carrier frequency of the incident signals, etc. When a signal is incident from more than one direction the spatial signature will be a linear combination of the array response vectors corresponding to the AOA of each path. This assumes that the multipath delay spread τ is small relative to $1/B$, where B is the signal bandwidth. If $\tau > 1/B$, each multipath component received by the array is uncorrelated with other arriving components, and each component is treated as a different signal. The case in which the delay spread is significant, but not so large that the multipath arrivals are uncorrelated, is not treated in this chapter.

An adaptive beamformer weights and sums the signals received by the array to form an estimate $y(n) = \mathbf{w}^H \mathbf{x}(n)$ of the desired signal, where $\mathbf{w}$ is the $M \times 1$ complex vector of beamformer weights and $\mathbf{w}^H$ is, as usual, the Hermitian transpose. This process is illustrated conceptually in Figure 1.1. The desired behavior of adaptive beamforming is most easily visualized for environments that lack multipath. In these environments an adaptive beamformer seeks to steer a beam towards the AOA of the desired signal while simultaneously steering nulls towards the AOA's of the interfering signals.

Computation of the optimal (maximum output SNR, minimum MSE, etc.) weight vector requires either a training signal or precise knowledge of the desired signal spatial signature. In practice, a training signal is not always available. Furthermore the spatial signature may be impractical to obtain. In such situations the CMA may be used to obtain a nearly optimal weight vector if the desired signal has constant modulus.

The constant modulus property can in general be exploited by minimizing the non-linear cost function

$$F_{(p,q)} = \langle\, (|y(n)|^p - 1)^q \,\rangle \tag{3.3}$$

where $\langle\, \cdot \,\rangle$ denotes time average. The form of (3.3) generally makes analysis of CMA difficult. The LSCMA is a block update iterative technique for minimizing the $F_{(1,2)}$ cost function given by

$$F_{(1,2)} = \left\langle\, (|y(n)| - 1)^2 \,\right\rangle \tag{3.4}$$

The LSCMA is implemented as follows. At the kth iteration, N temporal samples of the beamformer output are generated using the current weight vector $\mathbf{w}_k$. This gives

$$y_k(n) = \mathbf{w}_k^H \mathbf{x}(n) \tag{3.5}$$

The initial weight vector $\mathbf{w}_0$ can be taken as $\mathbf{w}_0 = [1\ 0\ 0\ \cdots 0\,]^T$ if no a-priori information is available. Other initialization methods are considered in [67], where it is shown that the dominant eigenvectors of the observed data covariance matrix are good choices for initial weight vectors. The kth signal estimate is then hard limited to yield

$$d_k(n) = \frac{y_k(n)}{|y_k(n)|} \tag{3.6}$$

and a new weight vector is formed according to

$$\mathbf{w}_{k+1} = \mathbf{R}_{xx}^{-1} \mathbf{r}_{xd} \tag{3.7}$$

where

$$\mathbf{R}_{xx} = \left\langle\, \mathbf{x}(n)\, \mathbf{x}^H(n) \,\right\rangle_N \tag{3.8}$$

and

$$\mathbf{r}_{xd} = \langle\, \mathbf{x}(n)\, d_k^*(n) \,\rangle_N \tag{3.9}$$

In the above expressions, $\langle\, \cdot \,\rangle_N$ denotes a time average over $0 \le n \le N-1$. The updated weight vector $\mathbf{w}_{k+1}$ minimizes the mean square error

$$\left\langle\, \left| d_k(n) - \mathbf{w}_{k+1}^H \mathbf{x}(n) \right|^2 \,\right\rangle_N \tag{3.10}$$

The iteration described by (3.5),(3.6), and (3.7) is continued until either the change in the weight vector is smaller than some threshold, or until the envelope variance of the output signal is deemed sufficiently small. In a stationary environment, the LSCMA iteration can be performed using a new block of data, or can be re-applied to the same block of observed data. The former approach is referred to as *dynamic* LSCMA in [43], while the latter approach is referred to as *static* LSCMA.

The only parameter which must be selected when using the LSCMA is the block size N. Selection of N will depend on many factors, including the acceptable latency of the update procedure, the rate at which the signal environment is changing, and the available processing power. Also, the number of temporal samples of array data must be equal to or greater than the number of antennas so that the linear system of equations is overdetermined. The LSCMA block size is similar to the SGD adaptation step parameter in that smaller block sizes yield faster convergence at the cost of higher variance in the output SNR. By faster convergence we mean that fewer temporal samples of data need be processed to achieve steady-state. An interesting feature of the LSCMA is that the output SIR as a function of the number of iterations is nearly independent of data block size. That is, the LSCMA typically converges in 5 to 10 iterations regardless of the block size. The effect of finite block size on the behavior of the LSCMA is addressed in more detail in Section 3.6.

An aside is in order here on the computational complexity of the LSCMA. Direct minimization of MSE by solving a set of simultaneous linear equations, as in (3.7), is often considered impractical for real-time applications because of the computational load. Certainly this may be true for adaptive equalization applications, where the number of filter coefficients can be large. However, the computational load is more reasonable for beamforming applications since adaptation of a relatively small number of coefficients is required.

The LSCMA bears a strong resemblance to the classical least squares method that can be used when a known training signal is available. The LSCMA can be interpreted as a least squares method that uses a pseudo-training signal that is derived from the observed data. This is the viewpoint taken here. We essentially determine the quality of the pseudo-training signal in some representative situations. A different motivation for the LSCMA is presented next, along with a discussion of the relationship of the LSCMA to other existing algorithms.

The LSCMA can be viewed as a method where a signal estimate is alternately projected onto the set of CM signals and the space spanned by the observed data. In this way the LSCMA resembles the Gerchberg-Saxton algorithm (GSA) [68], as noted by Wang, et al. [69] and Van der Veen [48]. The problem solved by GSA is to recover the magnitude and phase of a signal when only the magnitude of the signal and the magnitude of its Fourier transform are known. This problem arises in many applications, including speech and image processing

(see, e.g., [70]). The GSA projects a signal estimate in an alternating fashion onto time domain and frequency domain property sets. The principal similarity of GSA and LSCMA is that both employ a projection onto a non-convex set. This is in contrast to Projections Onto Convex Sets (POCS) [70, 71, 72, 73, 74]. POCS can be used to find a signal estimate which satisfies multiple properties. The POCS takes an initial signal estimate and projects it in an alternating fashion onto the various property sets being exploited. The POCS is guaranteed to converge when the sets are closed and convex. The GSA and the LSCMA can be viewed as belonging to a class of algorithms that are sometimes known as the Method of Generalized Projections (MGP). The GSA is the archetype of MGP, and some authors use 'Gerchberg-Saxton' as a generic term to describe any MGP. MGP convergence cannot be assured in general, although in some cases MGP exhibits the 'error reduction' property. This implies that each iteration of a MGP reduces, or at worst does not increase, the cost function being minimized. The error reduction property of the LSCMA is described in the original LSCMA paper [43], but the relationship of the LSCMA to the GSA and other MGP approaches was not recognized.

3.3 Analysis Framework

In this section we describe the general framework used to analyze the LSCMA. A key assumption is that the interference is uncorrelated with the desired signal. We essentially describe a simple way to measure the quality of the pseudo-training signal, $d(n)$. If no background noise is present, the quality of the beamformer output $y(n)$ will be identical to the quality of $d(n)$. When background noise is present, the quality of $y(n)$ is dependent on the quality of $d(n)$ and the optimal output SINR. The optimal output SINR is in turn dependent on many factors, including the array geometry, the number of antennas, the number of incident signals, and the angle of arrival of each signal.

The beamformer output signal at the kth iteration can be expressed, to within a multiplicative constant, as

$$y_k(n) = \mathbf{w}_k^H \mathbf{x}(n) = s(n) + gz(n) \tag{3.11}$$

where $s(n)$ is the constant modulus desired signal, $z(n)$ is noise and interference, and the SINR of the beamformer output is controlled by g. Both the desired signal $s(n)$ and the interference $z(n)$ have unit variance. The hard-limiter output $d_k(n)$ will contain three components: (1) one component which is correlated with the desired signal; (2) one component which is correlated with the interference; and (3), one component which is correlated with neither the signal nor the interference. This last component is the result of intermodulation

between the signal and the interference. We can express the hard-limiter output as

$$d_k(n) = \frac{y_k(n)}{|y_k(n)|} = \alpha s(n) + \beta z(n) + \xi(n) \tag{3.12}$$

where the scalars α and β control the desired signal power and the interference power, respectively, and $\xi(n)$ contains the intermodulation terms.

We will now examine the relationship between the SINR in $d_k(n)$ and the SINR in the updated beamformer output y_{k+1}. We initially assume that no background noise is present, and that the array has sufficient degrees of freedom to completely remove the interference. Given these assumptions, the optimal beamformer output SINR is infinite. These assumptions are clearly not realistic, but this helps provide insight into the behavior of the LSCMA.

As the block size $N \to \infty$, the updated weight vector $\mathbf{w}_{k+1}$ minimizes the MSE between $y_{k+1}(n)$ and $d_k(n)$,

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} |y_{k+1}(n) - d_k(n)|^2 = \mathcal{E}\Big\{ |y_{k+1}(n) - d_k(n)|^2 \Big\} \tag{3.13}$$

We can express the updated beamformer output as

$$y_{k+1}(n) = \mathbf{w}_{k+1}^H \mathbf{x}(n) = \alpha' s(n) + \beta' z(n) \tag{3.14}$$

which, together with (3.12), allows the MSE to be written as

$$\begin{aligned} \mathcal{E}\Big\{ |y_{k+1}(n) - d_k(n)|^2 \Big\} &= \mathcal{E}\Big\{ |(\alpha' - \alpha)s(n) + (\beta' - \beta)z(n) - \xi(n)|^2 \Big\} & (3.15)\\ &= |\alpha' - \alpha|^2 \sigma_s^2 + |\beta' - \beta|^2 \sigma_z^2 & (3.16) \end{aligned}$$

where we have made use of the fact that $s(n)$, $z(n)$, and $\xi(n)$ are mutually uncorrelated. Clearly the MSE is minimized for $\alpha' = \alpha$, $\beta' = \beta$. This implies that the signal component in the updated beamformer output will match the magnitude and *phase* of the signal component in the hard-limiter output. Thus the MSE between $d_k(n)$ and the updated beamformer output $y_{k+1}(n)$ is minimized when

$$y_{k+1}(n) = \mathbf{w}_{k+1}^H \mathbf{x}(n) = \alpha s(n) + \beta z(n) \tag{3.17}$$

In order to find α and β, we calculate the cross-correlation of $s(n)$ and $z(n)$, respectively, with $d(n)$. Note that

$$\begin{aligned} R_{sd} &\triangleq \mathcal{E}\{ s(n) d^*(n) \} & (3.18)\\ &= \mathcal{E}\{ s(n) (\alpha s(n) + \beta z(n) + \xi(n))^* \} & (3.19)\\ &= \alpha^* & (3.20) \end{aligned}$$

Similarly we have

$$R_{zd} \triangleq \mathcal{E}\{\, z(n)d^*(n) \,\} = \beta^* \tag{3.21}$$

The SINR ρ in the hard-limiter output d_k is

$$\rho\,(d_k) = \frac{|\alpha|^2}{|\beta|^2} = \frac{|R_{sd}|^2}{|R_{zd}|^2} \tag{3.22}$$

since we model $s(n)$ and $z(n)$ as having unit variance. Thus, the output SIR of the updated LSCMA weight vector can be determined from R_{sd} and R_{zd}. This requires that the probability density function (PDF) of the signal and the interference be known.

To further illustrate the concepts behind this analysis framework, we will apply the LSCMA to a simple environment containing two uncorrelated complex sinusoids. The array configuration consists of two antennas, with the interelement spacing equal to $\lambda/2$, where λ is the carrier wavelength. One sinusoid, with a frequency of 5/1024, is incident from broadside to the array, which we define as 0°. This sinusoid is treated as the desired signal. The second sinusoid, with a frequency of -31/1024, is incident from 30°. This sinusoid is treated as the interfering signal. The amplitude of the first sinusoid is unity, and the amplitude of the second sinusoid is 0.9. The LSCMA is applied to this environment with the initial weight vector $\mathbf{w}_0 = [\ 1\ 0\ ]$. Thus the initial SIR is approximately -0.9 dB. The LSCMA block size N is set to 1024 samples. The periodogram of the initial beamformer output $y_0(n)$ is shown in Figure 3.1. The next step in the LSCMA is to hard-limit the beamformer output. The periodogram of the hard-limited beamformer output is shown in Figure 3.2. Note that the original sinusoidal frequencies are still present, along with intermodulation products. Also note that the relative amplitude of the desired sinusoid is now slightly higher relative to the interfering sinusoid. The exact change in relative amplitude is calculated later in Subsection 3.3.2. The next step in the LSCMA is to update the weight vector using the hard-limiter output in the same manner as a training signal. Figure 3.3 shows the periodogram of the updated beamformer output. As discussed earlier, the amplitude of each sinusoid in the updated beamformer output matches the amplitude of the corresponding sinusoids in the hard-limiter output. The intermodulation products are orthogonal to the signals present in the array data, and so have no effect on the weight update. It can be seen that the SIR in the updated beamformer output is approximately 3 dB higher than the initial SIR.

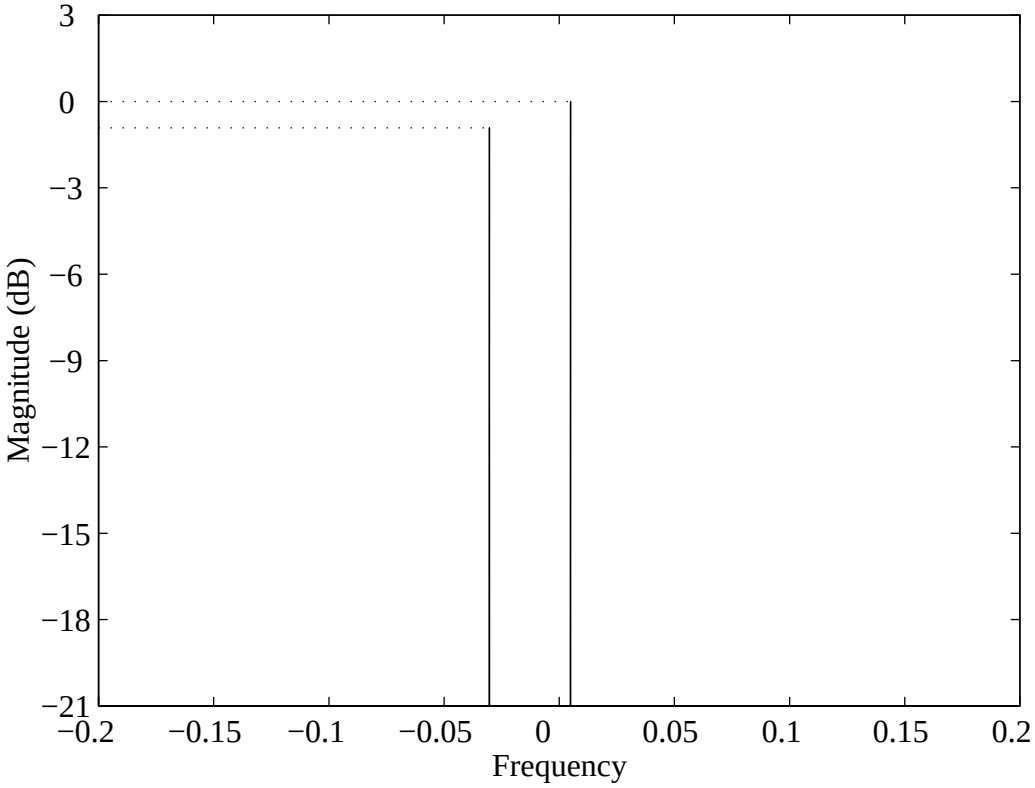

Figure 3.1: Periodogram of the initial beamformer output for the simple two-sinusoid environment. The initial SIR of 0.9 dB is indicated by the dotted horizontal lines.

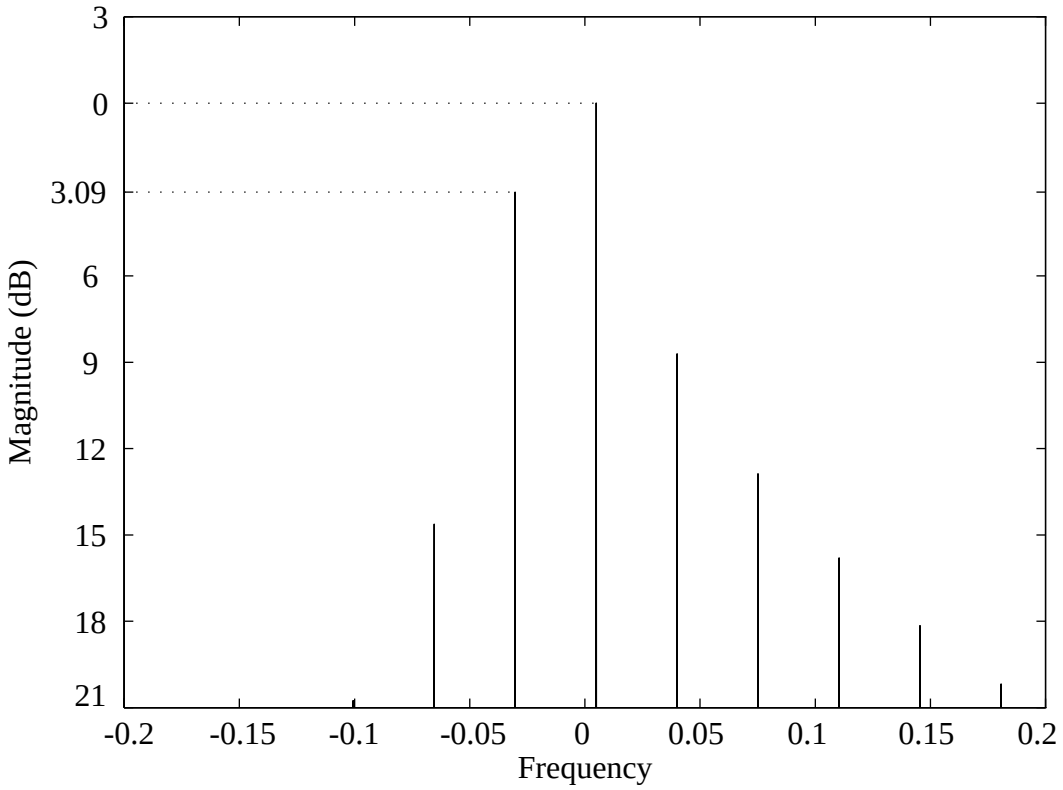

Figure 3.2: Periodogram of the hard-limited beamformer output for the simple two-sinusoid environment. The SIR of 3.09 dB is indicated by the dotted horizontal line. Note that the calculation of SIR does *not* take into account the intermodulation terms.

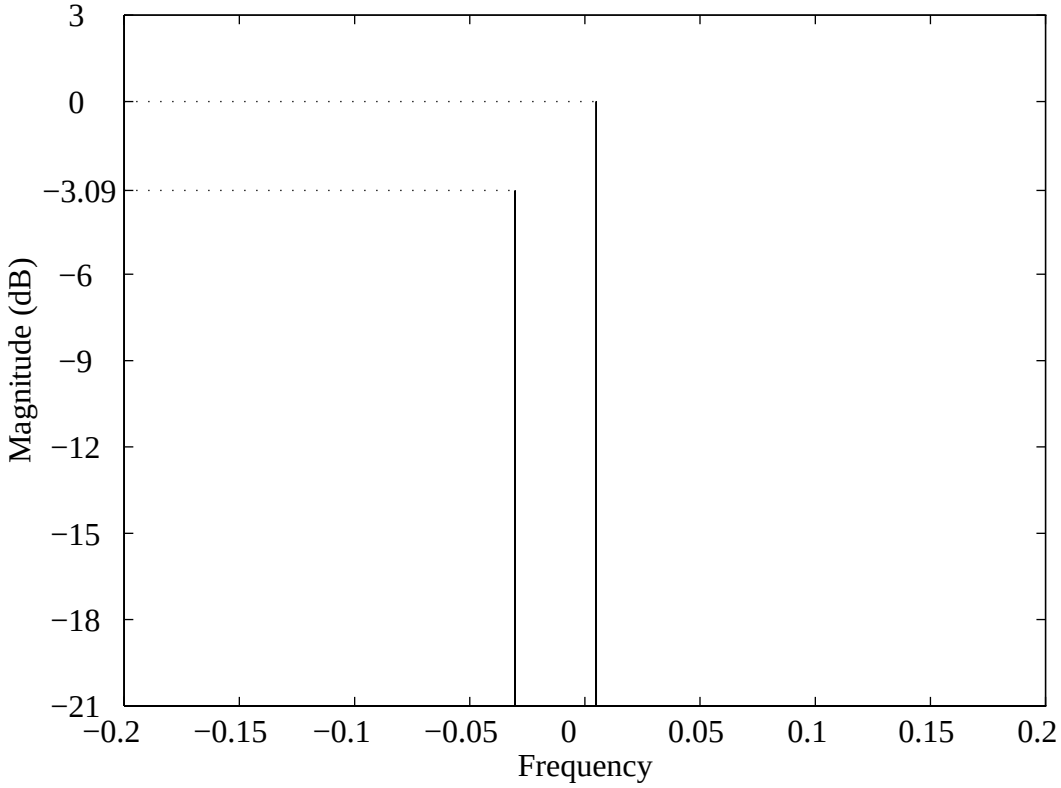

Figure 3.3: Periodogram of the updated beamformer output for the simple two-sinusoid environment. The SIR of 3.09 dB is *identical* to the SIR in the hard-limiter output.

When background noise is present the interference and noise cannot be completely removed by beamforming. Independent thermal noise generated by each of the M receivers required for the M antennas in the array is a common source of background noise. The relationship between the SINR in $d_k(n)$ and the updated LSCMA output SINR is then somewhat more complicated.

We now derive an expression for the output SINR of the updated LSCMA weight vector when background noise is present. This will be shown to be dependent only on the optimal output SINR, the initial SINR, and the SINR gain provided by the hard limit non-linearity. The observed data is modeled as

$$\mathbf{x}(n) = \mathbf{a}s(n) + \mathbf{q}(n) \tag{3.23}$$

where $\mathbf{a}$ is the spatial signature of the desired signal and $\mathbf{q}(n)$ contains the noise and interference. We assume that $\mathbf{R}_{qq}$ is equal to the identity matrix. There is no loss of generality since whitening the data has no effect on the LSCMA. The cross-correlation vector $\mathbf{R}_{xd}$ is given by

$$\mathbf{R}_{xd} = c\mathbf{a}\mathbf{a}^H\mathbf{w}_0 + \mathbf{w}_0 \tag{3.24}$$

where $\mathbf{w}_0$ is the initial weight vector, and c is the square root of the SINR gain, with

$$c \triangleq \frac{\alpha}{\beta\sqrt{\rho_k}} \tag{3.25}$$

The covariance matrix of the data is

$$\mathbf{R}_{xx} = \mathbf{a}\mathbf{a}^H + \mathbf{I} \tag{3.26}$$

By the matrix inversion lemma

$$\mathbf{R}_{xx}^{-1} = \mathbf{I} - \frac{\mathbf{a}\mathbf{a}^H}{1+\rho} \tag{3.27}$$

where ρ is the optimal output SINR (4.36). The updated weight vector $\mathbf{w}_{k+1}$ is

$$\begin{aligned}
\mathbf{w}_{k+1} &= \mathbf{R}_{xx}^{-1}\mathbf{r}_{xd} && (3.28)\\
&= \left(\mathbf{I} - \frac{\mathbf{a}\mathbf{a}^H}{1+\rho}\right)\left(c\mathbf{a}\mathbf{a}^H\mathbf{w}_0 + \mathbf{w}_0\right) && (3.29)\\
&= \left(\frac{c-1}{1+\rho}\mathbf{a}\mathbf{a}^H + \mathbf{I}\right)\mathbf{w}_0 && (3.30)
\end{aligned}$$

The output SINR of the updated weight vector is

$$\begin{aligned}
\mathrm{SINR}_{k+1} &= \frac{\left|\mathbf{a}^H\mathbf{w}_1\right|^2}{\mathbf{w}_1^H\mathbf{w}_1} && (3.31)\\
&= \frac{\left\{\rho\left(\frac{c-1}{1+\rho}\right)+1\right\}^2\left|\mathbf{a}^H\mathbf{w}_0\right|^2}{\rho\left(\frac{c-1}{1+\rho}\right)^2\left|\mathbf{a}^H\mathbf{w}_0\right|^2 + 2\left(\frac{c-1}{1+\rho}\right)\left|\mathbf{a}^H\mathbf{w}_0\right|^2 + \mathbf{w}_0^H\mathbf{w}_0} && (3.32)
\end{aligned}$$

Since the initial SINR ρ_0 is

$$\rho_0 = \frac{\left|\mathbf{a}^H \mathbf{w}_0\right|^2}{\mathbf{w}_0^H \mathbf{w}_0} \tag{3.33}$$

the output SINR of the updated LSCMA weight vector can be written

$$\mathrm{SINR}_{k+1} = \frac{\left\{\rho\left(\frac{c-1}{1+\rho}\right)+1\right\}^2 \rho_0}{\rho\rho_0\left(\frac{c-1}{1+\rho}\right)^2 + 2\rho_0\left(\frac{c-1}{1+\rho}\right)+1} \tag{3.34}$$

We argued earlier that the output SINR of the updated LSCMA beamformer is equal to the SINR in the hard limited signal $d_k(n)$ if no background noise is present. This can be verified by letting the optimal output SINR ρ approach infinity in (4.34). It is straightforward to show that

$$\lim_{\rho\to\infty} \mathrm{SINR}_{k+1} = c^2\rho_k = \frac{\alpha^2}{\beta^2} \tag{3.35}$$

which supports the argument made earlier. Also note that when $c = 1$ the hard limit operation provides no gain, and

$$\mathrm{SINR}_{k+1}\Big|_{c=1} = \rho_k \tag{3.36}$$

Here the output SINR of the updated weight vector equals the initial output SINR, as expected.

In order for LSCMA to converge, the hard-limiter must emphasize the desired signal relative to the noise and interference. The effect of hard-limiting and other non-linear operations on communication signals and noise has been a topic of study since the 1950's, e.g., see [75, 76] and references therein. A central motivation for this work is to understand the effect of non-linear amplifiers on communication signals, which are commonly used in satellite transponders. Non-linear processing has also been studied as a possible means for reducing the effects of noise and interference, e.g., [77, 78]. These studies have clearly shown that hard-limiting and filtering a constant envelope signal will increase the SNR, even when the intermodulation components are considered. In fact, for a constant envelope signal, the hard-limiter becomes the optimal nonlinearity as the SNR tends to infinity [79].

3.3.1 High SIR

We first examine the situation where the beamformer output SIR is high, as might be the case near LSCMA convergence. We model $y(n)$ as

$$y(n) = s(n) + gz(n) = e^{j\phi(n)} + gm(n)e^{j\psi(n)} \tag{3.37}$$

where $\phi(n)$ is the phase of the desired signal $s(n)$, and $m(n)$ and $\psi(n)$ are the magnitude and phase, respectively, of the unit-variance interference term, $z(n)$. The scalar g controls the SIR, and we assume $g \ll 1$. Note that we have assumed for convenience that the desired signal has unit amplitude in the beamformer output. This has no effect on the behavior of the LSCMA, since any scaling of $y(n)$ is removed by hard-limiting.

The cross-correlation of $s(n)$ and $d(n)$ is

$$R_{sd} = \mathcal{E}\{\, s(n)\, d^*(n)\,\} = \mathcal{E}\left\{\, \frac{s(n)y^*(n)}{|y(n)|}\,\right\} \tag{3.38}$$

Using the binomial approximation $(1+r)^{-1/2} \simeq 1 - r/2$,

$$\begin{aligned} \frac{1}{|y(n)|} &= \frac{1}{\sqrt{y(n)y^*(n)}} \\ &= \left(1 + g^2\, m^2(n) + 2g\, m(n)\cos(\phi(n) - \psi(n))\right)^{-1/2} \\ &\simeq 1 - gm(n)\cos\Delta(n) \end{aligned} \tag{3.39}$$

where $\Delta(n) = \phi(n) - \psi(n)$. Before proceeding further we consider the PDF of $\Delta(n)$.

We are concerned here with the PDF of the phase *difference* of two independent complex baseband signals for the case in which the PDF of the phase of each signal is uniform over $(-\pi, \pi]$. The desired PDF is obtained by convolving two uniform PDFs, which results in a triangular-shaped PDF over $(-2\pi, 2\pi]$. Since the phase wraps ($e^{j\Delta} = e^{j2\pi\Delta}$) the PDF of Δ is uniform over $(-\pi, \pi]$. This is true even if the received signals have the same modulation format and identical carrier frequencies.

The cross-correlation of $s(n)$ and $d(n)$ can now be approximated as

$$\begin{aligned} R_{sd} &\simeq \mathcal{E}\Big\{\, e^{j\phi(n)}\left(e^{-j\phi(n)} + gm(n)e^{-j\psi(n)}\right)(1 - gm(n)\cos\Delta(n))\,\Big\} \\ &\simeq \mathcal{E}\Big\{\, \left(1 + gm(n)e^{j\Delta(n)}\right)(1 - gm(n)\cos\Delta(n))\,\Big\} \\ &\simeq 1 \end{aligned} \tag{3.40}$$

where the magnitude of the interfering signal $m(n)$ and the phase difference $\Delta(n)$ are assumed independent. The result that $R_{sd} \simeq 1$ is intuitively appealing since the SIR is high.

The cross-correlation of $z(n)$ and $d(n)$ is

$$\begin{aligned} R_{zd} &\simeq \mathcal{E}\Big\{\, m(n)e^{j\psi(n)}\left(e^{-j\phi(n)} + gm(n)e^{-j\psi(n)}\right)(1 - gm(n)\cos\Delta(n))\,\Big\} \\ &\simeq \mathcal{E}\Big\{\, \left(m(n)\, e^{-j\Delta(n)} + g\, m^2(n)\right)(1 - gm(n)\cos\Delta(n))\,\Big\} \\ &\simeq g\mathcal{E}\Big\{\, m^2(n)\,\Big\} - (g/2)\mathcal{E}\Big\{\, m^2(n)\,\Big\} \\ &\simeq g/2 \end{aligned} \tag{3.41}$$

The output SIR (3.22) now becomes

$$\text{SIR} = \frac{|R_{sd}|^2}{|R_{zd}|^2} = 4/g^2 \tag{3.42}$$

Since the input SIR is $1/g^2$, the ratio of the output SIR to the input SIR is 4, so that the SIR increases by 6 dB. This result for high SIR holds for any CM signal with uncorrelated cochannel noise and interference and will be observed in the simulation results to follow.

3.3.2 Two Complex Sinusoid Environment

We now examine the behavior of the LSCMA in an environment where the antenna array receives two orthogonal complex sinusoids in the absence of background noise. We show that if the SIR at iteration k is known, the LSCMA output SIR can be predicted exactly for all later iterations. We also show that these results are a very good approximation with sinusoids having arbitrary, but well separated, frequencies. The results presented here are deterministic. Other results presented later examine the mean behavior of the LSCMA using a probabilistic framework.

The beamformer output signal obtained with the existing LSCMA weight vector is modeled as

$$y(n) = s(n) + gz(n) = e^{j\omega_1 n} + ge^{j\omega_2 n} \tag{3.43}$$

where $\omega_i = 2\pi k_i/N$ for integer k_i. The parameter g determines the relative power of the sinusoids. In (3.43), $s(n)$ and $z(n)$ represent the desired signal and the interferer, respectively. The amplitude of the desired signal in $y(n)$ is assumed to be unity, which has no effect on the behavior of the LSCMA.

The temporal cross-correlation of the desired signal, $s(n)$, and the hard-limiter output signal, $d(n)$, is

$$\begin{aligned} \hat{R}_{sd} &\triangleq \langle\, s(n)\, d^*(n)\,\rangle_N \\ &= \left\langle \frac{1 + ge^{j\Delta n}}{|y(n)|} \right\rangle_N \end{aligned} \tag{3.44}$$

where $\Delta \triangleq \omega_1 - \omega_2$. We note that since y is periodic, $1/|y|$ is also periodic and may therefore be expressed as a Fourier series. The period of $1/|y(n)|$ is $2\pi/\Delta$. The function is real and even, so the Fourier series is given by

$$\frac{1}{|y(n)|} = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos k\Delta n \tag{3.45}$$

where

$$a_k = \frac{2\Delta}{\pi} \int_0^{\pi/\Delta} \frac{\cos k\Delta n}{\sqrt{1+g^2+2g\cos\Delta n}}\, dn \tag{3.46}$$

The Fourier coefficients given by (3.46) are independent of Δ, which implies that the results to follow hold true for any frequencies ω_1 and ω_2, if these frequencies lead to orthogonal sinusoids. Substituting (3.45) into (3.44) yields

$$\begin{aligned} \hat{R}_{sd} &= \left\langle \left(1+ge^{j\Delta n}\right)\left(\frac{a_0}{2}+\sum_{k=1}^{\infty} a_k \cos k\Delta n\right)\right\rangle_N \\ &= \frac{a_0}{2} + \left\langle ge^{j\Delta n}\sum_{k=1}^{\infty} a_k \cos k\Delta n \right\rangle_N \\ &= \frac{a_0+ga_1}{2} \end{aligned} \tag{3.47}$$

The Fourier coefficients a_0 and a_1 may be found using numerical integration. Using a similar approach for $\hat{R}_{zd}$ yields

$$\begin{aligned} \hat{R}_{zd} &= \left\langle \left(e^{-j\Delta n}+g\right)\left(\frac{a_0}{2}+\sum_{k=1}^{\infty} a_k \cos k\Delta n\right)\right\rangle_N \\ &= \frac{ga_0+a_1}{2} \end{aligned} \tag{3.48}$$

The output SIR of the hard-limiter is

$$\mathrm{SIR}_{out} = \frac{|R_{sd}|^2}{|R_{zd}|^2} = \left(\frac{a_0+ga_1}{ga_0+a_1}\right)^2 \tag{3.49}$$

and the SIR gain is

$$\frac{\mathrm{SIR}_{out}}{\mathrm{SIR}_{in}} = \left(\frac{a_0+ga_1}{g(ga_0+a_1)}\right)^2 \tag{3.50}$$

Figure 3.4 shows the SIR gain (3.50) as a function of input SIR [1]. Note that the SIR gain tends asymptotically to 6 dB as predicted by the high SIR analysis.

The results presented in Figure 3.4 are now used to predict the output SIR of the LSCMA in an environment containing a sinusoidal desired signal and a sinusoidal interferer. A two element beamformer is simulated with the inter-element spacing equal to one-half the carrier wavelength, λ. The desired signal is incident from broadside with an amplitude of one, and the interferer is incident from 30° off broadside with an amplitude of 0.9. The initial LSCMA weight vector is set to $[\,1\;0\,]^T$, so that the initial SIR $\simeq +0.9$ dB. The block size is 1024 samples, with $f_1 = 5/1024$ and $f_2 = -31/1024$. Table 3.1 compares the predicted and

[1] The Fourier series coefficients were computed with the MATLAB™ numerical integration routine **quad8**

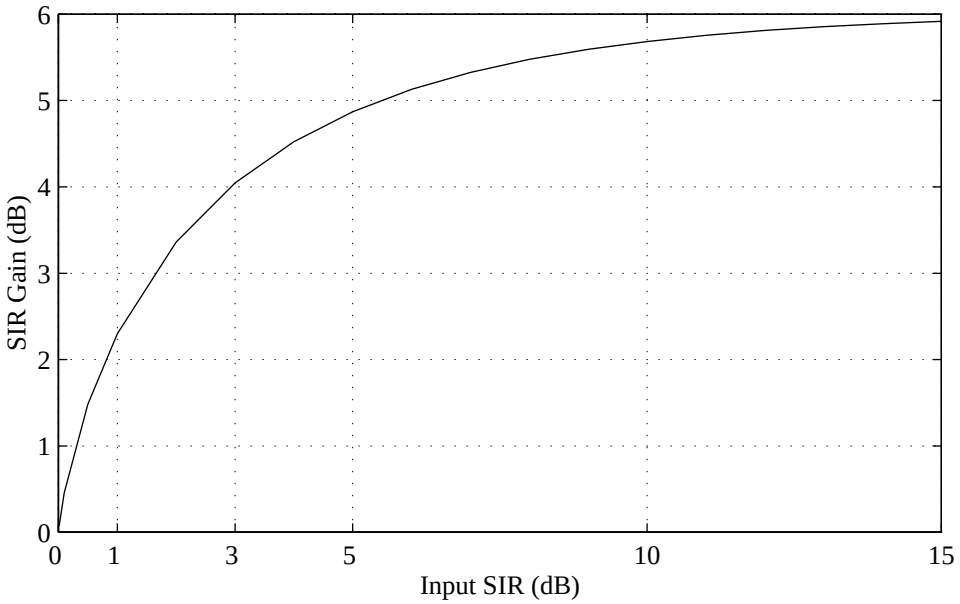

Figure 3.4: Improvement in SIR achieved by one iteration of LSCMA with sinusoidal desired signal and sinusoidal interferer.

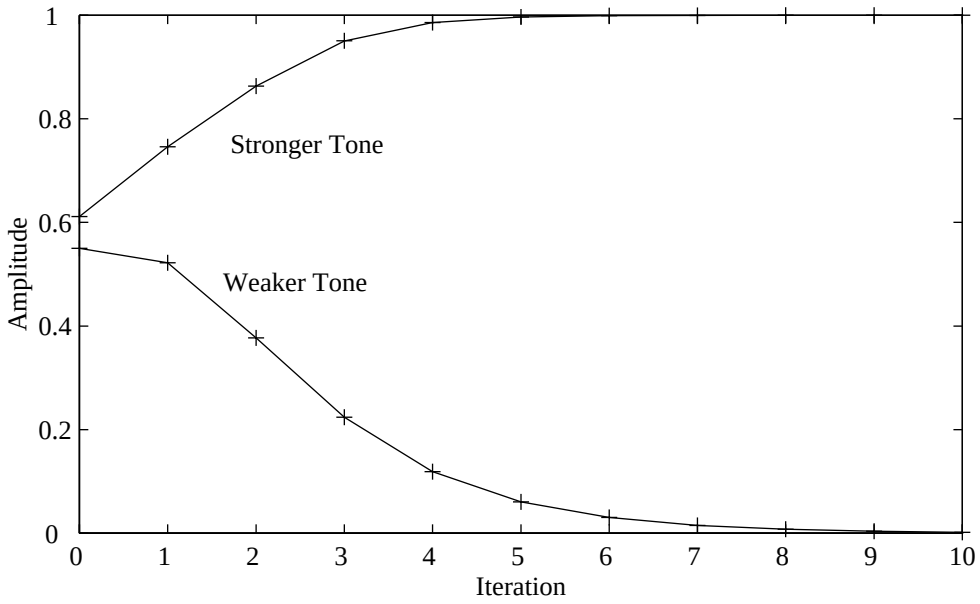

Figure 3.5: Amplitude of both complex sinusoids in the beamformer output as a function of the number of LSCMA iterations. Solid line indicates predicted amplitude, '+' indicates amplitude measured in simulation.

measured output SIR at several iterations of the algorithm, which shows excellent agreement with theory. These same results are presented in an alternative manner in Figure 3.5. This figure shows the amplitude of each sinusoid as a function of the number of LSCMA iterations. A similar figure, showing the behavior of the SGD CMA in an environment with two sinusoids, appears in [66].

Iteration	Output SIR (dB)		SIR Gain
	Measured	Theory	
0	0.915	0.915	-
1	3.097	3.095	2.181
2	7.195	7.193	4.099
3	12.554	12.552	5.358
4	18.390	18.388	5.837
5	24.363	24.361	5.973
6	30.372	30.370	6.009
7	36.389	36.387	6.018
8	42.409	42.407	6.020

Table 3.1: Comparison of predicted and measured LSCMA output SIR in an environment containing two complex sinusoids.

We now briefly consider the case in which the two complex sinusoids are not orthogonal. The cross-correlation between the two sinusoids will be small if the two sinusoids are well separated in frequency. For this case we would expect the preceding analysis to be a good approximation to the observed behavior. This is supported by the results shown in Table 3.2. The simulation parameters are the same as in Table 3.1, except that $f_1 = 0.5/1024$ and $f_2 = 3/1024$. The agreement with theory is still very good even though the sinusoids are not orthogonal.

3.3.3 CM Signal with CM Interference

We now consider an environment where the antenna array receives a CM desired signal and a CM interferer in the absence of background noise. Unlike the deterministic framework employed in the sinusoidal environment, we now rely on a probabilistic framework. These results therefore describe the mean behavior of the LSCMA. The initial beamformer output signal $y(n)$ is modeled as

$$y(n) = e^{j\phi(n)} + ge^{j\psi(n)} \tag{3.51}$$

Iteration	Output SIR (dB)		SIR Gain
	Measured	Theory	
0	0.915	0.915	-
1	3.084	3.095	2.17
2	7.119	7.193	4.035
3	12.358	12.552	5.239
4	18.054	18.388	5.697
5	23.885	24.361	5.831
6	29.752	30.370	5.867
7	35.627	36.387	5.876
8	41.505	42.407	5.878

Table 3.2: Comparison of predicted and measured LSCMA output SIR in an environment containing two complex sinusoids when the sinusoids are not orthogonal.

where $\phi(n)$ and $\psi(n)$ denote the phase of the desired signal and interfering signal, respectively, and g determines the relative power. We assume that $\phi(n)$ and $\psi(n)$ are independent random variables uniformly distributed over $(-\pi, \pi]$.

The cross-correlation of the desired signal $s(n)$ and the hard-limiter output signal $d(n)$ is

$$R_{sd} = \int_{-\pi}^{\pi} \frac{1 + ge^{j\Delta}}{\sqrt{1 + g^2 + 2g\cos\Delta}}\, p(\Delta)\, d\Delta \tag{3.52}$$

where $\Delta \triangleq \phi(n) - \psi(n)$ and $p(\Delta) = 1/2\pi$, $-\pi < \Delta \leq \pi$, is the PDF of Δ. We simplify the expression for R_{sd} by noting that $e^{j\Delta} = \cos\Delta + j\sin\Delta$, and odd functions when integrated over $-\pi$ to π yield zero. We thus obtain

$$R_{sd} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1 + g\cos\Delta}{\sqrt{1 + g^2 + 2g\cos\Delta}}\, d\Delta \tag{3.53}$$

Similarly, the cross-correlation of the interferer $z(n)$ and the hard-limiter output signal can be expressed as

$$R_{zd} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{g + \cos\Delta}{\sqrt{1 + g^2 + 2g\cos\Delta}}\, d\Delta \tag{3.54}$$

The SIR (3.22) is calculated by evaluating both (3.53) and (3.54) by numerical integration. The resulting SIR gain is shown in Figure 3.6. The expected gain of hard-limiting in this environment is *identical* to the deterministic gain achieved in the sinusoidal signal environment. The mean SIR gain measured from simulation when the LSCMA is applied to an environment containing two FM signals with low-pass Gaussian messages is also shown in

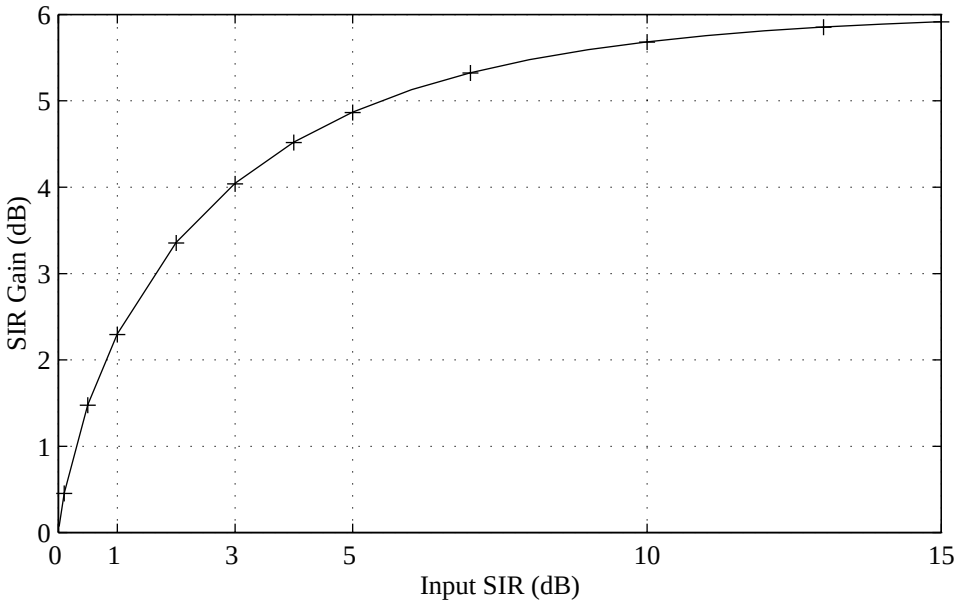

Figure 3.6: Improvement in output SIR achieved with one iteration of LSCMA with an FM desired signal and an FM interferer. Solid line indicates theoretical gain, '+' indicates mean gain measured in simulations.

the Figure 3.6. The array configuration and AOA of the signal and interference are the same as those used previously in the sinusoidal environment. The signals are generated by low-pass filtering Gaussian noise to a normalized bandwidth of 0.125, and then frequency modulating using a frequency deviation of 0.1. The baseband FM signals have no carrier frequency offset. The mean SIR gain is measured from 1000 Monte Carlo trials using $N = 256$ samples. Figure 3.6 shows excellent agreement between the theoretical and measured SIR gain.

As further evidence of the applicability of the above derivation to other CM modulation formats, we apply LSCMA to an environment with two QPSK signals. In order for the QPSK signals to be CM, we assume that they have the same symbol timing, and that they have been appropriately match-filtered and sampled baud-synchronously. We also assume that the QPSK signals have no carrier frequency offset. Thus each QPSK signal takes on one of four values which are drawn randomly from the set $\left\{ \pm 1/\sqrt{2}\,, \pm i/\sqrt{2} \right\}$. The carrier phase of each QPSK signal is randomly drawn from a uniform distribution for each trial. Simulation results are shown in Figure 3.7. All simulation parameters are the same as those used in the FM signal environment. Once again, there is excellent agreement between the predicted and observed behavior of LSCMA.

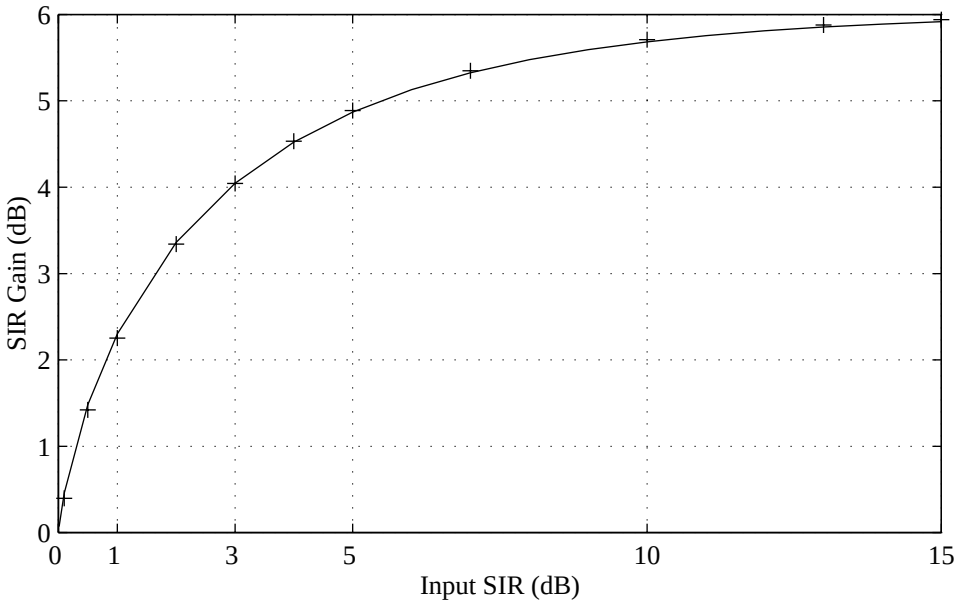

Figure 3.7: Improvement in output SIR achieved with one iteration of LSCMA with a QPSK desired signal and a QPSK interferer. Solid line indicates theoretical gain, '+' indicates mean gain measured in simulations.

In most practical applications, PSK signals with non-rectangular pulse shape are used in order to reduce the signal bandwidth. Unless the PSK signal is appropriately match-filtered and sampled baud synchronously it will not be CM. Accurate estimation of symbol timing is difficult in the presence of strong co-channel interference. However, this is precisely the sort of environment where adaptive beamforming would be applied. Thus the preceding assumption that the desired QPSK signal is sampled baud synchronously will not be valid in general. One solution to this problem is to apply the LSCMA to oversampled data, i.e., sample the digital signal at a rate higher than the symbol rate, since the CMA can be applied to non-constant modulus signals [40, 39]. The symbol timing can be re-estimated as the LSCMA converges. Ultimately the LSCMA can be adapted using only the baud-synchronous (constant modulus) samples instead of the oversampled (non-constant modulus) data. The analysis framework described here can be used to determine the performance of the LSCMA with pulse-shaped PSK signals. However, such an analysis is beyond our scope, and we instead rely on simulation to gain some insight into this issue. We would expect that the LSCMA will still converge with a pulse-shaped QPSK signal, but that the convergence will be slower.

We examine the behavior of LSCMA with two pulse-shaped $\pi/4$ QPSK signals. This modulation is commonly used in cellular and PCS applications. The signals are sampled at 8 times

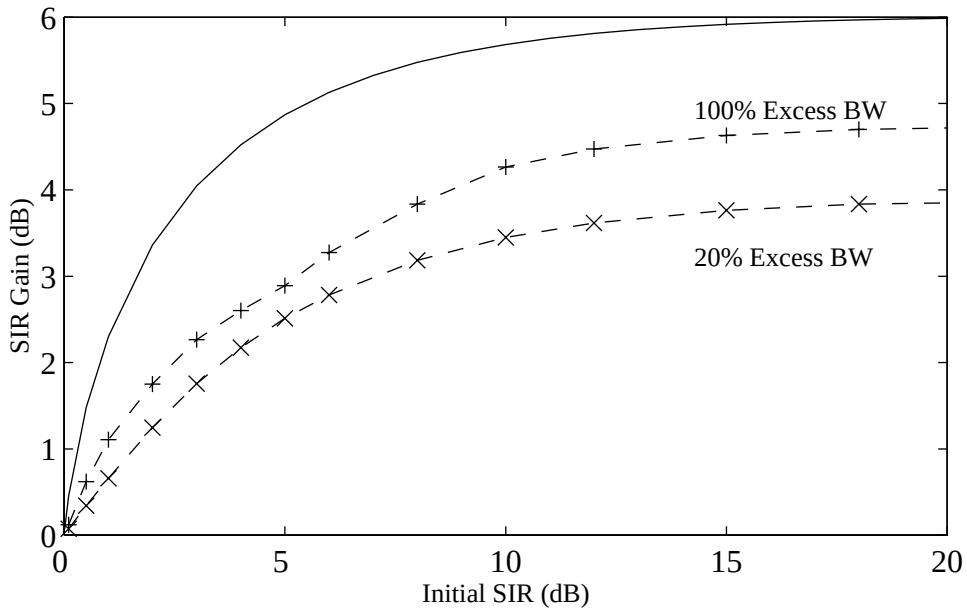

Figure 3.8: SIR gain of LSCMA with non-constant modulus pulse-shaped $\pi/4$ QPSK signals. The results are parametric in the percent of excess bandwidth for each signal. The dotted curves are based on Monte Carlo simulation, the solid curve is the theoretical result for a CM desired signal with CM interference.

the symbol rate, and Nyquist-type pulse shaping is used. It is well known that signals having lower excess bandwidth have higher modulus variation. Thus we would expect LSCMA to converge more quickly with higher excess bandwidth signals. This is verified by Figure 3.8, which shows the SIR gain for both 20% and 100% excess bandwidth. The simulation parameters used to generate these results are the same as those used previously. The SIR gain for the 100% excess bandwidth signal appears to asymptotically approach ≈ 4.7 dB as the initial SIR becomes high. The SIR gain for the 20% excess bandwidth signal appears to approach ≈ 3.9 dB as the initial SIR becomes high. These results indicate that the LSCMA will converge more slowly with a non-constant modulus signal than with a CM signal.

3.3.4 CM Signal with Gaussian Interference

We now examine the behavior of the LSCMA with a CM signal and Gaussian interference. These results are of interest since the distribution of a large number of co-channel interferers, as might be encountered in CDMA applications, will tend toward Gaussian by the central

limit theorem. The input to the hard-limiter is expressed as

$$y(n) = s(n) + gz(n) = e^{j\phi(n)} + g\,m(n)\,e^{j\psi(n)} \tag{3.55}$$

where $s(n)$ is an angle-modulated signal and $z(n)$ is unit-variance complex Gaussian interference. Note that $\psi(n)$ is uniformly distributed over $(-\pi, \pi]$ while $m(n)$ is Rayleigh distributed with PDF

$$r(m) = \begin{cases} 2me^{-m^2} & m > 0 \\ 0 & m < 0 \end{cases} \tag{3.56}$$

The cross-correlation of $s(n)$ and $d(n)$ is

$$R_{sd} = \frac{1}{\pi}\int_0^\infty Q(m)\,me^{-m^2}\,dm \tag{3.57}$$

where

$$Q(m) = \int_{-\pi}^{\pi} \frac{1 + g\,m\cos\Delta}{\sqrt{1 + g^2m^2 + 2g\,m\cos\Delta}}\,d\Delta \tag{3.58}$$

In a similar fashion it can be shown that

$$R_{zd} = \frac{1}{\pi}\int_0^\infty P(m)\,me^{-m^2}\,dm \tag{3.59}$$

where

$$P(m) = \int_{-\pi}^{\pi} \frac{m\cos\Delta + gm^2}{\sqrt{1 + g^2m^2 + 2g\,m\cos\Delta}}\,d\Delta \tag{3.60}$$

Both (3.57) and (3.59) are evaluated by numerical integration and used to obtain the SIR gain shown in Figure 3.9. The SIR gain as measured from simulations is also shown in Figure 3.9 and verifies the theoretical analysis. The simulation parameters are the same as those used previously. As before the SIR gain tends to 6 dB as the input SIR becomes high. Note that the SIR gain is greater than 0 dB even for an input SIR of -10 dB. This would seem to indicate that LSCMA can be expected to converge even at low initial input SIR. However, it is important to bear in mind that these results are based on probabilistic notions. The SIR gain will be a random variable for finite block size and Figure 3.9 shows only the expected value of this random variable.

We have now calculated α and β for several important signal and interference distributions. At this point we have sufficient information to study the (1,2)-CMA cost function using these expressions for α and β.

3.4 Cost Function Analysis

In order to perform a thorough analysis of any blind algorithm, it is necessary to find the stationary points of the cost function. A stationary point occurs when the gradient with

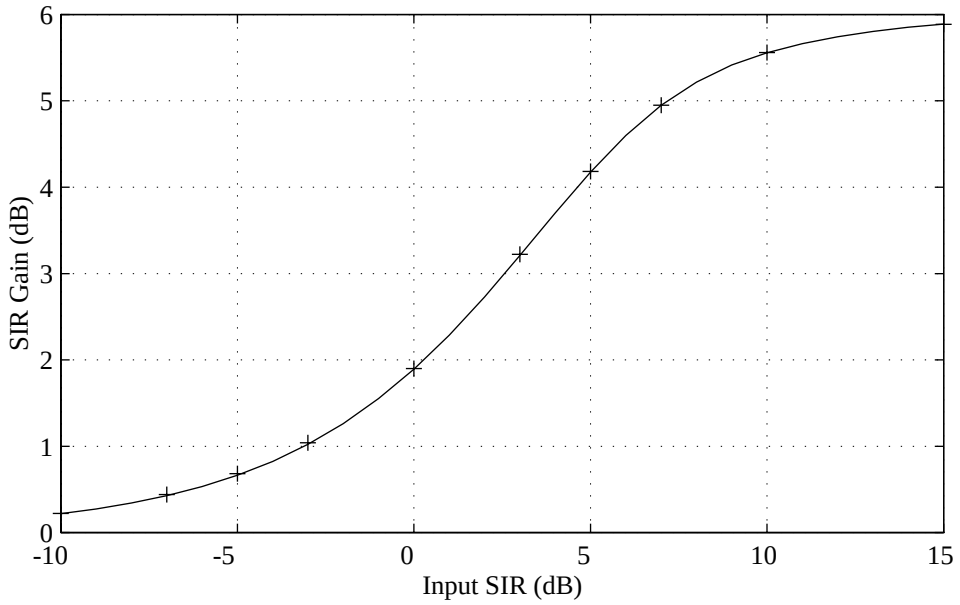

Figure 3.9: Improvement in output SIR achieved with one iteration of LSCMA for a CM signal plus Gaussian interference. Solid line indicates theoretical gain, '+' indicates mean gain measured in simulations with an FM signal and a Gaussian interferer.

respect to the weight vector is equal to zero. These stationary points are important because they correspond to local minima, local maxima, or saddlepoints of the cost function. The gradient of a blind cost function with respect to the weight vector cannot, in general, be solved for directly. If this could be accomplished, the weight vector that minimizes the cost function could be solved for directly without the need for iterative algorithms. However, it is sometimes possible to express the cost function in terms of the output SINR and other parameters as opposed to the weight vector. This can simplify the analysis.

We show in Appendix A that for the (1,2)-CMA cost function

$$F(\rho) = 2 - \frac{2}{\sqrt{1+\rho}} \left(\alpha^* \sqrt{\rho} + \beta^*\right) \tag{3.61}$$

where α and β are as defined earlier, and ρ is the beamformer output SINR. Note that α and β are dependent on the SINR ρ.

Figure 3.10 shows the (1,2) CMA cost function as calculated from (3.61) for the case of a CM signal and CM interference. Simulation results are included to support the analytic results. These simulation results are based on a CM signal and a CM interferer. The signal and interference are white, and the phase in each case is random and uniformly distributed. The (1,2) CMA cost function was measured with a data length of 1024 samples, and averaged

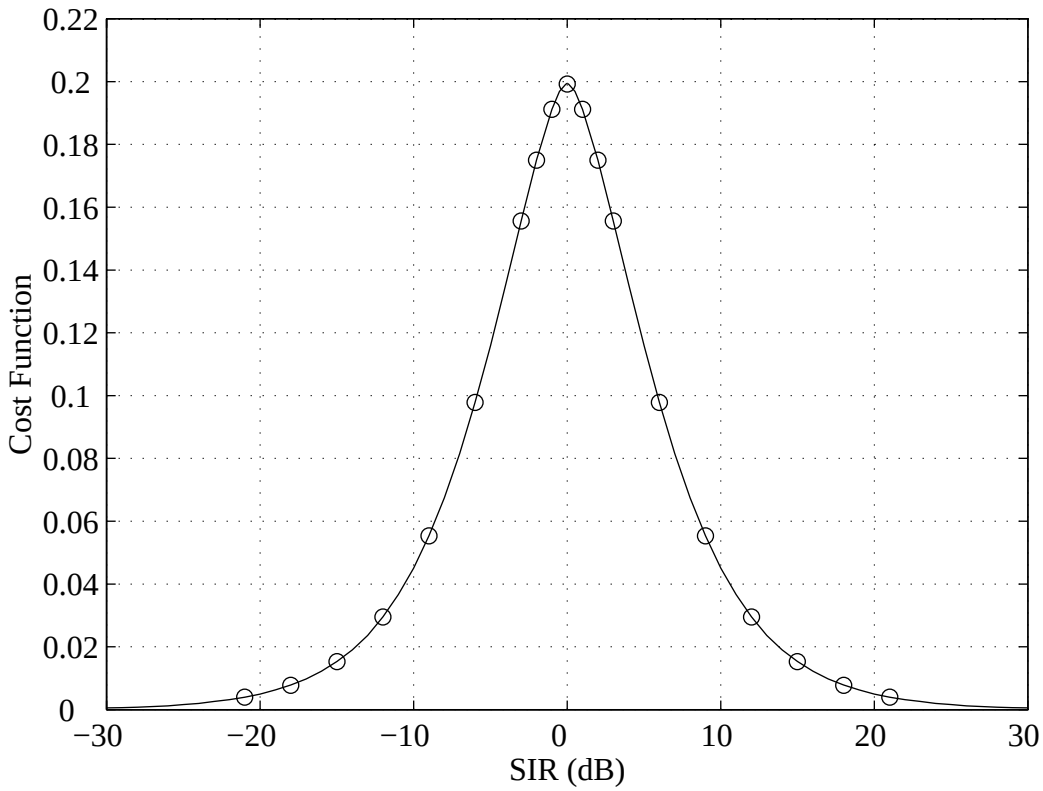

Figure 3.10: (1,2) CMA cost function versus SIR with a CM signal and CM interference. The solid line shows the analytic expression, the '$\circ$' show simulation results.

over 1000 independent Monte Carlo trials.

Note that the cost function in Figure 3.10 is symmetric about 0 dB SIR. This is to be expected, since the CMA cost function cannot distinguish between a CM signal and a CM interferer. Also note that the cost function has a global maximum at 0 dB SIR. Since the gradient is small in the neighborhood of 0 dB SIR, a gradient search algorithm, such as steepest descent, will converge slowly in this environment if the initial SIR is low.

Figure 3.11 shows the (1,2) CMA cost function as calculated from (3.61) for the case of a CM signal and Gaussian noise and interference. Simulation results are again included to support the analytic expression (3.61). The simulation parameters are identical to those used previously, except that the noise is complex Gaussian as opposed to CM. As expected, the cost function grows small as the SIR grows large. However, the behavior for low SIR is very different for Gaussian noise than for CM noise. In Gaussian noise, the cost function remains large, but the gradient approaches zero as the SIR grows small. Since a gradient-based algorithm seeks to find a point in the cost function where the gradient is zero, a

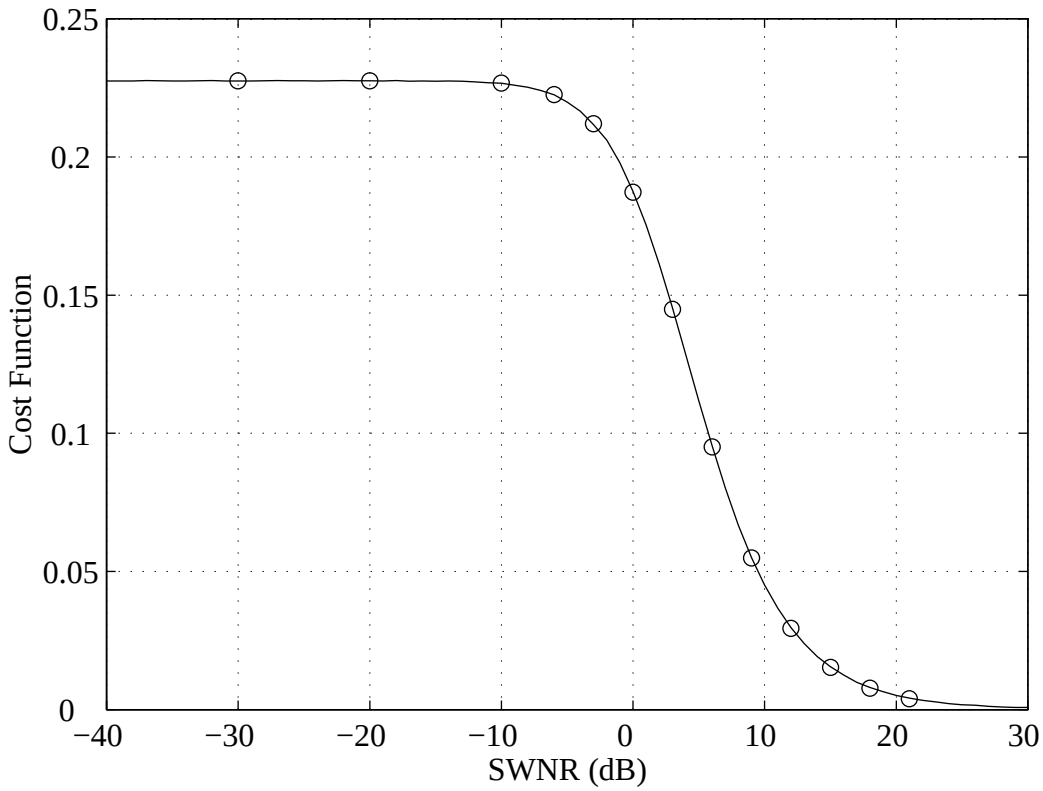

Figure 3.11: (1,2) CMA cost function versus SIR with a CM signal and Gaussian interference. The solid line shows the analytic expression, the '∘' show simulation results.

gradient-based CMA may become trapped in a low output SIR state. This is known as *noise capture* [80]. In noise capture, the output of a CMA-adapted array consists of Gaussian background noise; any CM signals received by the array are nulled.

3.5 Inclusion of Background Noise

In this section we examine the effect of background noise on the behavior of the LSCMA. We assume that the noise has a complex circularly symmetric Gaussian distribution and is uncorrelated from sensor to sensor. We consider one environment where the interference is Gaussian, and another where the interference is CM. In both cases the desired signal is QPSK and is assumed to have been match filtered and sampled so that it is CM. All simulation results are based on 1000 trials with the LSCMA block size equal to 256 symbols. The array is linear with eight elements and uniform interelement spacing equal to $\lambda/2$. The

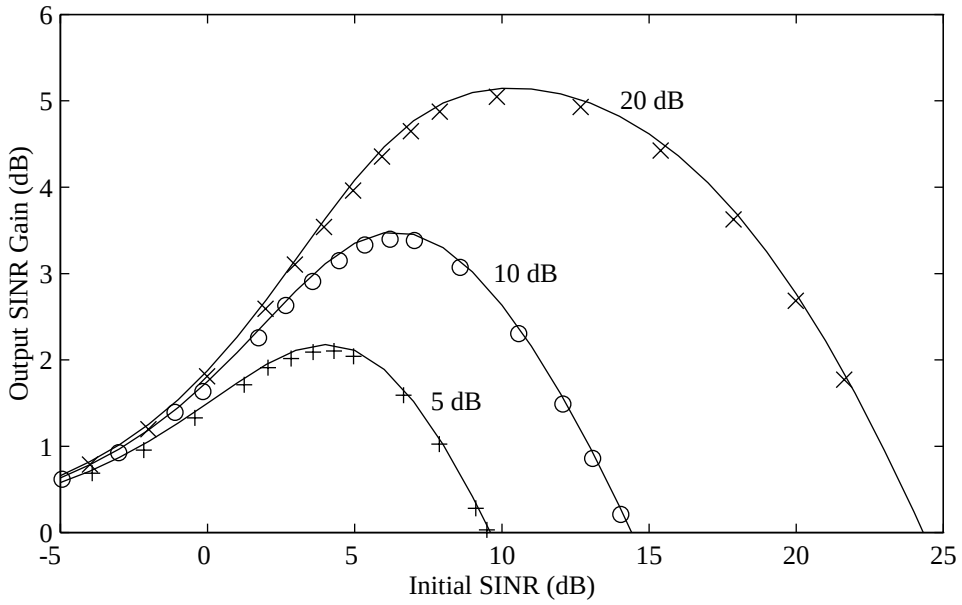

Figure 3.12: Improvement in output SIR achieved with one iteration of LSCMA with a QPSK desired signal received with a Gaussian interferer and Gaussian background noise. Solid lines indicate theoretical result, '+','o', and 'x' denote mean gain from simulation.

signal power is measured relative to the unit variance background noise, and is termed the Signal to White Noise Ratio (SWNR).

First consider the case where the desired signal is incident from 0° and a single Gaussian interferer is incident from 5°. Since the noise and interference is Gaussian, the SINR gain from hard-limiting is given by (3.57) and (3.59). The LSCMA output SINR is related to the SINR gain from hard-limiting by (4.34). The mean LSCMA SINR gain is presented in Figure 3.12 for SWNR equal to 5, 10, and 20 dB. The power of the interferer is kept equal to the power of the desired signal. The figure shows excellent agreement between the measured and predicted SINR gain. As the SWNR increases, the optimal output SINR increases, and the SINR gain approaches the gain obtained when no background noise is present.

Now consider the case where the interference is CM. Since the interference is not Gaussian, calculation of the SINR gain is tedious and it is appropriate to make some approximations. When the beamformer output SINR is low, the dominant source of distortion is the CM interferer, and the SIR gain from hard-limiting can be accurately predicted by the results for CM interference, given by (3.53) and (3.54). As the output SINR becomes higher, the interferer is nulled, and the background noise becomes the dominant source of distortion. However, we have shown that in all cases the SINR gain from hard-limiting approaches 6 dB

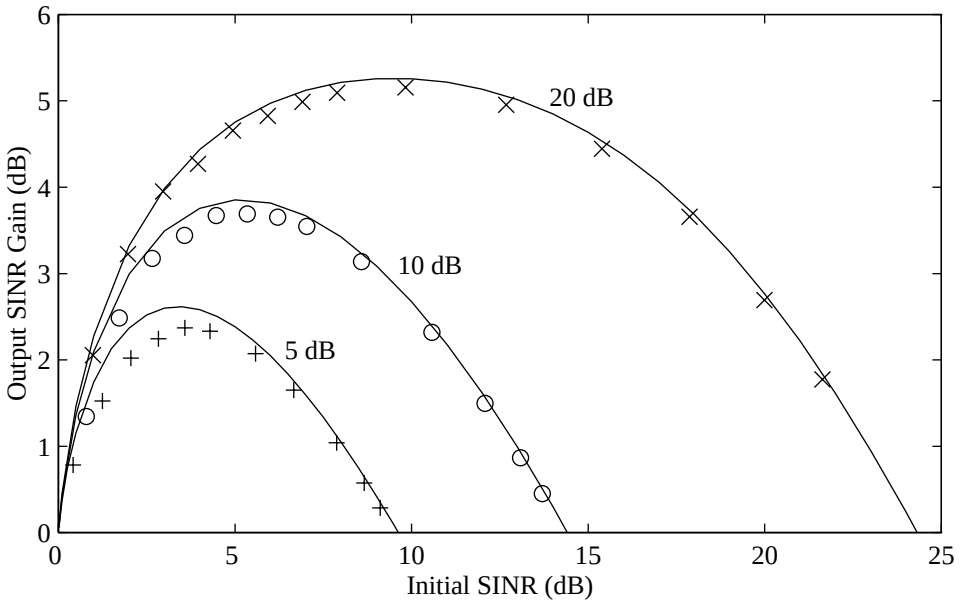

Figure 3.13: Improvement in output SIR achieved with one iteration of LSCMA with a QPSK desired signal received with a QPSK interferer and Gaussian background noise. Solid lines indicate theoretical result, '+','o', and 'x' denote mean gain from simulation.

as the SINR becomes high. Therefore the behavior of LSCMA in this case can be predicted by using the results for CM interference (3.53) and (3.54) together with (4.34). Simulation results for an environment similar to that described above, except that the Gaussian interferer is replaced with a CM QPSK interferer, are presented in Figure 3.13. This figure shows very good agreement between the approximate theoretical result and the simulation results.

3.6 Finite Block Size

The effect of finite block size on classical least squares beamforming (with a known training signal) has been considered previously [28, 81]. While it should be possible to apply some of these same techniques to the analysis of the LSCMA, we instead rely on simulation to obtain some intuition regarding the effect of finite block size. The mean behavior of the LSCMA should be very close to that predicted by the analysis, even for small block size. Furthermore the variance of the SIR gain should decrease with increasing block size. This is verified by Figure 3.14, which shows the distribution of SIR gain versus the LSCMA block size with an FM desired signal and an equal power Gaussian interferer. The results are based on 1000 trials. Note that a significant number of trials had SIR gain of less than 0 dB,

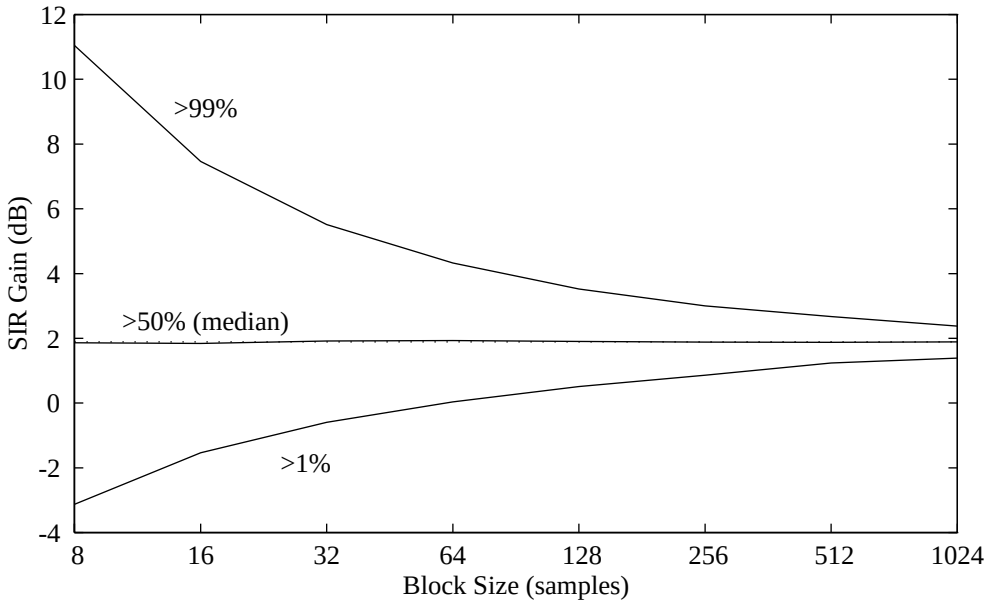

Figure 3.14: SIR gain of LSCMA versus block size with an FM desired signal and 0 dB Gaussian interferer. Dotted line indicates theoretical gain. Upper and lower traces define region where 98% of trials fell.

i.e., the Gaussian interferer was emphasized. The LSCMA will not converge to the desired solution in these cases, but will instead capture the Gaussian interferer. However, for block sizes of 64 samples and higher, no trials were observed to have negative SIR gain. Thus the algorithm tends to behave in the desired manner as the block size increases. Proper selection of the initial weight vector can also help ensure LSCMA convergence to the desired solution.

3.7 Conclusions

We have examined the convergence behavior of the LSCMA in some simple environments. The results are derived by calculating the improvement in SIR caused by hard-limiting a CM signal plus additive noise and interference. These results help to explain why LSCMA converges, and are helpful in explaining the general behavior of LSCMA. However, we have made several simplifying assumptions which must be addressed in order to extend the analysis to more realistic situations.

We have assumed that the desired signal and the interfering signals are not correlated. This does not in general allow the direct analysis of correlated multipath environments. However,

the analysis presented here is valid for the case where the multipath delay spread is very small. In this case, the delayed paths are highly correlated with the desired path, and the overall effect is simply to change the spatial signature.

Many applications require the extraction of multiple signals. Algorithms such as Multi-Target CMA [46], Multistage CMA [45], and Iterative Least Squares with Projection [47] can be used for this purpose. The results presented here can form a basis for analysis of these multi-signal extraction techniques. Clearly the variance and distribution of output SINR obtained with the LSCMA is also an important area for investigation.

We finally comment on the hard-limit non-linearity. For high SIR, the hard-limiter is the optimal non-linearity when the desired signal has a constant envelope [79]. However, at low SIR other non-linearities can yield greater SIR gain. Thus it is possible that non-linear functions other than the hard-limit can be used to develop blind adaptive algorithms which converge faster for low initial SINR.

Chapter 4

Decision Directed Algorithm in Gaussian Interference

Summary

We examine the convergence of an existing decision directed adaptive algorithm in an adaptive beamforming application. This analysis allows us to compare the convergence rate of a decision directed algorithm and a constant modulus algorithm analytically. The signal environment which we examine consists of a desired PSK signal received at multiple antennas in the presence of additive Gaussian co-channel interference. Results are presented for M-ary PSK in general and for BPSK and QPSK in particular. As expected, the results show that the decision directed algorithm converges rapidly for moderately high initial SINR. The block-update algorithm considered here can converge even when the initial symbol error rate is very high. The effect of carrier phase offset on algorithm performance is examined, and a modified algorithm that offers faster convergence with unknown carrier phase is presented.

4.1 Introduction

The focus of this chapter is on the convergence properties of an algorithm which we will refer to as the Least Squares Decision Directed Algorithm (LSDDA). The LSDDA is a fast-converging, block update algorithm which has been studied by Swindlehurst et al. in [9]. The results presented in [9] have been widely cited by other authors, including [10, 11, 12, 13, 14, 15, 16]. It is argued in [9] that the LSDDA will converge in one iteration as the block size grows large, independent of the initial SNR. In contrast we argue that multiple LSDDA

iterations are required for convergence, even for infinite block size (unless the initial SNR is high).

We make several simplifying assumptions in order to make the analysis of the LSDDA tractable. First, we assume that the signal is uncorrelated with the noise and interference. Thus we do not consider frequency selective channels in general. We also assume that the carrier frequency of the desired signal is known to the receiver. Finally, we assume that the symbol timing is known to the receiver. Clearly there are many situations where these assumptions are not reasonable. However, these assumptions are made as a first step towards understanding the behavior of the LSDDA.

This chapter is organized as follows. First an overview of the LSDDA is given along with a description of the array data model. The Least Squares Constant Modulus Algorithm (LSCMA) [43, 44], which is very similar to the LSDDA, is also reviewed. Then the framework used to analyze the performance of the LSDDA is presented. This framework has also been used to study the LSCMA [63, 7]. Since our analysis yields distinctly different results from those reported in [9], Section 4.4 presents a detailed comparison of our approach and the approach employed in [9]. The behavior of the LSDDA with a PSK signal received in Gaussian noise and interference is considered in Section 4.5, including an analysis of the effect of carrier phase offset. Results are presented for BPSK, QPSK, and general M-ary PSK. The relative convergence rate of the LSCMA and the LSDDA is also studied. This is believed to be the first *analytic* comparison of the convergence rate of CMA and a decision directed algorithm. Section 4.7 demonstrates that the LSDDA analysis framework proposed here may prove useful for a full analysis of the LSDDA for finite block size. We conclude with a summary of the results presented here and describe some directions for future work.

4.2 Overview of the Decision Directed Algorithm

This section presents an overview of the LSDDA in an adaptive beamforming application. We first describe the model used for the observed data and then describe the LSDDA in detail. The objective is to obtain an estimate of a desired signal in the presence of cochannel interference using an array of antennas. Letting the $M \times 1$ vector $\mathbf{x}(n)$ represent the signals and noise received at an array of M antennas at discrete time n gives

$$\mathbf{x}(n) = \mathbf{a}_s s(n) + \mathbf{q}(n) \tag{4.1}$$

where the $M \times 1$ column vector $\mathbf{a}_s$ is the spatial signature of the unit variance desired signal $s(n)$, and the $M \times 1$ vector $\mathbf{q}(n)$ contains noise and interference. The power of the noise and

interference is normalized so that

$$\mathrm{tr}\left(\mathcal{E}\left\{\,\mathbf{q}(n)\,\mathbf{q}^H(n)\,\right\}\right) = \mathrm{tr}\left(\mathbf{R}_{qq}\right) = M \tag{4.2}$$

where $\mathrm{tr}(\cdot)$ denotes the matrix trace. Note that, in this model, the amplitude of the received signal is incorporated in the spatial signature. In the absence of multipath, the spatial signature is only dependent on the angle of arrival, the carrier frequency, and the array geometry. In this case it is often referred to as an array response vector. When flat fading multipath is present, the spatial signature is a linear combination of the array response vectors corresponding to each path taken by the desired signal. This assumes that the delay spread is small relative to the symbol rate. If the delay spread is large, those paths having large differential delay can be incorporated into the noise and interference term. A linear estimate $y(n)$ of the desired signal is given by $y(n) = \mathbf{w}^H\mathbf{x}(n)$, where $\mathbf{w}$ is the $M \times 1$ vector of beamformer weights and $\mathbf{w}^H$ is the Hermitian transpose.

The LSDDA is implemented as follows. First N temporal samples of the data observed at M antennas are collected and arranged to form the $M \times N$ matrix $\mathbf{X}$. The nth column of $\mathbf{X}$ is the data observed at time index n. For simplicity we assume that the samples are taken baud synchronously with one sample per symbol. Denote the $M \times 1$ vector of beamformer weights at the kth iteration by $\mathbf{w}_k$. This weight vector is used to form a beamformer output

$$\mathbf{y}_k = \mathbf{w}_k^H\mathbf{X} \tag{4.3}$$

where $\mathbf{y}_k$ is the $1 \times N$ row vector of beamformer output samples. If no a-priori information is available, the initial weight vector $\mathbf{w}_0$ may be taken as $\mathbf{w}_0 = [1\ 0\ 0\ \cdots 0]^T$. The use of other initial weight vectors has been reported in [67], where the dominant eigenvectors of the spatial covariance matrix have been shown to speed convergence. The beamformer output $y_k(n)$ is then passed to a demodulator which estimates the transmitted symbols. We denote the $1 \times N$ row vector of estimated symbols as $\mathbf{d}_k$. For example, for a BPSK signal the demodulator output $d_k(n)$ is defined as

$$d_k(n) = \begin{cases} 1 & \text{if Real}\left\{y_k(n)\right\} > 0 \\ -1 & \text{if Real}\left\{y_k(n)\right\} < 0 \end{cases} \tag{4.4}$$

The demodulator will naturally be specific to the modulation format of the desired signal. In any case the demodulator output is a non-linear function of the demodulator input. A new weight vector $\mathbf{w}_{k+1}$ is then found which minimizes

$$\left\| \mathbf{d}_k - \mathbf{w}_{k+1}^H\mathbf{X} \right\|_2^2 \tag{4.5}$$

where $\| \mathbf{e} \|_2^2 = |e_0|^2 + \cdots + |e_{N-1}|^2$. Thus the new weight vector is given by

$$\mathbf{w}_{k+1} = \left(\mathbf{X}\mathbf{X}^H\right)^{-1}\mathbf{X}\mathbf{d}_k^H. \tag{4.6}$$

The weight vector update can also be expressed as

$$\mathbf{w}_{k+1} = \hat{\mathbf{R}}_{xx}^{-1}\hat{\mathbf{r}}_{xd} \tag{4.7}$$

where

$$\hat{\mathbf{R}}_{xx} = \sum_{n=0}^{N-1} \mathbf{x}(n)\,\mathbf{x}^H(n) \tag{4.8}$$

and

$$\hat{\mathbf{r}}_{xd} = \sum_{n=0}^{N-1} \mathbf{x}(n)\,d_k^*(n) \tag{4.9}$$

The iteration described by (4.3),(4.4) (using the appropriate demodulator), and (4.7) is continued until no changes are made in the estimated symbols $\mathbf{d}$.

The LSDDA is designed to extract a single signal from the array data. An algorithm similar to the LSDDA can be derived for extracting $L > 1$ signals [9]. Assume that the array receives L signals and that the background noise is white and Gaussian. Then the Maximum Likelihood (ML) estimator of the signals is found by minimizing

$$\left\| \mathbf{X} - \hat{\mathbf{A}}\mathbf{D} \right\|_F^2 \tag{4.10}$$

where

$$\| \mathbf{A} \|_F = \sqrt{\sum_i \sum_j |a_{ij}|^2} \tag{4.11}$$

is the Frobenious matrix norm,

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \vdots \\ \mathbf{d}_L \end{bmatrix} \tag{4.12}$$

is the $L \times N$ matrix of estimated signals, and

$$\hat{\mathbf{A}} = [\ \hat{\mathbf{a}}_1\ \hat{\mathbf{a}}_2\ \cdots \hat{\mathbf{a}}_L\ ] \tag{4.13}$$

is the $M \times L$ corresponding set of estimated spatial signatures. Direct minimization of the ML cost function (4.10) requires a costly multidimensional search over the signal space [14]. One suboptimal approach to finding the ML solution is to estimate the signals and the spatial signature matrix in an alternating fashion. This is sometimes referred to as an *alternating projections* approach. This can be accomplished with the following iterative procedure:

1. At the kth iteration, generate a set of L beamformer outputs $\mathbf{Y}_k$ with

$$\mathbf{Y}_k = \mathbf{W}_k^H\mathbf{X} \tag{4.14}$$

   where each column of $\mathbf{W}_k$ is a beamformer weight vector.

2. Demodulate each of the beamformer outputs, denoting the demodulation function by $f(\cdot)$, so that

$$\mathbf{D}_k = f\left(\mathbf{Y}_k\right) \tag{4.15}$$

3. Estimate a new set of spatial signatures

$$\hat{\mathbf{A}}_{k+1} = \mathbf{X}\mathbf{D}_k^H \left(\mathbf{D}_k\mathbf{D}_k^H\right)^{-1} \tag{4.16}$$

4. Form a new set of weight vectors

$$\mathbf{W}_{k+1} = \hat{\mathbf{A}}_{k+1}\left(\hat{\mathbf{A}}_{k+1}^H\hat{\mathbf{A}}_{k+1}\right)^{-1} \tag{4.17}$$

The LSSDA as described applies multiple iterations to the *same* block of data. The LSDDA can also be implemented so that a new block of data is used for each weight vector update. This may be beneficial in some applications. For example, the LSDDA may be more likely to converge if each update uses an independent block of data, especially if the block size is small. However, it is more computationally efficient to apply multiple iterations to the same set of data. Direct solution of a set of simultaneous linear equations, as in (4.7), requires that the data covariance matrix, or equivalently the data, be orthogonalized. Once the data has been orthogonalized, the cost of multiple iterations is not much higher than a single iteration. Loosely speaking, once the covariance matrix $\mathbf{R}_{xx}$ has been inverted, each update only requires a demodulation, cross-correlation, and matrix-vector multiply. In practice, of course, a matrix inverse is rarely explicitly calculated.

An aside is in order here on the computational complexity of the LSDDA. Direct minimization of MSE by solving a set of simultaneous linear equations is often considered impractical for real-time applications because of the computational load. Certainly this can be true for adaptive equalization applications, where the number of filter coefficients can be large. However, the computational load is more reasonable for beamforming applications since adaptation of a relatively small number of coefficients is required.

An example of LSDDA convergence is shown in Figure 4.1. This figure shows the mean LSDDA output SINR versus the number of weight vector updates for one particular environment. The environment consists of a QPSK desired signal and a single Gaussian interfering signal received at a 4 element uniform linear array. The antenna spacing is equal to $\lambda/2$, where λ is the wavelength corresponding to the QPSK carrier frequency. The QPSK signal is incident from 10° off broadside, and the Gaussian interferer is incident from 65°. Spatially and temporally white complex Gaussian background noise is also present in the observed data. The desired signal is 6.3 dB stronger than the background noise, and the Gaussian interferer is 6 dB stronger than the desired signal. (In the remainder of this chapter we

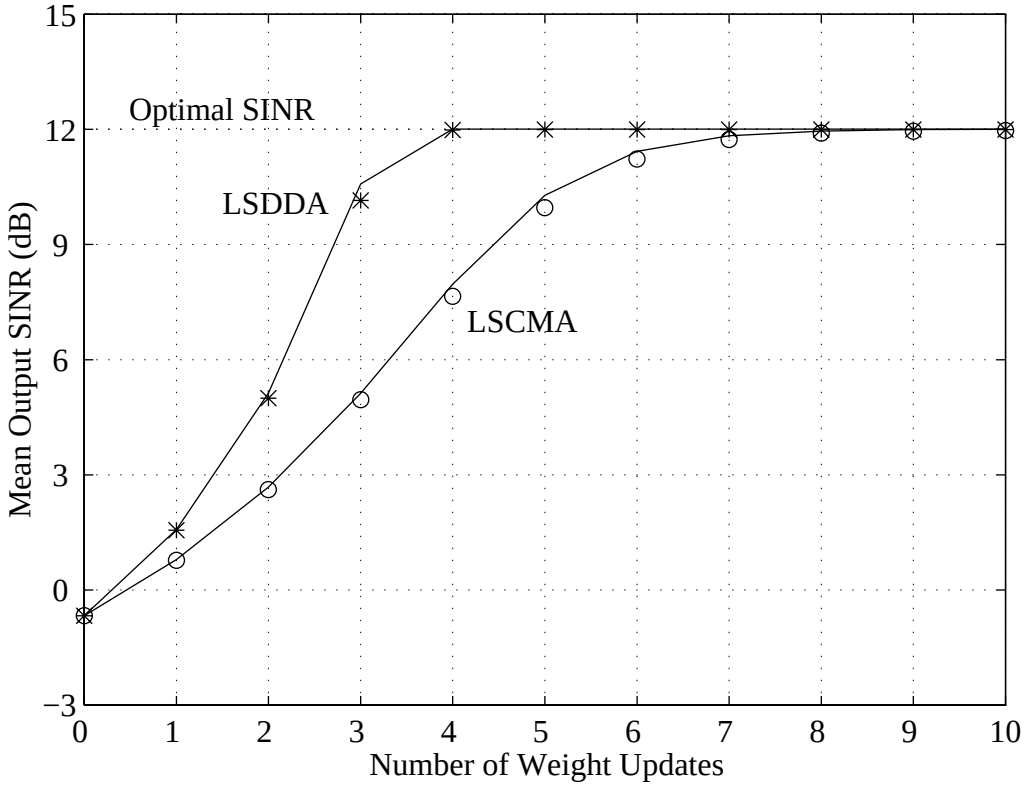

Figure 4.1: Mean output SINR of LSDDA and LSCMA versus the number of weight vector updates. Solid lines show theoretically predicted output SINR, the '$*$' and '$\circ$' show results from simulation.

will denote signal strength relative to the white background noise as the Signal to White Noise Ratio, or SWNR). The optimal output SINR is 12.0 dB. The initial weight vector is $\mathbf{w}_0 = [\,1\ 0\ 0\ 0\,]^T$. This gives an initial output SINR of $\approx$ -0.67 dB. The corresponding symbol error rate is approximately 0.18. The LSDDA block size is 1024 symbols, and results are based on 1000 independent trials. The results of each trial are based on multiple LSDDA iterations applied to the same block of data. The figure shows that the LSDDA has converged after 4 iterations. It was found that all 1000 trials converged to the desired solution, even though the initial SINR is very low.

Figure 4.1 also shows results for the LSCMA. The LSCMA weight update is nearly identical to the LSDDA update described by (4.3),(4.4), and (4.7). The difference is that LSCMA uses

$$c_k(n) = \frac{y_k(n)}{|y_k(n)|} \tag{4.18}$$

in place of hard decision $d_k(n)$. Figure 4.1 shows that LSDDA converges more quickly than

the LSCMA, at least in this particular environment. However, it may be misleading to conclude that LSDDA is always to be preferred over LSCMA. First, the carrier frequency of the desired signal will not always be known. Second, the LSDDA has knowledge of the received signal carrier frequency and carrier *phase*, so the LSDDA results in 4.1 are optimistic.

The main goal of this chapter is to predict and understand the observed behavior of the LSDDA. To show that progress has been made towards reaching this goal, Figure 4.1 includes an approximate theoretical prediction for the mean LSDDA output SINR at each iteration based on results derived in this chapter. (The theoretical prediction is more accurately described as an upper bound on the mean LSDDA output SINR. This bound grows tighter as the block size N grows larger.) The theoretical approximation to the observed behavior is very good. The approach used to obtain this theoretical approximation is described in the next section.

4.3 Analysis Framework

In this section we describe the general framework used to analyze the LSDDA. This framework has also been used to analyze the LSCMA [63, 7]. Central to this framework is an interpretation of demodulation as a non-linear operator applied to the beamformer output. We essentially describe a simple way to measure the SINR of $d_k(n)$. We then relate the SINR of $d_k(n)$ to the output SINR achieved with the updated beamformer weight vector $\mathbf{w}_{k+1}$. In general, the updated output SINR is dependent on the SINR of $d_k(n)$, the initial SINR, and the optimal output SINR. The optimal output SINR is in turn dependent on many factors, including the array geometry, the number of antennas, the number of incident signals, and the angle of arrival of each signal.

Our approach is to determine the expected values of $\hat{\mathbf{R}}_{xx}$ and $\hat{\mathbf{r}}_{xd}$ and insert these expressions into (4.7). This gives the asymptotic (as $N \to \infty$) value of $\mathbf{w}_{k+1}$. The corresponding output SINR is then calculated. This yields the asymptotic output SINR of the updated weight vector. A clear distinction must be made between the *mean* and the *asymptotic* output SINR of $\mathbf{w}_{k+1}$. The mean output SINR is defined as the ensemble average output SINR observed over multiple independent realizations of the data for finite N. The mean output SINR will be dependent on N and the number of antennas in the array. In contrast, the asymptotic output SINR is *not* dependent on N or on the number of antennas. However, the asymptotic output SINR is shown to be very close to the mean output SINR observed in simulations when N is significantly greater than the number of antennas. An exact determination of the mean output SINR for finite N would be much more complicated than the approach used here. An analogy can be made to the use of a training sequence in a

classical Least Squares (LS) approach. The use of a finite duration training sequence means that the output SINR of the LS beamformer will be strictly less than the optimal value. As a general rule of thumb, the duration of the training sequence must be approximately twice the number of antennas in order to achieve reasonable performance [28].

The beamformer output signal at the kth iteration can be expressed as

$$\begin{aligned} y_k(n) &= \mathbf{w}_k^H \mathbf{x}(n) && (4.19)\\ &= \mathbf{w}_k^H \mathbf{a} + \mathbf{w}_k^H \mathbf{q}(n) && (4.20)\\ &= \sqrt{\rho_k}\, e^{j\theta} s(n) + z(n) && (4.21) \end{aligned}$$

where $\sqrt{\rho_k}$ is the real amplitude of the signal in the beamformer output, θ is the output signal phase, and $z(n) = \mathbf{w}_k^H \mathbf{q}(n)$ is the residual noise plus interference term. Note that we can scale $y_k(n)$ in any convenient manner because multiplying the beamformer output by any *real* scalar has no effect on the PSK demodulator output. That is, if $f(\cdot)$ denotes the demodulation function for a PSK signal,

$$f(cy(n)) = f(y(n)) \quad \forall \quad \text{real} \quad c \tag{4.22}$$

This would not be the case for, e.g., a QAM demodulator. Since we limit our analysis to PSK signals, we assume in the remainder that the beamformer output has been normalized so that $z(n)$ has unit variance. This can be written as

$$\sigma_z^2 = \mathbf{w}^H \mathbf{R}_{qq} \mathbf{w} = 1 \tag{4.23}$$

Thus the beamformer output SINR is simply ρ_k .

The next step in the LSDDA is to demodulate $y_k(n)$. Clearly demodulation is a non-linear operation. In general, when a signal plus additive noise is passed through a non-linearity, the output will contain the signal, the noise, and intermodulation products of the signal and noise. The demodulator output may therefore be partially correlated with the signal and partially correlated with the interference. The non-linear operation may also create components that are correlated with neither the signal nor the interference. Thus we allow a model for $d_k(n)$ that may contain three components: (1) one component which is correlated with the desired signal; (2) one component which is correlated with the interference; and (3), one component which is correlated with neither the signal nor the interference. We model the demodulator output as

$$d_k(n) = \alpha s(n) + \beta z(n) + \xi(n) \tag{4.24}$$

where the scalars α and β are the complex amplitude of the signal and residual noise and interference components in the demodulator output, respectively, and $\xi(n)$ contains those

terms that are uncorrelated with both $s(n)$ and $z(n)$. In the model (4.24), the term $\xi(n)$ is by definition uncorrelated with both the signal and the interference. In order to find α and β, we calculate the cross-correlation of $s(n)$ and $z(n)$, respectively, with $d(n)$. Note that

$$\begin{aligned} R_{sd} &\triangleq \mathcal{E}\{\, s(n)d^*(n)\,\} && (4.25)\\ &= \mathcal{E}\{\, s(n)\left(\alpha s(n)+\beta z(n)+\xi(n)\right)^*\,\} && (4.26)\\ &= \alpha^* && (4.27)\end{aligned}$$

Similarly we have

$$R_{zd} \triangleq \mathcal{E}\{\, z(n)d^*(n)\,\} = \beta^* \tag{4.28}$$

The SINR ρ in the demodulator output d_k is

$$\rho\left(d_k\right) = \frac{\left|\alpha\right|^2}{\left|\beta\right|^2} = \frac{\left|R_{sd}\right|^2}{\left|R_{zd}\right|^2} \tag{4.29}$$

since we model $s(n)$ and $z(n)$ as having unit variance.

We now seek to relate the SINR of d_k to the output SINR of the updated LSDDA weight vector $\mathbf{w}_{k+1}$. Let us initially assume that no background noise is present, and that the number of received signals is less than the number of antennas. In this case the beamformer has sufficient degrees of freedom to place any desired gain on the interference, independent of the gain on the signal. The interference can be completely removed by the beamformer, and the optimal beamformer output SINR is infinite. As the block size $N \to \infty$, the updated weight vector $\mathbf{w}_{k+1}$ minimizes the MSE between $y_{k+1}(n)$ and $d_k(n)$,

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \left|y_{k+1}(n) - d_k(n)\right|^2 = \mathcal{E}\Big\{\, \left|y_{k+1}(n) - d_k(n)\right|^2 \,\Big\} \tag{4.30}$$

We can express the updated beamformer output as

$$y_{k+1}(n) = \mathbf{w}_{k+1}^H \mathbf{x}(n) = \alpha' s(n) + \beta' z(n) \tag{4.31}$$

which, together with (3.12), allows the MSE to be written as

$$\begin{aligned} \mathcal{E}\Big\{\, \left|y_{k+1}(n) - d_k(n)\right|^2 \,\Big\} &= \mathcal{E}\Big\{\, \left|(\alpha'-\alpha)s(n) + (\beta'-\beta)z(n) - \xi(n)\right|^2 \,\Big\} && (4.32)\\ &= \left|\alpha'-\alpha\right|^2 \sigma_s^2 + \left|\beta'-\beta\right|^2 \sigma_z^2 && (4.33)\end{aligned}$$

where we have made use of the fact that $s(n)$, $z(n)$, and $\xi(n)$ are mutually uncorrelated. Clearly the MSE is minimized for $\alpha' = \alpha$, $\beta' = \beta$. This implies that the signal component in the updated beamformer output will match the magnitude and *phase* of the signal component in the demodulator output. This also implies that the MSE between $d_k(n)$ and the updated

beamformer output $y_{k+1}(n)$ is minimized when the SIR of $y_{k+1}(n)$ is equal to the SIR of $d_k(n)$ (4.29). Thus, the output SIR of the updated LSDDA weight vector can be determined from R_{sd} and R_{zd}. This in turn requires that the probability density function (pdf) of the signal and the interference be known.

When background noise is present, the interference and noise cannot be completely removed by beamforming. Independent thermal noise generated by each of the M receivers required for the M antennas in the array is a common source of background noise. The relationship between the SINR in $d_k(n)$ and the updated LSDDA output SINR is then more complicated. We show in [7] that the output SINR of the updated LSDDA weight vector is

$$\rho_{k+1} = \frac{\left\{\gamma\left(\frac{G-1}{1+\gamma}\right)+1\right\}^2 \rho_k}{\gamma\rho_k\left(\frac{G-1}{1+\gamma}\right)^2 + 2\rho_k\left(\frac{G-1}{1+\gamma}\right)+1} \tag{4.34}$$

where

$$\rho_k \triangleq \frac{\left|\mathbf{a}^H\mathbf{w}_k\right|^2}{\mathbf{w}_k^H\mathbf{R}_{qq}\mathbf{w}_k} \tag{4.35}$$

is the initial beamformer output SINR and $\mathbf{w}_k$ is the initial weight vector. The optimal output SINR γ is given by

$$\gamma \triangleq \mathbf{a}^H\mathbf{R}_{qq}^{-1}\mathbf{a} \tag{4.36}$$

The noise and interference matrix $\mathbf{R}_{qq}$ has full rank, since the optimal output SINR would otherwise be infinite. Finally,

$$G^2 \triangleq \frac{|\alpha|^2}{|\beta|^2\rho_k} \tag{4.37}$$

is the SINR gain provided by the demodulator.

We argued earlier that the output SINR of the updated LSDDA beamformer is equal to the SINR in $d_k(n)$ if no background noise is present. This can be verified by letting the optimal output SINR γ approach infinity in (4.34). It is straightforward to show that

$$\lim_{\gamma\to\infty}\rho_{k+1} = G^2\rho_k = \frac{|\alpha|^2}{|\beta|^2} \tag{4.38}$$

which supports the argument made earlier. We would also expect that as the SINR of $d_k(n)$ grows large, the output SINR of the updated weight vector will approach the optimal output SINR. This can be verified by letting G as defined in (4.37) approach infinity in (4.34). This yields

$$\lim_{G\to\infty}\rho_{k+1} = \gamma \tag{4.39}$$

as expected.

In the material to follow we determine the SINR gain G^2 provided by demodulating a BPSK, QPSK, and M-ary PSK signal received in Gaussian noise and interference. We also find the SINR gain and the output phase when the carrier phase is not known at the receiver.

4.4 LSDDA Analysis by Swindlehurst et al.

It is argued in [9] that the LSDDA will converge in one iteration as $N \to \infty$. Since we draw a distinctly different conclusion, it is important to review the analysis of [9] and identify the key differences in these two approaches. To summarize, the analysis presented in [9] implicitly assumes that any decision errors are uncorrelated with the noise. In contrast, we implicitly assume that the decision errors are *correlated* with the noise.

To begin, we first demonstrate through simulation that the LSDDA may require more than one iteration for convergence, even as the block size $N \to \infty$. We re-examine the behavior of the LSDDA in the environment corresponding to Figure 4.1. Recall that Figure 4.1 shows the LSDDA output SINR versus the number of weight vector updates with the block size $N = 1024$ QPSK symbols. Figure 4.2 shows the distribution of the LSDDA output SINR after one update, with the block size varied from 8 symbols to 1024 symbols. These results are based on 10,000 independent trials. The figure shows that the LSDDA output SINR after one update asymptotically approaches 1.57 dB, which is much less than the optimal output SINR of 12.0 dB.

Let us now review the approach used in [9]. The output of the demodulator is modeled as

$$d(n) = s(n) + e(n) \tag{4.40}$$

where $s(n)$ is an M-ary PSK signal and $e(n)$ is the error signal. When a correct decision is made, $d(n) = s(n)$, and $e(n) = 0$. It is then noted that, "When an error occurs, with high probability it will be because the symbol was associated with an immediately adjacent point on the signal constellation", in which case

$$d(n) = e^{\pm j2\pi/M} s(n) \tag{4.41}$$

Thus, when an error occurs, the error signal can be written as

$$e(n) = \left(e^{\pm j2\pi/M} - 1\right) s(n) \tag{4.42}$$

Continuing to follow [9], if we let b denote the probability of a symbol error, and assume that each of the two most likely demodulation errors is equally likely, we have

$$e(n) = \Delta(n) s(n) \tag{4.43}$$

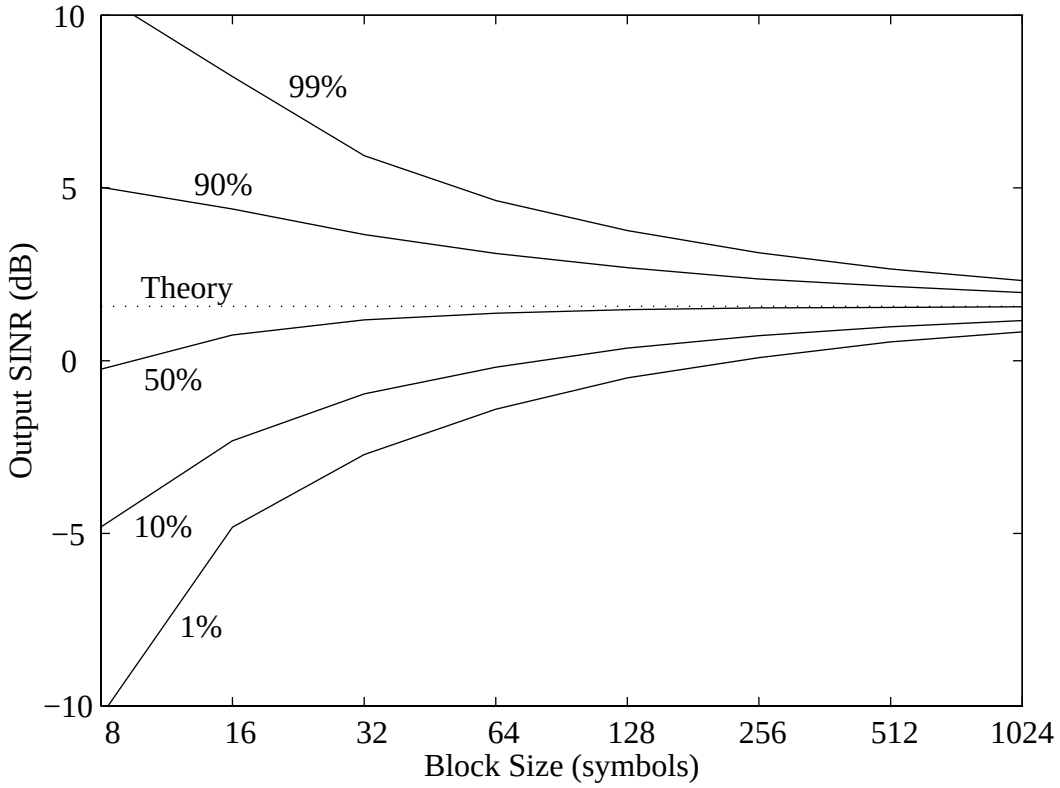

Figure 4.2: Empirical distribution of the LSDDA output SINR versus block size. The environment contains a QPSK desired signal with Gaussian interference. The asymptotic output SINR predicted by analysis is -1.94 dB.

where

$$\Delta(n) = \begin{cases} 0 & \text{with probability} \quad 1-b \\ \left(e^{+j2\pi/M} - 1\right) & \text{with probability} \quad b/2 \\ \left(e^{-j2\pi/M} - 1\right) & \text{with probability} \quad b/2 \end{cases} \tag{4.44}$$

In order for the updated weight vector $\mathbf{w}_{k+1}$ to converge to the optimal solution as $N \to \infty$, we must have

$$\lim_{N\to\infty} \hat{\mathbf{R}}_{xd} = c\mathbf{a} \tag{4.45}$$

where $\mathbf{a}$ is the spatial signature of the desired signal and c can be any non-zero scalar. We have, using the expression (4.43),

$$\begin{aligned} \mathbf{R}_{xd} &= \mathcal{E}\{\, \mathbf{x}(n)d^*(n)\,\} && (4.46)\\ &= \mathcal{E}\{\, (\mathbf{a}s(n) + \mathbf{q}(n))\,(s(n) + e(n))^*\,\} && (4.47)\\ &= \mathcal{E}\{\, (\mathbf{a}s(n) + \mathbf{q}(n))\,(s^*(n) + \Delta^*(n)s^*(n))\,\} && (4.48)\\ &= \mathbf{a} + \mathbf{a}\mathcal{E}\{\,\Delta(n)\,\} + \mathcal{E}\{\,\mathbf{q}(n)\Delta^*(n)s^*(n)\,\} && (4.49) \end{aligned}$$

where we have assumed that $s(n)s^*(n) = 1$, and that the signal $s(n)$ and the interference $q(n)$ are uncorrelated. The term involving $\mathcal{E}\{\,\Delta(n)\,\}$ is not of interest, since this is proportional to $\mathbf{a}$. The term of critical interest is

$$\mathcal{E}\{\,\mathbf{q}(n)\Delta^*(n)s^*(n)\,\} \tag{4.50}$$

If this term is equal to zero, then $\lim_{N\to\infty} \hat{\mathbf{R}}_{xd} = c\mathbf{a}$. If this term is non-zero, the noise and interference contributes to $\mathbf{R}_{xd}$, and more than one update will be required even for $N \to \infty$. It is argued in [9] that, " while $\mathbf{q}(n)$ may be correlated with $\Delta(n)$, the signal $s(n)$ is uncorrelated with both $\Delta(n)$ and $\mathbf{q}(n)$", hence (4.50) will be zero. However, by definition (4.44),

$$\Delta(n) = \frac{e(n)}{s(n)} \tag{4.51}$$

This leads to

$$\begin{aligned} \mathcal{E}\{\,\mathbf{q}(n)\Delta^*(n)s^*(n)\,\} &= \mathcal{E}\left\{\, \mathbf{q}(n)\frac{e^*(n)}{s^*(n)}s^*(n)\,\right\} && (4.52)\\ &= \mathcal{E}\{\,\mathbf{q}(n)e^*(n)\,\} && (4.53) \end{aligned}$$

This shows that assuming that (4.50) is equal to zero is equivalent to assuming that the error signal $e(n)$ is uncorrelated with the noise $\mathbf{q}(n)$. The analysis framework presented here in Section 4.3 does not explicitly define an error sequence. However, we will show that we

implicitly assume that the error signal $e(n)$ is *correlated* with the noise $\mathbf{q}(n)$. Using our model (3.12) for $d(n)$, we have

$$\begin{aligned} e(n) &= d(n) - s(n) && (4.54)\\ &= (\alpha - 1)s(n) + \beta \mathbf{w}^H \mathbf{q}(n) + \xi(n) && (4.55) \end{aligned}$$

and

$$\begin{aligned} \mathcal{E}\{\, \mathbf{q}(n)e^*(n)\,\} &= \mathcal{E}\Big\{\, \mathbf{q}(n)\left((\alpha^* - 1)s^*(n) - \beta^* \mathbf{q}^H(n)\mathbf{w} + \xi^*(n)\right)\Big\} && (4.56)\\ &= \beta^* \mathbf{R}_{qq}\mathbf{w} && (4.57)\\ &\neq 0 && (4.58) \end{aligned}$$

4.5 PSK Signal with Gaussian Interference

In this section we examine the behavior of the LSDDA when the desired signal is PSK and the interference is Gaussian. Results are also presented for the LSCMA for comparison. BPSK, QPSK, and general M-ary PSK are considered. Clearly it is unlikely that a single Gaussian interferer will be encountered in practice, except perhaps in an anit-jam application. However, the performance of the LSDDA is not determined by the distribution of each individual interferer, but by the distribution of the *sum* of all interferers plus the background noise. The distribution of a sum of many co-channel interferers will tend asymptotically towards a Gaussian distribution, even if each interferer has a constant modulus.

4.5.1 BPSK Signal

Before we address the behavior of the LSDDA with a BPSK signal, we note that the conjugate SCORE algorithm [54, 55] is a much more effective algorithm for extracting a BPSK signal with known carrier frequency. The conjugate SCORE algorithm exploits the conjugate cyclostationarity of the BPSK signal exhibited at twice the carrier frequency. This property is manifested when a BPSK signal is squared, causing a spectral line to be regenerated at twice the carrier frequencny. The SCORE algorithm is guaranteed to converge to the optimal solution, except for finite data effects. Given that SCORE is the perferred algorithm, it may be questioned whether the analysis of LSDDA for a BPSK signal is even necessary. One motivation for this analysis is that general insight into the LSDDA are obtained. Another motivation is that the analysis of BPSK must be included for completeness and to compare to results obtained for QPSK and general M-ary PSK.

We now examine the behavior of the LSDDA with a BPSK signal and Gaussian interference. Before the algorithm has converged, the beamformer output will contain both the desired signal and the interference. Thus the beamformer output is proportional to

$$y(n) = \sqrt{\rho_k}\, s(n) + z(n). \tag{4.59}$$

The BPSK desired signal $s(n)$ takes on the values $+1$ and -1 with equal probability. We assume initially that the signal carrier phase is known to the receiver, so that $\sqrt{\rho_k}$ is *real*. The interferer $z(n)$ is complex, circularly symmetric Gaussian with variance equal to unity. Thus the real and imaginary parts of $z(n)$ are independent, Gaussian distributed random variables with each having variance equal to $1/2$. The hard decisions $d(n)$ are given by

$$d(n) = \text{sign}(\, y_r \,) = \text{sign}(\, \sqrt{\rho_k}\, s(n) + z_r(n) \,) \tag{4.60}$$

where y_r is the real component of y and z_r is the real component of z. We show in Appendix C that the cross correlation R_{sd} of the BPSK signal s with the demodulator output d is

$$R_{sd} = \text{erf}(\sqrt{\rho_k}). \tag{4.61}$$

Appendix C also shows that the cross correlation R_{zd} of the Gaussian noise z with the demodulator output d is

$$R_{zd} = \frac{1}{\sqrt{\pi}} e^{-\rho_k}\,. \tag{4.62}$$

Using (4.29) together with (4.61) and (4.62), the SINR of the hard decisions $d_k(n)$ is

$$\frac{R_{sd}^2}{R_{zd}^2} = \pi\, e^{\,2\rho_k}\ \left(\text{erf}(\sqrt{\rho_k})\right)^2. \tag{4.63}$$

If the SINR at the demodulator output is higher than the SINR at the demodulator input, the LSDDA beamformer output SINR will improve with each iteration, and the LSDDA will converge. Thus we are interested in the SINR gain (4.37) provided by BPSK demodulation, which is

$$G^2 = \frac{R_{sd}^2}{R_{zd}^2 \rho_k} \tag{4.64}$$

$$= \frac{\pi}{\rho_k}\, e^{\,2\rho_k}\ \left(\text{erf}\,(\sqrt{\rho_k})\right)^2. \tag{4.65}$$

The SINR gain (4.65) grows large as the initial SINR ρ_k increases, since $\text{erf}(\sqrt{\rho_k}) \to 1$ as $\sqrt{\rho_k}$ grows large. This makes sense since the probability of symbol error approaches zero as $\sqrt{\rho_k}$ grows large. For small $\sqrt{\rho_k}$,

$$R_{sd} = \text{erf}(\sqrt{\rho_k}) \approx 2\sqrt{\frac{\rho_k}{\pi}}. \tag{4.66}$$

We also have that $R_{zd} \approx \frac{1}{\sqrt{\pi}}$. Therefore the SINR gain for a low SINR BPSK signal is

$$\text{SINR}_{gain}\Big|_{g \ll 1} \approx 4. \tag{4.67}$$

This indicates that, even for very low initial SINR, each LSDDA update will increase the output SINR by least 6 dB when the desired signal is BPSK. (This assumes that no background noise is present. When background noise is present, the beamformer output SINR can never exceed the optimal SINR, and the 6 dB improvement per update may not be possible). This is a direct consequence of the fact that the Gaussian noise is *complex* while the desired BPSK signal is *real*. That is, if the above derivation is repeated under the assumption that the noise and interference is real with unit variance, the BPSK SINR gain approaches unity as the initial SINR grows small. We note that modeling the noise as complex rather than real is more appropriate. It is difficult to imagine a scenario where all noise and interference would be in-phase with the desired BPSK signal.

We must emphasize that this 6 dB minimum gain per iteration with a BPSK signal was derived under the assumption that the *carrier phase is known* to the receiver. If the carrier phase is not known, the LSDDA can behave very differently, as discussed in Subsection 4.6.

The SINR gain function for BPSK (4.65) is shown in Figure 4.3 as a function of input SINR. The gain at low SINR approaches 6 dB as $\rho_k \to 0$, as predicted by the low SINR approximation given above. Furthermore the SINR gain increases very rapidly as the input SINR increases, as would be expected. The figure also includes results from Monte Carlo simulation. The simulation generates a noisy set of BPSK symbols, demodulates the symbols according to (4.60), and calculates the cross-correlation of the BPSK symbols and the Gaussian noise with the demodulator output. A simplified block diagram of the simulation is shown in Figure 4.4. The simulation results are based on 1000 independent trials and a sample size of $N = 1024$ symbols. Figure 4.3 shows very good agreement between the theoretical and measured SINR gain provided by the demodulator.

We now show that the BPSK SINR gain (4.65), along with (4.34), can be used to predict the output SINR of the LSDDA. We consider an environment similar to the QPSK environment corresponding to Figure 4.1 and Figure 4.2. The only difference is that the QPSK signal is replaced with a BPSK signal. The theoretical output SINR at each iteration k is found by the following procedure. Given the initial output SINR ρ_0,

1. Calculate the BPSK SINR gain (4.65), which is a function of ρ_k;
2. Calculate the updated output SINR ρ_{k+1} (4.34);
3. Let $k = k + 1$ and return to Step #1.

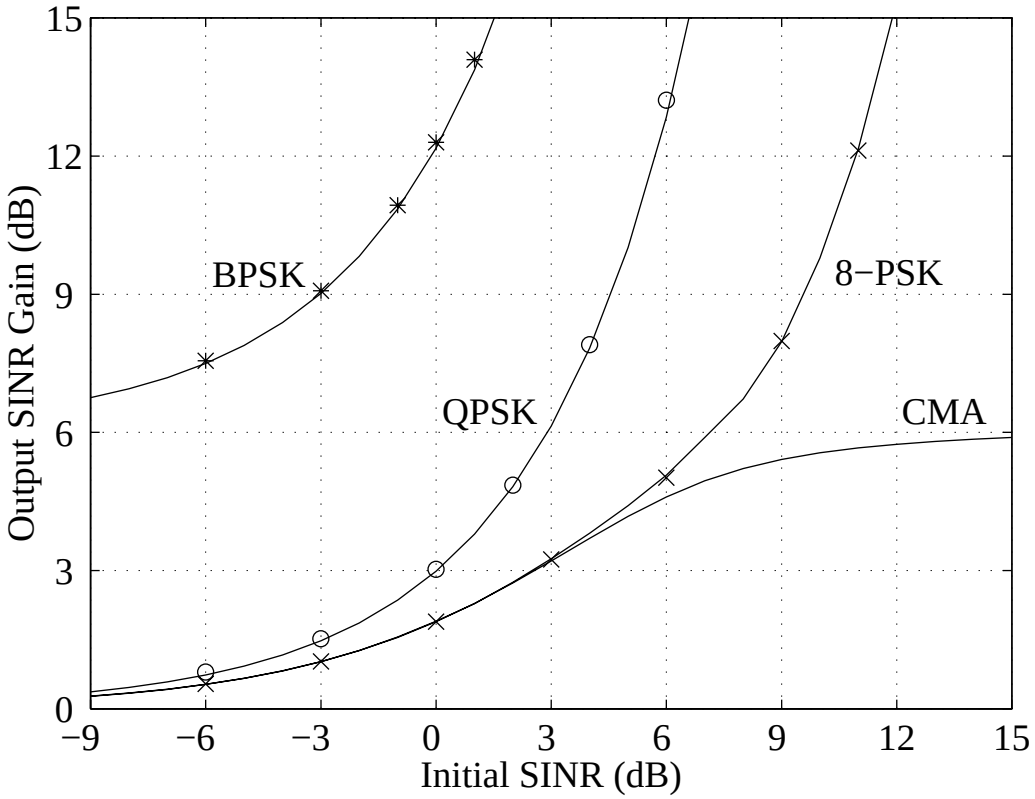

Figure 4.3: Improvement in output SIR achieved with one iteration of LSDDA with a PSK desired signal and Gaussian interference. Solid lines denote gain predicted by analysis, '$*$','$\circ$', and '$\times$' denote mean gain measured from simulation. Gain achieved with CMA shown for comparison.

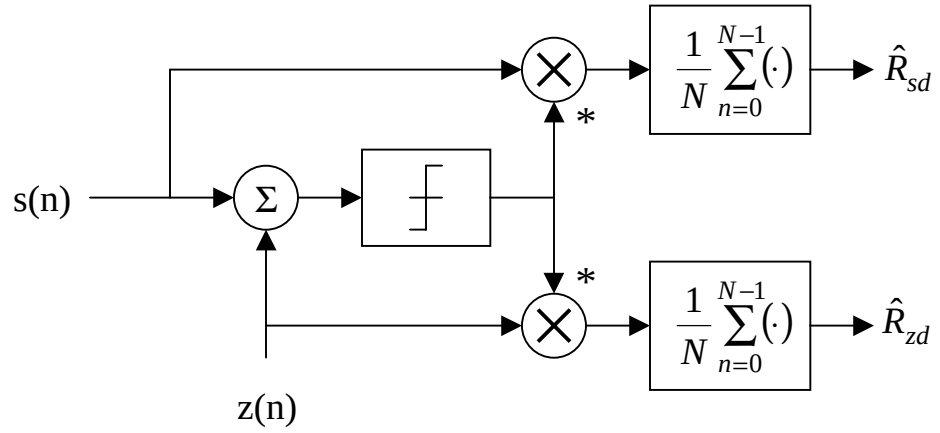

Figure 4.4: Block diagram of the simulation used to support SINR gain calculations.

Figure 4.6 shows the predicted output SINR of the LSDDA versus the number of weight vector updates. Results are shown for several different values of initial SINR ρ_0. Supporting results from simulation are also included. The simulation results are based on $N = 1024$ bits and 1000 independent trials. The agreement between the theoretical asymptotic output SINR and the mean output SINR from simulation is very good *for this block size*. An important simulation issue is the method used to calculate an initial weight vector $\mathbf{w}_0$ that achieves a desired output SINR ρ_0. There can be an infinite number of weight vectors that achieve a certain sub-optimal output SINR. The method used here is described in Appendix B. Some examples of the corresponding initial beampatterns are shown in Figure 4.5, which includes the optimal beampattern for comparison.

Figure 4.6 shows the mean output SINR of the LSDDA. The variance and distribution of the LSDDA output SINR is also of interest. Calculation of these parameters is beyond the scope of this dissertation. However, simulation results are presented in Figure 4.7 that illustrate the distribution of the output SINR as the LSDDA converges in the BPSK environment. This figure shows the empirical 90% confidence interval for the output SINR whose mean is shown in Figure 4.6. A true confidence interval would be based on theoretical considerations, so that the empirical nature of these results must be emphasized. The LSDDA is initialized with the same weight vector over all 1000 interations, so there is no variance in the initial output SINR. One trend that is evident in Figure 4.6 is that the output SINR variance increases as the initial SINR decreases. Also note that the variance decreases as the LSDDA nears convergence. The variance at convergence would be expected to be very close to the variance achieved with a known training sequence. We would also expect the variance to decrease with increasing block size, as demonstrated in Figure 4.2 for the QPSK environment.

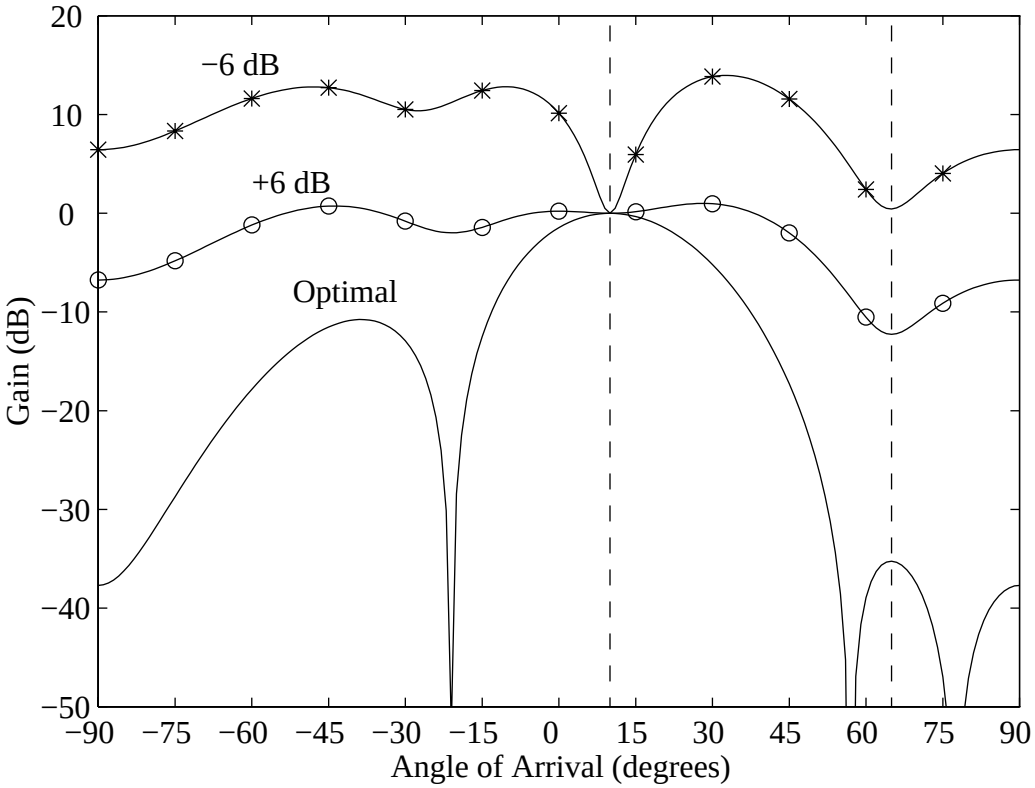

Figure 4.5: Beampatterns used to obtain certain initial output SINR.

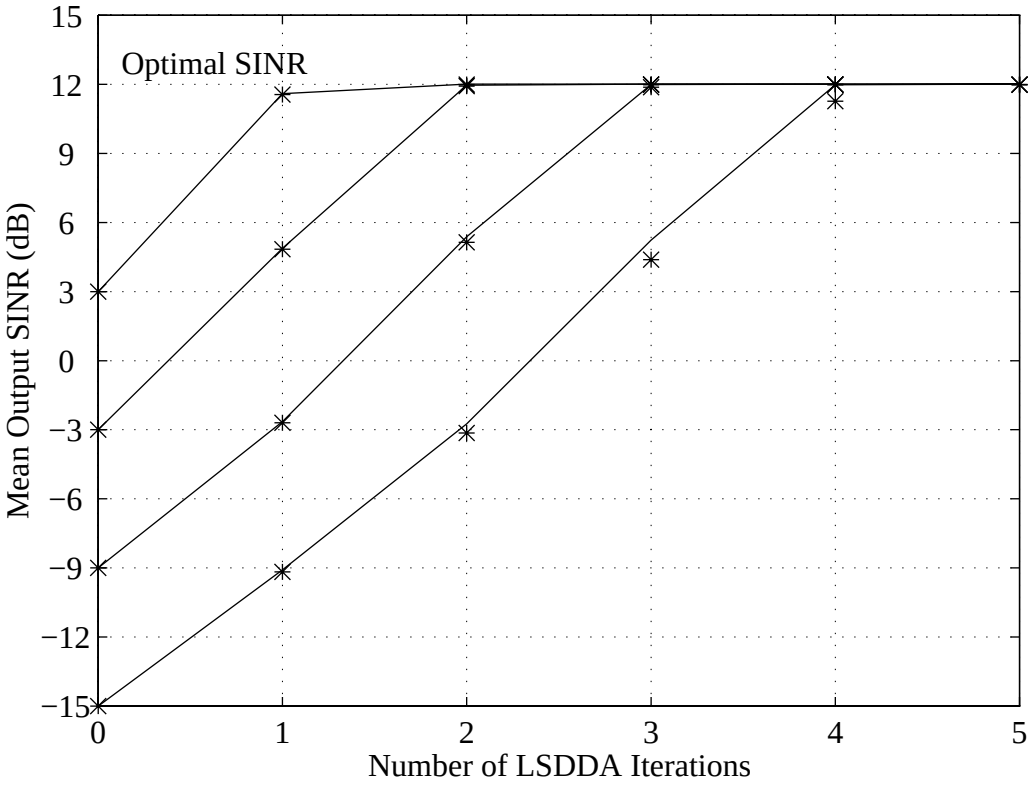

Figure 4.6: Theoretical output SINR of the LSDDA beamformer versus the number of iterations. The signal is BPSK and the background noise and interference is Gaussian. Solid lines show theoretically predicted output SINR, '$*$' show results from simulation.

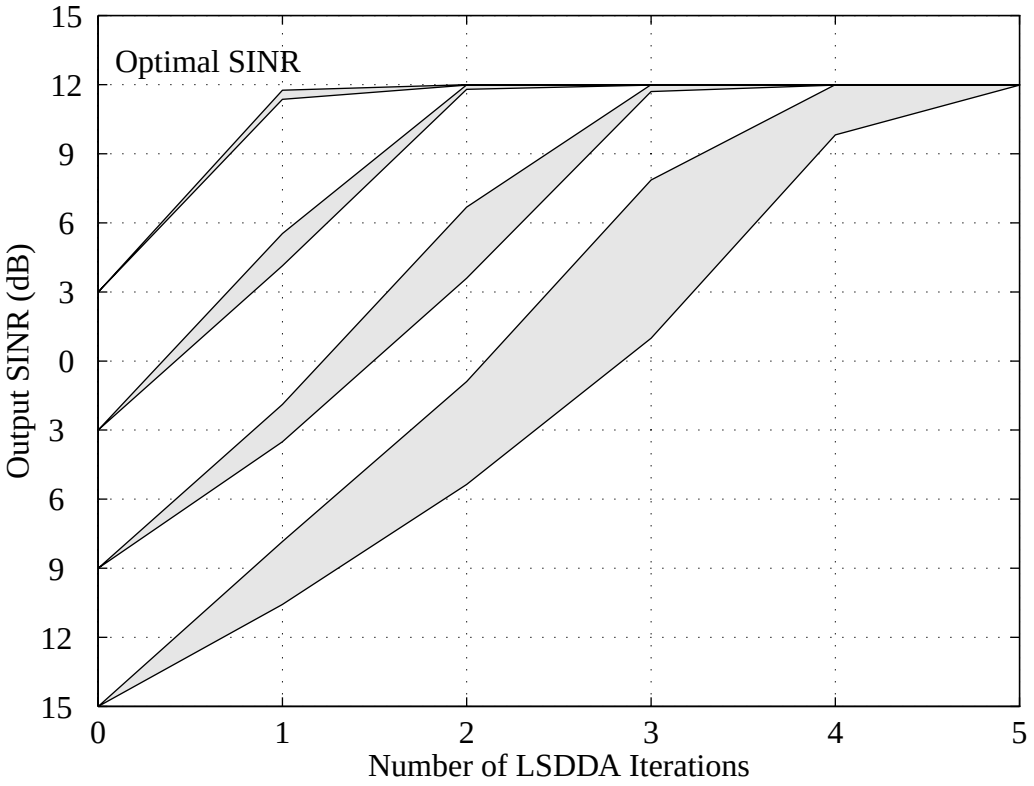

Figure 4.7: Output SINR of the LSDDA with a BPSK signal. The shaded regions show where 90% of the 1000 trials fell.

4.5.2 QPSK Signal

We now consider the case where the signal is QPSK and the interference is Gaussian. The QPSK signal is modeled by

$$s = e^{\,j(k\pi/2 + \pi/4)} \tag{4.68}$$

where k is an integer random variable taking on the values 0, 1, 2, or 3 with equal probability. With the QPSK signal defined as above, the hard decisions $d(n)$ are given by

$$d = \text{sign}(y_r) + j\,\text{sign}(y_i) \tag{4.69}$$

where y_r and y_i are the real and imaginary parts, respectively, of the beamformer output y. We show in Appendix D that the cross correlation R_{sd} of the desired signal and the hard decisions is

$$R_{sd} = \sqrt{2}\ \text{erf}\left(\sqrt{\rho_k\ /2}\right). \tag{4.70}$$

It is also shown that the cross correlation R_{zd} of the Gaussian noise and the hard decisions is

$$R_{zd} = \frac{2}{\sqrt{\pi}} e^{-\rho_k\ /2}. \tag{4.71}$$

The demodulator output SINR when the input is a QPSK signal plus Gaussian noise is obtained by substituting (4.70) and (4.71) into (4.29). This yields

$$\frac{R_{sd}^2}{R_{zd}^2} = \frac{\pi}{2} e^{\rho_k} \left(\text{erf}\left(\sqrt{\rho_k\ /2}\right)\right)^2 \tag{4.72}$$

The SINR gain for QPSK demodulation is

$$G^2 = \frac{\pi}{2\rho_k} e^{\rho_k} \left(\text{erf}\left(\sqrt{\rho_k\ /2}\right)\right)^2 \tag{4.73}$$

Figure 4.3 shows the QPSK SINR gain (4.73), along with supporting results from Monte Carlo simulation. The simulation results are obtained using the same method used to support the BPSK SINR gain calculations. Clearly the QPSK SINR gain becomes large as the input SINR increases, as would be expected. However, the QPSK gain is much lower than the BPQK gain, especially at low SINR. For small ρ_k , the QPSK output SIR gain is approximately

$$\text{SIR}_{gain}\Big|_{g \ll 1} \approx \frac{\pi}{2\rho_k} \left(\sqrt{\frac{\rho_k}{2}}\,\frac{2}{\sqrt{\pi}}\right)^2 \tag{4.74}$$

$$\approx 1. \tag{4.75}$$

Because the SINR gain for QPSK is very small for low input SINR, the LSDDA is susceptible to *noise capture* when extracting a QPSK signal. The noise capture properties of CMA have

been examined by several authors [80, 39]. It has been shown that noise capture corresponds to a saddlepoint in the CMA cost function. Noise capture can be viewed from the following perspective. Assume that the initial LSDDA weight vector nulls the incident signal so that the output is dominated by Gaussian noise. The SINR gain will be very small in such a case. A single LSDDA update causes very little change to the beamformer weight vector. Therefore the LSDDA may stay in this noise capture state indefinitely. By contrast, since the SINR gain for BPSK is always greater than 6 dB, noise capture is not possible with the LSDDA when applied to BPSK. As an aside, a noise capture weight vector lies in the signal nullspace of the observed data covariance matrix. Therefore in order to avoid noise capture it makes sense to use an initial weight vector which lies in the signal subspace, as proposed in [67].

Given that the QPSK gain is lower than the BPSK gain, but higher than the hard-limit gain, we would expect the LSDDA with a QPSK signal to converge more quickly than the LSCMA, but not as quickly as the LSDDA with a BPSK signal. Figure 4.8 verifies this anticipated result. Figure 4.8 shows the theoretical output SINR versus the number of LSDDA iterations for the QPSK environment. Results are shown for several different initial SINRs. The mean output SINR measured in simulation is very close to the asymptotic output SINR predicted by theory. The output SINR distribution observed in simulations is shown in Figure 4.9.

4.5.3 M-ary PSK

We now consider the case where the desired signal is M-ary PSK. The desired signal is modeled as

$$s = e^{j2\pi k/M} \tag{4.76}$$

where k is an integer random variable taking on the values $0 \leq k \leq M-1$ with equal probability and M is the number of symbols. Consider the case where the transmitted signal phase is equal to zero, so that $s = 1$. The demodulator input is then equal to

$$y(n) = \sqrt{\rho_k} + z_r(n) + jz_i(n) \tag{4.77}$$

where $\sqrt{\rho_k}$ is the signal amplitude. The cross-correlation of s with the hard decisions is given by

$$\mathcal{E}\{\, s\, d^* \mid s = 1 \,\} = \int_0^{2\pi} f^*(\phi)\, p_\phi(\phi)\, d\phi \tag{4.78}$$

where $f(\phi)$ is the PSK demodulation function expressed as a function of the received signal phase. We show in Appendix E that $p_\phi(\phi)$, the pdf of the phase of y, is given by

$$p_\phi(\phi) = \int_0^\infty r e^{-(r^2 - 2\sqrt{\rho_k} r\cos\phi + \rho_k)}\, dr \tag{4.79}$$

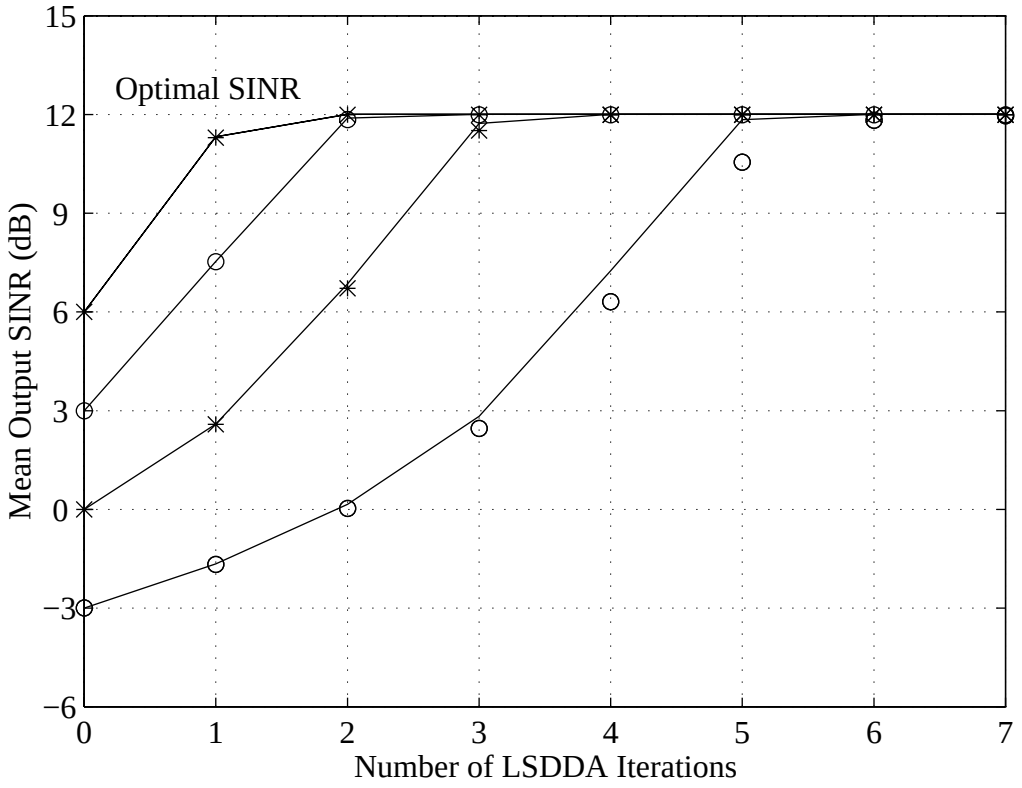

Figure 4.8: Theoretical output SINR of the LSDDA beamformer as a function of number of iterations. The desired signal is QPSK and the background noise and interference is Gaussian. Results are shown for several different values of initial output SINR.

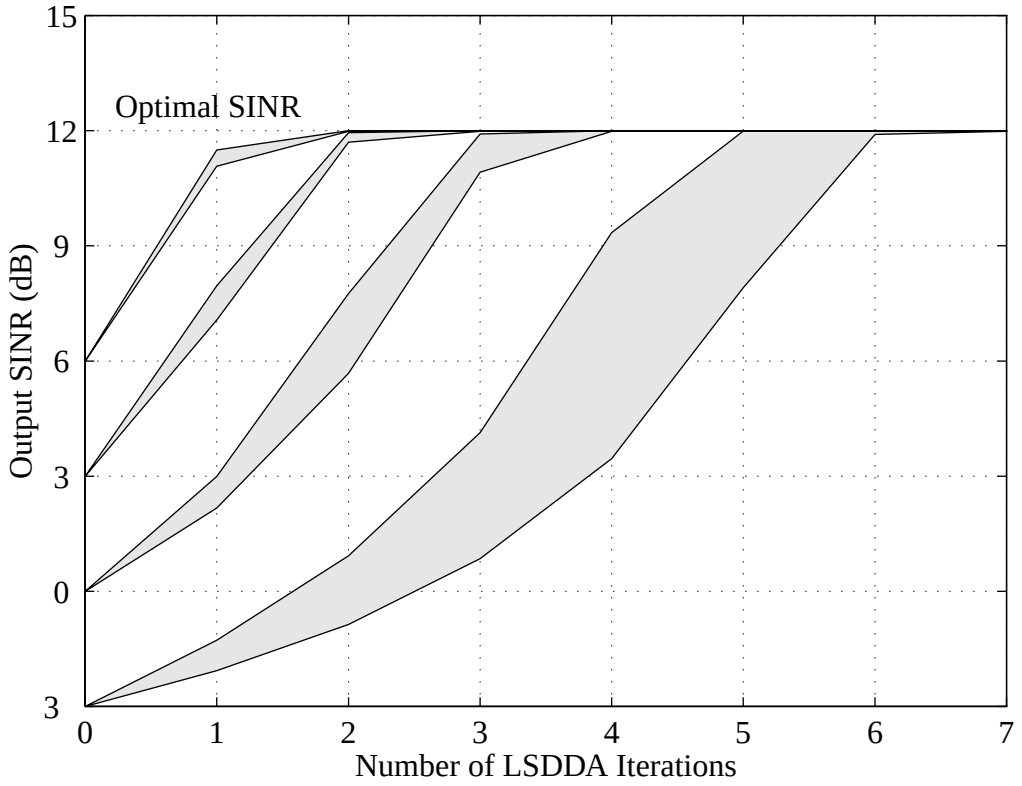

Figure 4.9: Output SINR of the LSDDA with a QPSK signal. The shaded regions show where 90% of the 1000 trials fell.

The PSK demodulation function can be defined as

$$f(\phi) = e^{j2\pi k/M} \quad \text{where} \quad \frac{2\pi k - \pi}{M} < \phi < \frac{2\pi k + \pi}{M}. \tag{4.80}$$

Thus

$$\mathcal{E}\{\, s\, d^* \mid s = 1 \,\} = \sum_{k=0}^{M-1} e^{-j2\pi k/M} \int_{(2\pi k-\pi)/M}^{(2\pi k+\pi)/M} p_\phi(\phi)\, d\phi \tag{4.81}$$

Since each symbol is equally likely, the cross-correlation R_{sd} is equal to the conditional cross-correlation given by (4.81).

We now consider the cross-correlation of z with the hard decisions d. Again consider the case where $s = 1$. We first describe the hard decisions as a function of the Gaussian interference z. We operate in polar coordinates and use $z = me^{j\theta}$ where m is the magnitude of z and θ is the angle of z. The angle of the demodulator input is

$$\angle y = \tan^{-1} \frac{y_i}{y_r} = \tan^{-1} \frac{m \sin\theta}{m\cos\theta + \sqrt{\rho_k}}. \tag{4.82}$$

The demodulation function is then given by

$$g\left(m, \theta, \sqrt{\rho_k}\right) = e^{j2\pi k/M} \quad \text{where} \quad \frac{2\pi k - \pi}{M} < \angle y < \frac{2\pi k + \pi}{M} \tag{4.83}$$

and k is an integer. The conditional cross-correlation can then be expressed as

$$\begin{aligned} \mathcal{E}\{\, z\, d^* \mid s = 1 \,\} &= \int_{-\pi}^{\pi} \int_0^{\infty} m e^{j\theta} g^*\left(m, \theta, \sqrt{\rho_k}\right)\, p_\theta(\theta)\, p_m(m)\, d\theta\, dm & (4.84) \\ &= \frac{1}{\pi} \int_{-\pi}^{\pi} \left\{ \int_0^{\infty} g^*\left(m, \theta, \sqrt{\rho_k}\right)\, m^2 e^{-m^2}\, dm \right\} e^{j\theta}\, d\theta & (4.85) \end{aligned}$$

Since each symbol is equally likely, the cross-correlation R_{zd} is equal to the conditional cross-correlation given by (4.85).

The SINR gain provided by 8-ary PSK demodulation (4.37) can be determined using numerical integration to evaluate (4.81) and (4.85). Results for 8-PSK are shown in Figure 4.3. Supporting results from Monte Carlo simulation are also shown. The simulation results and the analytic results predicted by (4.81) and (4.85) agree very well. The simulation approach is identical to that used for BPSK and QPSK signals, except that the desired signal is 8-ary PSK. Note that the SINR gain for 8-ary PSK falls between the gain for QPSK and the gain for hard-limiting. This is the expected result. As the number of symbols M in the PSK constellation increases, the process of M-ary demodulation becomes more similar to hard-limiting. The SINR gain for 8-ary PSK is within 1 dB of the SINR gain for hard-limiting up to ≈ 7 dB input SINR.

We expect the behavior of the LSDDA with an 8-ary PSK signal in Gaussian interference to be predictable in the same manner as previously shown for BPSK and QPSK signals. for brevity, full simulation results of the LSDDA with an 8-ary PSK signal are not included here.

4.5.4 Comparison with CMA

In this section we compare the performance of the LSDDA beamformer with the LSCMA beamformer. The convergence of the LSCMA has been examined in [63, 7]. As noted previously, the LSCMA and the LSDDA are very similar. The only difference is that LSCMA uses a hard-limit non-linearity, while LSDDA uses a demodulator.

The hard-limited beamformer output $c_k(n)$ was defined in (4.18). The cross-correlation R_{sc} of a constant modulus desired signal $s(n)$ with $c(n)$ can be shown to be [63, 7]

$$R_{sc} = \frac{1}{\pi} \int_0^{\infty} \int_{-\pi}^{\pi} \frac{\sqrt{\rho_k} + m \cos \Delta}{\sqrt{\rho_k \ + m^2 + 2m\sqrt{\rho_k} \cos \Delta}} \, d\Delta \, m e^{-m^2} \, dm \tag{4.86}$$

It can also be shown that the cross-correlation R_{zc} of the Gaussian interference $z(n)$ with the hard-limited beamformer output $c(n)$ is

$$R_{zc} = \frac{1}{\pi} \int_0^{\infty} \int_{-\pi}^{\pi} \frac{\sqrt{\rho_k}\, m \cos \Delta + m^2}{\sqrt{\rho_k \ + m^2 + 2m\sqrt{\rho_k} \cos \Delta}} \, d\Delta \, m e^{-m^2} \, dm \tag{4.87}$$

Both (4.86) and (4.87) are evaluated by numerical integration and used to predict the SINR gain shown in Figure 4.3. One important feature of the LSCMA is that the output SIR gain tends asymptotically to 6 dB as the input SIR becomes high.

From the figure we can see that the LSDDA always yields higher output SIR than the LSCMA. This is intuitive since the LSDDA exploits more information than the LSCMA. However, for QPSK and higher order PSK there is very little difference between the LSDDA and the LSCMA for low initial SINR. The expected difference in SINR gain is shown explicitly in Table 4.1. This is a significant result, since it shows analytically that there is little to be gained by using a decision directed algorithm over CMA at low SINR, even when the carrier frequency and phase are known perfectly. In practice, the carrier frequency and phase will not be known. This provides some justification for the common practice of using CMA to bootstrap a decision directed algorithm. In particular, CMA can be used to raise the output SINR to a point where the carrier frequency and symbol timing can be estimated accurately.

We will now present a direct comparison of the number of iterations required for convergence of LSDDA and LSCMA. The number of LSDDA iterations required for convergence is clearly

SIR_{in}	ΔG^2 (dB)	
	QPSK	8-PSK
-6 dB	0.2	0.0
-3 dB	0.5	0.0
0 dB	1.1	0.0
3 dB	2.9	0.0
6 dB	8.2	0.5
9 dB	22.0	2.6

Table 4.1: Difference in the SINR gain achived with hard-limiting (CMA) versus QPSK demodulation and 8-ary PSK demodulation.

dependent on the modulation format and the initial SINR ρ_0. A plot that directly shows the number of LSDDA iterations required for convergence as a function of initial SINR is presented in Figure 4.10. The LSDDA was deemed to have converged when the output SINR was within 1 dB of the optimal. This convergence criterion is somewhat arbitrary, but this provides a simple way to determine relative convergence rates. The figure shows that, with a BPSK signal, the LSDDA will converge in one iteration if the initial SINR is greater than ≈ 1.94 dB. It should be emphasized that this 1.94 dB threshold is dependent on the optimal output SINR. Figure 4.10 also includes supporting results from simulation. The simulation results are based on 1000 trials with 1024 symbols, and are presented as the mean number of iterations required. For example, if 100 trials converge in 2 iterations, while 900 trials converge in 3 iterations, the mean number of iterations for convergence is $(2 * 100 + 3 * 900)/1000 = 2.9$. This figure also includes results for LSDDA with a QPSK signal and LSCMA. In this environment, the LSDDA with a QPSK signal will converge in one iteration if the initial SINR is greater than 5.57 dB. In contrast, the LSCMA requires that the initial SINR be greater than 9 dB in order to converge in one iteration.

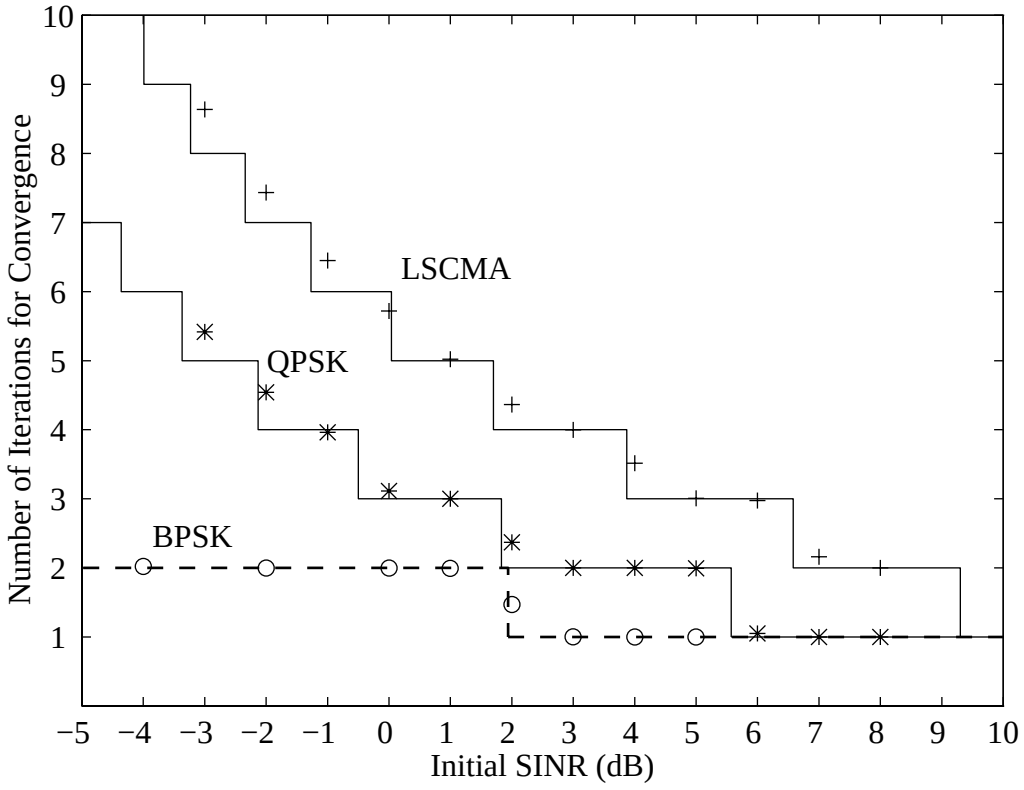

Figure 4.10: Number of iterations required for convergence, comparing LSCMA, LSDDA with a QPSK signal, and LSDDA with a BPSK signal. The solid lines denote the theoretically predicted value, the '+', '∗', and '∘' show results from simulation.

4.6 Behavior with Carrier Phase Offset

To this point we have assumed that the carrier frequency and carrier phase are known. We now investigate the effect of carrier phase offset on the performance of the LSDDA. The discussion here is limited to BPSK and QPSK desired signals, although the general conclusions will be valid for higher order PSK as well.

When the signal phase is known to the receiver, we have shown that R_{sd} is *real*, so that the phase of the signal in the beamformer output is not preserved. In this case the output SINR at iteration $k+1$ is only dependent on the SINR at iteration k. When the carrier phase is not known to the receiver, or when there is carrier phase offset, R_{sd} will in general be *complex*. Thus the output SINR at iteration $k+1$ is dependent on the SINR at iteration k and the phase of R_{sd}. Furthermore the output phase at iteration $k+1$ will be dependent on ρ_k and and the output phase at iteration k.

4.6.1 BPSK

We first consider the effect of carrier phase offset on the LSDDA when the desired signal is BPSK. We model the beamformer output signal as

$$y(n) = \sqrt{\rho_k}\, s(n)\, e^{j\theta} + z(n) \tag{4.88}$$

where the BPSK signal s takes on the values $+1$ and -1 with equal probability and θ is the carrier phase offset. Since the demodulator operates only on the real part of the beamformer output, the effect of carrier phase offset is equivalent to scaling the signal amplitude by $\cos\theta$. Thus we can write an expression for the output SINR gain for BPSK by simply substituting $\sqrt{\rho_k}\cos\theta$ for $\sqrt{\rho_k}$ in (4.65). This yields

$$G^2 = \frac{\pi}{\rho_k}\, e^{2\rho_k\, \cos^2\theta}\, \left(\mathrm{erf}(\sqrt{\rho_k}\cos\theta)\right)^2 \tag{4.89}$$

Note that we have defined SINR here as the SINR measured at the antenna, not as the SINR into the demodulator. The SINR gain (4.89) as a function of carrier phase offset θ is shown in Figure 4.11, along with supporting results from Monte Carlo simulation. Note that the SINR gain for BPSK demodulation with carrier phase offset is *not* bounded from below by 6 dB, as was the case when the carrier phase was known. However, this does not necessarily imply that LSDDA is susceptible to noise capture in this environment. Since R_{sd} is real, the phase of the signal component in the beamformer output will have zero offset. That is, we expect the LSDDA with a BPSK signal to perfectly correct for any carrier phase offset, unless the initial phase offset is $\pm\pi/2$, in which case there is no signal present in the the

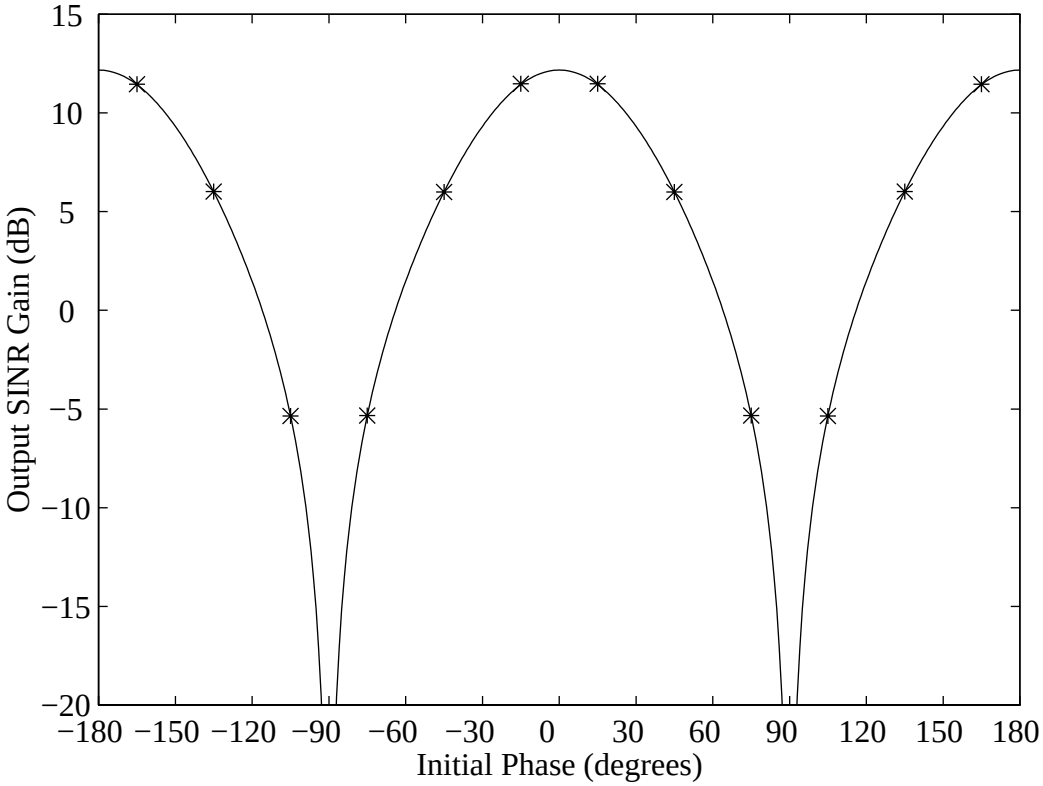

Figure 4.11: Output SINR gain for a BPSK signal in Gaussian interference with carrier phase offset. Solid lines denote gain predicted by analysis, '$*$' denotes mean gain measured from simulation. Results are for an input SINR of 0 dB.

demodulator input. Even in this case, however, random fluctuations due to finite block size may cause a small fraction of the signal to appear in the output. One a small component of the signal appears in the beamformer output, the output phase will be correct. We have shown that the LSDDA with a BPSK signal will converge even for very low initial SINR. Thus the LSDDA with a BPSK signal might be expected to converge so long as the initial phase (into the demodulator) is not exactly equal to $\pm\pi/2$.

To show that we have correctly anticipated the behavior of the LSDDA, we examine a situation where the phase is such that the SINR gain is less than unity. Let the initial weight vector be $\mathbf{w}_0 = [\,1\ 0\ 0\ 0\,]^T$, which gives an initial output SINR of -0.67 dB in the simulation environment. After one update, we expect the output SINR to be -3.10 dB, with the output phase equal to zero. That is, the output SINR initially *decreases*, but the output phase should be correct. After this first update the LSDDA should behave as described earlier for a BPSK signal with known carrier phase. Figure 4.12 shows the expected output SINR of

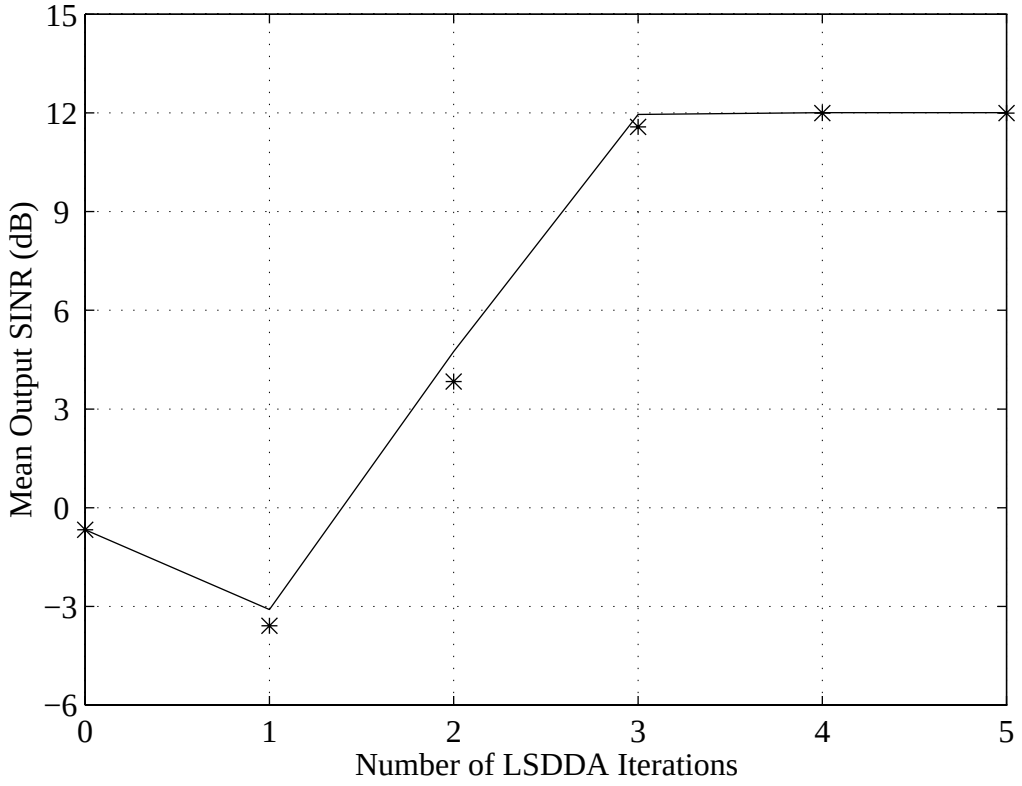

Figure 4.12: Comparison of expected and observed behavior of the LSDDA with a BPSK signal having initial output phase equal to 70°. The '∗' denote results from simulation.

the LSDDA based on these arguments, along with supporting results from simulation. The simulation results are again based on 1024 symbols and 1000 independent trials. The figure shows that the LSDDA does in fact decrease the output SINR initially, but then converges very rapidly.

A key conclusion is that the behavior of the LSDDA is indeed dependent on the initial phase of the signal estimate. Therefore it is possible that the performance of the LSDDA can be improved by the addition of a carrier phase estimator that is not decision directed. This is investigated in the following material, and it is shown that a simple non-linear phase estimator can dramatically improve LSDDA performance in some cases.

4.6.2 QPSK

We now consider the effect of carrier phase offset on the performance of LSDDA with a QPSK signal. The QPSK signal is modeled as described previously in (4.68). The cross-correlation

R_{sd} may be written as

$$R_{sd} = \sum_{k=0}^{3} p_{s_k}(k)\, \mathcal{E}\{\, s\, d^* \mid s = s_k \,\} \tag{4.90}$$

$$= \frac{1}{4} \sum_{k=0}^{3} \mathcal{E}\{\, s\, d^* \mid s = s_k \,\} \tag{4.91}$$

Following the derivation presented in Appendix D it can be shown that

$$\mathcal{E}\{\, s\, d^* \mid s = s_k \,\} = s_k \,\mathrm{erf}\left(\mathrm{Re}\left\{\sqrt{\rho_k}\, s_k\, e^{\,j\theta}\right\}\right) - j\, s_k \,\mathrm{erf}\left(\mathrm{Im}\left\{\sqrt{\rho_k}\, s_k\, e^{\,j\theta}\right\}\right) \tag{4.92}$$

where $\mathrm{Re}\{\cdot\}$ and $\mathrm{Im}\{\cdot\}$ denote real part and imaginary part, respectively. We have

$$\mathrm{Re}\left\{\sqrt{\rho_k}\, s_k\, e^{\,j\theta}\right\} = \sqrt{\rho_k}\cos\left(k\pi/2 + \pi/4 + \theta\right) \tag{4.93}$$

$$\mathrm{Im}\left\{\sqrt{\rho_k}\, s_k\, e^{\,j\theta}\right\} = \sqrt{\rho_k}\sin\left(k\pi/2 + \pi/4 + \theta\right) \tag{4.94}$$

The cross-correlation R_{zd} may be written as

$$R_{zd} = \sum_{k=0}^{3} p_{s_k}(k)\, \mathcal{E}\{\, z\, d^* \mid s = s_k \,\} \tag{4.95}$$

$$= \frac{1}{4} \sum_{k=0}^{3} \mathcal{E}\{\, z\, d^* \mid s = s_k \,\} \tag{4.96}$$

Following the derivation presented in Appendix D it can be shown that

$$R_{zd} = \frac{1}{4\sqrt{\pi}} \sum_{k=0}^{3} \left\{\exp\left(-\rho_k\, \cos^2\psi\right) + \exp\left(-\rho_k\, \sin^2\psi\right)\right\} \tag{4.97}$$

where

$$\psi = k\pi/2 + \pi/4 + \theta. \tag{4.98}$$

The QPSK SINR gain in the presence of carrier phase offset is shown in Figure 4.13. Supporting results from Monte Carlo simulation are included. Results are shown for several different values of initial SINR. As mentioned earlier, the SINR gain is not sufficient to determine the behavior of the LSDDA with carrier phase offset – we must also consider the phase of the updated beamformer output. The angle of the updated beamformer output corresponding to the SINR gain curves shown in Figure 4.13 is shown in Figure 4.14.

Figure 4.15 shows that we can accurately predict LSDDA behavior with a QPSK signal and phase offset. The environment and simulation parameters are identical to those considered previously, except that the initial output phase is 30°. It is evident that carrier phase offset can slow LSDDA convergence. This is shown more directly in Figure 4.16. This figure shows the number of LSDDA iterations required for convergence in the same signal environment, with the carrier phase offset varied from -45° to +45°. The initial SINR is o dB. Note that the required number of interations increases dramatically as the initial carrier phase approaches 45°. A simple modification that can greatly improve LSDDA convergence is introduced next.

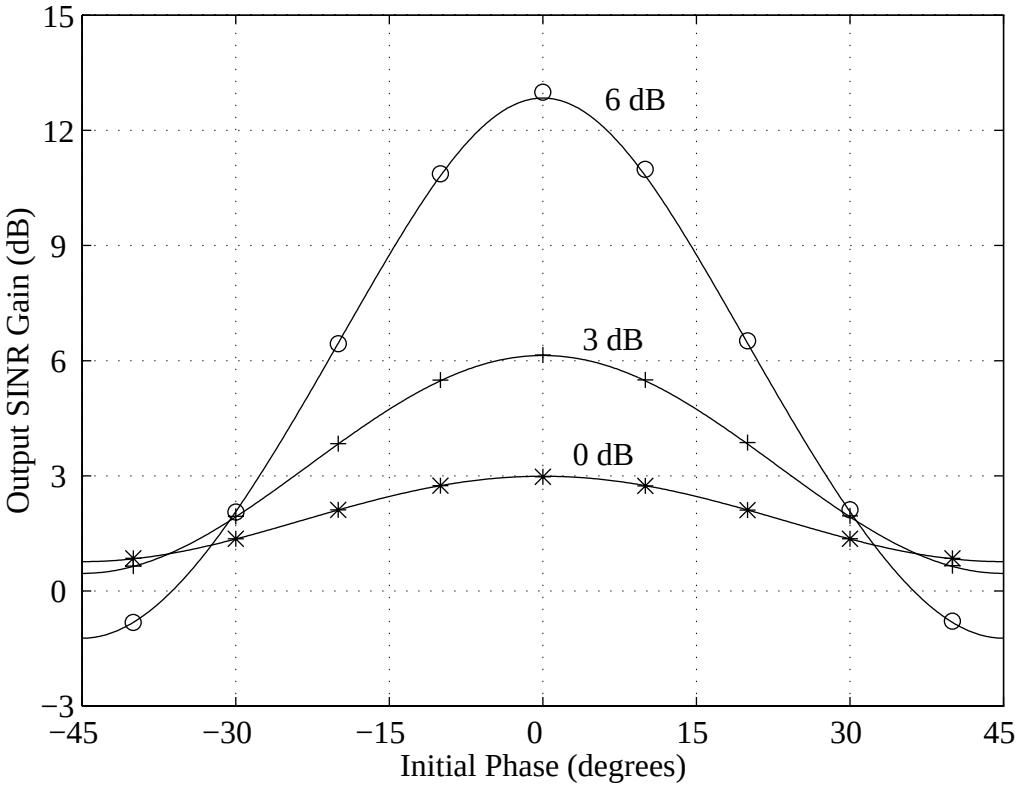

Figure 4.13: Output SINR gain for a QPSK signal in Gaussian interference with carrier phase offset. Solid lines denote gain predicted by analysis, '$\circ$', '$+$', and '$*$' denote mean gain measured from simulation. Results are parametric in input SINR.

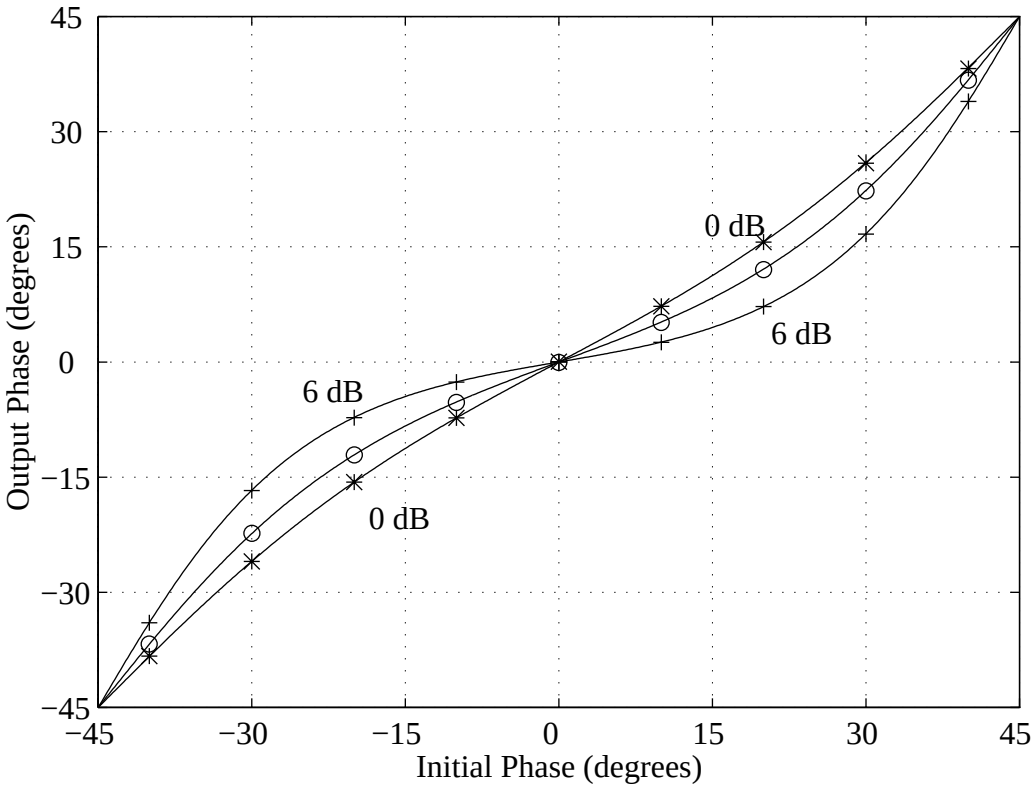

Figure 4.14: Output phase for a QPSK signal in Gaussian interference with carrier phase offset. Solid lines denote phase predicted by analysis, '$\circ$', '$+$', and '$*$' denote mean gain measured from simulation.

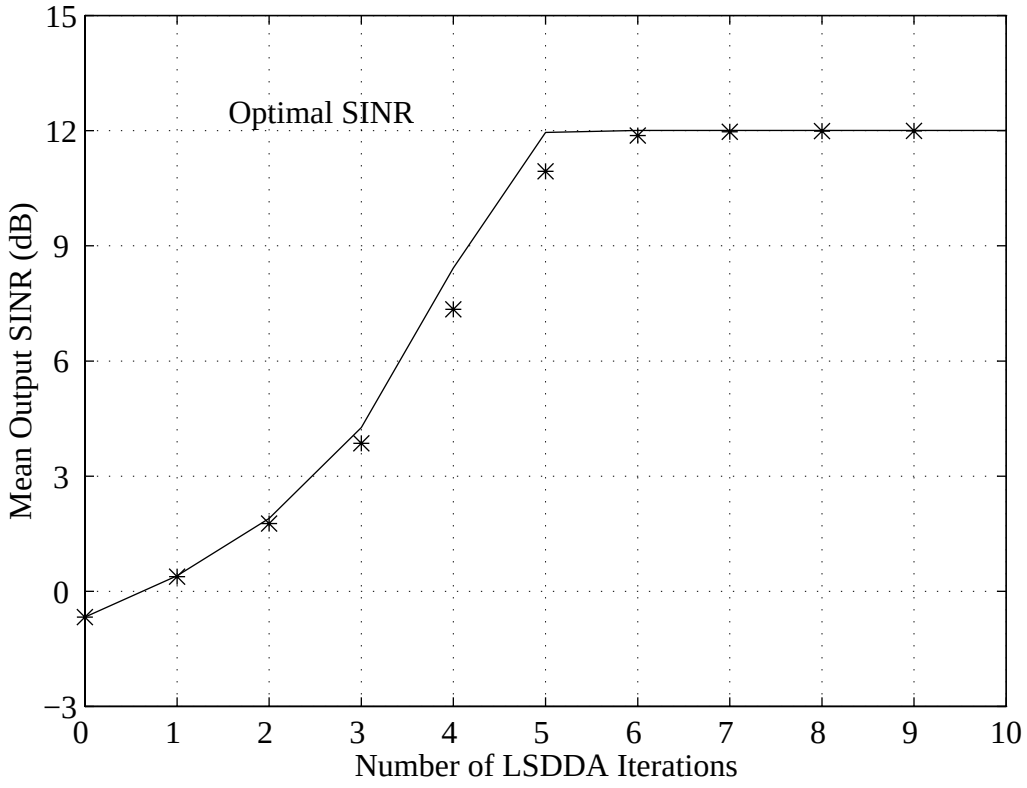

Figure 4.15: Comparison of expected and observed behavior of the LSDDA with a QPSK signal having initial output phase equal to 30°. The '$*$' denote results from simulation.

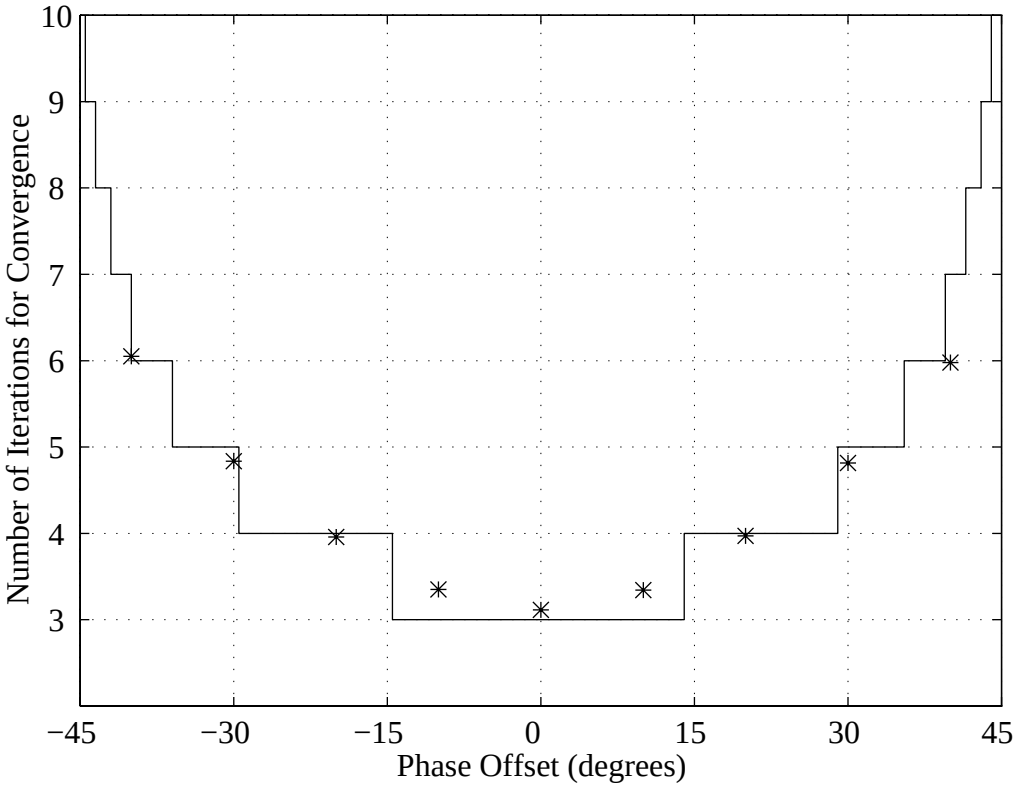

Figure 4.16: Number of iterations required for LSDDA convergence with a QPSK signal as a function of initial carrier phase offset. The intial SINR is 0 dB. The solid line denotes the theoretically predicted value, the '$*$' shows results from simulation.

4.6.3 Modified LSDDA with Carrier Phase Tracking

We have shown that the initial carrier phase offset can have an impact on the LSDDA convergence rate. A simple remedy to this problem is to add a phase compensation step to the algorithm. Before the signal is demodulated, the QPSK carrier phase is estimated by raising the output to the 4th power and averaging. This removes the modulation from the signal. This phase estimate is then removed from the beamformer output. The kth iteration of the modified LSDDA for a QPSK signal is then

1. Form a beamformer output $y_k(n) = \mathbf{w}_k^H \mathbf{x}(n)$.
2. Estimate the phase offset θ of the signal with
$$\hat{\theta} = \angle \sum_{n=0}^{N-1} y_k^4(n) \tag{4.99}$$
where $\angle$ denotes the angle of the summation.
3. Remove the estimated phase offset from the beamformer output
$$\tilde{y}_k(n) = y_k(n) \exp\left(-j\hat{\theta}\right) \tag{4.100}$$
4. Demodulate $\tilde{y}_k(n)$ to form $d_k(n)$
5. Solve for the updated weight vector $\mathbf{w}_{k+1} = \hat{\mathbf{R}}_{xx}^{-1} \hat{\mathbf{r}}_{xd}$

Figure 4.17 compares the output SINR of the existing and the modified versions of LSDDA. The figure shows that the speed of convergence is nearly the same as when the carrier phase is known. The effectiveness of this modified LSDDA in other environments, with smaller N, etc., remains an open issue.

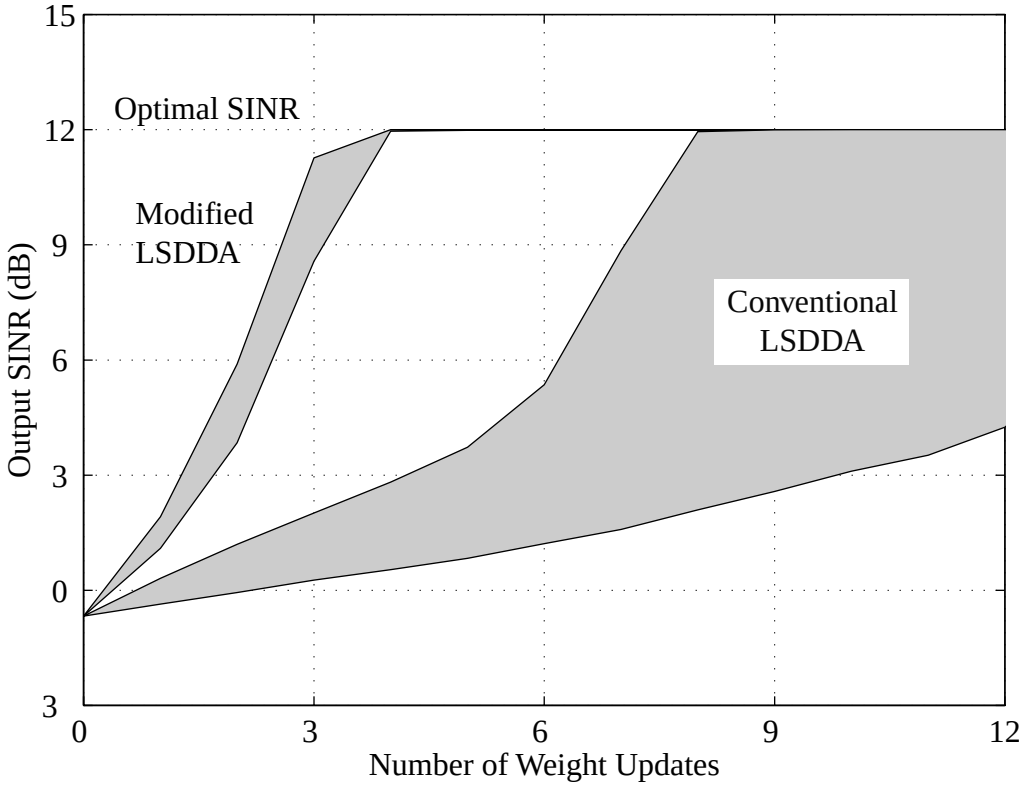

Figure 4.17: Comparison of conventional LSDDA and LSDDA with a separate carrier phase estimation step in the QPSK environment. The initial output phase is 45°. The shaded areas define the region where 98% of the trials fell.

4.7 Statistics for Finite Block Size

The main focus of this chapter is the *mean* behavior of the LSDDA. Hence we have not to this point considered the effect of finite data block size. This is in general a difficult problem to analyze, but some behavior can be readily identified.

To begin, we note that if no decision errors are made, the LSDDA will converge in one iteration to the best solution possible from the given block of data. Some deviation from the optimal solution will be unavoidable due to finite-data effects. A well-established rule of thumb is that the block size must be greater than twice the number of antennas in order for the mean output SINR to be within 3 dB of the optimal. The probability of making no decision errors over a block of N received symbols is given by

$$(1-p)^N \tag{4.101}$$

where p is the probability of making a symbol decision error. For very small block sizes there is a significant probability of selecting the correct symbols, even if the symbol error probability p is fairly high. As the block size increases, the probability of making no decision errors decreases rapidly unless p is very small.

Some insight into the behavior of the LSDDA and LSCMA for finite block size can be obtained by examining the bias and variance of the cross-correlation vector $\mathbf{R}_{xd}$, which may be viewed as an estimate of the SOI spatial signature $\mathbf{a}$. A spatial signature estimate can also be used directly in the following applications:

1. As a weight vector in a diversity combining system;
2. As a beamformer weight vector, which can be especially effective in CDMA applications.

4.7.1 Estimators with Known Waveform

Before proceeding with the derivation of the mean and variance of blind spatial signature estimators, we derive the mean and variance of estimators that exploit a known waveform. The reason for progressing in this fashion is to allow us to compare the performance of blind and known waveform-based parameter estimators. We would expect the performance of the blind estimators to approach the performance of the known waveform estimators as the SNR becomes large. First we examine the estimator of the (scalar) amplitude of a known waveform in Gaussian noise. Next we derive the mean and variance of the spatial signature estimator for a known waveform in unknown noise and interference. We expect the results for

the vector parameter case to reduce to the results for the scalar case if the vector dimension is set to one. We make several important assumptions about the signal and additive noise in this work. First we assume that the signal is constant modulus, i.e., $|s(n)| = 1 \forall n$. This is a simplifying assumption, but it is important to note that a wide class of common modulation formats are constant modulus, including FM, FSK, and PSK. We also assume that the signal and noise are uncorrelated. It is interesting that the mean and variance of these estimators are not dependent on the distribution of the noise and interference as long as the waveform is constant modulus. Of course, the values of α and β *are* dependent on the distribution of the signal and the noise.

First consider the scalar case where the data is modeled as

$$x(n) = as(n) + z(n) \tag{4.102}$$

where a is an unknown complex constant, $s(n)$ is the known modulated signal, and $z(n)$ is complex Gaussian noise. The signal and the noise are assumed to be white and have unit variance.

Define the estimate of a with a known waveform as

$$\hat{a} = \frac{1}{N} \sum_{n=0}^{N-1} x(n) s^*(n) \tag{4.103}$$

Clearly this is an unbiased estimator since

$$\mathcal{E}\{\, \hat{a} \,\} = a \tag{4.104}$$

The covariance of this estimator is

$$\begin{aligned} \mathcal{E}\{\, \hat{a}\hat{a}^* \,\} &= \mathcal{E}\left\{ \frac{1}{N^2} \left(\sum_{n=0}^{N-1} (as(n) + z(n)) s^*(n) \right) \left(\sum_{m=0}^{N-1} (as^*(m) + z^*(m)) s(m) \right) \right\} && (4.105) \\ &= \frac{1}{N^2} \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} \mathcal{E}\{\, (as(n) + z(n)) s^*(n) (as^*(m) + z^*(m)) s(m) \,\} && (4.106) \end{aligned}$$

This expression can be evaluated by examining the terms where $n \neq m$ and the terms where $n = m$ separately. For $n \neq m$, we have

$$\begin{aligned} & \mathcal{E}\{\, (as(n) + z(n)) s^*(n) (as^*(m) + z^*(m)) s(m) \,\} && (4.107) \\ &= \mathcal{E}\{\, (as(n) + z(n)) s^*(n) \,\} \mathcal{E}\{\, (as^*(m) + z^*(m)) s(m) \,\} && (4.108) \\ &= a^2 && (4.109) \end{aligned}$$

For $n = m$, we have

$$\begin{aligned} & \mathcal{E}\{\, (as(n) + z(n)) s^*(n) (as^*(m) + z^*(m)) s(m) \,\} && (4.110) \\ &= \mathcal{E}\{\, (as(n) + z(n)) (as^*(n) + z^*(n)) \,\} && (4.111) \\ &= a^2 + 1 && (4.112) \end{aligned}$$

where we have made use of the fact that $s(n)s^*(n) = 1$. We note that there are $N(N-1)$ terms with $n \neq m$, and N terms with $n = m$. This yields

$$\begin{aligned} \operatorname{cov} \hat{a} &= \frac{N-1}{N} a^2 + \frac{1}{N}\left(a^2 + 1\right) && (4.113) \\ &= a^2 + \frac{1}{N} && (4.114) \end{aligned}$$

We are also interested in the variance, given by

$$\begin{aligned} \operatorname{var} \hat{a} &\triangleq \mathcal{E}\{\, \hat{a}\hat{a}^* \,\} - |\mathcal{E}\{\, \hat{a} \,\}|^2 && (4.115) \\ &= \frac{1}{N} && (4.116) \end{aligned}$$

We now derive the mean and variance of a spatial signature estimator with known waveform. The array data is modeled as

$$\mathbf{x}(n) = \mathbf{a}s(n) + \mathbf{q}(n) \tag{4.117}$$

where $s(n)$ is the constant modulus signal, $\mathbf{a}$ is the spatial signature of the signal $s(n)$, and $\mathbf{q}(n)$ is the $M \times 1$ vector of noise and interference. The power of the received signal is incorporated in the spatial signature. The background noise and interference covariance matrix is therefore assumed to satisfy

$$\operatorname{trace}\{\mathbf{R}_{qq}\} = \mathcal{E}\Big\{\, \mathbf{q}(n)\mathbf{q}^H(n) \,\Big\} = M \tag{4.118}$$

The spatial signature estimate is given by

$$\hat{\mathbf{a}} \triangleq \frac{1}{N} \sum_{n=0}^{N-1} \mathbf{x}(n) s^*(n) \tag{4.119}$$

The mean value is

$$\mathcal{E}\{\, \hat{\mathbf{a}} \,\} = \mathcal{E}\{\, \mathbf{x}(n)s^*(n) \,\} = \mathbf{a} \tag{4.120}$$

As expected, the estimator is unbiased. The covariance is

$$\mathcal{E}\Big\{\, \hat{\mathbf{a}}\hat{\mathbf{a}}^H \,\Big\} = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} \mathcal{E}\Big\{\, \left(\mathbf{a}s(n) + \mathbf{q}(n)\right) s^*(n) \left(\mathbf{a}^H s^*(m) + \mathbf{q}^H(m)\right) s(m) \,\Big\} \tag{4.121}$$

For $n = m$, and asssuming that $s(n)s^*(n) = 1$, we have

$$\mathcal{E}\Big\{\, \left(\mathbf{a}s(n) + \mathbf{q}(n)\right) \left(\mathbf{a}^H s^*(n) + \mathbf{q}^H(n)\right) \,\Big\} = \mathbf{a}\mathbf{a}^H + \mathbf{R}_{qq} \tag{4.122}$$

For $n \neq m$,

$$\mathcal{E}\{\, \left(\mathbf{a}s(n) + \mathbf{q}(n)\right) s^*(n) \,\}\, \mathcal{E}\Big\{\, \left(\mathbf{a}^H s^*(m) + \mathbf{q}^H(m)\right) s(m) \,\Big\} = \mathbf{a}\mathbf{a}^H \tag{4.123}$$

This yields

$$\mathcal{E}\Big\{\,\hat{\mathbf{a}}\hat{\mathbf{a}}^H\,\Big\} = \mathbf{a}\mathbf{a}^H + \frac{1}{N}\mathbf{R}_{qq} \tag{4.124}$$

Note that if the dimension of $\mathbf{a}$ is set to one, the results for the vector parameter case reduce to the results for the scalar parameter case. The variance of the known waveform spatial signature estimator is then

$$\mathcal{E}\Big\{\,\hat{\mathbf{a}}\hat{\mathbf{a}}^H\,\Big\} - \mathcal{E}\{\,\hat{\mathbf{a}}\,\}\,\mathcal{E}\Big\{\,\hat{\mathbf{a}}^H\,\Big\} = \frac{1}{N}\mathbf{R}_{qq} \tag{4.125}$$

The total variance of each element in the estimated spatial signature is

$$\begin{aligned}\mathcal{E}\Big\{\,\hat{\mathbf{a}}^H\hat{\mathbf{a}}\,\Big\} - \mathcal{E}\Big\{\,\hat{\mathbf{a}}^H\,\Big\}\,\mathcal{E}\{\,\hat{\mathbf{a}}\,\} &= \frac{1}{N}\mathrm{trace}\,\{\mathbf{R}_{qq}\} &(4.126)\\ &= \frac{M}{N} &(4.127)\end{aligned}$$

4.7.2 Blind Estimators

We now turn our attention to deriving the mean and variance of blind parameter estimators for both scalar and vector parameters. First consider the scalar case where the data is modeled as described by (4.102). We are interested in the mean and variance of a blind estimate of a, which is given by

$$\hat{a} = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\,d^*(n) \tag{4.128}$$

where $d(n)$ is obtained by either a constant modulus mapping of the data or by demodulating the data. In either case, $d(n)$ is some nonlinear function of the data. As previously established , we can model the demodulator output signal as

$$d(n) = \alpha s(n) + \beta z(n) + \gamma(n) \tag{4.129}$$

In general, α and β will be complex. Since the signal and noise are uncorrelated and have unit variance, the mean value of $\hat{a}$ is

$$\begin{aligned}\mathcal{E}\{\,\hat{a}\,\} &= \mathcal{E}\{\,x(n)\,d^*(n)\,\} &(4.130)\\ &= \alpha^* a + \beta^* &(4.131)\end{aligned}$$

This clearly shows that the estimator is *biased*. However, this bias may be small if the SNR is high. For example, for a finite alphabet mapping, $\alpha \to 1$ and $\beta \to 0$ as the SNR grows high, i.e., as the probability of symbol decision error grows small. In practice the absolute bias is

not very revealing. The bias relative to the parameter being estimated is more revealing. For example, an absolute bias of 1 is large if $a = 1$, but is relatively small if $a = 100$. Therefore we are interested in the relative bias defined as

$$\frac{\mathcal{E}\{\,\hat{a}\,\} - a}{a} = \frac{\alpha^* a + \beta^* - a}{a} \tag{4.132}$$

The covariance is found as follows:

$$\begin{aligned}
\operatorname{cov}\hat{a} &\triangleq \mathcal{E}\{\,\hat{a}\hat{a}^*\,\} && (4.133)\\
&= \mathcal{E}\left\{ \frac{1}{N^2}\left(\sum_{n=0}^{N-1}(as(n)+z(n))d^*(n)\right)\left(\sum_{m=0}^{N-1}(as^*(m)+z^*(m))d(m)\right)\right\} && (4.134)\\
&= \frac{1}{N^2}\sum_{n=0}^{N-1}\sum_{m=0}^{N-1}\mathcal{E}\{\,(as(n)+z(n))d^*(n)(as^*(m)+z^*(m))d(m)\,\} && (4.135)
\end{aligned}$$

We assume that $s(n)$ and z(n) are white and uncorrelated with each other. For $n \neq m$, we have

$$\begin{aligned}
&\mathcal{E}\{\,(as(n)+z(n))d^*(n)(a^*s^*(m)+z^*(m))d(m)\,\} && (4.136)\\
&\quad= \mathcal{E}\{\,(as(n)+z(n))d^*(n)\,\}\,\mathcal{E}\{\,(as^*(m)+z^*(m))d(m)\,\} && (4.137)\\
&\quad= |\alpha^* a + \beta^*|^2 && (4.138)
\end{aligned}$$

For $n = m$, we have

$$\begin{aligned}
&\mathcal{E}\{\,(as(n)+z(n))d^*(n)(a^*s^*(m)+z^*(m))d(m)\,\} && (4.139)\\
&\quad= \mathcal{E}\{\,(as(n)+z(n))(a^*s^*(n)+z^*(n))\,\} && (4.140)\\
&\quad= |a|^2 + 1 && (4.141)
\end{aligned}$$

where we have made use of the fact that $d(n)d^*(n) = 1$. We note that there are $N(N-1)$ terms with $n \neq m$, and N terms with $n = m$. This yields

$$\operatorname{cov}\hat{a} = \frac{N-1}{N}|\alpha^* a + \beta^*|^2 + \frac{1}{N}\left(|a|^2 + 1\right) \tag{4.142}$$

The variance is given by

$$\begin{aligned}
\operatorname{var}\hat{a} &\triangleq \mathcal{E}\{\,\hat{a}\hat{a}^*\,\} - |\mathcal{E}\{\,\hat{a}\,\}|^2 && (4.143)\\
&= \frac{N-1}{N}|\alpha^* a + \beta^*|^2 + \frac{1}{N}\left(|a|^2+1\right) - |\alpha^* a + \beta^*|^2 && (4.144)\\
&= \frac{1}{N}\left(|a|^2+1\right) - \frac{1}{N}|\alpha^* a + \beta^*|^2 && (4.145)
\end{aligned}$$

Several features of the variance bear further discussion. First we note that the variance of the blind estimate approaches the variance of the known waveform estimate as $\alpha \to 1$

and $\beta \rightarrow 0$. A somewhat surprising result is that the *variance is not dependent on the intermodulation term* $\gamma(n)$.

Now consider the case where a spatial signature is to be estimated. The blind spatial signature estimate is

$$\hat{\mathbf{a}} = \frac{1}{N} \sum_{n=0}^{N-1} \mathbf{x}(n) d^*(n) \tag{4.146}$$

where $d(n)$ is, in general, derived as a non-linear function of a linear combination of the array data. In particular,

$$\begin{aligned} d(n) &= f\left(y(n)\right) && (4.147)\\ &= f\left(\mathbf{w}^H \mathbf{x}(n)\right) && (4.148) \end{aligned}$$

where $\mathbf{w}$ are the linear combiner (i.e., beamformer) weights. We are free to scale the weights or $d(n)$ in any convenient manner, since the only effect will be to multiply the estimated spatial signature by a simple scalar. Therefore we choose to scale $\mathbf{w}$ so that the scalar time series $\mathbf{w}^H \mathbf{q}(n)$ is unit power,

$$\mathbf{w}^H \mathbf{R}_{qq} \mathbf{w} = 1 \tag{4.149}$$

This allows us to write a model for $d(n)$ as

$$\begin{aligned} d(n) &= f\left(\mathbf{w}^H \mathbf{a} s(n) + \mathbf{w}^H \mathbf{q}(n)\right) && (4.150)\\ &= \alpha s(n) + \beta \mathbf{w}^H \mathbf{q}(n) + \gamma(n) && (4.151) \end{aligned}$$

The mean of the blind spatial signature estimate is then

$$\begin{aligned} \mathcal{E}\{\,\hat{\mathbf{a}}\,\} &= \mathcal{E}\{\,\mathbf{x}(n) d^*(n)\,\} && (4.152)\\ &= \mathcal{E}\Big\{\,\left(\mathbf{a} s(n) + \mathbf{q}(n)\right)\left(\alpha^* s^*(n) + \beta^* \mathbf{q}^H(n) \mathbf{w}\right)\Big\} && (4.153)\\ &= \alpha^* \mathbf{a} + \beta^* \mathcal{E}\Big\{\,\mathbf{q}(n)\mathbf{q}^H\,\Big\}\,\mathbf{w} && (4.154)\\ &= \alpha^* \mathbf{a} + \beta^* \mathbf{R}_{qq} \mathbf{w} && (4.155) \end{aligned}$$

Clearly the blind estimator is biased. The bias is dependent on the relative power of the signal component and noise component in $d(n)$, since $\hat{\mathbf{a}} \rightarrow \mathbf{a}$ as $\beta \rightarrow 0$. The bias is also dependent on the spatial distribution of the noise and interference, and the weight vector used to derive $d(n)$. It is instructive to examine the mean of $\hat{\mathbf{a}}$ when the optimal weight vector

$$\mathbf{w}_{\text{opt}} = \mathbf{R}_{qq}^{-1} \mathbf{a} \tag{4.156}$$

is used to derive $d(n)$. In this case we obtain

$$\begin{aligned} \mathcal{E}\{\,\hat{\mathbf{a}}\,\} &= \alpha^* \mathbf{a} + \beta^* \mathbf{R}_{qq} \mathbf{w}_{\text{opt}} && (4.157)\\ &= \alpha^* \mathbf{a} + \beta^* \mathbf{R}_{qq} \mathbf{R}_{qq}^{-1} \mathbf{a} && (4.158)\\ &= \left(\alpha^* + \beta^*\right) \mathbf{a} && (4.159) \end{aligned}$$

so that the estimated spatial signature differs only by a simple scale factor.

The covariance of the blind estimator is

$$\mathcal{E}\Big\{\,\hat{\mathbf{a}}\hat{\mathbf{a}}^H\,\Big\} = \frac{1}{N^2}\sum_{n=0}^{N-1}\sum_{m=0}^{N-1}\mathcal{E}\Big\{\,(\mathbf{a}s(n)+\mathbf{q}(n))\,d^*(n)\left(\mathbf{a}^H s^*(m)+\mathbf{q}^H(m)\right)d(m)\,\Big\} \tag{4.160}$$

We again treat the cases $n \neq m$ and $n = m$ separately. For $n \neq m$,

$$\begin{aligned}
&\mathcal{E}\Big\{\,(\mathbf{a}s(n)+\mathbf{q}(n))\,d^*(n)\left(\mathbf{a}^H s^*(m)+\mathbf{q}^H(m)\right)d(m)\,\Big\} && (4.161)\\
&\quad= \mathcal{E}\{\,(\mathbf{a}s(n)+\mathbf{q}(n))\,d^*(n)\,\}\,\mathcal{E}\Big\{\left(\mathbf{a}^H s^*(m)+\mathbf{q}^H(m)\right)d(m)\,\Big\} && (4.162)\\
&\quad= (\alpha^*\mathbf{a}+\beta^*\mathbf{R}_{qq}\mathbf{w})\left(\alpha\mathbf{a}^H+\beta\mathbf{w}^H\mathbf{R}_{qq}\right) && (4.163)
\end{aligned}$$

For $n = m$,

$$\begin{aligned}
&\mathcal{E}\Big\{\,(\mathbf{a}s(n)+\mathbf{q}(n))\,d^*(n)\left(\mathbf{a}^H s^*(m)+\mathbf{q}^H(m)\right)d(m)\,\Big\} && (4.164)\\
&\quad= \mathcal{E}\Big\{\,(\mathbf{a}s(n)+\mathbf{q}(n))\left(\mathbf{a}^H s^*(m)+\mathbf{q}^H(m)\right)\Big\} && (4.165)\\
&\quad= \mathbf{a}\mathbf{a}^H+\mathbf{R}_{qq} && (4.166)
\end{aligned}$$

Collecting the $n = m$ and $n \neq m$ terms, we obtain

$$\mathcal{E}\Big\{\,\hat{\mathbf{a}}\hat{\mathbf{a}}^H\,\Big\} = \frac{N-1}{N}(\alpha^*\mathbf{a}+\beta^*\mathbf{R}_{qq}\mathbf{w})\left(\alpha\mathbf{a}^H+\beta\mathbf{w}^H\mathbf{R}_{qq}\right)+\frac{1}{N}\left(\mathbf{a}\mathbf{a}^H+\mathbf{R}_{qq}\right) \tag{4.167}$$

This expression reduces to the previously obtained result for scalar $\hat{a}$ if we set $\mathbf{R}_{qq} = 1$ and $\mathbf{w} = 1$.

The variance of the blind spatial signature estimate is

$$\mathcal{E}\Big\{\,\hat{\mathbf{a}}\hat{\mathbf{a}}^H\,\Big\} - \mathcal{E}\{\,\hat{\mathbf{a}}\,\}\,\mathcal{E}\Big\{\,\hat{\mathbf{a}}^H\,\Big\} = \frac{1}{N}\left(\mathbf{a}\mathbf{a}^H+\mathbf{R}_{qq}\right)-\frac{1}{N}(\alpha^*\mathbf{a}+\beta^*\mathbf{R}_{qq}\mathbf{w})\left(\alpha\mathbf{a}^H+\beta\mathbf{w}^H\mathbf{R}_{qq}\right) \tag{4.168}$$

4.7.3 Simulation Results

To support these derivations we will examine the specific problem of estimating the unknown amplitude of a QPSK signal with unknown message received in additive white complex Gaussian noise. We assume that the QPSK signal is at complex baseband, i.e., the carrier frequency is known, and we also assume that the carrier *phase* is known. The effect of carrier phase offset is to modify the values of α and β, so carrier phase offset can be accounted for in a straightforward way. The QPSK signal is modeled as having been sampled baud synchronously with zero ISI. Figure 4.18 shows the relative bias given by (4.132) along with

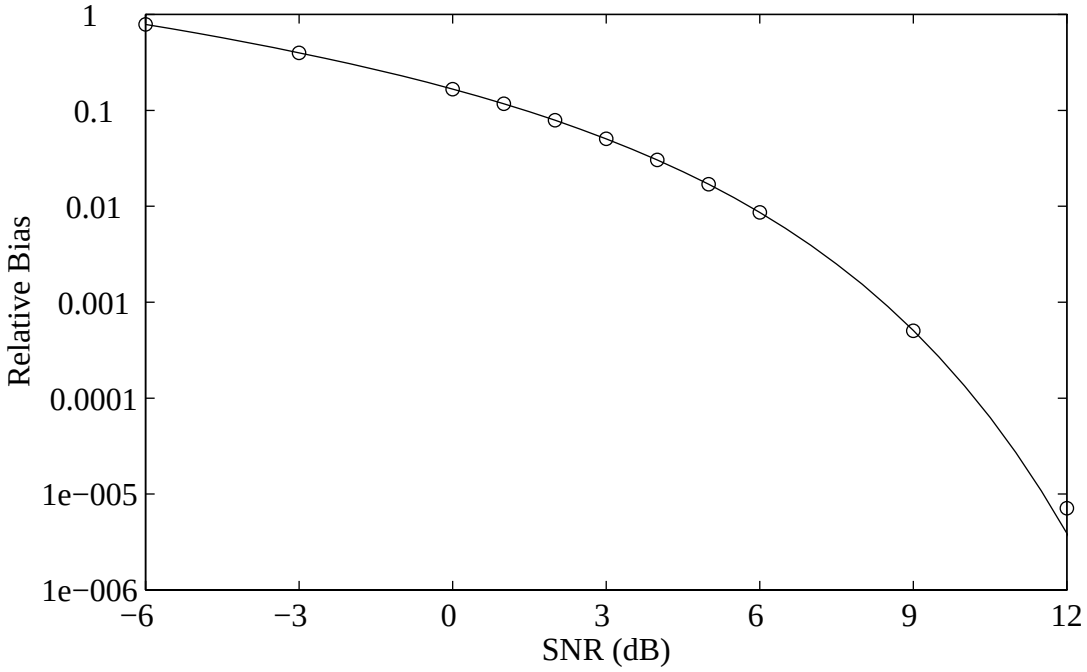

Figure 4.18: Relative bias of a decision directed estimate of signal amplitude as a function of SNR with a QPSK signal and Gaussian noise. Solid line denotes theoretical bias, '∘' denotes bias measured from simulation.

results from Monte Carlo simulation. The simulation results are based on 10,000 independent trials with $N = 1024$ symbols. The agreement between theory and simulation is excellent. Clearly the relative bias of the estimator is small for moderately high SNR. To examine the overall performance of the blind estimator, we use the Root Mean Square Error (RMSE), defined as

$$\text{RMSE} = \sqrt{\mathcal{E}\left\{ |a - \hat{a}|^2 \right\}} \tag{4.169}$$

The RMSE is convenient since it includes both bias and variance. Figure 4.19 shows the RMSE of the blind decision-directed estimate of the QPSK amplitude as a function of Signal to Noise Ratio (SNR). The SNR is defined here as the ratio of signal variance to noise variance.

We have now obtained the mean and variance of the estimator. The distribution is also of interest. Direct calculation of the distribution of $\hat{a}$ from basic principles is certainly possible. We take the simpler approach of assuming that, by the Central Limit Theorem, the distribution of $\hat{a}$ will tend to Gaussian as N becomes large.

We have shown that the general analysis framework introduced in this document can be used

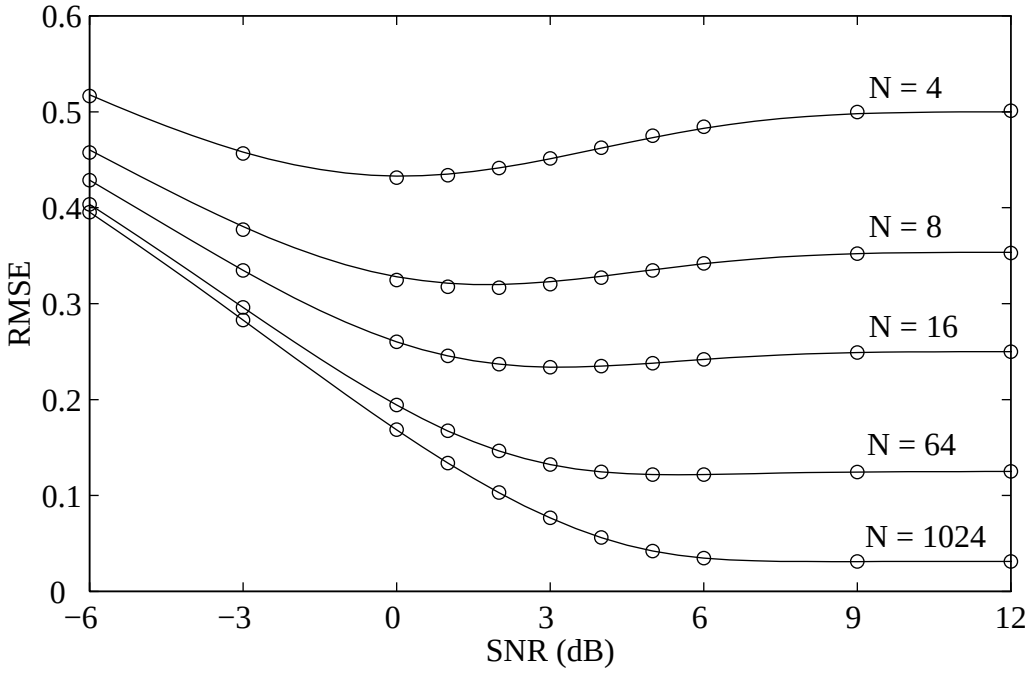

Figure 4.19: RMSE of a decision directed estimate of signal amplitude as a function of SNR with a QPSK signal and Gaussian noise. Solid lines denote theoretical RMSE, '$\circ$' denote RMSE measured from simulation. Results are parametric in number of symbols processed N.

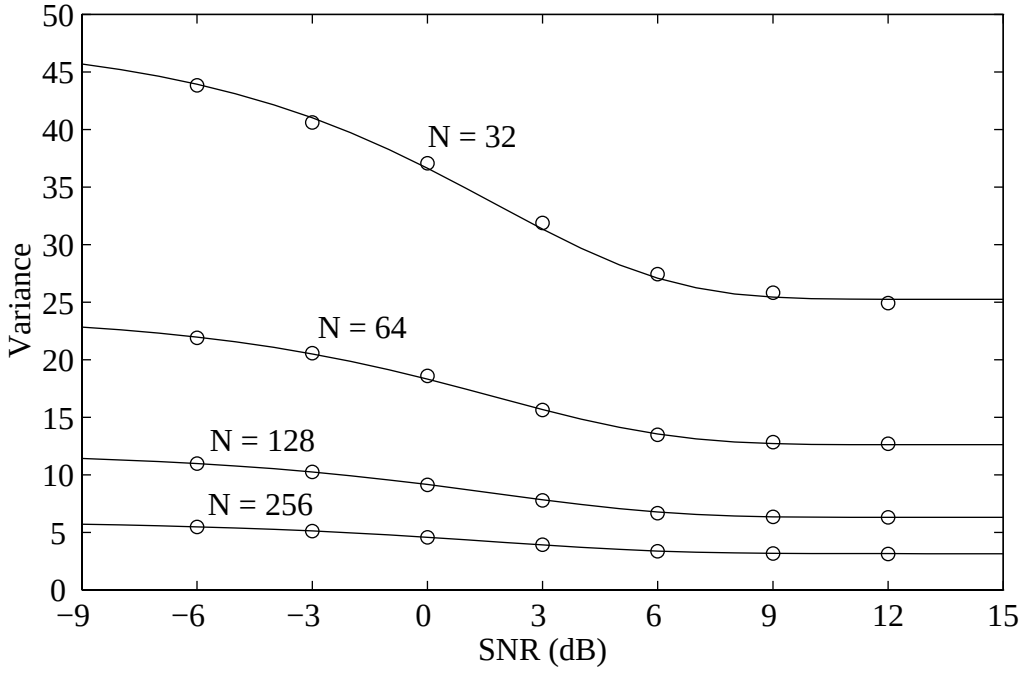

Figure 4.20: Variance of a decision directed estimate of a QPSK signal spatial signature in Gaussian noise. Solid lines denote theoretical variance, '$\circ$' denote variance measured from simulation. Results are parametric in number of symbols processed N.

to determine the mean and variance of the cross-correlation vector $\mathbf{R}_{xd}$. We have found that the mean and variance of a blind spatial signature estimator based on CMA or DDA is not dependent on the intermodulation terms. However, the intermodulation terms affect the statistics of the weight vector, and the corresponding output SINR distribution. This has implications for the design of new algorithms. Specifically, the selection of a non-linearity that gives excellent SIR, but also generates strong intermodulation terms, may not give better performance.

4.7.4 Finite Block Size Analysis of Non-Linear Least Squares

An exact analysis of the LSCMA and LSDDA for finite block size N should be possible, at least in principal. One approach is to re-express the weight vector update as a set of simultaneous linear equations. This yields

$$\mathbf{w} = \hat{\mathbf{R}}_{xx}^{-1}\hat{\mathbf{R}}_{xd} \tag{4.170}$$

$$\sum_{n=0}^{N-1}\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w} = \sum_{n=0}^{N-1}\mathbf{x}(n)d^*(n) \tag{4.171}$$

$$\sum_{n=0}^{N-1}\mathbf{x}(n)y^*(n) = \sum_{n=0}^{N-1}\mathbf{x}(n)d^*(n) \tag{4.172}$$

$$\sum_{n=0}^{N-1}\mathbf{x}(n)\left(y(n)-d(n)\right)^* = 0 \tag{4.173}$$

$$\sum_{n=0}^{N-1}\mathbf{x}(n)e^*(n) = 0 \tag{4.174}$$

where

$$e(n) \triangleq y(n)-d(n) \tag{4.175}$$

$$= y(n)-\alpha s(n)-\beta z(n)-\gamma(n) \tag{4.176}$$

$$= \mathbf{w}^H\mathbf{a}s(n)+\mathbf{w}^H\mathbf{q}(n)-\alpha s(n)-\beta z(n)-\gamma(n) \tag{4.177}$$

This yields

$$\sum_{n=0}^{N-1}\mathbf{x}(n)\left((\mathbf{w}^H\mathbf{a}-\alpha)s(n)+(1-\beta)\mathbf{w}^H\mathbf{q}(n)-\gamma(n)\right)^* = 0 \tag{4.178}$$

$$(\mathbf{a}^H\mathbf{w}-\alpha)\sum_{n=0}^{N-1}\mathbf{x}(n)s^*(n)+(1-\beta)\sum_{n=0}^{N-1}\mathbf{x}(n)\mathbf{q}^H(n)\mathbf{w}-\sum_{n=0}^{N-1}\mathbf{x}(n)\gamma^*(n) = 0 \tag{4.179}$$

Note that the intermodulation terms $\gamma(n)$ do indeed have an effect on the statistics of the weight vector, even though they *do not* affect the mean and variance of the cross-correlation

vector $\mathbf{R}_{xd}$. This shows that while we can ignore the intermodulation terms when considering the asymptotic algorithm performance, they cannot be ignored in a finite-time analysis.

4.8 Cost Function Analysis

As we noted earlier in Section 3.4, where the (1,2) CMA cost function was presented, it is important to examine the properties of any blind cost function. We apply the same methodology previously used to calculate the CMA cost function to the decision-directed cost function.

We show in Appendix A that the decision-directed cost function has the general form

$$F(\rho,\theta) = 2 - \frac{2}{\sqrt{1+\rho}}\text{Real}\left\{\alpha^* e^{j\theta}\sqrt{\rho} + \beta^*\right\} \tag{4.180}$$

where ρ is the beamformer output SINR and θ is the phase at the demodulator input. Note that α and β are dependent on ρ and also on θ.

We begin by considering the simplest decision-directed cost function, that of a BPSK signal with known carrier phase in Gaussian noise. By evaluating (4.180) with α given by (4.61) and β given by (4.62), we obtain the function shown in Figure 4.21. The resulting function decreases monotonically with increasing SIR. Recall that we showed earlier that noise capture is not possible with a BPSK signal.

We next consider a QPSK signal in Gaussian noise with known carrier phase. By evaluating (4.180) with α given by (4.70) and β given by (4.71), we obtain the function shown in Figure 4.22. Recall that noise capture *is* possible with a QPSK signal. It can be seen that the gradient tends to zero as the SIR grows small, which is consistent with susceptibility to noise capture.

Finally we consider a more interesting situation, that of a QPSK signal with carrier phase offset in Gaussian noise. The decision directed cost function is shown in Figure 4.23. Note that the cost function does not necessarily decrease wth increasing SIR. In particular, if the carrier offset is large, the cost function *increases* as the SIR increases. Clearly this is not a desirable characteristic for a blind cost function. This would indicate that the LSDDA is more susceptible to noise capture when the carrier phase is unknown. However, if the SIR is reasonably large, the gradient is large in the phase dimension. This indicates that, in the presence of carrier phase offset, the LSDDA will tend to correct the phase of the output signal more quickly than it will optimize output SINR.

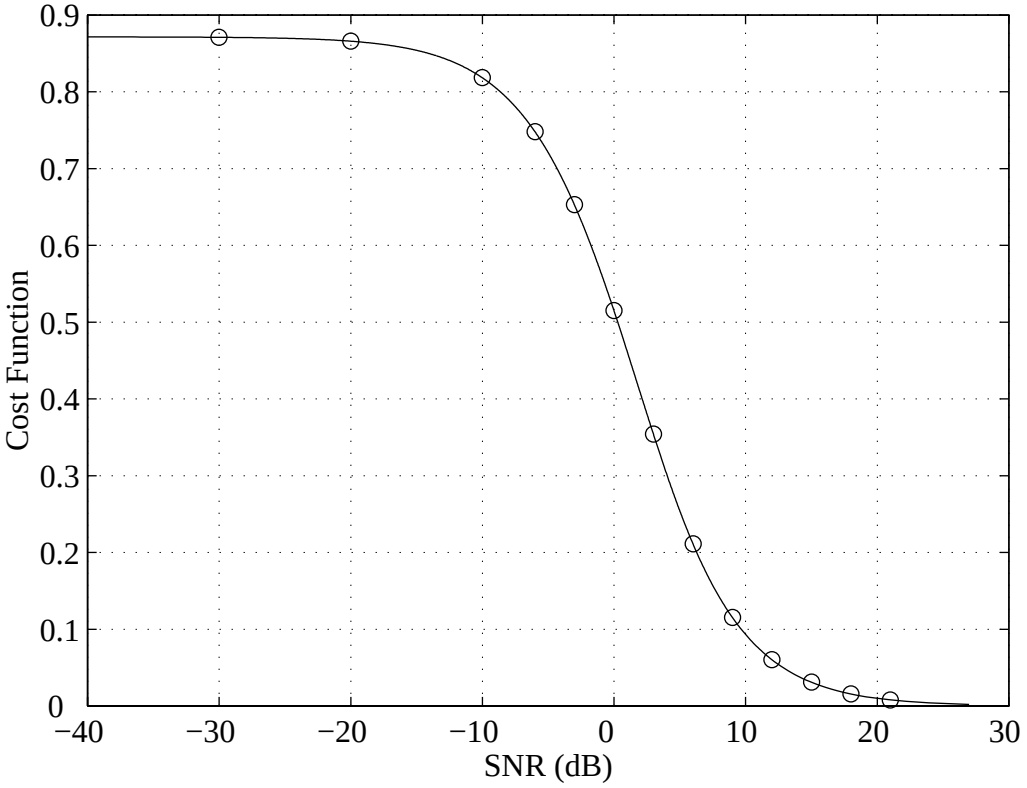

Figure 4.21: Decision-directed cost function for a BPSK signal with known carrier phase in complex Gaussian noise.

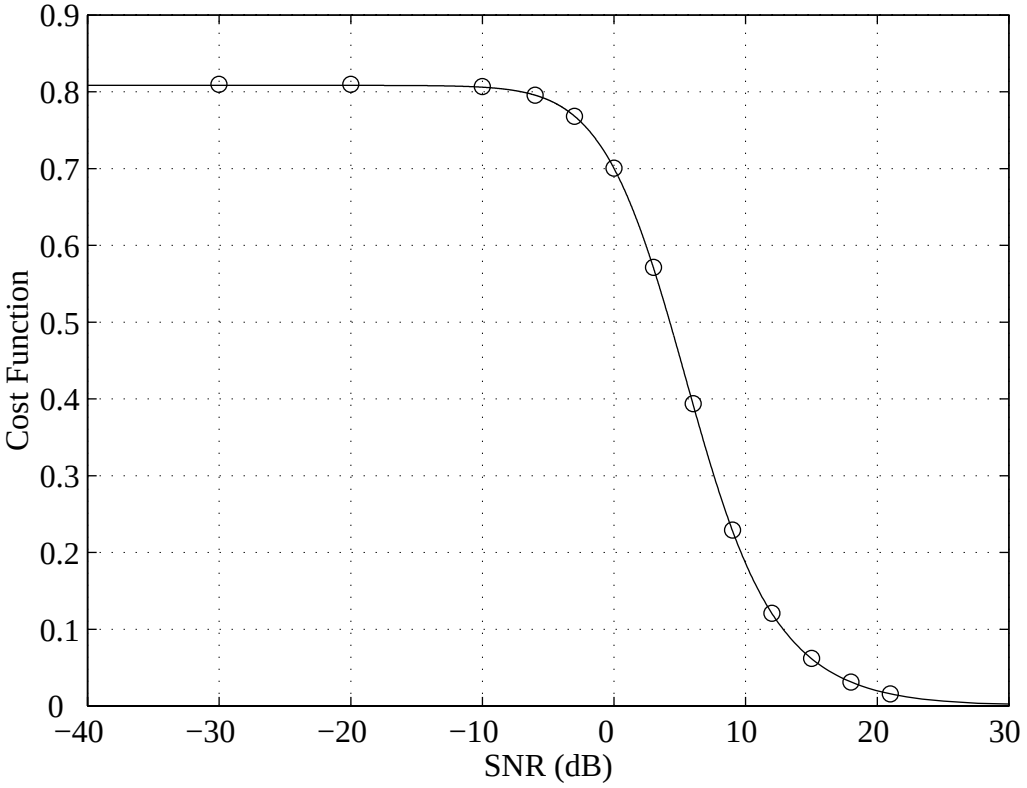

Figure 4.22: Decision-directed cost function for a QPSK signal with known carrier phase in complex Gaussian noise.

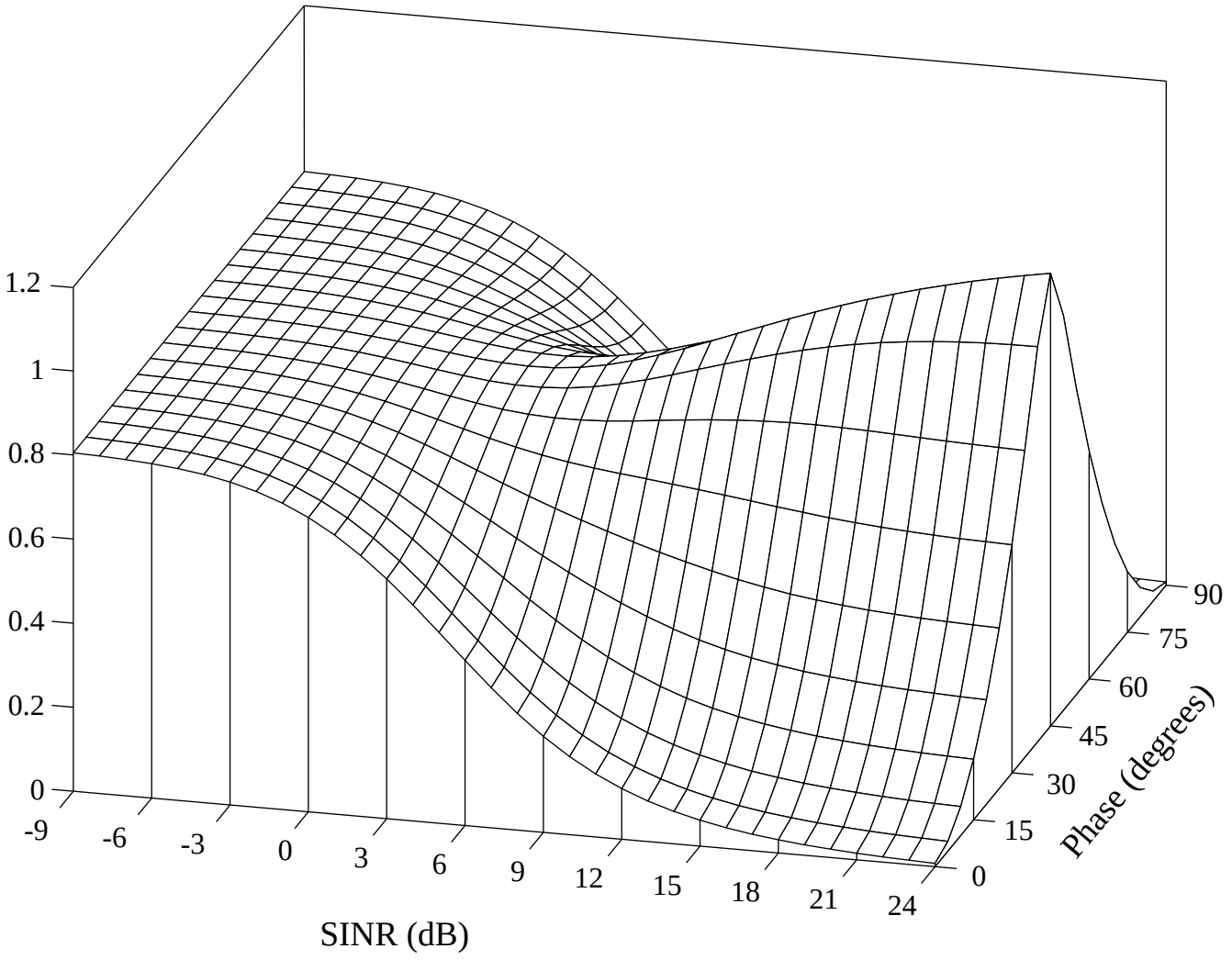

Figure 4.23: Decision-directed cost function for a QPSK signal in complex Gaussian noise as a function of carrier phase offset and SINR.

4.9 Conclusions

Analytic expressions have been presented for the mean improvement in output SIR achieved with each iteration of a block-update decision directed adaptive algorithm known as the Least Squares Decision Directed Algorithm (LSDDA). We have obtained results for M-ary PSK in general, and for BPSK and QPSK in particular. We have also analyzed the effect of carrier phase offset. The analysis framework described here can be used to predict the mean behavior of the LSDDA in *any* environment, so long as the pdf of the noise and interference is known. Using results previously established in Chapter 3, we are able to compare the performance of the LSDDA and the Least Squares Constant Modulus Algorithm (LSCMA).

As would be expected, the LSDDA converges very rapidly for moderately high initial SIR. When the interference is Gaussian, the improvement in output SIR obtained with the LSDDA is bounded from below by the improvement obtained with the LSCMA. This makes sense intuitively since the decision directed algorithm exploits more information than the CMA. The performance of the LSDDA and LSCMA at low SIR is very similar (except for the special case of a BPSK desired signal). This gives some theoretical justification for the common practice of using CMA to bootstrap a decision directed algorithm. An intuitively appealing result is that a decision directed algorithm behaves more and more like CMA as the number of symbols in the PSK constellation increases.

The LSDDA can be susceptible to the phenomenon known as *noise capture*. Noise capture occurs when the algorithm extracts Gaussian noise and interference and nulls the desired signal. This phenomenon has been studied extensively for CMA, but the existence of noise capture in decision directed algorithms has not been recognized previously.

Some directions for future work include:

- extension of the analysis to other modulation formats, especially FSK and M-ary orthogonal modulation;
- extensions to the popular Stochastic Gradient Descent (SGD) and Normalized SGD versions of the DDA;
- extension to extraction of multiple signals, e.g., Multi-Target DDA, ILSP.
- extensions to equalization applications, where the primary source of distortion is frequency selective multipath, not co-channel interference;
- obtain results for finite block size using results from multivariate statistics;
- extension to wideband space-time processing;

Chapter 5

Decision Directed Algorithm in Constant Modulus Interference

Summary

In Chapter 4 we studied the behavior of the LSDDA when the interference has a complex Gaussian distribution. We now consider the case where the interference has constant envelope. We show that the LSDDA behavior with constant modulus interference is quite different from the behavior with Gaussian interference.

5.1 BPSK with Constant Modulus Interference

We first consider the situation where the desired signal is BPSK and the interference is constant modulus (CM). We initially assume that the carrier phase of the BPSK signal is known to the receiver. We model the beamformer output signal as

$$y = \sqrt{\rho_k}\, e^{jk\pi} + e^{j\phi} \tag{5.1}$$

where $k = 0, 1$ and ϕ is the phase of the CM interferer. We assume that the CM interference has uniform phase distribution. This is equivalent to assuming that the carrier phase is a uniformly distributed random variable. Since the BPSK demodulator operates on only the real part of the beamformer output, we only need the PDF of the real part of the interferer. This is given by

$$p_{z_r}(z_r) = \frac{1}{\pi\sqrt{1 - z_r^2}} \tag{5.2}$$

over $-1 \leq z_r \leq 1$, with $p_{z_r}(z_r)$ equal to zero outside this interval.

The cross-correlation of the desired BPSK signal $s(n)$ with the hard-decisions $d(n)$ is shown in Appendix F to be

$$\mathcal{E}\{\, s\, d\,\} = \frac{2}{\pi} \arcsin \sqrt{\rho_k} \tag{5.3}$$

Appendix F shows that the cross-correlation of the CM interferer with the hard decisions is

$$\mathcal{E}\{\, z\, d\,\} = \frac{2}{\pi} \sqrt{1 - \rho_k} \tag{5.4}$$

The SIR in the demodulator output is thus given by

$$\mathrm{SIR}_{out} = \frac{\left(\arcsin \sqrt{\rho_k}\right)^2}{1 - \rho_k} \tag{5.5}$$

The ratio of output SIR over input SIR is

$$\mathrm{SIR}_{gain} = \frac{\left(\arcsin \sqrt{\rho_k}\right)^2}{\rho_k\, (1 - \rho_k\,)} \tag{5.6}$$

Figure 5.1 shows the SIR gain for BPSK given by (5.6) versus the initial beamformer output SIR ρ_k . Results from scalar Monte Carlo simulation are also included to support the derivations. The simulation results are based on 1000 trials with a BPSK desired signal and a tone interferer. The frequency of the sinusoid is 13/1024, with uniformly distributed random phase. The simulation results agree very well with theory.

A key property of the BPSK SIR gain is that the SIR gain is always greater than unity (0 dB). As the initial SIR grows small, the SIR gain approaches unity. This is similar to the behavior of the LSCMA with Gaussian noise. Thus we expect that the LSDDA with a BPSK signal is susceptible to capture of a CM interferer, such as a CW signal or another co-channel BPSK signal. Since the SIR gain only asymptotically approaches 0 dB, we would expect the CM capture of LSDDA here to correspond to a saddlepoint in the LSDDA cost function, and not a true minima. When the SIR is higher than 0 dB there will be no decision errors. Therefore the SIR gain tends asymptotically towards infinity as the initial SIR approaches 0 dB.

5.2 QPSK with Constant Modulus Interference

We now examine the behavior of the LSDDA with a QPSK desired signal received in CM interference. It is shown in Appendix G that the cross-correlation of the desired signal $s(n)$

with the demodulator output is

$$\mathcal{E}\{\, s\, d^* \,\} = \frac{2\sqrt{2}}{\pi} \arcsin \sqrt{\rho_k \ /2} \tag{5.7}$$

Appendix G shows that the cross-correlation of the interfering signal $z(n)$ with the demodulator output is

$$\mathcal{E}\{\, z\, d^* \,\} = \frac{4}{\pi} \sqrt{1 - \rho_k \ /2} \tag{5.8}$$

The output SIR is

$$\mathrm{SIR}_{out} = \frac{\left(\arcsin \sqrt{\rho_k \ /2}\right)^2}{2 - \rho_k} \tag{5.9}$$

The ratio of output SIR over input SIR is

$$\mathrm{SIR}_{gain} = \frac{\left(\arcsin \sqrt{\rho_k \ /2}\right)^2}{\rho_k \ (2 - \rho_k \ )} \tag{5.10}$$

Figure 5.1 shows the SIR gain for QPSK given by (5.10) versus the initial beamformer output SIR ρ_k . Supporting results from Monte Carlo simulation are also included. The simulation is identical to that used with the BPSK signal, except that a QPSK signal is used. Again, the simulation results agree very well with theory. Note that, as was the case with Gaussian interference, the performance with QPSK is much worse than that obtained with BPSK.

Figure 5.1 shows that the behavior of the LSDDA with a QPSK signal in CM interference is unexpected. The SIR gain for QPSK with a constant modulus interferer is less than one for initial SIR less than approximately 1.203 dB. If the initial SIR is less than this value, the LSDDA is likely to capture the CM interferer instead of the desired QPSK signal. This will occur even though the LSDDA has perfect knowledge of the carrier frequency and carrier phase of the desired QPSK signal.

Also note that SIR gain for LSDDA with a QPSK signal approaches -6 dB as the SIR grows small. This is very similar to the behavior of LSCMA with two CM signals. Thus we expect the CM capture here to correspond to a true minimum in the LSDDA cost function.

Figure 5.2 compares the behavior of the LSDDA and the LSCMA with a QPSK signal received in strong CM interference. The simulation environment consists of a QPSK signal incident from 10° with 16.5 dB SWNR. A complex sinusoid interferer is received with 16.0 dB SWNR from 65°. The array has two elements with $\lambda/2$ spacing, and the LSDDA block size is set to 1024 symbols. The initial weight vector is $\mathbf{w}_0 = [\,1\ 0]$, which gives an initial SINR of approximately +0.4 dB. The analysis presented above predicts that the LSDDA will capture the CM interferer, since the initial SINR is less than the capture threshold of 1.2 dB. However,

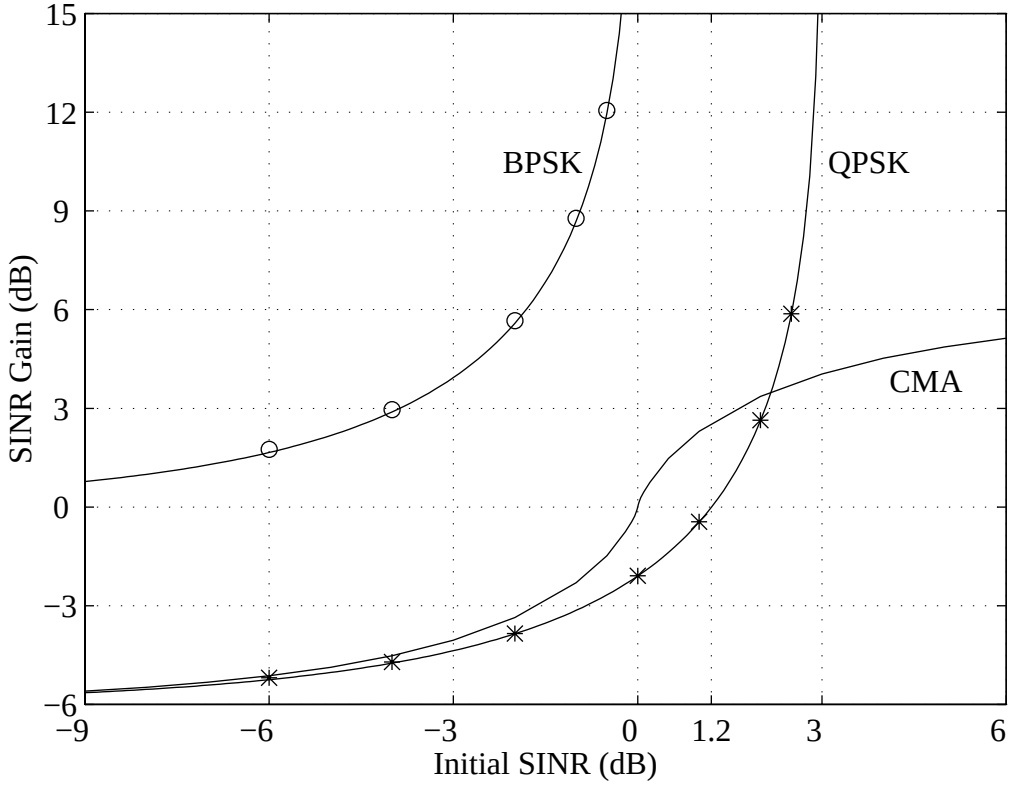

Figure 5.1: Output SINR gain achieved with one iteration of LSDDA with a PSK desired signal and constant modulus interference. Solid lines denote gain predicted by theory, '$*$' and '$\circ$' denote mean gain measured from simulations. Gain achieved with CMA shown for comparison.

it is important to bear in mind that this is an asymptotic analysis, and the asymptotic results are only an approximation to the mean behavior of the algorithms with a finite amount of data. Thus, due to the effects of finite collect time, and Gaussian background noise, the LSDDA does not *always* capture the CM interferer in this simulation. Out of 1000 trials, 89.9% resulted in capture of the interferer. In contrast, the LSCMA always converged to the desired QPSK-capture solution in this environment. Despite the fact that the simulation results do not exactly agree with the asymptotic analysis, the analysis did accurately predict that capture of a CM interferer was more likely with LSDDA than with LSCMA.

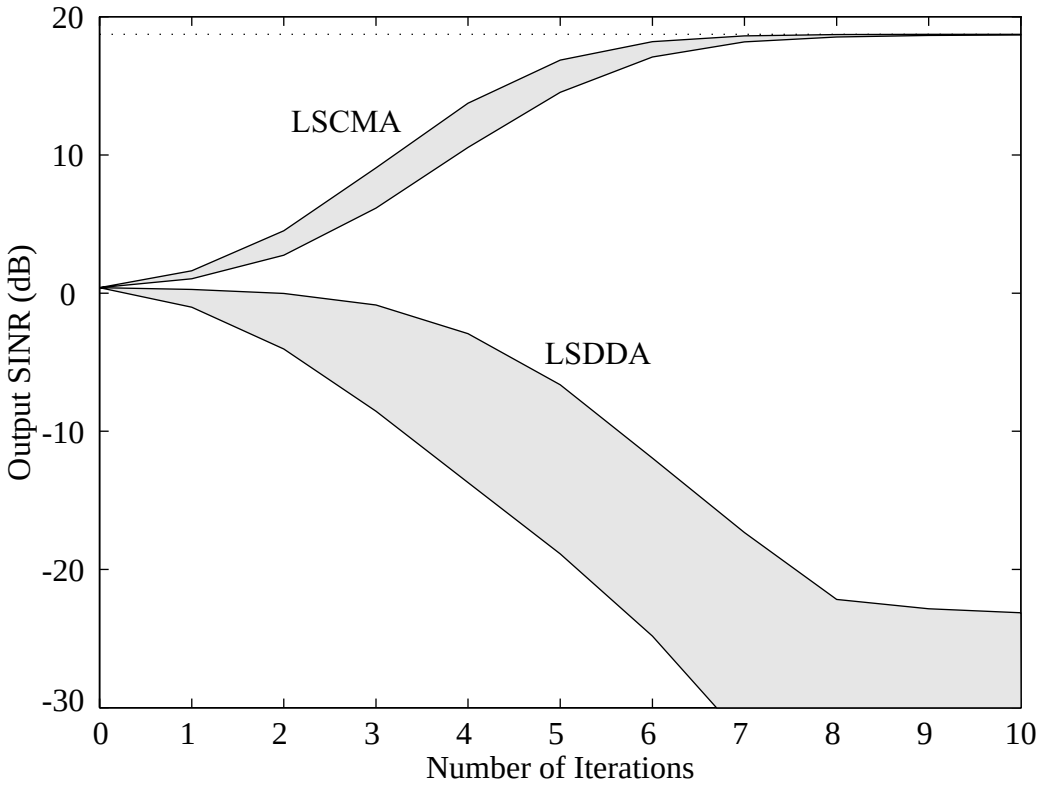

Figure 5.2: Output SINR of LSDDA and LSCMA with a QPSK desired signal and a sinusoid interferer, with the initial SINR equal to +0.4 dB. The shaded regions show where 95% of trials fell.

5.3 Constant Modulus Interference and Phase Offset

We have assumed to this point that the carrier phase of the desired signal is known. We will now consider the more general case where the carrier phase is either unknown, or there is some error in carrier phase knowledge.

When the signal is BPSK, the effect of carrier phase offset is the same as reducing the signal amplitude by $\cos\theta$, where θ is the phase offset. The behavior with a QPSK signal and CM interference is more complicated. We model the beamformer output as

$$y(n) = \sqrt{\rho} e^{jk\pi/2} e^{j\pi/4} e^{j\theta} + e^{j\phi} \tag{5.11}$$

where $k = 0, 1, 2, 3$ is an integer random variable, θ is the phase offset of the signal, and ϕ is the random phase modulation of the CM interferer. The cross-correlation R_{sd} may be written as

$$\begin{aligned} R_{sd} &= \sum_{k=0}^{3} p_{s_k}(k)\, \mathcal{E}\{\, s\, d^* \mid s = s_k \,\} & (5.12) \\ &= \frac{1}{4} \sum_{k=0}^{3} \mathcal{E}\{\, s\, d^* \mid s = s_k \,\} & (5.13) \end{aligned}$$

We have

$$\begin{aligned} \mathcal{E}\{\, s\, d^* \mid s = s_k \,\} &= e^{jk\pi/2} e^{j\pi/4} \frac{1}{2\pi} \int_0^{2\pi} \left[\operatorname{sign}\left\{ \sqrt{\rho} \cos\left(\frac{k\pi}{2} + \frac{\pi}{4} + \theta \right) + \cos\phi \right\} + \right. & (5.14) \\ &\quad \left. j \operatorname{sign}\left\{ \sqrt{\rho} \sin\left(\frac{k\pi}{2} + \frac{\pi}{4} + \theta \right) + \sin\phi \right\} \right] d\phi \end{aligned}$$

Let us define the function $f(\cdot)$ as

$$f(x) \triangleq \int_o^{2\pi} \operatorname{sign}(x + \cos\phi)\, d\phi \tag{5.15}$$

Then it is straightforward to show that

$$f(x) = \begin{cases} -2\pi & x < -1 \\ 4 \arcsin(x) & -1 < x < +1 \\ 2\pi & +1 > x \end{cases} \tag{5.16}$$

This allows the cross-correlation of the QPSK signal with the demodulator output to be written as

$$R_{sd} = \frac{e^{j\pi/4}}{2\pi} \sum_{k=0}^{3} e^{jk\pi/2} \left\{ f\left(\sqrt{\rho_k} \cos\left(\frac{k\pi}{2} + \frac{\pi}{4} + \theta \right) \right) - j f\left(\sqrt{\rho_k} \sin\left(\frac{k\pi}{2} + \frac{\pi}{4} + \theta \right) \right) \right\} \tag{5.17}$$

We now turn our attention to calculation of R_{zd}. The conditional cross-correlation is

$$\begin{aligned}\mathcal{E}\{\, z\, d^* \mid s = s_k \,\} &= \frac{1}{2\pi}\int_0^{2\pi} e^{j\phi}\left[\operatorname{sign}\left\{\sqrt{\rho}\cos\left(\frac{k\pi}{2}+\frac{\pi}{4}+\theta\right)+\cos\phi\right\}+\right. \\ &\qquad \left. j\operatorname{sign}\left\{\sqrt{\rho}\sin\left(\frac{k\pi}{2}+\frac{\pi}{4}+\theta\right)+\sin\phi\right\}\right] d\phi\end{aligned} \tag{5.18}$$

It is straightforward to show that

$$\int_0^{2\pi}\cos\phi\operatorname{sign}\{x+\cos\phi\}\,d\phi = 4\sqrt{1-x^2} \tag{5.19}$$

This shows that the cross-correlation of the CM interferer with the demodulator output is

$$R_{zd} = \frac{1}{2\pi}\sum_{k=0}^{3}\left\{4\sqrt{1-\cos^2\left(\frac{k\pi}{2}+\frac{\pi}{4}+\theta\right)}+4\sqrt{1-\sin^2\left(\frac{k\pi}{2}+\frac{\pi}{4}+\theta\right)}\right\} \tag{5.20}$$

To support these derivations, we will examine the magnitude and phase of R_{sd} and R_{zd} before proceeding further. Figure 5.3 shows the magnitude of R_{sd} versus carrier phase offset for a QPSK signal in CM interference with the SIR equal to 0.5 dB. Since this SIR is less than 1.2 dB, this corresponds to a situation where we expect LSDDA to capture the CM interference. Supporting results from simulation are included in Figure 5.3. These simulations are based on 1000 Monte Carlo trials, with each trial based on 1024 QPSK symbols and a complex sinusoidal interferer. Figure 5.5 shows the magnitude of R_{zd} for the same scenario. Again, the results from simulation agree very well with the results from analysis.

The phase of R_{sd} is examined in Figure 5.4. This figure shows the difference between the signal phase at the demodulator input and the demodulator output. Note that if the phase difference is zero, the demodulator will not alter the phase of the signal. The phase difference in Figure 5.4 is equal to zero at integer multiples of $\pi/2$, as expected. A decision directed algorithm is insensitive to phase shifts of $\pi/2$. However, Figure 5.4 shows that the phase difference is also equal to zero at integer multiples of $\pi/4$. This is an indication that the LSDDA may have an undesirable local minima where the output phase of the QPSK signal is offset by $\pi/$ from what is expected.

Finally, we examine the SIR gain verusus phase offset for this scenario. This is shown in Figure 5.6. Note that the SIR gain at zero phase offset is less than 0 dB, even though the input SIR is positive. Also note that the SIR gain at odd integer multiples of $\pi/4$ is greater than one. However, this does *not* prove that the LSDDA will converge to the desired solution if the phase offset is an integer multiple of $\pi/4$. To do this we must examine the SIR gain as a function of initial SIR with the carrier phase offset set to $\pi/4$. This is shown in Figure 5.7.

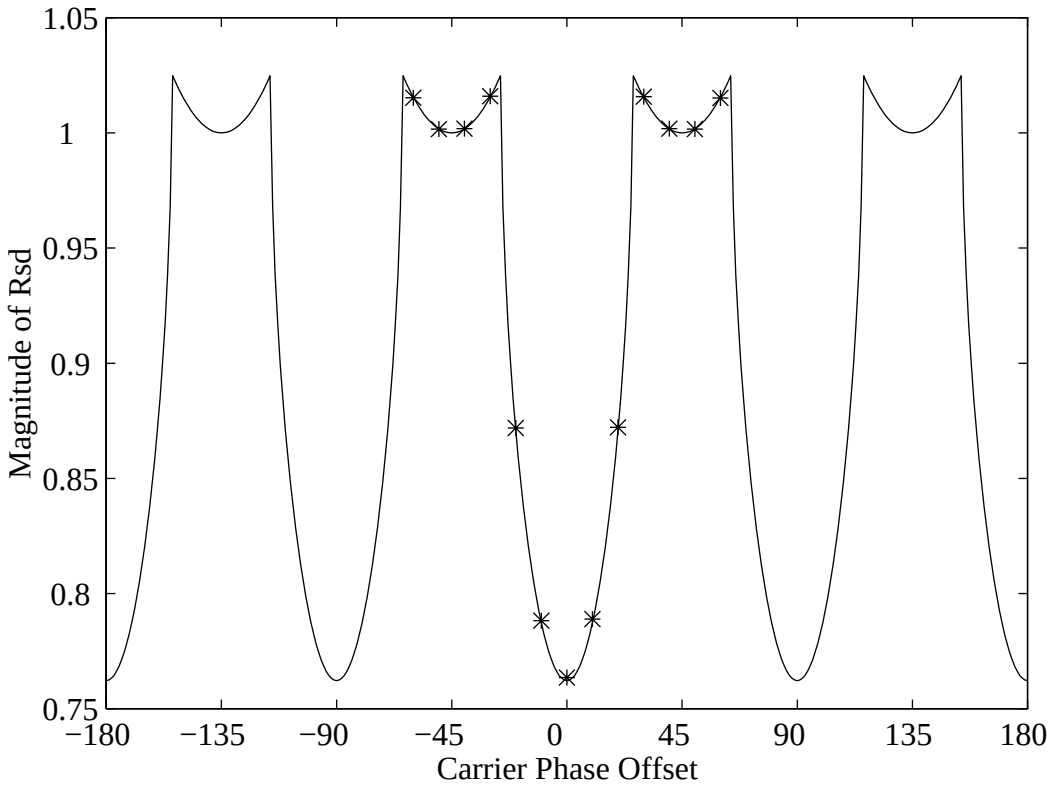

Figure 5.3: Magnitude of R_{sd} with a QPSK signal and CM interference versus the carrier phase offset at the demodulator input. The SIR at the demodulator input is +0.5 dB. Solid line denotes theoretical expression, '$*$' denotes result measured from simulations.

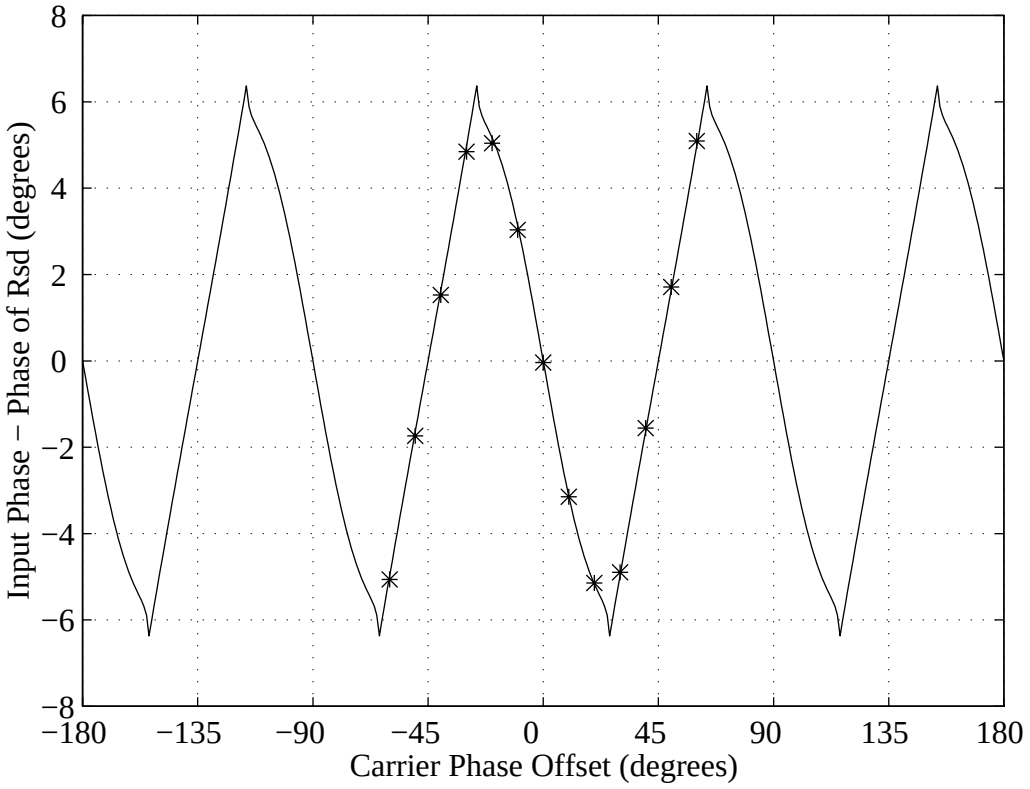

Figure 5.4: Difference in the angle of s at the demodulator input and demodulator output, with a QPSK signal and CM interference, versus the carrier phase offset at the demodulator input. The SIR at the demodulator input is +0.5 dB. Solid line denotes theoretical expression, '$*$' denotes result measured from simulations.

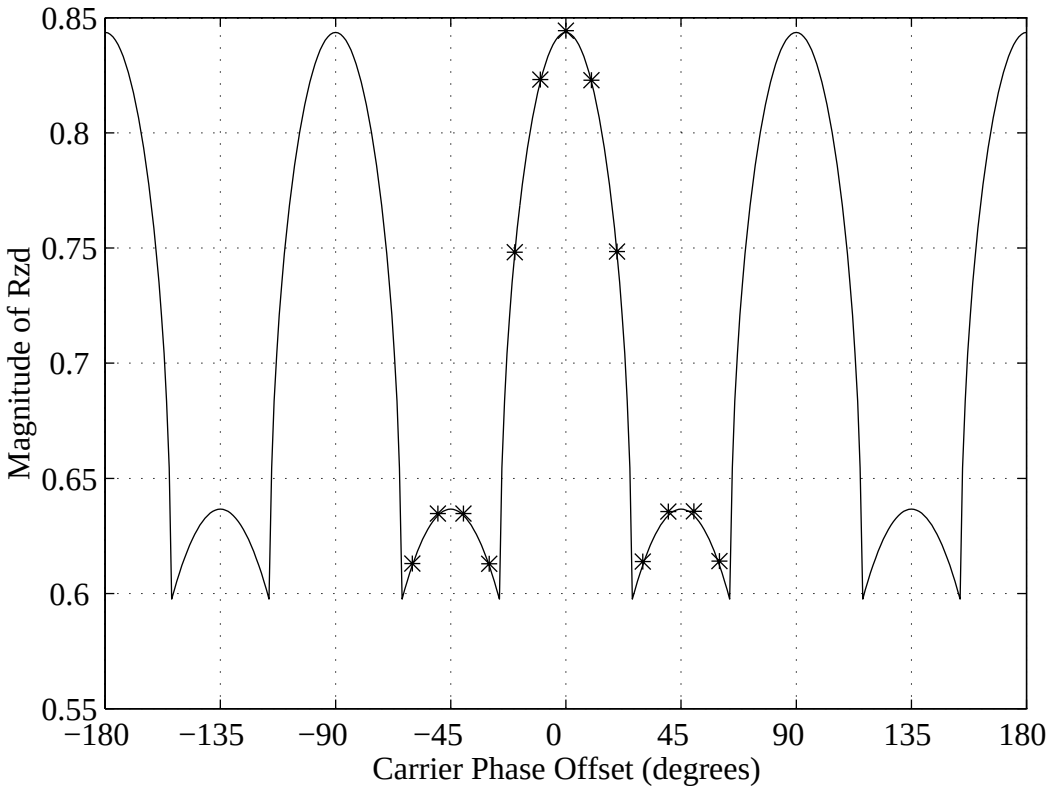

Figure 5.5: Magnitude of R_{zd} with a QPSK signal and CM interference versus the carrier phase offset at the demodulator input. The SIR at the demodulator input is +0.5 dB. Solid line denotes theoretical expression, '$*$' denotes result measured from simulations.

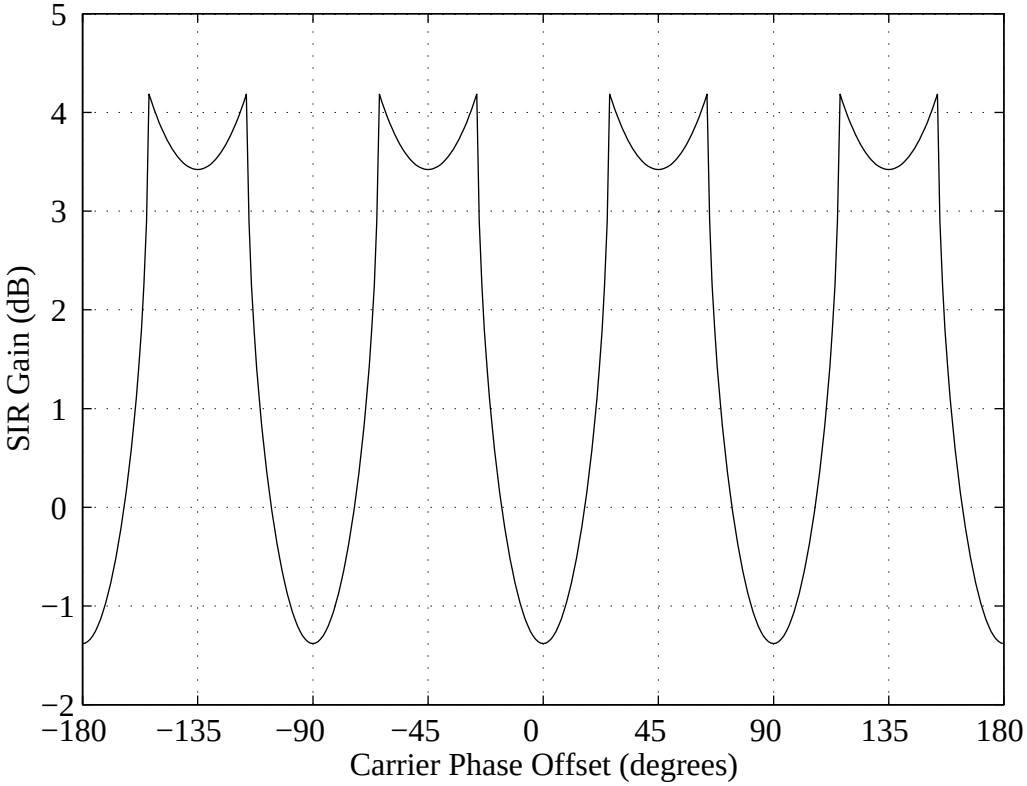

Figure 5.6: SIR gain with a QPSK signal and CM interference versus the carrier phase offset at the demodulator input. The SIR at the demodulator input is +0.5 dB.

Note that the SIR gain is equal to zero for an input SIR of approximatley 4 dB. The signal phase at the demodulator output remains equal to the phase at the input. This shows that the SIR will remain the same, and the signal phase will remain the same, so the LSDDA will remain at this solution. Thus this is a local minimum of the cost function. This is a new and unanticipated result.

We now perform a simulation of the LSDDA to see if it will in fact converge to this local minimum. We apply the LSDDA an environment with a 20.5 dB SWNR QPSK signal at 10°, and an uncorrelated 20 dB SWNR CM interfering signal at 65°. The phase of the CM interferer is random with uniform distribution. The initial carrier phase offset is 30°, and the LSDDA block size is 1024 symbols. Due to the effects of background noise and finite data block size, the LSDDA does not always converge to the local minimum identified above. However, for some realizations of data the LSDDA can converge to the solution shown in the I-Q plot of Figure 5.8. As a second example, we apply the LSDDA to a similar environment, except that the interfering signal is an uncorrelated QPSK signal. Again, the LSDDA is

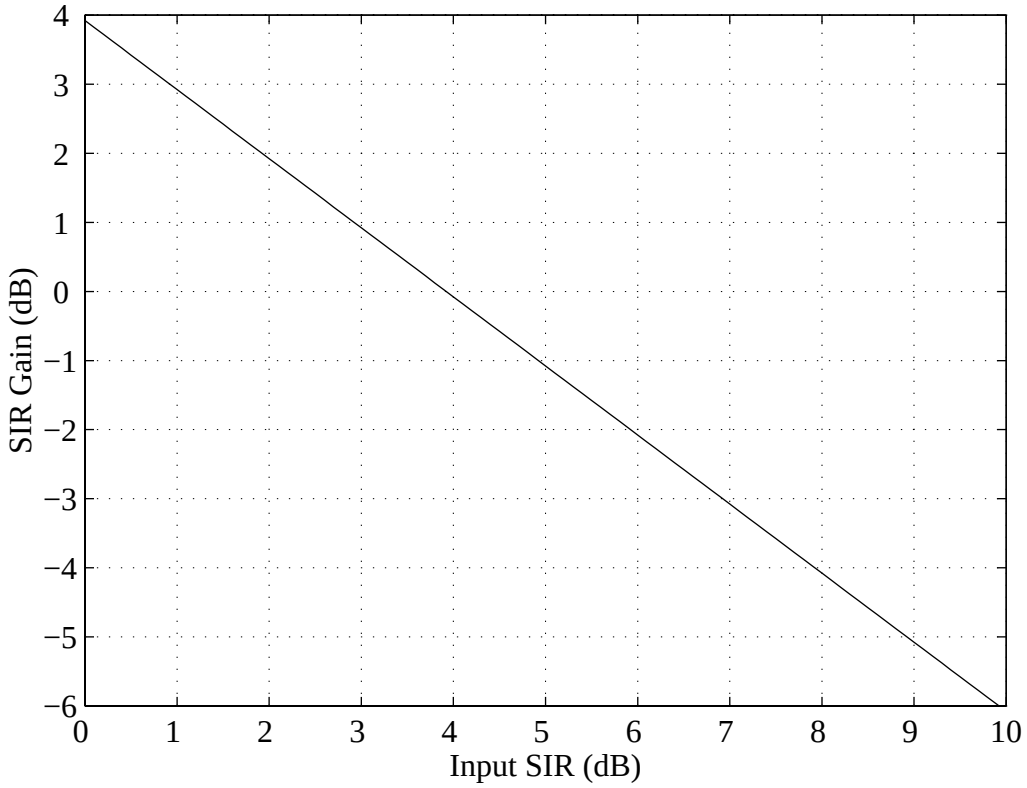

Figure 5.7: SIR gain with a QPSK signal and CM interference versus the SIR at the demodulator input. The carrier phase at the demodulator input is 45°.

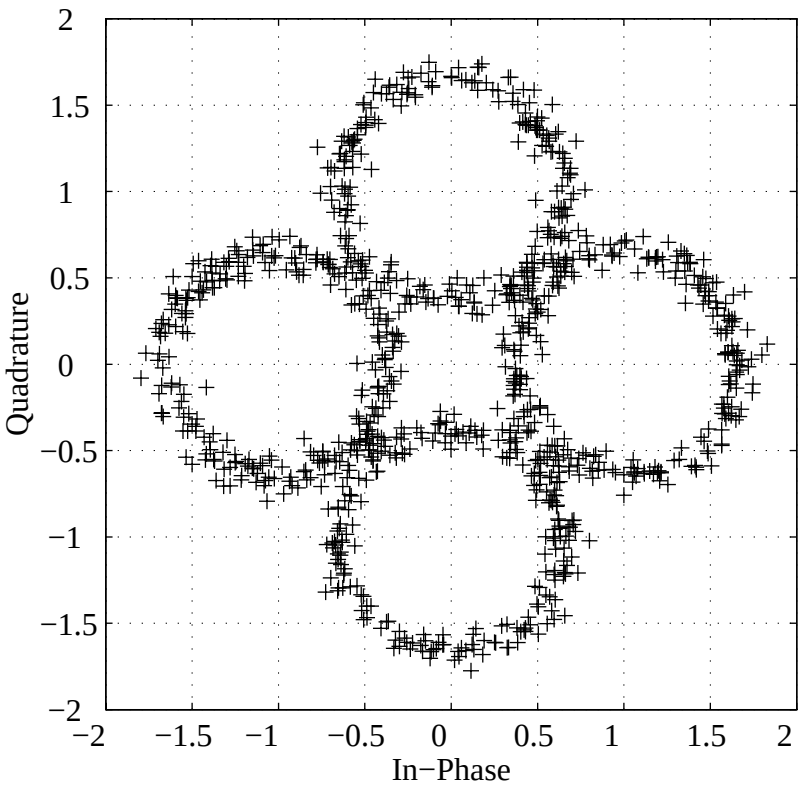

Figure 5.8: I-Q plot of the LSDDA beamformer when the algorithm converges to an undesired local minimum. The desired signal is QPSK, and the interferer is CM with random phase.

not always trapped in the local minimum, but it does sometimes converge to the solution illustrated in Figure 5.9. In order to better understand the behavior of the LSDDA in this environment, we examine the LSDDA cost function in the next section.

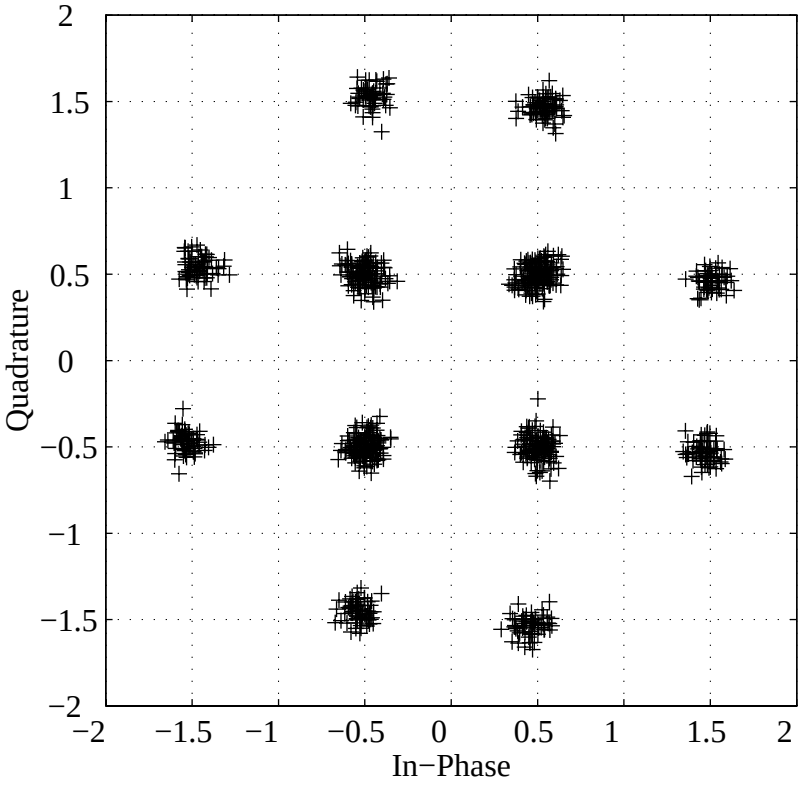

Figure 5.9: I-Q plot of the LSDDA beamformer when the algorithm converges to an undesired local minimum. The input is two independent QPSK signals.

5.4 Cost Function Analysis

We show in Appendix A that the decision-directed cost function has the general form

$$F(\rho,\theta) = 2 - \frac{2}{\sqrt{1+\rho}}\text{Real}\left\{\alpha^* e^{j\theta}\sqrt{\rho} + \beta^*\right\} \tag{5.21}$$

where ρ is the beamformer output SINR and θ is the phase at the demodulator input. Note that α and β are dependent on ρ and also on θ.

We begin with the simple case of a QPSK signal in CM interference. By evaluating (5.21) with α given by (5.7) and β given by (5.8), we obtain the function shown in Figure 5.10. The cost function peaks near an SIR of +1.5 dB, and decreases for both SIR greater than +1.5 dB and *less than* +1.5 dB. This shows clearly that a local minimum exists for capture of the CM interferer. Thus the DDA can be said to exhibit tone capture, just as CMA [66].

We next consider the more complicated cost function that arises when a QPSK signal having carrier offset is received in CM interference. By evaluating (5.21) with α given by (5.17) and β given by (5.20), we obtain the function shown in Figure 5.11. This figure shows a possible local minimum corresponding to an output SINR of approximately 4 dB and output phase of 45°. Figure 5.12 shows the cost function versus SIR with the phase fixed at 45°. The cost function minimum occurs at an output SINR of 3.92 dB. Figure 5.13 shows the cost function versus output phase with the SIR fixed at 3.92 dB. This figure shows that a local minimum is present at 45°. Since a minimum occurs in both the SIR and the output phase, we conclude that a local minimum exists in the DDA cost function at an output SIR of 3.92 dB and output phase of 45°. This corresponds to the undesirable LSDDA convergence shown in Figures 5.9 and 5.8. This verifies the existence of the undesired solution.

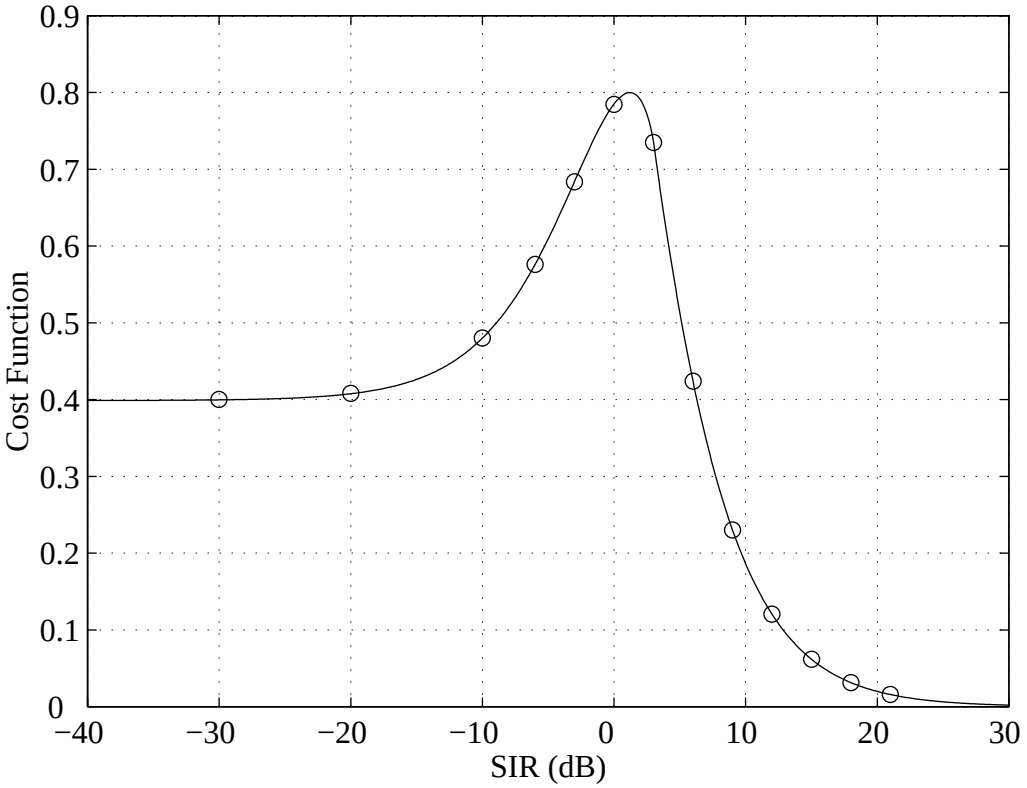

Figure 5.10: Decision-directed cost function for a QPSK signal with known carrier phase in constant modulus interference.

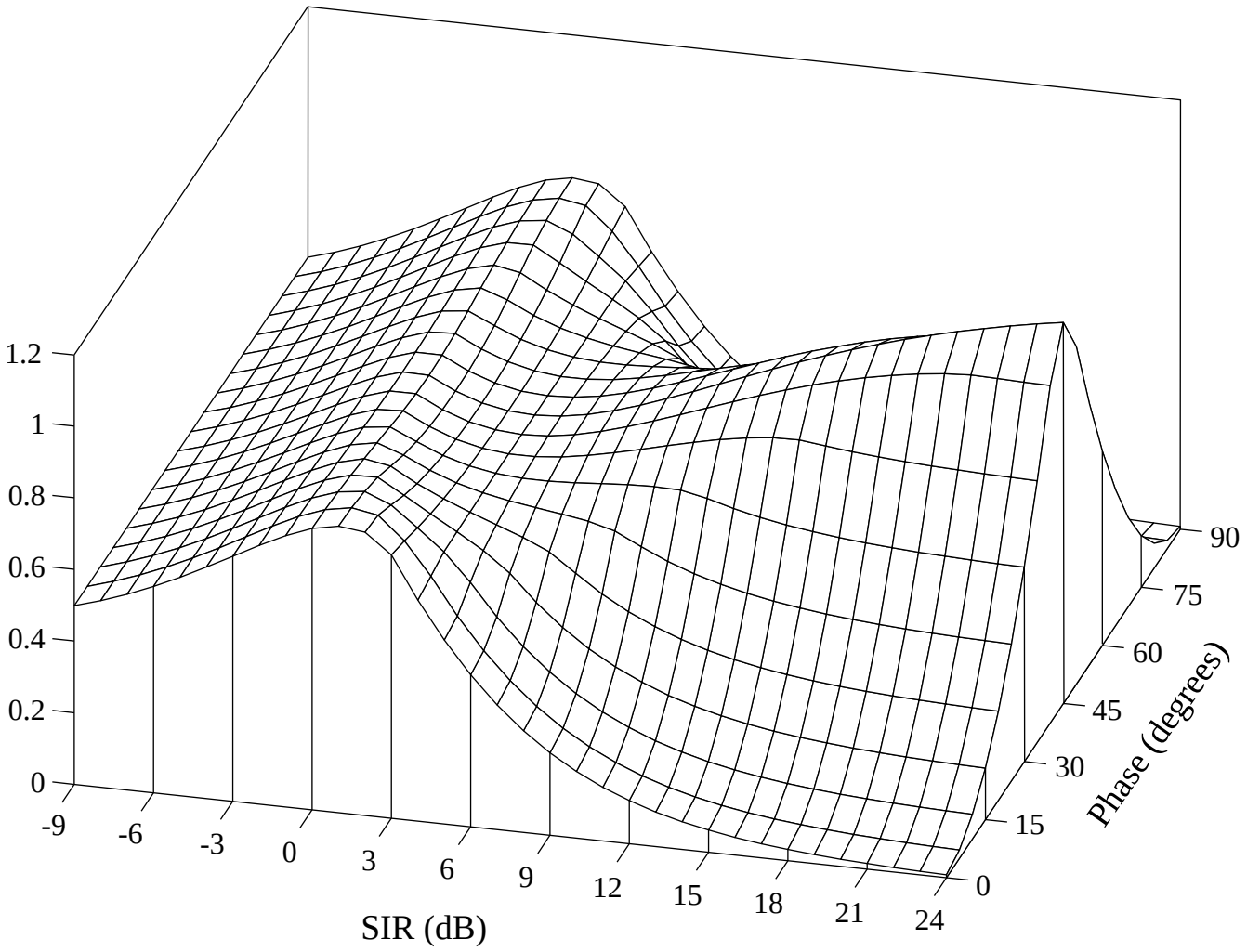

Figure 5.11: Decision-directed cost function for a QPSK signal in constant modulus interference as a function of carrier phase offset and SIR.

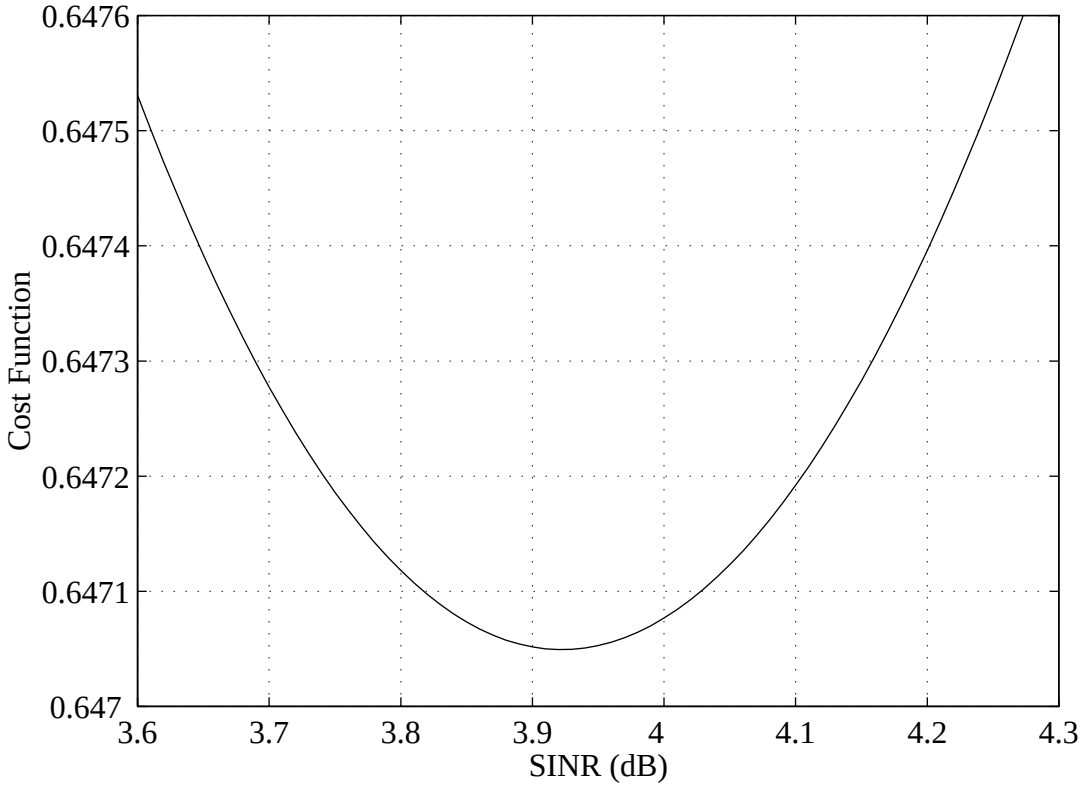

Figure 5.12: Decision-directed cost function for a QPSK signal in constant modulus interference as a function of SINR, corresponding to output phase of 45°.

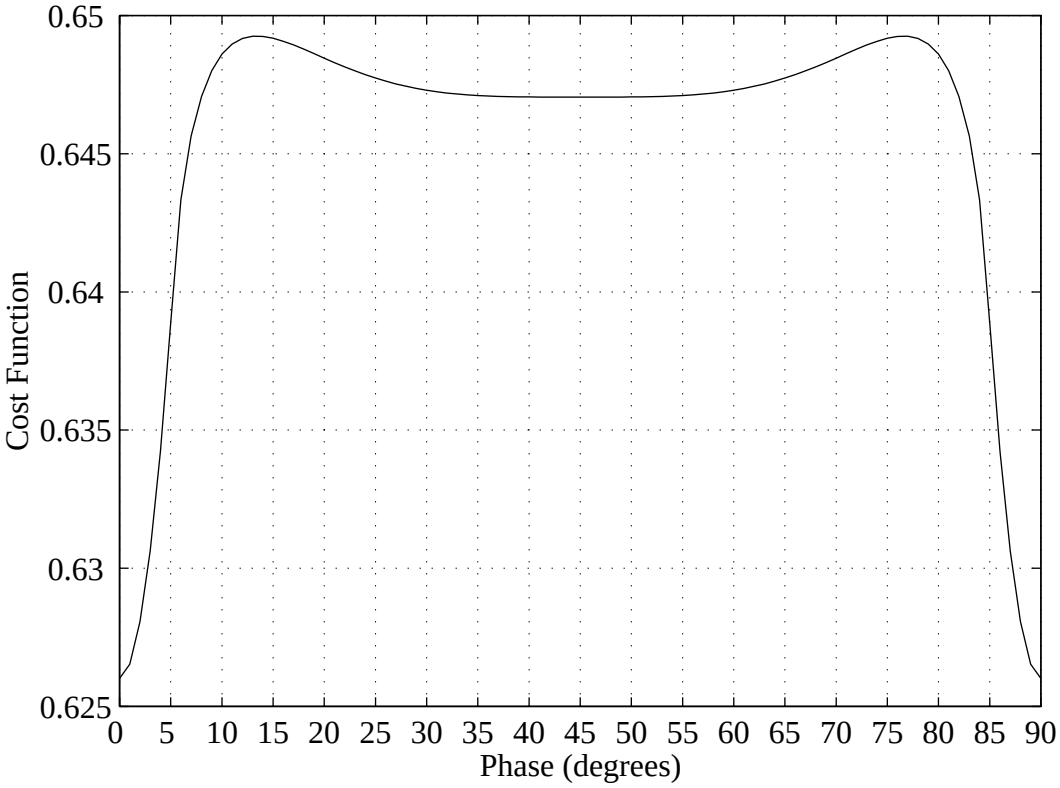

Figure 5.13: Decision-directed cost function for a QPSK signal in constant modulus interference as a function of phase, corresponding to output SINR of 3.92 dB.

5.5 Conclusions

We have shown that the behavior of the LSDDA with CM interference is much more complicated than in Gaussian interference. In particular, the LSDDA is susceptible to capture of a CM interferer when the initial SINR is low. Furthermore, the LSDDA is *more* susceptible to capture of a CM interferer than LSCMA. This non-intuitive behavior is predicted by the analysis presented here.

The LSDDA is also susceptible to being trapped in a local minima when desired signal is QPSK and the interference is CM. This local minima corresponds to a state where the both the QPSK signal and the CM interferer are present in the demodulator output. The LSCMA does not exhibit this undesired behavior.

Based on the results obtained here, it appears that the CMA has more reliable convergence properties compared to the LSDDA when the initial SINR is low and the interference is CM. It might be postulated that this conclusion would generalize to other more complicated, and hence more realistic, environments.

Chapter 6

Directions for Future Research

Summary

This chapter outlines several potential remaining research topics to be explored. Some of these ideas are more well-developed than others. Significant work has been accomplished in some research areas, including the Least Squares General Modulus Algorithm, introduced in Section 6.4. The discussion presented in this chapter can be viewed as a 'shopping list' for research topics, and is meant to be a starting point for discussions.

6.1 Constant Modulus Algorithm

In this section we outline additional research that can be performed with the CMA using some of the basic principles outlined in this dissertation.

6.1.1 Stochastic Gradient Descent

The SGD algorithm has been studied extensively for the case where a known training signal is available. However, similar results for blind SGD algorithms are generally not available, mainly because an appropriate model for the 'training signal' used in a blind algorithm has not been identified. We will now seek to show that the model for the hard-limiter output introduced in the document may be helpful in analyzing certain SGD versions of CMA, particularly the (1,2) version. Specifically, the model

$$d(n) = \alpha s(n) + \beta z(n) + \xi(n) \tag{6.1}$$

can be used for the hard-limiter output, and then existing results on SGD algorithms with known training signals can be applied. We will now show that this procedure should be possible in principal.

We know that the SGD algorithm converges to the LS solution in the limit as $n \rightarrow \infty$ for sufficiently small adaptive step size μ. By analogy, we would expect the SGD CMA to converge to the solution found after one LSCMA update in the limit as $n \rightarrow \infty$ for sufficiently small adaptive step size μ. That is, consider the signal $d_0(n)$, where

$$y_0(n) = \mathbf{w}_0^H \mathbf{x}(n) \tag{6.2}$$

$$d_0(n) = \frac{y_0(n)}{|y_0(n)|} \tag{6.3}$$

and $\mathbf{w}_0$ is the initial weight vector. The updated LSCMA weight vector is

$$\mathbf{w}_1 = \mathbf{R}_{xx}^{-1} \mathbf{R}_{xd_0} \tag{6.4}$$

If we implement the SGD algorithm using $d_0(n)$ as the training signal, then the weight vector found by the SGD method should approach the weight vector found by the LSCMA in one iteration. To demonstrate that this is true, we return to a simple environment used to help analyze the LSCMA: two orthogonal complex sinusoids with no background noise. The behavior of the LSCMA in this environment is deterministic – the SGD behavior in this environment should be also. We simulate a two-element $\lambda/2$ array, with one sinusoid incident from 0°, and a second sinusoid incident from 65°. The sinusoid at 0° has frequency equal to 5/1024, and power equal to 9 dB. The sinusoid at 65° has frequency equal to -31/1024, and power equal to 6 dB. When the LSCMA is applied to this environment, the output SIR after one iteration is 7.04 dB. If the same training signal $d_0(n)$ is used by a SGD algorithm instead a direct LS algorithm (as used by LSCMA), we would expect the SGD algorithm to converge to the same output SIR of 7.04 dB. This is verified by the results shown in Figure 6.1. A block update version of the (1,2) SGD CMA is implemented, where the weight vector update is

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mu \langle \, \mathbf{x}(n) e^*(n) \, \rangle \tag{6.5}$$

and $\langle \cdot \rangle$ denotes a time average over the entire data block of $N = 1024$ temporal samples. The adaptation step size $\mu = 0.02$, and the initial weight vector $\mathbf{w}_0 = [\ 1\ 0\ ]^T$. This shows that the non-linear model for the hard-limiter output should also be useful for analyzing SGD versions of CMA.

6.1.2 Pulse-Shaped Signals

It should be possible to use the same general framework to analyze CMA performance with non-CM signals, such as pulse-shaped PSK. The analysis procedure would still be the same,

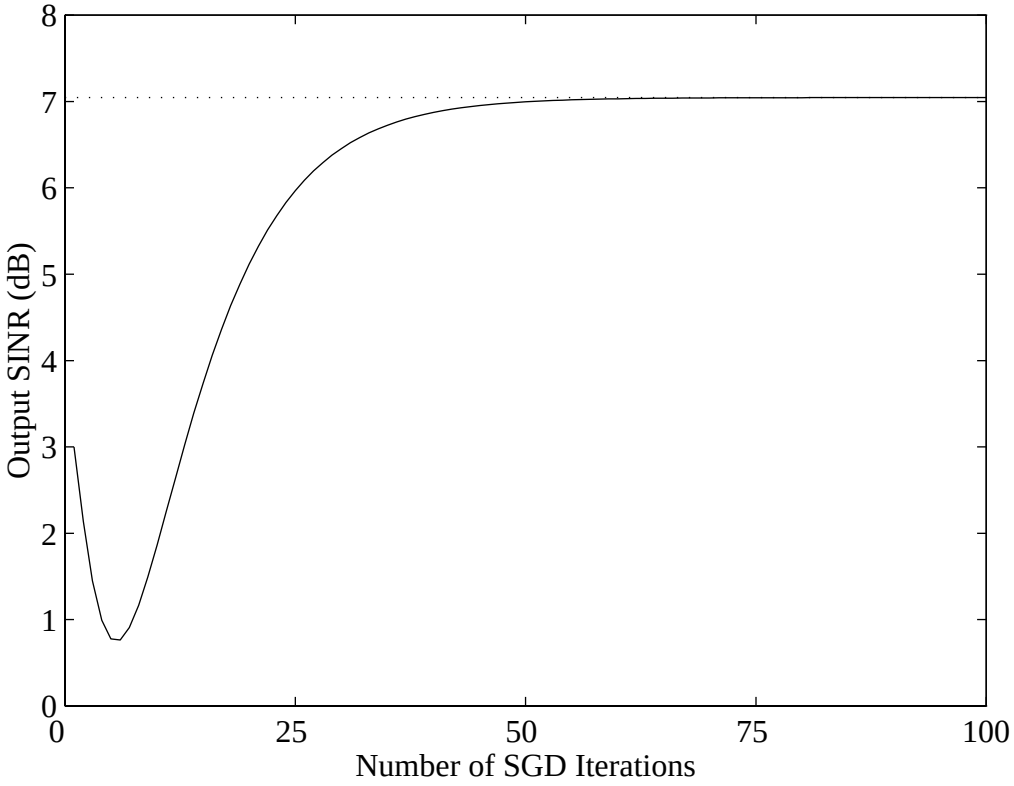

Figure 6.1: Output SIR of the (1,2) SGD CMA with a fixed training signal, derived by hard-limiting the initial beamformer output. The LSCMA output SIR after one iteration is indicated by the horizontal dotted line. This shows that the SGD CMA converges to the same solution as the LSCMA.

relying on the calculation of R_{sd} and R_{zd}. The only difficulty would be that an expression for the amplitude distribution of a pulse-shaped PSK signal would be needed. This pdf does not appear to be well-known, but may be found after a more diligent literature search. If this proves to be impractical, an alternative approach would be to examine CMA performance with an AM signal. The simplest case would be a tone-modulated AM signal. It may prove useful to consider the AM case initially as a starting point at any rate. The analysis of LSCMA for AM signals might be sufficient for a peer-reviewed journal publication. The analysis of LSCMA for pulse-shaped signals is of interest to a broader audience, and would probably require more effort, and so would probably be sufficient for publication in a peer-reviewed journal publication.

6.2 Decision Directed Algorithm

In this section we outline future research for the analysis and development of decision directed algorithms.

6.2.1 Cost Function Stationary Points

In order to perform a thorough analysis of decision-directed (DD) algorithms, or in fact any blind algorithm, it is necessary to find the stationary points of the cost function. A stationary point of the cost function would correspond to a weight vector for which the cost function gradient with respect to the weight vector is equal to zero. The DD cost function can be written as

$$F(\mathbf{w}) = \mathcal{E}\left\{ \left| \mathbf{w}^H \mathbf{x}(n) - f\left(\mathbf{w}^H \mathbf{x}(n)\right) \right|^2 \right\} \tag{6.6}$$

where $f(\cdot)$ denotes the appropriate demodulation function. The gradient of the cost function with respect to the weight vector cannot be solved for directly. If this could be accomplished, the weight vector that minimizes the cost function (6.6) could be solved for directly without the need for iterative algorithms. Nevertheless, a method has been identified to find the stationary points of (6.6). This method calls for expressing the cost function in terms of the signal amplitude, noise and interference amplitude, signal phase, and noise phase. In the case of Gaussian noise and interference, the cost function is not dependent on the noise phase, since the noise is assumed to be circularly symmetric.

We will set up the problem for the simplest case of a PSK signal with known carrier phase in Gaussian noise and interference. Note that the beamformer output $\mathbf{w}^H \mathbf{x}(n)$ can be expressed in terms of a signal component $s(n)$ and noise component $z(n)$ as

$$\mathbf{w}^H \mathbf{x}(n) = \sqrt{\rho} s(n) + z(n) \tag{6.7}$$

where ρ is the SINR. If we define

$$e(n) = \mathbf{w}^H \mathbf{x}(n) - f\left(\mathbf{w}^H \mathbf{x}(n)\right) \tag{6.8}$$

then we can write, using the general framework developed here,

$$e(n) = \sqrt{\rho} s(n) + z(n) + \xi(n) - \alpha(\rho) s(n) - \beta(\rho) z(n) \tag{6.9}$$

Recall that for the special case of known carrier phase, the variables α and β are *real*. Also recall that we have defined $s(n)$ and $z(n)$ earlier so that they have unit variance. Then we can express (6.6) in terms of output SINR ρ as

$$\begin{aligned} F(\rho) &= \mathcal{E}\left\{ |e(n)|^2 \right\} && (6.10) \\ &= \left(\rho - 2\sqrt{\rho}\alpha(\rho) + \alpha^2(\rho)\right) + \left(1 - 2\sqrt{\rho}\beta(\rho) + \beta^2(\rho)\right) + \sigma_\xi^2 && (6.11) \end{aligned}$$

To find the stationary points, we take the derivative of (6.11) with respect to ρ and set the result equal to zero. In order to determine whether the stationary points are minima, maxima, or saddlepoints, we must examine the second derivative. We expect to find one stationary point for BPSK, corresponding to a global minimum for SOI capture. We expect to find two stationary points for QPSK: one corresponding to a global minimum for SOI capture, and one corresponding to a saddlepoint for noise capture. Results for higher order PSK may be intractable. To consider the effect of carrier phase offset, the DD cost function can be expressed in terms of output SINR ρ and the phase offset θ. The case of constant modulus interference (when the interference has random phase) with carrier phase offset must take into account signal amplitude, signal phase, and interference amplitude. If the interference is PSK with non-random phase ϕ, the cost function must also include this variable.

6.2.2 Investigation of Other Decision Functions

The use of a hard-decision non-linearity may not be optimal. Other non-linearities may yield better performance. In particular, the so-called null-zone non-linearity might be worth investigating. This function is defined as

$$f(x) = \begin{cases} -1 & x < -c \\ 0 & -c \leq x \leq +c \\ +1 & x > +c \end{cases} \tag{6.12}$$

and is illustrated in Figure 6.2. This function in effect places zero weight on decisions that may have high probability of error. It should be straightforward, at least conceptually, to investigate other decision functions. The resulting integrals may not be as easily evaluated as for the hard-decision case, but we may rely on numerical integration in such cases.

Many other decision functions could be investigated as well, including the magnitude non-linearities discussed in Section 6.4.

6.2.3 QAM and Orthogonal Modulation

Results for the decision-directed algorithm have only been obtained for PSK modulation. It should be straightforward to obtain similar results for QAM, since this is a linear modulation like PSK. Extension to orthogonal modulation would also be very desirable. Examples of orthogonal modulation include binary FSK, and the 64-ary orthogonal modulation based on Walsh functions used in the IS-95 reverse link. Extension to orthogonal modulation may not be quite as straightforward as extension to QAM, but should not be too difficult.

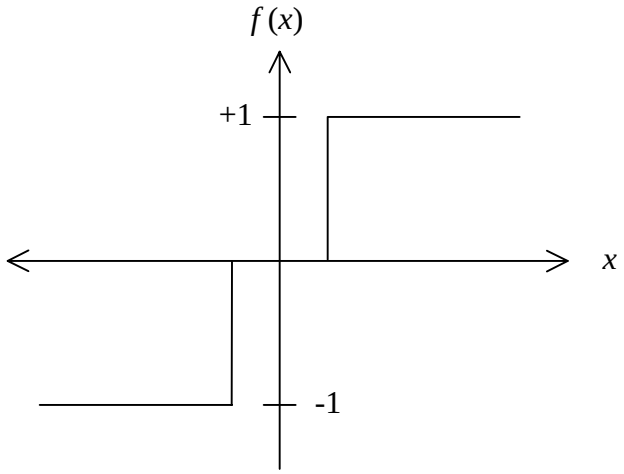

Figure 6.2: Null-zone decision non-linearity.

6.2.4 Extensions for Multiple Signals

The LSDDA is designed to extract a single signal. Other algorithms exist which are designed to simultaneously extract multiple signals. These algorithms typically converge more rapidly since they exploit more information about the environment. Two approaches for extracting multiple signals are the

1. Multi-Target LSDDA - This algorithm essentially applies multiple LSDDA beamformers to the same set of received data, and performs an additional layer of processing to ensure that each beamformer extracts a unique signal. This is typically done with a so-called 'soft orthogonalization', where the output signals of each independent LSDDA beamformer are made to have low correlation rather than zero correlation. This is a generalization of the MT-CMA described in [46].

2. Iterative Least Squares with Projection (ILSP) - This algorithm uses an alternating projections approach, alternating between estimating the transmitted signals by beamforming and making hard decisions, and using these estimated signals to estimate the spatial signatures via (2.65). This algorithm is reviewed in Subsection 2.5.2.

It should be straightforward to investigate at least the mean convergence of both the MT-LSDDA and the ILSP algorithms using the general framework introduced here. The only additional work needed would be to determine the correlation of two signal estimates $d_1(n)$ and $d_2(n)$.

6.2.5 Extension to SGD

The general non-linear model introduced may be useful for investigating the popular Stochastic Gradient Descent (SGD) decision-directed algorithms. The argument for why this should be true follows that presented for generalizing the LSCMA analysis to SGD CMA.

6.3 LSCMA and LSDDA Statistics for Finite N

The analytic results presented for the LSCMA and LSDDA to this point have been for the *asymptotic* behavior. Exact calculation of the mean and distribution of the output SINR for finite data block size N would of course be of great interest, but would be much more challenging to obtain. However, the mean and variance of the cross-correlation vector $\mathbf{R}_{xd}$ used in LSCMA and LSDDA are relatively easy to find, and are presented in Section 4.7. This demonstrates that the non-linear model for the hard-limiter output and PSK demodulator output can be used in principal to determine the exact behavior of LSCMA and LSDDA for finite N. Such an analysis would rely on principles from multivariate statistics.

6.3.1 Decision-Directed Phase Tracking

One interesting topic would be to apply the methods used to analyze the LSDDA to the analysis of a decision-directed carrier phase tracker. A literature search for such an analysis has not been performed, so it is not known if such an analysis has already been performed. However, it should be fairly straightforward to take the results presented in Section 4.7 for a decision-directed estimate of signal amplitude and generalize them to a decision-directed estimate of *complex* signal amplitude.

6.4 General Modulus Algorithm

In this section we introduce a novel blind adaptive algorithm that is motivated by the general framework used to analyze the LSCMA and the LSDDA. This algorithm is referred to here as the General Modulus Algorithm (GMA). This algorithm replaces the hard-limit non-linearity used in CMA with a general magnitude-only non-linearity. One particular version of the GMA simply raises the modulus to a certain power. This version is particularly easy to analyze, and offers some potential for faster convergence relative to CMA. We also consider the possibility of using other non-linearities to derive a training signal, with the

goal of improving the rate and reliability of algorithm convergence.

6.4.1 Overview of the Least Squares General Modulus Algorithm

The LSCMA essentially uses the signal

$$d(n) = e^{j\phi(n)} \tag{6.13}$$

as a training signal, where

$$\phi(n) = \angle y(n) \tag{6.14}$$

is the angle of the beamformer output $y(n) = \mathbf{w}^H\mathbf{x}(n)$. This is a magnitude-only non-linearity, since the phase is not modified. Now consider the possibility of using the signal

$$d(n) = m^p(n)e^{j\phi(n)} \tag{6.15}$$

as a training signal, where $\phi(n)$ is the phase as defined above,

$$m(n) = |y(n)| \tag{6.16}$$

is the magnitude, and p is some real scalar. This algorithm will satisfy the Bussgang criterion [82]. Note that with $p = 0$, the algorithm reduces to the LSCMA. Also note that with $p = 1$, we have $d(n) = y(n)$, so the 'non-linearity' has no effect.

6.4.2 SINR Gain for High Initial SINR

The SINR gain provided by the non-linearity defined in (6.15) can be derived directly for the case where the input SINR is high. We use the same approach as employed in Subsection 3.3.1 to analyze the LSCMA for high SINR. Let the initial beamformer output be

$$\begin{aligned} y(n) &= s(n) + gz(n) && (6.17)\\ &= e^{j\phi(n)} + gm(n)e^{j\psi(n)} && (6.18) \end{aligned}$$

where $\phi(n)$ is the phase of the desired signal, while $m(n)$ and $\psi(n)$ are the magnitude and phase, respectively, of the noise and interference. The 'training signal' $d(n)$ can be written as

$$d(n) = y(n)\left[y(n)y^*(n)\right]^u \tag{6.19}$$

where

$$u = \frac{p-1}{2} \tag{6.20}$$

We first examine the cross-correlation of the desired signal $s(n)$ with the training signal $d(n)$. We have

$$\begin{aligned} R_{sd} &\triangleq \mathcal{E}\{\, s(n)d^*(n)\,\} && (6.21)\\ &= \mathcal{E}\{\, s(n)y^*(n)\,[y(n)y^*(n)]^u\,\} && (6.22)\\ &= \mathcal{E}\Big\{\, \Big(1+gm(n)e^{j\Delta}\Big)\,\Big[1+2gm(n)\cos\Delta+g^2m^2(n)\Big]^u\,\Big\} && (6.23)\end{aligned}$$

where $\Delta = \phi(n) - \psi(n)$. For high SINR, we have $g \ll 1$, and we therefore ignore terms containing g^2. Furthermore, the binomial expansion gives

$$(1+x)^u \approx 1+ux \tag{6.24}$$

so that

$$R_{sd} \approx \mathcal{E}\Big\{\, \Big(1+gm(n)e^{j\Delta}\Big)\,\Big(1+2m(n)u\cos\Delta+g^2m^2(n)\Big)\,\Big\} \tag{6.25}$$

If we again ignore terms containing g^2, and assume that Δ is uniformly distributed over $(0, 2\pi]$, we have

$$R_{sd} \approx 1 \tag{6.26}$$

We now calculate the cross-correlation of the interference and noise $z(n)$ with the training signal. Using similar arguments as made above for R_{sd}, we have

$$\begin{aligned} R_{zd} &\triangleq \mathcal{E}\{\, z(n)d^*(n)\,\} && (6.27)\\ &= \mathcal{E}\Big\{\, \Big(m(n)e^{-j\Delta}+gm^2(n)\Big)\,\Big[1+2gm(n)\cos\Delta+g^2m^2(n)\Big]^u\,\Big\} && (6.28)\\ &\approx \mathcal{E}\Big\{\, \Big(m(n)e^{-j\Delta}+gm^2(n)\Big)\,(1+2ugm(n)\cos\Delta)\,\Big\} && (6.29)\\ &\approx g\mathcal{E}\Big\{\, m^2(n)\,\Big\}+ug\mathcal{E}\Big\{\, m^2(n)\,\Big\} && (6.30)\\ &\approx g(1+u) && (6.31)\end{aligned}$$

We have also made use of the fact that $z(n)$ is unit variance. The output SIR in the training signal (6.15) is therefore

$$\mathrm{SIR} = \frac{R_{sd}^2}{R_{zd}^2} \approx \frac{1}{g^2\,(1+u)^2} \tag{6.32}$$

Since the input SIR is $1/g^2$, the SIR gain is

$$\mathrm{SIR}_{\mathrm{gain}} = \frac{\mathrm{SIR}_{\mathrm{output}}}{\mathrm{SIR}_{\mathrm{input}}} = \frac{1}{(1+u)^2} \tag{6.33}$$

Expressing this as a function of p where p is defined in (6.15), we have

$$\mathrm{SIR}_{\mathrm{gain}} = \frac{4}{(1+p)^2} \tag{6.34}$$

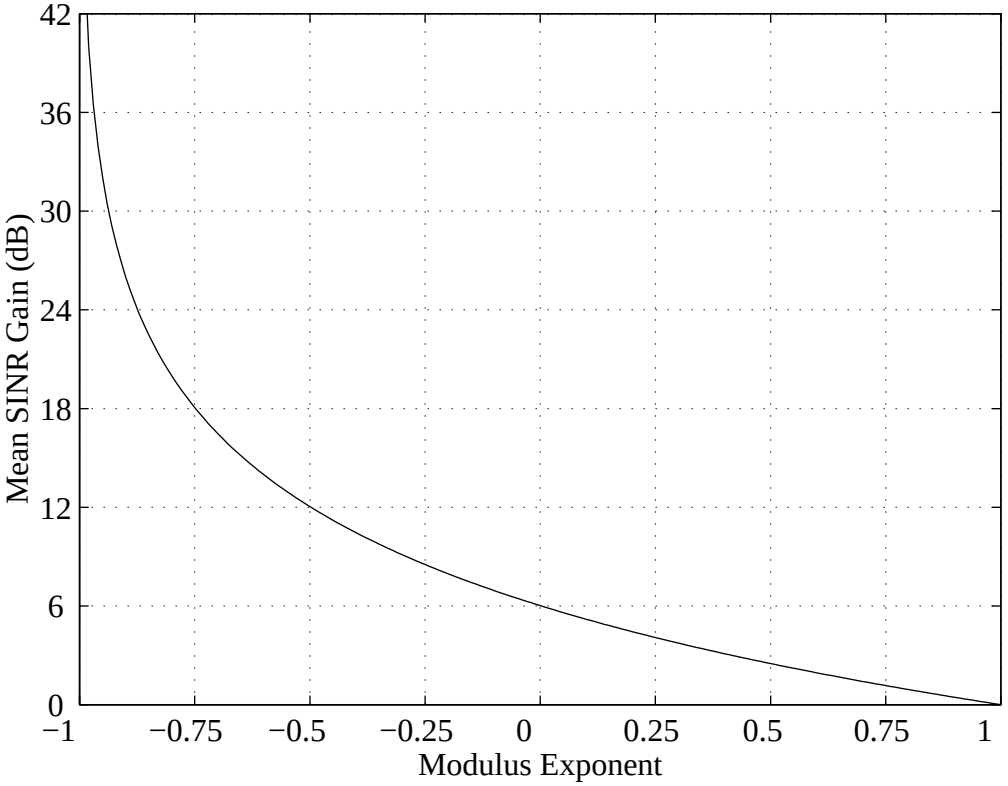

Figure 6.3: Theoretical SINR gain per iteration at high SINR as a function of modulus exponent.

Note that the SINR gain is equal to 4 (i.e., 6 dB) for $p = 0$, which is consistent with the results for LSCMA obtained in Subsection 3.3.1.

A plot of the GMA SINR gain for high initial SINR is shown in Figure 6.3. Note that the SINR gain tends to infinity as $p \to -1$. Previous analysis indicates that the LSGMA would converge in a single iteration if the SINR gain is infinite. However, the mean analysis does not consider the effect of the intermodulation terms. These intermodulation terms have no effect for $N \to \infty$, but do have an effect for finite N. Thus the behavior for finite N is not immediately obvious. Preliminary simulation results have indicated that the LSGMA does hold promise for faster convergence relative to LSCMA.

6.5 Extension to Equalization Applications

The analysis framework developed here has only been applied to the adaptive beamforming problem. It should be possible to apply the same principles to an equalization or RAKE application.

6.6 Exploitation of Multiple Non-Linearities

One way to improve the performance of blind adaptive algorithms may be to exploit multiple cost functions. For example, it may be beneficial to jointly exploit the CM property and the finite-alphabet property exhibited by PSK signals. One approach to exploiting multiple non-linearities is to use the Recursive PCCA framework.

6.7 Beamforming with Non-Coherent Demodulation

A common approach to finding a nearly optimal set of antenna weights is to minimize the mean squared error between the received signal and a known training sequence. In order to calculate the Least Squares (LS) weight vector, the system must know the carrier frequency and symbol timing of the desired signal. However, estimation of these parameters is particularly difficult in low SINR situations. Therefore it is important to investigate adaptive beamforming methods which are either robust to synchronization parameter estimation error, or eliminate the need for knowledge of these parameters.

6.8 Update Rate Requirements

The update rate requirements for an adaptive array can have great impact on system requirements. If the environment is changing rapidly, the weight must be updated very rapidly, and converge rapidly, which places high demands on the system hardware and software. The update rate requirements for adaptive arrays in fading environments have not been thoroughly studied. Some preliminary results have been obtained in this area, based on the following procedure. For some time-varying, multipath-fading environment, calculate the optimal weight vector for time $t = 0$. Then determine the output SINR with this fixed weight vector as a function of time. Then measure the time interval over which the output SINR is acceptable. This is one way to describe the rate at which the weight vector must be

updated. This issue can be studied using various models for the environment, and can also be used to study live data collected in field experiments.

6.9 Adaptive Arrays for CDMA Applications

Most of the wok presented in this document has centered on conventional, non-spread spectrum modulation. However, the use of Code Division Multiple Access (CDMA) schemes in mobile wireless applications is growing rapidly. There are several motivations for this. First, the use of a wideband spread spectrum modulation allows resolution of closely spaced multipath components, which can then be combined using a RAKE receiver. This provides a form of diversity which combats fading and improves performance. A second motivation is that the wider bandwidth typically employed in CDMA systems allows for the transmission of higher data rates. Therefore the use of adaptive antenna arrays in CDMA may be a worthwhile field for further investigation.

One promising approach has been proposed by Naguib [53]. We will refer to this approach as the Code Gated Algorithm (CGA) due to its similarity to time-gated and frequency-gated algorithms. The CGA calls for setting the weight vector equal to the eigenvector corresponding to the dominant eigenvalue of the generalized eigenequation

$$\mathbf{R}_{xx}\mathbf{w} = \lambda \mathbf{R}_{\text{despr}}\mathbf{w} \tag{6.35}$$

The covariance matrix of the observed array data asymptotically approaches

$$\mathbf{R}_{xx} = \sigma_s^2 \mathbf{a}_s \mathbf{a}_s^H + \mathbf{R}_{qq} \tag{6.36}$$

The covariance matrix of the despread array data asymptotically approaches

$$\mathbf{R}_{\text{despr}} = G\sigma_s^2 \mathbf{a}_s \mathbf{a}_s^H + \mathbf{R}_{qq} \tag{6.37}$$

where G is the spreading gain of the system. The dominant eigenvector of (6.35) maximizes the quantity

$$F = \frac{\mathbf{w}^H \mathbf{R}_{xx} \mathbf{w}}{\mathbf{w}^H \mathbf{R}_{\text{despr}} \mathbf{w}} \tag{6.38}$$

$$= \frac{G\sigma_s^2 \left|\mathbf{w}^H \mathbf{a}_s\right|^2 + \mathbf{w}^H \mathbf{R}_{qq} \mathbf{w}}{\sigma_s^2 \left|\mathbf{w}^H \mathbf{a}_s\right|^2 + \mathbf{w}^H \mathbf{R}_{qq} \mathbf{w}} \tag{6.39}$$

Define the output SINR as

$$\gamma = \frac{\sigma_s^2 \left|\mathbf{w}^H \mathbf{a}_s\right|^2}{\mathbf{w}^H \mathbf{R}_{qq} \mathbf{w}} \tag{6.40}$$

Environment #1	
Array Geometry	4 Element Uniform Linear Array $\lambda/2$ Spacing
Data Modulation	QPSK
Spreading Code Modulation	QPSK
SOI AOA (degrees)	30
SOI SWNR	9 dB
Spreading Gain	32 (15.05 dB)
Number of Interferers	9
SNOI AOA (degrees)	-50 -45 -40 -35 -30 -20 10 20 45
SNOI SWNR	9 dB for all
Max SINR	11.42 dB

Table 6.1: Parameters for a simple stationary environment with zero delay spread and no multipath.

Then by dividing the numerator and denominator of (6.39) through by $\mathbf{w}^H\mathbf{R}_{qq}\mathbf{w}$, we obtain

$$F = \frac{G\gamma + 1}{\gamma + 1} \tag{6.41}$$

Now note that F is a monotonically increasing function of γ if $G > 1$. For this reason maximizing F is equivalent to maximizing the output SINR γ. Thus the CGA weight vector maximizes the beamformer output SINR. A block diagram of the CGA is presented in 6.4.

Some preliminary results on the relative performance of several CDMA beamforming algorithms have been obtained. A key figure of merit for any adaptive algorithm is the integration time required for convergence. To determine the required integration time we examine the mean output SINR versus integration time in a simple stationary environment. The environment parameters are listed below in Table 6.9.

The mean output SINR of several algorithms of interest are shown in Table 6.9.

	Mean Output SINR in Stationary Environment					
N	LS-Chip	CGA	Beam	CMA	LS-Symbol	DDA
4	11.09	11.06	10.58	5.36	3.36	3.34
8	11.26	11.24	10.90	9.25	9.04	9.04
16	11.35	11.34	11.04	10.41	10.45	10.45
32	11.38	11.38	11.13	10.91	10.98	10.98
64	11.40	11.40	11.17	11.15	11.20	11.19
128	11.41	11.41	11.19	11.28	11.31	11.31

Table 6.2: Relative algorithm performance in a stationary environment with 4 antennas. 'LS-Chip' refers to a block Least Squares method with known message operating on data sampled at the chip rate, CGA refers to the Code-Gated Algorithm, 'Beam' refers to the beam-steered method, CMA refers to the block Least Squares Constant Modulus Algorithm, 'LS-Symbol' refers to a block Least Squares method with known message operating on data sampled at the symbol rate, DDA refers to the block Least Squares Decision Directed Algorithm.

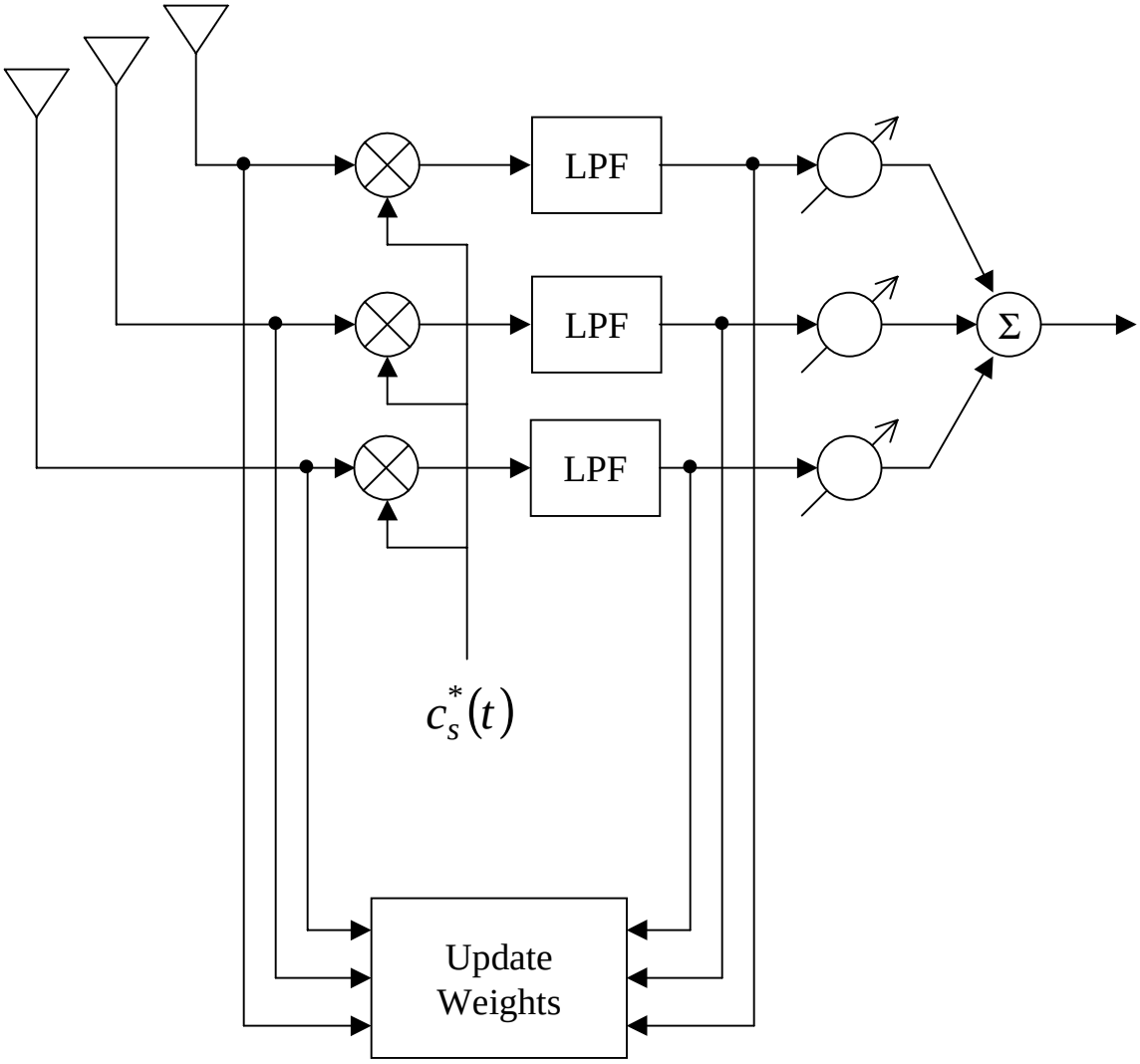

Figure 6.4: Block diagram of the Code-Gated Algorithm (CGA) for CDMA signals with known spreading code.

Appendix A

General Expression for Blind Adaptive Cost Functions

A general method for calculating the cost functions of a certain class of blind cost functions is described. This class includes the popular Constant Modulus Algorithm (CMA) and Decision Directed Algorithm (DDA). The method described here calls for expressing the cost function in terms of readily accessible signal parameters, such as SNR and carrier phase. This approach is described in detail below, and expressions are obtained for several important cost functions and signal/interference combinations, including:

1. (1,2) CMA cost function for CM signal and CM interference;
2. (1,2) CMA cost function for CM signal and Gaussian interference;
3. decision-directed cost function for a BPSK signal in Gaussian noise;
4. decision-directed cost function for a QPSK signal in Gaussian noise;
5. decision-directed cost function for a QPSK signal in CM interference.

Many blind adaptive algorithms have a cost function of the form

$$F(\mathbf{w}) = \mathcal{E}\Big\{ \left| y(n) - f\left(y(n)\right)\right|^2 \Big\} \tag{A.1}$$

where

$$y(n) = \mathbf{w}^H \mathbf{x}(n) \tag{A.2}$$

and $f(\cdot)$ is some memoryless non-linearity. For example, the (1,2) CMA cost function

$$F_{(1,2)} = \mathcal{E}\Big\{ \left(|y(n)| - 1\right)^2 \Big\} \tag{A.3}$$

can also be written as

$$F_{(1,2)} = \mathcal{E}\left\{ \left| y(n) - \frac{y(n)}{|y(n)|} \right|^2 \right\} \tag{A.4}$$

A decision-directed cost function also has this form, where $f(\cdot)$ denotes the appropriate demodulation function. For example, for a BPSK decision-directed cost function

$$f(y) = \text{sign}\,(y_r) \tag{A.5}$$

where y_r is the real component of $y(n)$. A decision-directed cost function for a QPSK signal uses the non-linearity

$$f(y) = \text{sign}\,(y_r) + j\,\text{sign}\,(y_i) \tag{A.6}$$

where y_i is the imaginary component of $y(n)$.

The first step in our analysis is to note that the cost function (A.1) can be written as

$$F(\mathbf{w}) = \sigma_y^2 - R_{yd} - R_{dy} - \sigma_d^2 \tag{A.7}$$

where

$$\sigma_y^2 = \mathcal{E}\{\, y(n)y^*(n)\,\} \tag{A.8}$$

and

$$d(n) = f\,(y(n)) \tag{A.9}$$

The cost function is dependent on the power of the linear combiner output $y(n)$. However, this has no effect on the symbol error rate or the SINR. Furthermore the output power is easily controlled so that it is equal to the demodulator output power. We therefore assume that $\sigma_y^2 = \sigma_d^2$. We also assume, without loss of generality, that $\sigma_d^2 = 1$. This gives

$$F = 2 - 2\text{Real}\,\{R_{yd}\} \tag{A.10}$$

This shows that minimizing the cost function F is equivalent to maximizing the real part of the cross-correlation between the beamformer output y and the demodulator output d. We now calculate R_{yd}.

The beamformer output $y(n) = \mathbf{w}^H\mathbf{x}(n)$ can be expressed as the sum of a signal component $s(n)$ and a noise component $z(n)$ as

$$\mathbf{w}^H\mathbf{x}(n) = gs(n) + cz(n) \tag{A.11}$$

where

$$\rho = \frac{|g|^2}{|c|^2} \tag{A.12}$$

is the SINR. Note that we allow g to be complex in order to allow for carrier phase offset at the demodulator input. That is,

$$g = |g|\ e^{j\theta} \tag{A.13}$$

where θ is the carrier phase offset. We model the phase of the noise as being a uniformly distributed random variable, so the phase of the noise at the demodulator input has no impact. Therefore we assume that c is real. Without loss of generality, let $\sigma_s^2 = \sigma_z^2 = 1$. Since we have constrained $\sigma_y^2 = 1$, we have

$$|g|^2 + c^2 = 1 \tag{A.14}$$

This in turn implies that

$$|g| = \sqrt{\frac{\rho}{1+\rho}} \tag{A.15}$$

$$c = \sqrt{\frac{1}{1+\rho}} \tag{A.16}$$

Using the previously established model for the demodulator output,

$$d(n) = \alpha s(n) + \beta z(n) + \xi(n) \tag{A.17}$$

the cross-correlation R_{yd} is

$$R_{yd} \triangleq \mathcal{E}\{\, y(n)d^*(n)\,\} \tag{A.18}$$

$$= \mathcal{E}\{\, (gs(n) + cz(n))\,(\alpha s(n) + \beta z(n) + \xi(n))^*\,\} \tag{A.19}$$

$$= g\alpha^* + c\beta^* \tag{A.20}$$

$$= \alpha^*\, e^{j\theta}\sqrt{\frac{\rho}{1+\rho}} + \beta^*\sqrt{\frac{1}{1+\rho}} \tag{A.21}$$

The cost function can now be expressed as a function of the SINR ρ and the phase at the demodulator input θ:

$$F(\rho,\theta) = 2 - \frac{2}{\sqrt{1+\rho}}\text{Real}\left\{\alpha^*\, e^{j\theta}\,\sqrt{\rho} + \beta^*\right\} \tag{A.22}$$

Note that α and β are dependent on the SINR ρ and also, in some cases, on the phase at the demodulator input θ.

The (1,2)-CMA cost function is *not* dependent on the phase of the signal. Furthermore, α and β are *real* for CMA. Thus for the (1,2)-CMA cost function (A.22) reduces to

$$F(\rho) = 2 - \frac{2}{\sqrt{1+\rho}}\left\{\alpha^*\,\sqrt{\rho} + \beta^*\right\} \tag{A.23}$$

The stationary points of F can be found by taking the partial derivatives with respect to the SINR ρ and phase θ, setting the resulting expressions equal to zero, and solving for ρ and θ.

Appendix B

Initial Weight Vector

In this appendix we describe a method for obtaining a beamformer weight vector that has a desired output SINR ρ . First, assume that the noise is white so that $\mathbf{R}_{qq} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Then express the desired weight vector as two orthogonal components, one which is proportional to the signal spatial signature, and one which is orthogonal to the spatial signature:

$$\begin{aligned} \mathbf{w} &= \mathbf{P}_a\mathbf{w} + \mathbf{P}_a^{\perp}\mathbf{w} && \text{(B.1)} \\ &= \mathbf{a} + \mathbf{w}_q && \text{(B.2)} \end{aligned}$$

Note that we can ignore any scaling of $\mathbf{w}$ since this has no effect on output SINR. Therefore we assume for convenience that $\mathbf{P}_a\mathbf{w} = \mathbf{a}$. The output SINR of this weight vector is

$$\begin{aligned} \rho &= \frac{\mathbf{w}^H\mathbf{a}\mathbf{a}^H\mathbf{w}}{\mathbf{w}^H\mathbf{w}} && \text{(B.3)} \\ &= \frac{(\mathbf{a}+\mathbf{w}_q)^H\,\mathbf{a}\mathbf{a}^H\,(\mathbf{a}+\mathbf{w}_q)}{(\mathbf{a}+\mathbf{w}_q)^H\,(\mathbf{a}+\mathbf{w}_q)} && \text{(B.4)} \\ &= \frac{\mathbf{a}^H\mathbf{a}\mathbf{a}^H\mathbf{a}}{\mathbf{a}^H\mathbf{a}+\mathbf{w}_q^H\mathbf{w}_q} && \text{(B.5)} \end{aligned}$$

This can be written as

$$\mathbf{w}_q^H\mathbf{w}_q = \frac{\mathbf{a}^H\mathbf{a}\mathbf{a}^H\mathbf{a} - \rho\mathbf{a}^H\mathbf{a}}{\rho} \qquad \text{(B.6)}$$

Thus, in order for $\mathbf{w}$ to have output SINR ρ, we generate a $\mathbf{w}$ according to (B.2) that satisfies (B.6). We can choose $\mathbf{w}_q$ to be any vector that lies in the nullspace of $\mathbf{a}$. For simplicity, we take $\mathbf{w}_q$ to be the first column of $\mathbf{P}_a^{\perp}$. That is,

$$\mathbf{w}_q = \mathbf{P}_a^{\perp}\mathbf{g} \qquad \text{(B.7)}$$

where

$$\mathbf{g} = [1\ 0\ 0\ \cdots 0]^T \tag{B.8}$$

In general, the noise and interference will not be spatially white. This can be handled by performing the above procedure on the whitened spatial signature

$$\tilde{\mathbf{a}} = \mathbf{R}_{qq}^{-1/2}\mathbf{a} \tag{B.9}$$

The resulting weight vector is then multiplied by $\mathbf{R}_{qq}^{-1/2}$ to obtain the desired weight vector.

Appendix C

BPSK Signal in Gaussian Noise

In this appendix we derive expressions for R_{sd} and R_{zd} when s is BPSK, z is complex circularly symmetric Gaussian, and d is the output of a BPSK hard-decision device with

$$y = \sqrt{\rho_k}\, s + z \tag{C.1}$$
$$d = \operatorname{sign} \operatorname{Re}(z) \tag{C.2}$$

The cross-correlation of the BPSK signal s with the hard-decisions d is given by

$$\mathcal{E}\{\, s\, d\,\} = \int_{-\infty}^{\infty} s\, \operatorname{sign}(\sqrt{\rho_k}\, s + z_r)\; p_{z_r}(z_r)\, dz_r \tag{C.3}$$
$$= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} s\, \operatorname{sign}(\sqrt{\rho_k}\, s + z_r)\; e^{-z_r^2}\, dz_r \tag{C.4}$$

We consider the case where the desired signal is equal to $+1$. In this case

$$\mathcal{E}\{\, s\, d \mid s = +1\,\} = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \operatorname{sign}(\sqrt{\rho_k} + z_r)\; e^{-z_r^2}\, dz_r \tag{C.5}$$
$$= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{-\sqrt{\rho_k}} (-1)\; e^{-z_r^2}\, dz_r + \frac{1}{\sqrt{\pi}} \int_{-\sqrt{\rho_k}}^{\infty} (+1)\; e^{-z_r^2}\, dz_r. \tag{C.6}$$

The error function and complementary error function are defined as

$$\operatorname{erf}(x) \triangleq \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt \tag{C.7}$$
$$\operatorname{erfc}(x) \triangleq \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2}\, dt. \tag{C.8}$$

This allows the expression for R_{sd} to be written as

$$\mathcal{E}\{\, s\, d \mid s = +1\,\} = -\frac{1}{2} \operatorname{erfc}(\sqrt{\rho_k}) + \frac{1}{2} \operatorname{erf}(\sqrt{\rho_k}) + \frac{1}{2} \tag{C.9}$$

$$= \frac{1}{2}\left(1 - \operatorname{erf}(\sqrt{\rho_k})\right) + \frac{1}{2}\operatorname{erf}(\sqrt{\rho_k}) + \frac{1}{2} \tag{C.10}$$

$$= \operatorname{erf}(\sqrt{\rho_k}). \tag{C.11}$$

The same expression is obtained for the case where $s = -1$. Since each symbol is equally likely, the final expression for the cross-correlation of $s(n)$ and $d(n)$ is

$$R_{sd} = \operatorname{erf}(\sqrt{\rho_k}). \tag{C.12}$$

We now find the cross-correlation of the hard decisions $d(n)$ with the Gaussian noise $z(n)$. We again consider the case where the desired signal is equal to $+1$. In this case

$$\mathcal{E}\{\, z\, d \mid s = +1 \,\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (z_r + jz_i)\,\operatorname{sign}\left(\sqrt{\rho_k} + z_r\right)\, p_{z_r}(z_r)\, p_{z_i}(z_i)\, dz_r\, dz_i \tag{C.13}$$

$$= \int_{-\infty}^{\infty} z_r \operatorname{sign}\left(\sqrt{\rho_k} + z_r\right)\, p_{z_r}(z_r)\, dz_r \int_{-\infty}^{\infty} z_i\, p_{z_i}(z_i)\, dz_i \tag{C.14}$$

$$= \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty} z_r\, \operatorname{sign}\left(\sqrt{\rho_k} + z_r\right)\, e^{-z_r^2}\, dz_r \tag{C.15}$$

$$= \frac{1}{\sqrt{\pi}}\int_{-\infty}^{-\sqrt{\rho_k}} z_r\,(-1)\, e^{-z_r^2}\, dz_r + \frac{1}{\sqrt{\pi}}\int_{-\sqrt{\rho_k}}^{\infty} z_r\,(+1)\, e^{-z_r^2}\, dz_r \tag{C.16}$$

$$= \frac{1}{\sqrt{\pi}}\left(\frac{1}{2}e^{-z_r^2}\right)\Bigg|_{-\infty}^{-\sqrt{\rho_k}} + \frac{1}{\sqrt{\pi}}\left(-\frac{1}{2}e^{-z_r^2}\right)\Bigg|_{-\sqrt{\rho_k}}^{\infty} \tag{C.17}$$

$$= \frac{1}{\sqrt{\pi}}e^{-\rho_k}\,. \tag{C.18}$$

The same expression is obtained for the case where $s = -1$. Thus the final expression for the cross-correlation of $z(n)$ and $d(n)$ is

$$R_{zd} = \frac{1}{\sqrt{\pi}}e^{-\rho_k}\,. \tag{C.19}$$

Appendix D

QPSK Signal in Gaussian Noise

In this appendix we derive expressions for R_{sd} and R_{zd} for the the case of a QPSK signal in Gaussian noise. We first find an expression for R_{sd} for the case where $s = e^{j\pi/4}$. We have

$$\mathcal{E}\Big\{ s\,d^* \mid s = e^{j\pi/4} \Big\} \tag{D.1}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} sd^*\, p_{z_r}(z_r)\, p_{z_i}(z_i)\, dz_r\, dz_i \tag{D.2}$$

$$= e^{j\pi/4}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left[\text{sign}\left(\sqrt{\frac{\rho_k}{2}}+z_r\right) + j\,\text{sign}\left(\sqrt{\frac{\rho_k}{2}}+z_i\right)\right]^* p_{z_r}(z_r)\, p_{z_i}(z_i)\, dz_r\, dz_i \tag{D.3}$$

$$= e^{j\pi/4}\int_{-\infty}^{\infty}\text{sign}\left(\sqrt{\frac{\rho_k}{2}}+z_r\right) p_{z_r}(z_r)\, dz_r \int_{-\infty}^{\infty} p_{z_i}(z_i)\, dz_i$$
$$- je^{j\pi/4}\int_{-\infty}^{\infty}\text{sign}\left(\sqrt{\frac{\rho_k}{2}}+z_i\right) p_{z_i}(z_i)\, dz_i \int_{-\infty}^{\infty} p_{z_r}(z_r)\, dz_r \tag{D.4}$$

$$= e^{j\pi/4}\int_{-\infty}^{\infty}\text{sign}\left(\sqrt{\frac{\rho_k}{2}}+z_r\right) p_{z_r}(z_r) - je^{j\pi/4}\int_{-\infty}^{\infty}\text{sign}\left(\sqrt{\frac{\rho_k}{2}}+z_i\right) p_{z_i}(z_i)\, dz_i \tag{D.5}$$

From the results obtained for BPSK we know that

$$\frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty}\text{sign}(a+z)\, e^{-z^2}\, dz = \text{erf}(a) \tag{D.6}$$

Thus

$$\mathcal{E}\Big\{ s\,d^* \mid s = \sqrt{\rho_k}\, e^{j\pi/4} \Big\} = \sqrt{2}\,\text{erf}\left(\sqrt{\rho_k\,/2}\right). \tag{D.7}$$

Since the four symbols are equally likely, and the same expression is obtained for each of the four symbols, the cross correlation of the desired signal and the hard decisions is

$$R_{sd} = \sqrt{2}\,\text{erf}\left(\sqrt{\rho_k\,/2}\right). \tag{D.8}$$

We now find an expression for R_{zd} given that $s = e^{j\pi/4}$. We have

$$\mathcal{E}\Big\{ z\, d^* \mid s = e^{j\pi/4} \Big\} \tag{D.9}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} z d^*\, p(z_r)\, p(z_i)\, dz_r\, dz_i \tag{D.10}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (z_r + jz_i)\left[\operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_r\right) + j\operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_i\right)\right]^* p(z_r)\, p(z_i)\, dz_r\, dz_i \tag{D.11}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left[z_r \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_r\right) - jz_r \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_i\right)\right.$$
$$\left. -jz_i \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_r\right) + z_i \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_i\right)\right] p(z_r)\, p(z_i)\, dz_r\, dz_i \tag{D.12}$$

There are four terms to consider. For the first term

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} z_r \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_r\right) p(z_r)\, p(z_i)\, dz_r\, dz_i \tag{D.13}$$

$$= \int_{-\infty}^{\infty} z_r \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_r\right) p(z_r)\, dz_r \int_{-\infty}^{\infty} p(z_i)\, dz_i \tag{D.14}$$

$$= \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty} z_r \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_r\right) e^{-z_r^2}\, dz_r \tag{D.15}$$

For the second term

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} jz_r \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_i\right) p(z_r)\, p(z_i)\, dz_r\, dz_i \tag{D.16}$$

$$= \int_{-\infty}^{\infty} \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z_i\right) p(z_i)\, dz_i \int_{-\infty}^{\infty} z_r\, p(z_r)\, dz_r \tag{D.17}$$

$$= 0 \tag{D.18}$$

because the expected value of z_r is zero. By a similar argument the third term is also equal to zero. The fourth term is equal to the first term by inspection. Thus we have that

$$\mathcal{E}\Big\{ z\, d^* \mid s = e^{j\pi/4} \Big\} = \frac{2}{\sqrt{\pi}}\int_{-\infty}^{\infty} z \operatorname{sign}\left(\sqrt{\frac{\rho_k}{2}} + z\right) e^{-z^2}\, dz \tag{D.19}$$

By comparison with the results obtained for BPSK, and because the four symbols are equally likely, we have

$$R_{zd} = \frac{2}{\sqrt{\pi}} e^{-\rho_k/2}. \tag{D.20}$$

Appendix E

pdf of the Angle of a Signal plus Gaussian Noise

In this appendix we derive an expression for the pdf of the angle of a signal plus complex, unit variance Gaussian noise. This expression can be found in other sources, such as [25], and is included here for completeness. The received noisy signal $y(n)$ is modeled as

$$y(n) = \sqrt{\rho_k} + z_r(n) + jz_i(n) \tag{E.1}$$

where $\sqrt{\rho_k}$ is the signal and z_r and z_i are independent and Gaussian distributed, each having zero mean and variance equal to 1/2. The joint pdf of the real and imaginary parts of y is

$$\begin{aligned} p_{y_r,y_i}(y_r, y_i) &= p_{y_r}(y_r)\, p_{y_i}(y_i) && \text{(E.2)} \\ &= \left(\frac{1}{\sqrt{\pi}} e^{-(y_r - \sqrt{\rho_k})^2}\right) \left(\frac{1}{\sqrt{\pi}} e^{-y_i^2}\right) && \text{(E.3)} \\ &= \frac{1}{\pi} e^{-(y_r^2 - 2\sqrt{\rho_k} y_r + \rho_k + y_i^2)}. && \text{(E.4)} \end{aligned}$$

We now transform to polar coordinates

$$\begin{aligned} \phi &= \tan^{-1}\left(\frac{y_i}{y_r}\right) && \text{(E.5)} \\ r &= \sqrt{y_r^2 + y_i^2} && \text{(E.6)} \end{aligned}$$

where ϕ is the angle and r is the magnitude of y. Noting that

$$\begin{aligned} y_r &= r\cos\phi && \text{(E.7)} \\ y_i &= r\sin\phi && \text{(E.8)} \end{aligned}$$

and also noting that the Jacobian of the transformation is $1/r$, the joint pdf becomes

$$p_{r,\phi}(r,\phi) = \frac{r}{\pi} e^{-(r^2 - 2\sqrt{\rho_k}\, r\cos\phi + \rho_k)} \tag{E.9}$$

The pdf of ϕ is found by integrating $p_{r,\phi}(r,\phi)$ over r,

$$p_{\phi}(\phi) = \int_0^{\infty} r e^{-(r^2 - 2\sqrt{\rho_k}\, r\cos\phi + \rho_k)}\, dr \tag{E.10}$$

which is not easily expressed in closed form.

Appendix F

BPSK in Constant Modulus Interference

In this appendix we derive expressions for R_{sd} and R_{zd} when s is BPSK, z is constant modulus with random, uniformly distributed phase ϕ, and d is the output of a BPSK hard-decision device with

$$y = \sqrt{\rho_k}\, s + e^{j\phi} \tag{F.1}$$

$$d = \operatorname{sign} \operatorname{Re}\left(\sqrt{\rho_k}\, s + e^{j\phi}\right) \tag{F.2}$$

Since ϕ is uniformly distributed, the pdf of $\cos\phi$ is

$$p_z(z) = \frac{1}{\pi\sqrt{1-z^2}} \qquad -1 \le z \le 1 \tag{F.3}$$

First consider the cross-correlation of s and d.

$$R_{sd} \triangleq \mathcal{E}\{\, sd^* \,\} \tag{F.4}$$

$$= \int_{-\infty}^{\infty} s \operatorname{sign}(\sqrt{\rho_k}\, s + z_r)\, p_{z_r}(z_r)\, dz_r \tag{F.5}$$

Consider the case where $s = +1$. For clarity of notation, drop the subscript 'r' from z_r, with the understanding that we are only concerned with the real part of z. Then

$$\mathcal{E}\{\, sd^* \mid s = +1 \,\} = \int_{-\infty}^{\infty} \operatorname{sign}(\sqrt{\rho_k} + z)\, p_z(z)\, dz \tag{F.6}$$

$$= \frac{1}{\pi}\int_{-1}^{+1} \operatorname{sign}(\sqrt{\rho_k} + z)\, \frac{1}{\sqrt{1-z^2}}\, dz \tag{F.7}$$

$$= \frac{1}{\pi}\int_{-1}^{-\sqrt{\rho_k}} (-1)\, \frac{1}{\sqrt{1-z^2}}\, dz + \frac{1}{\pi}\int_{-\sqrt{\rho_k}}^{+1} (+1)\, \frac{1}{\sqrt{1-z^2}}\, dz \tag{F.8}$$

From standard integration tables,

$$\int \frac{dx}{\sqrt{a^2 - x^2}} = \arcsin \frac{x}{|a|} \tag{F.9}$$

Therefore

$$\begin{aligned} \mathcal{E}\{\, sd^* \mid s = +1 \,\} &= \frac{-1}{\pi} \arcsin z \Big|_{-1}^{-\sqrt{\rho_k}} + \frac{1}{\pi} \arcsin z \Big|_{-\sqrt{\rho_k}}^{1} && \text{(F.10)} \\ &= \frac{2}{\pi} \arcsin \sqrt{\rho_k} && \text{(F.11)} \end{aligned}$$

Since each BPSK symbol is equally likely, this is the final expression for R_{sd}.

Using a similar approach for R_{zd},

$$\begin{aligned} \mathcal{E}\{\, zd^* \mid s = +1 \,\} &= \int_{-\infty}^{\infty} z \operatorname{sign}(\sqrt{\rho_k} + z)\, p_z(z)\, dz && \text{(F.12)} \\ &= \frac{1}{\pi} \int_{-1}^{-\sqrt{\rho_k}} (-1) \frac{z}{\sqrt{1 - z^2}}\, dz + \frac{1}{\pi} \int_{-\sqrt{\rho_k}}^{+1} (+1) \frac{z}{\sqrt{1 - z^2}}\, dz && \text{(F.13)} \end{aligned}$$

From standard integration tables,

$$\int \frac{x\, dx}{\sqrt{a^2 - x^2}} = -\sqrt{a^2 - x^2} \tag{F.14}$$

Therefore

$$\begin{aligned} \mathcal{E}\{\, zd^* \mid s = +1 \,\} &= \frac{-1}{\pi}(-\sqrt{1 - z^2}) \Big|_{-1}^{-\sqrt{\rho_k}} + \frac{1}{\pi}(-\sqrt{1 - z^2}) \Big|_{-\sqrt{\rho_k}}^{1} && \text{(F.15)} \\ &= \frac{2}{\pi} \sqrt{1 - \rho_k} && \text{(F.16)} \end{aligned}$$

Since each BPSK symbol is equally likely, this is the final expression for R_{zd}.

Appendix G

QPSK in Constant Modulus Interference

In this appendix we derive expressions for R_{sd} and R_{zd} when s is QPSK, z is constant modulus with random, uniformly distributed phase, and d is the output of a QPSK hard-decision device with

$$y = \sqrt{\rho_k}\, s + z \tag{G.1}$$

$$d = \operatorname{sign}\operatorname{Re}(z) + j \operatorname{sign}\operatorname{Im}(z) \tag{G.2}$$

To calculate R_{sd}, we consider the case where $s = e^{j\pi/4}$. Then

$$\mathcal{E}\left\{ sd^* \mid s = e^{j\pi/4} \right\} = \frac{e^{j\pi/4}}{2\pi} \int_0^{2\pi} \left\{ \operatorname{sign}\left(\sqrt{\rho_k/2}\, + \cos\phi \right) - j \operatorname{sign}\left(\sqrt{\rho_k/2}\, + \cos\phi \right) \right\} d\phi \tag{G.3}$$

To clarify notation, define

$$\gamma = \arccos\left(-\sqrt{\rho_k/2}\, \right) \tag{G.4}$$

Examining each integral separately,

$$\int_0^{2\pi} \operatorname{sign}\left(\sqrt{\rho_k/2}\, + \cos\phi \right) d\phi = \int_{-\pi}^{-\gamma} (+1)\, d\phi + \int_{-\gamma}^{+\gamma} (-1)\, d\phi + \int_{+\gamma}^{\pi} (+1)\, d\phi \tag{G.5}$$

$$= 2\pi - 4\gamma \tag{G.6}$$

This gives

$$\mathcal{E}\left\{ sd^* \mid s = e^{j\pi/4} \right\} = \frac{e^{j\pi/4}}{2\pi} \left\{ (2\pi - 4\gamma) - j(2\pi - 4\gamma) \right\} \tag{G.7}$$

$$= \sqrt{2} \left(1 - \frac{2}{\pi} \arccos\left(-\sqrt{\rho_k/2}\, \right) \right) \tag{G.8}$$

$$= -\frac{2\sqrt{2}}{\pi} \arcsin\left(\sqrt{\rho_k/2}\, \right) \tag{G.9}$$

Turning our attention now to R_{zd},

$$\begin{aligned}
\mathcal{E}\Big\{ zd^* \mid s = e^{j\pi/4} \Big\} &= \frac{1}{2\pi}\int_0^{2\pi} \Big[\cos\phi \ \mathrm{sign}\left(\sqrt{\rho_k/2} + \cos\phi\right) && \text{(G.10)} \\
&\quad j\sin\phi \ \mathrm{sign}\left(\sqrt{\rho_k/2} + \cos\phi\right) + \\
&\quad j\cos\phi \ \mathrm{sign}\left(\sqrt{\rho_k/2} + \sin\phi\right) + \\
&\quad \sin\phi \ \mathrm{sign}\left(\sqrt{\rho_k/2} + \sin\phi\right) \Big] \ d\phi && \text{(G.11)} \\
&= -4\sqrt{1-\rho_k/2} && \text{(G.12)}
\end{aligned}$$

Bibliography

[1] B. D. van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering", *IEEE ASSP Magazine*, vol. 5, no. 2, pp. 4–24, Apr. 1988.

[2] J. Litva and T.K. Lo, *Digital Beamforming in Wireless Communications*, Artech, Boston, MA, 1996.

[3] L.C. Godara, "Application of antenna arrays to mobile communications. I. Performance improvement, feasibility, and system considerations", *Proc. of the IEEE*, , no. 7, pp. 1031–1060, July 1997.

[4] L.C. Godara, "Application of antenna arrays to mobile communications. II. Beamforming and direction-of-arrival considerations", *Proc. of the IEEE*, , no. 8, pp. 1195–1245, Aug 1997.

[5] R.B. Ertel, *Antenna Array Systems: Propagation and Performance*, PhD thesis, Viginia Tech, 1999.

[6] J.C. Liberti and T.S. Rappaport, *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*, Prentice Hall, Englewood Cliffs, NJ, 1999.

[7] T. E. Biedka, W. H. Tranter, and J. H. Reed, "Convergence analysis of the least squares constant modulus algorithm in interference cancellation applications", *IEEE Trans. on Commun.*, vol. 48, no. 3, pp. 491–501, March 2000.

[8] T.E. Biedka, J.H. Reed, and W.H. Tranter, "Mean convergence rate of a decision directed adaptive beamformer with Gaussian interference", *Proc. of the IEEE Sensor Array and Multichannel Signal Processing Workshop*, March 2000.

[9] A.L. Swindlehurst, S. Daas, and J. Yang, "Analysis of a decision directed beamformer", *IEEE Trans. on Signal Processing.*, vol. 43, no. 12, pp. 2920–2927, December 1995.

[10] C.M.S See, C.F.N. Cowan, and A. Nehorai, "Spatio-temporal channel identification and equalization in the presence of strong co-channel interference", *Signal Processing*, vol. 78, no. 2, pp. 127–138, Oct. 1999.

[11] A.V. Keethi and J.J. Shynk, "Separation of cochannel signals in TDMA mobile radio", *IEEE Trans. on Signal Processing.*, vol. 46, no. 10, pp. 2684–2697, Oct. 1998.

[12] A.-J. van der Veen, "Algebraic methods for deterministic blind beamforming", *Proc. of the IEEE*, vol. 86, no. 10, pp. 1987–2008, Oct. 1998.

[13] A.-J. van der Veen, S. Talwar, and A. Paulraj, "A subspace approach to blind space-time signal processing for wireless communication systems", *IEEE Trans. on Signal Processing.*, vol. 45, no. 1, pp. 173–190, Jan. 1997.

[14] S. Talwar, M. Viberg, and A. Paulraj, "Blind separation of synchronous co-channel digital signals using an antenna array: Part I. algorithms", *IEEE Trans. on Signal Processing.*, vol. 44, no. 5, May 1996.

[15] A.V. Keethi, J.J. Shynk, and A. Mathur, "Steady-state analysis of the multistage constant modulus array", *IEEE Trans. on Signal Processing.*, vol. 44, no. 4, pp. 948–962, Apr. 1996.

[16] J.J. Shynk and R.P. Gooch, "The constant modulus array for cochannel signal copy and direction finding", *IEEE Trans. on Signal Processing.*, vol. 44, no. 3, pp. 652–660, Mar. 1996.

[17] T.E. Biedka, J.H. Reed, W.H. Tranter, and A.L. Swindlehurst, "Convergence analysis of a decision directed beamformer. Part I: Gaussian interference", *to be submitted to IEEE Trans. on Commun.*, April 2000.

[18] T.E. Biedka, J.H. Reed, W.H. Tranter, and A.L. Swindlehurst, "Convergence analysis of a decision directed beamformer. Part II: Constant modulus interference", *to be submitted to IEEE Trans. on Commun.*, April 2000.

[19] J.D. Kraus, *Radio Astronomy*, Cygnus-Quasar Books, Powell, OH, 2nd edition, 1986.

[20] G. H. Golub and C. F. van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, second edition, 1989.

[21] A.J. Viterbi, *CDMA: Principles of Spread Spectrum Communication*, Addison-Wesley, Reading, MA, 1995.

[22] T.S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall, Upper Saddle River, NJ, 1996.

[23] R.B. Ertel, P. Cardieri, K.W. Sowerby, T.S Rappaport, and J.H. Reed, "Overview of spatial channel models for antenna array communication systems", *IEEE Personal Communcations*, vol. 5, pp. 10–22, Feb. 1998.

[24] M.C. Jeruchim, P. Balaban, and K.S. Shanmugan, *Simulation of Communication Systems*, Plenum Press, New York, NY, 1992.

[25] J. G. Proakis, *Digital Communications*, McGraw–Hill, New York, third edition, 1995.

[26] J. H. Winters, "Optimum combining in digital mobile radio with cochannel interference", *IEEE Trans. Vehic. Tech.*, vol. VT-33, no. 3, pp. 144–155, Aug. 1984.

[27] R.B. Ertel and J.H. Reed, "Generation of two equal power correlated rayleigh fading envelopes", *IEEE Commun. Letters*, vol. 2, no. 10, pp. 276–278, Oct. 1998.

[28] I. S. Reed, J. D. Mallett, and L. E. Brennan, "Rapid convergence rate in adaptive arrays", *IEEE Trans. Aerospace Electron. Syst.*, vol. AES-10, pp. 853–863, Nov. 1974.

[29] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*, Prentice-Hall, 1985.

[30] W.A. Gardner, "Learning characteristics of stochastic-gradient-descent algorithms: a general study, analysis, and critique", *Signal Processing*, vol. 6, no. 2, pp. 113–133, April 1984.

[31] W. A. Sethares, "The LMS family", in *Efficient System Identification and Signal Processing Algorithms*, N. Kalouptsidis and S. Theodoridis, Eds. Springer-Verlag, 1993.

[32] H. Krim and M. Viberg, "Two decades of array signal processing research: The parametric approach", *IEEE Signal Processing Magazine*, pp. 67–94, July 1996.

[33] R. T. Compton, "Pointing accuracy and dynamic range in a steered beam adaptive array", *IEEE Transactions on Aerospace and Electronics Systems*, vol. AES-16, no. 3, pp. 280–287, September 1980.

[34] J. W. Kim and C. K. Un, "Signal subspace method for beamsteered adaptive arrays", *Electronics Letters*, vol. 25, no. 16, pp. 1076–1077, August 1989.

[35] D. Feldman and L. J. Griffiths, "A constraint projection approach for robust adaptive beamforming", in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1991, pp. 1381–1384.

[36] R. O. Schmidt and R. E. Franks, "Multiple source DF signal processing: An experimental system", *IEEE Transactions on Antennas and Propagation*, vol. AP-34, no. 3, pp. 281–290, March 1986.

[37] B. G. Wahlberg, I. M. Y. Mareels, and I. Webster, "Experimental and theoretical comparison of some algorithms for beamforming in single receiver adaptive arrays", *IEEE Transactions on Antennas and Propagation*, vol. AP-39, no. 1, pp. 21–28, January 1991.

[38] J.-F. Cardoso, "Blind signal separation: Statistical principles", *Proc. of the IEEE*, , no. 10, pp. 2009–2026, Oct. 1998.

[39] B. G. Agee, "Convergent behavior of modulus-restoring adaptive arrays in gaussian interference environments", *Proc. of the Asilomar Conf. on Signals, Systems, and Computers*, pp. 818–822, Dec. 1988.

[40] J. Lundell and B. Widrow, "Application of the constant modulus adaptive beamformer to constant and nonconstant modulus signals", *Proc. of the Asilomar Conf. on Signals, Systems, and Computers*, pp. 432–436, Nov. 1987.

[41] J. R. Treichler and B. G. Agee, "A new approach to multipath correction of constant modulus signals", *IEEE Trans. on Acous., Speech, and Signal Process.*, vol. ASSP-31, no. 2, pp. 459–471, April 1983.

[42] R. Gooch and J. Lundell, "The CM array: An adaptive beamformer for constant modulus signals", *Proc. of Inter. Conf. on Acous., Speech, and Signal Process.*, pp. 2523–2526, April 1986.

[43] B. G. Agee, "The least-squares CMA: A new technique for rapid correction of constant modulus signals", *Proc. of Inter. Conf. on Acous., Speech, and Signal Process.*, pp. 953–956, April 1986.

[44] B.G. Agee, "Maximum likelihood approaches to blind adaptive signal extraction using narrowband arrays", *Proc. of the Asilomar Conf. on Signals, Systems, and Computers*, pp. 716–720, Nov. 1991.

[45] J. J. Shynk and R. P. Gooch, "Convergence properties of the multistage CMA adaptive beamformer", *Proc. of the Asilomar Conf. on Signals, Systems, and Computers*, pp. 622–626, Nov. 1993.

[46] B. G. Agee, "Blind separation and capture of communication signals using a multitarget constant modulus beamformer", *Proc. of IEEE Military Commun. Conf.*, pp. 340–346, May 1989.

[47] I. Parra, G. Xu, and H. Lui, "A least squares projective constant modulus approach", *Personal Indoor Mobile Radio Conference*, pp. 673–676, 1995.

[48] Alle-Jan van der Veen and A. Paulraj, "An analytical constant modulus algorithm", *IEEE Trans. Signal Processing*, vol. 44, no. 5, pp. 1136–1155, May 1996.

[49] A.J. van der Veen, "Algebraic methods for deterministic blind beamforming", *Proc. of the IEEE*, , no. 10, pp. 1987–2008, Oct. 1998.

[50] K. Bakhru and D. Torrieri, "The maximin algorithm for adaptive arrays and frequency-hopping communications", *IEEE Trans. on Antennas and Propagation*, vol. AP-32, pp. 919–928, Sept. 1984.

[51] B. G. Agee, "Fast acquisition of burst and transient signals using a predictive adaptive beamformer", *Proc. of IEEE Military Commun. Conf.*, pp. 19.3.1–19.3.6, 1989.

[52] M. Viberg, "Sensor array processing using gated signals", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 447–450, March 1989.

[53] A.F. Naguib, *Adaptive Antennas for CDMA Wireless Networks*, PhD thesis, Stanford University, 1996.

[54] B. G. Agee, S. V. Schell, and W. A. Gardner, "Spectral self-coherence restoral: A new approach to blind adaptive signal extraction", *Proceedings of the IEEE*, vol. 78, no. 4, pp. 756–767, April 1990.

[55] S. V. Schell and W. A. Gardner, "Programmable canonical correlation analysis: A flexible framework for blind adaptive spatial filtering", *IEEE Trans. on Signal Processing*, vol. 43, no. 12, pp. 2898–2908, Dec. 1995.

[56] S.V. Schell, *Exploitation of Spectral Correlation for Signal-Selective Direction Finding*, PhD thesis, University of California, Davis, CA, 1990.

[57] B. G. Agee, *The Property-Restoral Approach to Blind Adaptive Signal Extraction*, PhD thesis, University of California, Davis, CA, 1989.

[58] T. E. Biedka, "Subspace constrained SCORE algorithms", *Proc. of the Asilomar Conference on Signals, Systems, and Computers*, pp. 716–720, November 1993.

[59] M. F. Kahn, M. A. Mow, W. A. Gardner, and T. E. Biedka, "A recursive programmable canonical correlation analyzer", *Proc. of the Second Workshop on Cyclostationary Signals*, August 1994.

[60] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems", *IEEE Trans. on Commun.*, vol. COM-28, no. 11, pp. 1867–1875, Nov. 1980.

[61] S. V. Schell and W. A. Gardner, "Maximum likelihood and common factor analysis-based blind adaptive spatial filtering for cyclostationary signals", in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Minneapolis, MN, April 1993, pp. 292–295.

[62] D. M. Dlugos and R. A. Scholtz, "Acquisition of spread spectrum signals by an adaptive array", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-37, no. 8, pp. 1253–1270, Aug. 1989.

[63] T. E. Biedka, W. H. Tranter, and J. H. Reed, "Convergence analysis of the least squares constant modulus algorithm", *Proc. of the Asilomar Conf. on Signals, Systems, and Computers*, pp. 541–545, Nov. 1996.

[64] Z. Ding, R.A. Kennedy, B.D.O. Anderson, and C.R. Johnson, Jr., "Ill-convergence of Godard blind equalizers in data communication systems", *IEEE Trans. on Commun.*, vol. 39, no. 9, pp. 1313–1327, Sept. 1991.

[65] Y. Li and Z. Ding, "Convergence analysis of finite length blind adaptive equalizers", *IEEE Trans. Signal Processing*, vol. 43, no. 9, pp. 2120–2129, Sept. 1995.

[66] J. R. Treichler and M. G. Larimore, "The tone capture properties of CMA-based interference suppressors", *IEEE Trans. on Acous., Speech, and Signal Process.*, vol. ASSP-33, no. 4, pp. 946–958, August 1985.

[67] T. E. Biedka, "A comparison of initialization schemes for blind adaptive beamforming", *Proc. of Inter. Conf. on Acous., Speech, and Signal Process.*, pp. 1665–1668, May 1998.

[68] R.W. Gerchberg and W.O. Saxton, "A practical algorithm for the determination of phase from image and diffraction plane pictures", *Optik*, vol. 35, no. 2, pp. 237–246, 1972.

[69] Y. Wang, Y.C. Pati, Y.M. Cho, A. Paulraj, and T. Kailath, "A matrix factorization approach to signal copy of constant modulus signals arriving at an antenna array", *Proc. of the 28th Conf. on Information Science and Systems (Princeton, NJ)*, pp. 178–183, Mar. 1994.

[70] Henry Stark, Ed., *Image Recovery - Theory and Application*, Academic Press, 1987.

[71] D.C. Youla, "Generalized image restoration by the method of alternating projections", *IEEE Trans. on Circuits Syst.*, vol. CAS-25, pp. 694–702, Sept. 1978.

[72] D.C. Youla and H. Webb, "Image restoration by the method of convex projections: Part I, theory", *IEEE Trans. Medical Imaging*, vol. MI-1, pp. 81–94, Oct. 1982.

[73] H.J. Trussell and M.R. Civanlar, "The feasible solution in signal restoration", *IEEE Trans. on Acous., Speech, and Signal Process.*, vol. ASSP-32, no. 2, pp. 201–212, April 1984.

[74] Z. Kostic, M.I. Sezan, and E.L. Titlebaum, "Estimation of parameters of a multipath channel using set-theoretic deconvolution", *IEEE Trans. on Commun.*, vol. 40, no. 6, pp. 1006–1011, June 1992.

[75] N.M. Blachman, "Detectors, bandpass nonlinearities, and their optimization: Inversion of the Chebyshev transform", *IEEE Trans. on Information Theory*, vol. IT-17, no. 4, pp. 398–404, July 1971.

[76] N.M. Blachman, "The output signal-to-noise ratio of a bandpass limiter", *IEEE Trans. on Aerosp. and Electronic Systems*, p. 635, July 1968.

[77] K.J. Friederichs, "A novel canceller for strong CW and angle modulated interferers in spread spectrum receivers", *Proc. of IEEE Military Commun. Conf.*, pp. 32.4/1–4, Oct 1984.

[78] Y.C. Pati, G.G. Raleigh, and A. Paulraj, "Estimation of co-channel FM signals with multitarget adaptive phase-locked loops and antenna arrays", *Proc. of Inter. Conf. on Acous., Speech, and Signal Process.*, pp. 1741–1744, May 1995.

[79] N.M. Blachman, "Optimum nonlinearities and pessimum interference", *Proc. of the Second Workshop on Cyclostationary Signals*, pp. 5.1–9, Aug. 1994.

[80] J. R. Treichler and M. G. Larimore, "The noise capture properties of the constant modulus algorithm", *Proc. of Inter. Conf. on Acous., Speech, and Signal Process.*, pp. 1165–1168, August 1985.

[81] R. A. Monzingo and T. W. Miller, *Introduction to Adaptive Arrays*, John Wiley and Sons, NY, 1980.

[82] S. Haykin, *Adaptive Fitler Theory*, Prentice Hall, Englewood Cliffs, NJ, third edition, 1996.

Vita

Thomas E. Biedka

EDUCATION

Ph.D. in Electrical Engineering: October 2001, Virginia Tech, Blacksburg, VA. Dissertation Title: Analysis and Development of Blind Adaptive Beamforming Algorithms

M.S. in Electrical Engineering: December 1989, University of Pittsburgh, Pittsburgh, PA.

B.S. (with Honors) in Engineering Science: May 1987, Pennsylvania State University, University Park, PA.

EXPERIENCE

June 2000-Present: Tropian, Inc. Title: Senior Systems Engineer

- Systems engineer for an effort to demonstrate the viability of a polar modulator for generation of IS-95 and UMTS signals. This included an analysis of RF hardware impairments and finite precision DSP limitations. Also responsible for testing of prototype systems, including creation of LabWindows automated test software.
- Systems engineer for a mixed signal IC development. Responsible for development of VCO calibration techniques and definition of system requirements for analog signal processing paths. Developed novel techniques for time-aligning two analog waveforms.
- Responsible for modification and debugging of an ARM-based controller for cellphone baseband emulation for test and evaluation of a polar modulation IC.
- Responsible for definition, programming, and debugging of a LabWindows GUI control interface.

April 1990-June 2000: Raytheon Systems Company, Greenville TX Last Title: Senior Engineer

- TDOA-FDOA Geolocation System - Primarily responsible for DSP algorithm development, and supervision of real-time DSP software implementation. Assisting in hardware implementation and system level issues.
- CDMA Interference Cancellation - Responsible for the development and analysis of techniques that combine non-linear multi-user detection approaches and linear multi-sensor approaches.
- PACS Smart Antenna - Supervised software development and testing; assisted with field data collection; analyzed the data to determine channel parameters; assisted in system integration and test; responsible for presentations and quarterly progress reports to the customer and consortium members. This DARPA-sponsored program was performed in cooperation with consortium members Hughes Network Systems, Hughes Research Labs, and Stanford University.
- Interference Cancellation and Direction Finding System - Responsible for DSP algorithm development and performance analysis for a wireless communication application.
- Various other signal processing development and analysis applications.

Sept. 1988 - March 1990: University of Pittsburgh, Pittsburgh, PA

Teaching Assistant (Communications Classes) and Research Assistant (System Modeling)

May 1998 - August 1988: NCR, Inc., Cambridge OH

Designed and constructed a prototype low-cost proprietary Local Area Network.

PUBLICATIONS

[1] T. E. Biedka, J. H. Reed, and W. H. Tranter, Statistics of Blind Spatial Signature Estimators, Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, Nov. 2000

[2] T. E. Biedka, W. H. Tranter, and J. H. Reed, "Convergence Analysis of the Least Squares Constant Modulus Algorithm in Interference Cancellation Applications", IEEE Transactions on Communications, vol. 8, no. 43, pp. 491-501, March 2000.

[3] T. E. Biedka, J. H. Reed, and W. H. Tranter, "Mean Convergence Rate of a Decision Directed Adaptive Beamformer with Gaussian Interference", Proc. of the IEEE Sensor Array and Multichannel Signal Processing Workshop, March, 2000.

[4] T. Biedka, B. Holden, S. Thornton, W. Ferguson, R. Hammons, B. Johnson, S. Kailas, V. Liau, A. Paulraj, and S. Sandhu, "Implementation of a Prototype Smart Antenna for Low Tier PCS", Proc. of the IEEE Vehicular Technology Conference, Houston, TX, May 1999.

[5] S. Thornton, B. Holden, T. Biedka, W. Nolan, B. Johnson, W. Ferguson, V. Liau, A. Paulraj, S. Sandhu, "Design and Testing of a Prototype Smart Antenna System for Low-Tier PCS", Presented at the 5th Stanford Workshop on Smart Antennas, July 1998.

[6] T. E. Biedka, "A Comparison of Initialization Schemes for Blind Adaptive Beamforming" , Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 1665-1668, May 1998.

[7] K. J. Krizman, T. E. Biedka, and T. S. Rappaport, "Wireless Position Location: Fundamentals, Implementation Strategies, and Sources of Error" , Proc. of IEEE Vehicular Technology Conference, May 1997. (Also included in T.S. Rappaport, ed., Smart Antennas: Adaptive Arrays, Algorithms, and Wireless Position Location , IEEE Press, 1998)

[8] T. E. Biedka, W. H. Tranter, and J. H. Reed, "Convergence Analysis of the Least Squares Constant Modulus Algorithm" , in Proc of the Thirtieth Asilomar Conference on Signals, Systems, and Computers, Nov 1996.

[9] T. E. Biedka, J. H. Reed, and B. D. Woerner, "Direction Finding Methods for CDMA Systems" , in Proc of the Thirtieth Asilomar Conference on Signals, Systems, and Computers, Nov 1996.

[10] T. S. Rappaport, J. H. Reed, and T. E. Biedka, "Position Location and E-911 Techniques for Wireless Systems" , Tutorial presented at IEEE Intl. Conf. on Universal Personal Communications, Cambridge, MA, Oct 1996.

[11] T. E. Biedka, L. M. Mili, and J. H. Reed, "Robust Estimation of Cyclic Correlation in Contaminated Gaussian Noise" , Proc. of the Twenty-Ninth Asilomar Conference on Signals, Systems, and Computers, Nov 1995.

[12] T. E. Biedka, "A Method for Reducing Computations in Cyclostationarity-Exploiting Beamforming", Proc of the International Conference on Acoustics, Speech, and Signal Processing, May 1995.

[13] T. E. Biedka and M. F. Kahn, "Methods for Constraining a CMA Beamformer to Extract a Cyclostationary Signal" , Proc. of the Second Workshop on Cyclostationary Signals, Monterey, CA, August 1994.

[14] M. F. Kahn, M. A. Mow, W. A. Gardner, and T. E. Biedka, A Recursive Programmable Canonical Correlation Analyzer, Proc. of the Second Workshop on Cyclostationary Signals, Monterey, CA, August 1994.

[15] T. E. Biedka, "Subspace-Constrained SCORE Algorithms", Proc. of the Twenty Seventh Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, November, 1993.

[16] M. F. Kahn, T. E. Biedka, and N. B. Whitley, Progress in Cyclostationary-Signal Processing , Presented at the First Workshop on Cyclostationary Signal Processing, Yountville, CA, August 1992.

[17] T. E. Biedka and B. G. Agee, Subinterval Cyclic MUSIC - Robust DF with Error in Cycle Frequency Knowledge , Proc. of the Twenty Fifth Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, pp. 262-266, November 1991.