ANTENNA ARRAYS: PERFORMANCE LIMITS AND GEOMETRY OPTIMIZATION

by

Peter Joseph Bevelacqua

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

ARIZONA STATE UNIVERSITY

May 2008

ANTENNA ARRAYS: PERFORMANCE LIMITS AND GEOMETRY OPTIMIZATION

by

Peter Joseph Bevelacqua

has been approved

March 2008

Graduate Supervisory Committee:

Constantine A. Balanis, Chair
Joseph Palais
Abbas Abbaspour-Tamijani
James Aberle
Cihan Tepedelenlioglu

ACCEPTED BY THE GRADUATE COLLEGE

ABSTRACT

The radiation pattern of an antenna array depends strongly on the weighting method and the geometry of the array. Selection of the weights has received extensive attention, primarily because the radiation pattern is a linear function of the weights. However, the array geometry has received relatively little attention even though it also strongly influences the radiation pattern. The reason for this is primarily due to the complex way in which the geometry affects the radiation pattern. The main goal of this dissertation is to determine methods of optimizing array geometries in antenna arrays.

An adaptive array with the goal of suppressing interference is investigated. It is shown that the interference rejection capabilities of the antenna array depend upon its geometry. The concept of an interference environment is introduced, which enables optimization of an adaptive array based on the expected directions and power of the interference. This enables the optimization to perform superior on average, instead of for specific situations. An optimization problem is derived whose solution yields an optimal array for suppressing interference. Optimal planar arrays are presented for varying number of elements. It is shown that, on average, the optimal arrays increase the signal-to-interference-plus-noise ratio (SINR) when compared to standard arrays.

Sidelobe level is an important metric used in antenna arrays, and depends on the weights and positions in the array. A method of determining optimal sidelobe-minimizing weights is derived that holds for any linear array geometry, beamwidth, antenna type and scan angle. The positions are then optimized simultaneously with the optimal weights to determine the minimum possible sidelobe level in linear arrays.

Results are presented for arrays of varying size, with different antenna elements, and for distinct beamwidths and scan angles.

Minimizing sidelobes is then considered for 2D arrays. A method of determining optimal weights in symmetric 2D arrays is derived for narrowband and wideband cases. The positions are again simultaneously optimized with the weights to determine optimal arrays, weights and sidelobe levels. This is done for arrays with varying number of elements, beamwidths, bandwidths, and different antenna elements.

ACKNOWLEDGEMENTS

This work would not have been possible without my adviser, Dr. Constantine Balanis. Dr. Balanis let me into his research group and gave me funding to research array geometry, which ultimately led to the work presented here. His guidance and helpfulness were paramount in producing successful research; without this the work would not have been completed due to my youthful impatience and wavering trajectory.

I would like to thank Dr. Joseph Palais, Dr. Abbaspour-Tamijani, Dr. James Aberle and Dr. Cihan Tepedelenlioglu for taking the time to be on my research committee and for helpful suggestions along the way, specifically during my qualifying and comprehensive examinations. Thanks also to Dr. Gang Qian and Dr. Andreas Spanias for helping with my qualifying exam and in understanding Fourier Transforms.

My thanks go to my colleagues at ASU, including Zhiyong Huang, Victor Kononov, Bo Yang, and Aron Cummings. The presence of these people increased the quality of my research and life in various ways during my time at ASU.

This work is the culmination of approximately 10 years of college education. I am indebted to many people for academic, personal, and financial assistance along the way. Of these, I would like to thank Dr. Shira Broschat and Dr. John Schneider from Washington State, Dr. Lee Boyce from Stanford, and my parents. Many other people have in some way contributed to my education, but they are too numerous to list here.

TABLE OF CONTENTS

Page

CHAPTER Page

LIST OF TABLES

Table Page

Table Page

LIST OF FIGURES

Figure Page

I. INTRODUCTION

1.1. Overview

On December 12, 1901, Guglielmo Marconi successfully received the first trans-atlantic radio message [1]. The message was the Morse-code for the letter 'S' – three short clicks. This event was arguably the most significant achievements in early radio communication. This communication system, while technically functional, clearly had significant room for improvement.

A century of improvement in the field of wireless communication has occurred. The envelope has been pushed in every imaginable direction, with no letup in progress likely in the foreseeable future. Development in the fields of electronics, information theory, signal processing, and antenna theory have all contributed to the ubiquity of wireless communication systems today. However, despite the tremendous advances since the days of Marconi in each of these fields, the desire for improved wireless communication systems has not been quenched.

The concept of an antenna array was first introduced in military applications in the 1940s [2]. This development was significant in wireless communications as it improved the reception and transmission patterns of antennas used in these systems. The array also enabled the antenna system to be electronically steered – to receive or transmit information primarily from a particular direction without mechanically moving the structure.

As the field of signal processing developed, arrays could be used to receive energy (or information) from a particular direction while rejecting information or nulling out the energy in unwanted directions. Consequently, arrays could be used to mitigate

intentional interference (jamming) or unintentional interference (radiation from other sources not meant for the system in question) directed toward the communication system.

Further development in signal processing led to the concept of adaptive antenna arrays. These arrays adapted their radiation or reception pattern based on the environment they were operating in. This again significantly contributed to the capacity available in wireless communication systems.

While there has been a large amount of work on the signal processing aspects (and in conjunction, the electronics used to implement the algorithms), the physical geometry (or location of the antenna elements in the array) has received relatively little attention. The reason for this lies in the mathematical complexity of dealing with the optimization of the element positions for various situations. As shown in Chapter 2, understanding the influence of the element weighting (which is a major component of the signal processing involved in antenna arrays) is significantly simpler than understanding the effect of varying the positions of the elements.

Thanks to the tremendous advances in numerical computing, optimization of the element positions in an antenna array (for various situations) is now tractable. The primary goal of this dissertation is to study the influence of array geometry on wireless system performance. It will be shown that performance gains can be obtained via intelligent selection of the array geometry. Array geometry optimization can therefore be hoped to contribute to the continuing advancement of wireless communication system performance.

This dissertation is organized as follows. Chapter 2 introduces the main ideas and terminology used in understanding antenna arrays. Chapter 3 discusses various optimization methods used in this work. Chapter 4 discusses methods of choosing the weighting vector applied in the antenna array. Chapters 2-4 are primarily a collection of other's work.

Chapters 5-7 represent the author's original research for this dissertation. Chapter 5 deals with a specific problem in a wireless communication system, namely interference suppression in an adaptive array. An optimization problem is derived whose solution yields an optimal array for a given interference environment, as defined in that chapter. Solutions of this optimization problem (that is, array geometries) are presented for a specific situation and the gains in performance are illustrated.

Chapter 6 deals with the minimum possible sidelobe level for a linear antenna array with a fixed number of elements. A method of determining the optimal sidelobe-minimizing weight vector is determined that holds for an arbitrary antenna type, scan angle, and beamwidth. This method of weight selection, coupled with a geometrical optimization routine, yield a lower bound on sidelobe levels in linear antenna arrays. The minimum sidelobe levels of arrays with an optimized geometry are compared to those with a standard (or non-optimized) geometry. The methods are employed on arrays of varying size and beamwidths, and with different types of antenna elements.

Chapter 7 deals with the determination of minimum sidelobe levels in planar or two-dimensional arrays. The method of weight selection is extended from the linear to the planar case along with the geometrical optimization routine. Two-dimensional arrays

are optimized of varying sizes and beamwidths, and made up of different antenna types. The narrowband assumption is then discarded and optimal weights are derived for the wideband situation. Optimal geometries are then presented for the wideband case for arrays made up of both omnidirectional and patch antenna elements.

Chapter 8 summarizes the important results and presents conclusions based on the solutions. Finally, future problems of interest are discussed. The remainder of this chapter presents a literature survey of previous research on array geometry optimization.

1.2. Literature Survey

The first articles on improving array performance via geometry optimization dates back to the early 1960s. Unz [3] studied linear arrays in 1960 and noted that performance improvement could be obtained by holding the weights constant and varying the element positions. In 1960, King [4] proposed eliminating grating lobes via element placement in an array. In 1961, Harrington [5] considered small element perturbations in an attempt to synthesize a desired array pattern.

The concept of 'thinned arrays' was introduced in the early 1960s as well. It was noted that in large, periodically spaced antenna arrays, removing some of the elements did not noticeably degrade the array's performance. This method of altering an array's geometry was introduced by Skolnik et al. [6] and was first studied deterministically – attempting to systematically determine the minimum number of elements required to achieve a desired performance metric. For large arrays, the problem was tackled in a statistical fashion to avoid the excessive amount of computation time required to determine an optimal thinned array [7].

Stutzman [8] introduced a simple method of designing nonuniformly spaced linear arrays that is based on Gaussian-quadrature that involves fairly simple calculations. In addition, he showed that by appropriate scaling of the element spacings, some of the elements will lie in the region where the ideal source has a small excitation, and thus can be omitted from the array (another method of array thinning).

Array geometry plays a critical roll in the direction-finding capabilities of antenna arrays. Pillai et al. [9] shows that for linear aperiodic arrays, there exists an array that has superior spatial-spectrum estimation ability. Gavish and Weiss [10] compared array geometries based on how distinct the steering vectors are for distinct signal directions; they proposed that larger distinctions lead to less ambiguity in direction finding. Ang et al. [11] also evaluated the direction-finding performance of arrays by varying the elements' positions based on a genetic algorithm.

Antenna arrays are also used for diversity reception, or comparing signal power at spatially distinct locations and processing the signals based on their relative strength. A textbook proof analyzing uniformly distributed multipath components suggest arrays will exhibit good diversity characteristics if the antennas are separated by at least 0.4λ [12].

An analytical method of choosing a linear array geometry for a given set of weights is presented in [13]; this method was also extended to circular and spherical arrays [14]. This method requires a specified array pattern and set of weights; it then attempts to determine an array geometry that closely approximates the desired array pattern. The method does not guarantee a global optimum for the element positions. In [15] the weights are optimized and then linear array scaled to find an optimal geometry. A

method of perturbing element positions to place nulls in desired directions is described in [16].

Due to the large increase in the computational capability of computers, array geometry optimization has been under investigation recently using biologically inspired algorithms, such as Genetic Algorithms (GA). Khodier and Christodoulou [17] used the Particle Swarm Optimization (PSO) method to determine optimal sidelobe-minimizing positions for linear arrays assuming the weights were constant. In [18], PSO methods were used for planar array synthesis in minimizing sidelobes, along with null-placement. Tennant et al. [19] used a genetic algorithm to reduce sidelobes via element position perturbations. In [20], the authors demonstrate sidelobe minimization by choosing a geometry based on the Ant Colony Optimization (ACO) method.

In addition to geometry considerations, the minimum possible sidelobe level for an array is of interest. For linear, equally spaced arrays, the problem of determining the optimal weights was solved by Dolph and published in 1946 [21]. This method is known as the Dolph-Chebyshev method, because Dolph uses Chebyshev polynomials to obtain the excitation coefficients. The method returns the minimum possible null-to-null beamwidth for a specified sidelobe level (or equivalently, the minimum possible sidelobe level for a specified null-to-null beamwidth). This method has an implicit maximum array spacing for a given beamwidth [22]. Riblet [23] showed that for arrays with interelement spacing less than $\lambda/2$, there exists a set of weights that give a smaller null-to-null main beam than Dolph's method. However, Riblet only derives the results for arrays with an odd number of elements. The Dolph-Chebyshev method produces

sidelobes that have equal amplitudes. A more generalized version of Dolph's algorithm (called an equiripple filter) is also frequently used in the design of Finite Impulse Response (FIR) filters in the field of signal processing [24].

In 1953, DuHamel extended the work of Dolph to endfire linear arrays with an odd number of elements [25]. Dolph's work was also considered for the case of non-isotropic sensors; the problem was not solved for the general case [26]. The optimum sidelobe-minimizing weights for broadside, non-uniformly spaced symmetric linear arrays with real weights can now be found using linear programming [27]. The general case of non-uniform arrays with arbitrary scan angle, beamwidth and antenna pattern is derived in Chapter 6. In [28], the authors attempt to simultaneously optimize the weights and the positions of a 25-element linear array using a Simulated Annealing (SA) algorithm. They make no claim that their results are optimal, but do show the sidelobes lowered via the optimization method. Adaptive antenna arrays began with the work of Bernard Widrow in the 1960s [29]. Optimizing an adaptive antenna array's geometry was performed in [30] with regards to suppressing interference; this work is the subject of Chapter 5.

The effect of array geometry on wireless systems in urban environments using Multiple-Input Multiple-Output (MIMO) channels has been studied [31]. The array geometry is shown to have a significant impact on the MIMO channel properties, including the channel capacity. Because of the difficulty in examining array geometry and determining an optimal array, the impact of geometry on performance was studied by considering standard arrays such as the uniform linear array. The effect of array orientation on MIMO wireless channels was investigated in [32].

II. FUNDAMENTAL CONCEPTS OF ANTENNA ARRAYS

2.1. Introduction

An antenna array is a set of *N* spatially separated antennas. Put simply, an array of antennas does a superior job of receiving signals when compared with a single antenna, leading to their widespread use in wireless applications.

Arrays in practice can have as few as N=2 elements, which is common for the receiving arrays on cell phone towers. In general, array performance improves with added elements; therefore arrays in practice usually have more elements. Arrays can have several thousand elements, as in the AN/FPS-85 Phased Array Radar Facility operated by U. S. Air Force [33].

The array has the ability to filter the electromagnetic environment it is operating in based on the spatial variation of the signals present. There may be one signal of interest or several, along with noise and interfering signals. The methods by which an antenna array can process signals in this manner are discussed following an elementary discussion of antennas.

2.2. Antenna Characteristics

Throughout this dissertation, a Cartesian coordinate system with axis labels *x*, *y*, and *z* will be used along with spherical coordinates θ (polar angle ranging from 0 to π, measured off the z-axis) and ϕ (azimuth angle ranging from 0 to 2π, measured off the *x*-axis). The coordinates are illustrated in Figure 1.

A physical antenna has a radiation pattern that varies with direction. By reciprocity, the radiation pattern is the same as the antenna's reception pattern [34], so the two can be discussed interchangeably. The radiation pattern is also a function of frequency;

however, except where noted, it will be assumed a single frequency is of interest (described by the corresponding wavelength λ). The radiation pattern takes different shapes depending on how far the observation is from the antenna – these regions, in order of increasing distance from the antenna, are commonly called the reactive near-field region, the radiating near-field (Fresnel) region and the far-field (Fraunhofer) region [22]. For an antenna of maximum length *D*, the far-field region occurs when the following two conditions are met:

$$R > \frac{2D^2}{\lambda} \tag{2.1}$$

$$R >> \lambda . \tag{2.2}$$

For a modern cellular phone operating at 1.9 GHz with an antenna length of roughly D=4 cm, both inequalities are achieved for R>2 meters. In practice, antennas communicate in the far-field region, and this is assumed throughout.

The radiated far-zone field of an antenna will be described by the function $F(R,\theta,\phi)$. For example, the far-zone field radiated by a short dipole of length *L* with uniform current *I* is given by [33]:

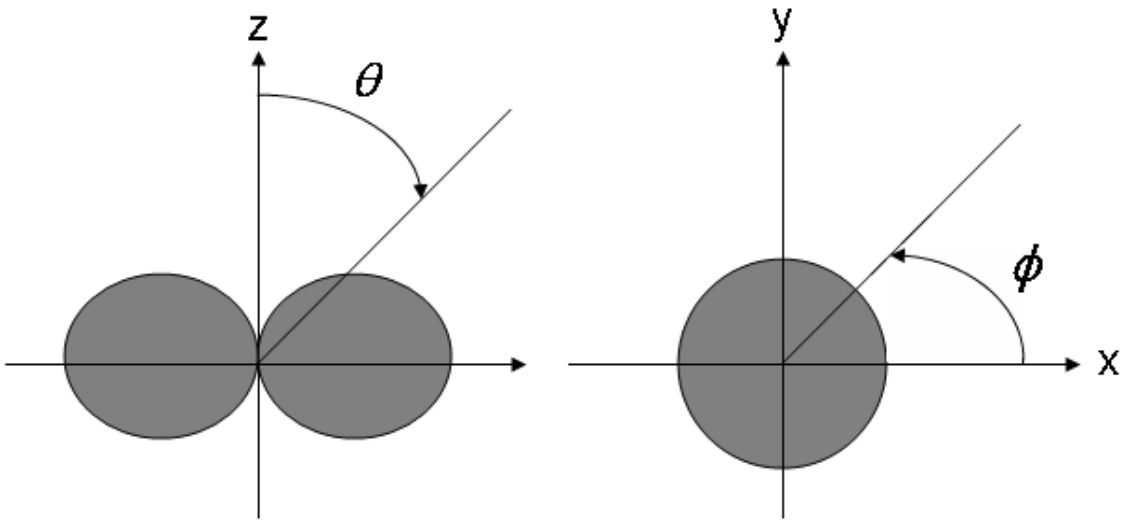
$$F(R,\theta,\phi) = \frac{jIL\eta_0 e^{-jkR}}{2R}\sin\theta \tag{2.3}$$

where $j = \sqrt{-1}$, η_0 is the impedance of free space, and $k = 2\pi/\lambda$ is the wavenumber.

The normalized field pattern will be of frequent interest in this work. This function, denoted by $f(\theta,\phi)$, describes the angular variation in the reception pattern of the antenna. For the short dipole, the normalized field pattern is expressed as

$$f(\theta,\phi) = \sin\theta . \tag{2.4}$$

This field pattern is plotted in Figure 1. The horizontal axis in Figure 1(a) can be the *x*- or *y*-axis; due to symmetry the elevation pattern will not change.



(a) Elevation Pattern (b) Azimuthal Pattern

Figure 1. (a) Elevation and (b) azimuthal patterns for a short dipole.

Directivity (or maximum directivity) is an important antenna parameter that describes how much more directional an antenna is from a reference source, usually an isotropic radiator. An antenna with a directivity of 1 (or 0 dB) would be an isotropic source; all actual antennas exhibit a directivity higher than this. The higher the directivity, the more pointed or directional the antenna pattern will be. Directivity, *D*, can be calculated from

$$D = \frac{4\pi}{\int_0^{2\pi}\int_0^{\pi} [f(\theta,\phi)]^2 \sin\theta \, d\theta \, d\phi}. \qquad (2.5)$$

The directivity of the short dipole discussed previously is 1.5 (1.76 dB).

Antennas are further described by their polarization. The polarization of an antenna is the same as the polarization of its radiated fields. The polarization of the radiated field is the figure traced out by the electric field at a fixed location in space as a function of time. Common polarizations are linear, elliptical and circular polarization. The polarization of the short dipole is linear.

If an antenna is attempting to receive a signal from an electromagnetic wave, it must be matched to the polarization of the incoming wave. If the wave is not matched to the antenna, part or all of the energy will not be detected by the antenna [22]. In this dissertation, unless otherwise noted, it will be assumed that the antennas are properly matched in polarization to the desired waves.

Further information on antennas can be found in several popular textbooks [22, 35-36]. The preceding discussion will be sufficient for the purposes in this work.

2.3. Wireless Communication

The primary purpose of antenna systems is for communication; however, they are also used for detection [37]. The information to be transmitted or received will be represented by $m(t)$. The message $m(t)$ will be assumed to be band-limited to B Hz, meaning almost all the energy has frequency content below B Hz. In the earlier days of radio, $m(t)$ had the information coded directly into the amplitude or frequency of the signal (as in AM or FM radio). Information today is primarily encoded into digital form, and $m(t)$ is a train of a discrete set of symbols representing 1s and 0s. The information is still encoded into the amplitude and phase of these symbols; however, the amplitudes and phases now take on a discrete set of values. In the most basic form of digital

communication, binary phase shift keying (BPSK), *m(t)* is either +1 or -1 (representing a 1 or a 0), so that the information is encoded into the phase. Note that *m(t)* can be complex, where the real part represents the in-phase component of the signal and the imaginary part corresponds to the quadrature component [38]. Digital communication is used because of its high data rate, lower probability of error than in analog communication (along with error-correcting codes), high spectral efficiency and high power efficiency [12].

The message *m(t)* is then modulated up to the frequency used by the antenna system. The transmitted signal *s(t)* is given by

$$s(t) = m(t)e^{j2\pi f_c t} \tag{2.6}$$

where f_c is the carrier (or center) frequency used by the antenna system. Note that in general $B << f_c$. Typically, the energy then lies within the frequency spectrum in a very narrow band around f_c, so that the transmitted signal is assumed to be a monochromatic plane wave. If the signal is sufficiently broadband that the narrowband assumption cannot be applied, the signal can be processed by filtering it into distinct narrow bands and processing each separately.

In the far field the narrowband signal will have the characteristics of a monochromatic plane wave. Assume that the wave is traveling in the direction defined by (θ, ϕ) relative to a reference point (for instance, a receiving antenna). The wavevector **k** is defined to represent the magnitude of the phase changes along the *x*-, *y*-, and *z*-directions:

$$\mathbf{k} = (k_x, k_y, k_z) = \frac{2\pi}{\lambda}(\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta). \tag{2.7}$$

The spatial variation of the signal can then be written as

$$S(x, y, z, t) = s(t)e^{-j(k_x x + k_y y + k_z z)}. \tag{2.8}$$

Defining the position vector as **R**=(*x,y,z*), (2.8) can be written more compactly as

$$S(\mathbf{R}, t) = s(t)e^{-j\mathbf{k}\cdot\mathbf{R}}. \tag{2.9}$$

Digital signal processors operating on a single antenna can only process signals based on their time variation. Space-time filters process signals based on their spatial and temporal variation [39]. In order to do spatial filtering, an array of sensors is required.

2.4. Antenna Arrays

The basic setup of an arbitrary antenna array is shown in Figure 2. The location of the n^{th} antenna element is described by the vector $\mathbf{d}_n$, where

$$\mathbf{d}_n = \begin{bmatrix} x_n & y_n & z_n \end{bmatrix}. \tag{2.10}$$

The set of locations of an *N*-element antenna array will be described by the *N*-by-3 matrix **D**, where

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \vdots \\ \mathbf{d}_N \end{bmatrix}. \tag{2.11}$$

When the array is linear (for example, all elements placed along the z-axis), the matrix **D** can be reduced to a vector.

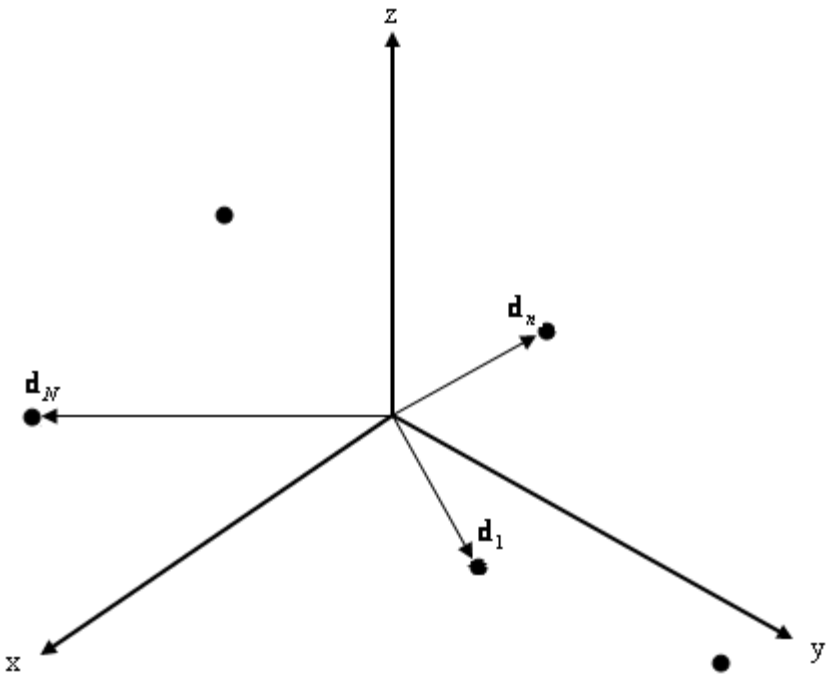


Figure 2. Arbitrary antenna array geometry.

Let the output from the n^{th} antenna at a specific time be X_n. Then the output from antenna *n* is weighted (by w_n), and summed together to produce the antenna array output, *Y*, as shown in Figure 3. See chapter 3 for a discussion of weighting methods. The array output can be written as

$$Y = \sum_{n=1}^{N} w_n X_n \,. \tag{2.12}$$

Defining

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{bmatrix} \tag{2.13}$$

and

$$\mathbf{W} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix}, \tag{2.14}$$

then (2.12) can be rewritten in compact form as

$$Y = \mathbf{W}^T \mathbf{X}, \tag{2.15}$$

where T represents the transpose operator.

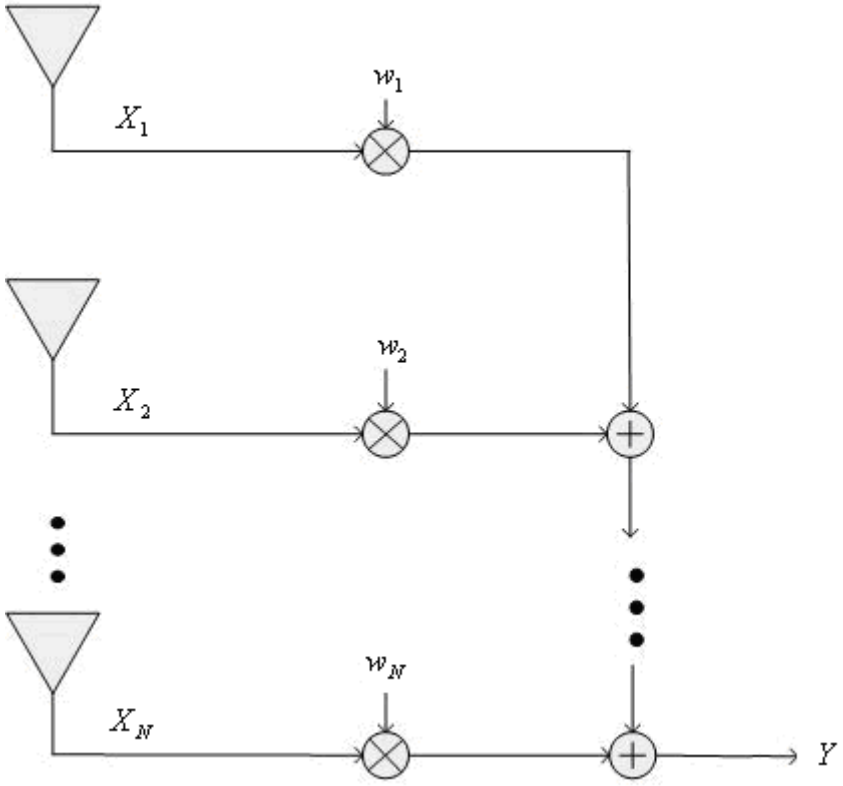


Figure 3. Spatial processing of antenna array signals.

2.5. Spatial Processing Using Antenna Arrays

Suppose the transmitted signal given by (2.9) is incident upon an *N*-element antenna array. Let the normalized field pattern for each antenna be described as a function of the wavevector (**k**) and be represented by *f*(**k**). The array output is then

$$y(t) = \sum_{n=1}^{N} w_n s(t) e^{-j\mathbf{k}\cdot\mathbf{d}_n} f(\mathbf{k}). \tag{2.16}$$

If the elements are identical, (2.16) reduces to

$$y(t) = s(t) f(\mathbf{k}) \left(\sum_{n=1}^{N} w_n e^{-j\mathbf{k}\cdot\mathbf{d}_n} \right). \tag{2.17}$$

The quantity in parenthesis is referred to as the *array factor* (*AF*). Hence, the output is proportional to the transmitted signal, multiplied by the element factor and the array factor. This factoring is commonly called pattern multiplication, and it is valid for arrays with identical elements oriented in the same direction.

A very general form for the output of an array is when there are *G* incident signals (with wavevectors $\mathbf{k}_i$, $i = 1, 2, \ldots G$) incident on *N* antennas with distinct patterns (given by $f_i(\mathbf{k})$, $i = 1, 2, \ldots, N$). Then the output is

$$y(t) = \sum_{n=1}^{N} \sum_{i=1}^{G} s_i(t) f_n(\mathbf{k}_i) w_n e^{-j\mathbf{k}_i\cdot\mathbf{d}_n}. \tag{2.18}$$

For one-dimensional arrays with elements along the z-axis (linear array),

$$\mathbf{d}_n = (0, 0, z_n). \tag{2.19}$$

Using (2.7), the *AF* reduces to

$$AF = \sum_{n=1}^{N} w_n e^{-j\frac{2\pi}{\lambda} z_n \cos\theta} . \tag{2.20}$$

The one-dimensional array factor is only a function of the polar angle. Hence, the array can filter signals based on their polar angle θ but cannot distinguish arriving signals based on the azimuth angle ϕ.

For two-dimensional arrays with elements on the *x*-*y* plane, the array factor becomes [22]

$$AF = \sum_{n=1}^{N} w_n e^{-j\frac{2\pi}{\lambda}(x_n \sin\theta\cos\phi + y_n \sin\theta\sin\phi)} . \tag{2.21}$$

The array factor is a function of both spherical angles and can therefore filter signals based on their azimuth and elevation angles.

The effect of the array on the received signal as a function of the angle of arrival is now illustrated by examining the array factor. An *N*-element array will be analyzed. For simplicity let $w_n = 1$ for all *n*, and let $\mathbf{d}_n = (0,\ 0,\ n\lambda/2)$. Then (2.19) reduces to

$$AF = \sum_{n=1}^{N} e^{-jn\pi\cos\theta} . \tag{2.22}$$

Using the identity

$$\sum_{n=0}^{N-1} c^n = \frac{1-c^N}{1-c}, \tag{2.23}$$

it follows that (2.21) can be written as

$$AF = e^{j\pi\cos\theta}\left(\frac{1-e^{-jN\pi\cos\theta}}{1-e^{-j\pi\cos\theta}}\right). \tag{2.24}$$

After factoring, the above equation simplifies to

$$AF = e^{j\pi\cos\theta}\left(\frac{e^{-j\frac{N\pi}{2}\cos\theta}}{e^{-j\frac{\pi}{2}\cos\theta}}\right)\frac{\sin\left(\frac{N\pi\cos\theta}{2}\right)}{\sin\left(\frac{\pi\cos\theta}{2}\right)}. \tag{2.25}$$

The magnitude of the array factor is plotted in Figure 4 for an array with N=5 elements, normalized so that the peak of the array factor is unity or 0 dB. The magnitude of the array factor shows that the array will receive (or transmit) the maximum energy when $\theta = 90°$. Manipulation of the weights will allow the array factor to be tailored to a desired pattern, which is the subject of Chapter 3. In addition, the response of the array factor is strongly influenced by the specific geometry (**D**) used. Selection of the weights is a simpler problem, as they array factor is a linear function of the weights. The array factor is a much more complicated function of the element positions; hence, optimizing array geometry is highly non-linear and exponentially more difficult.

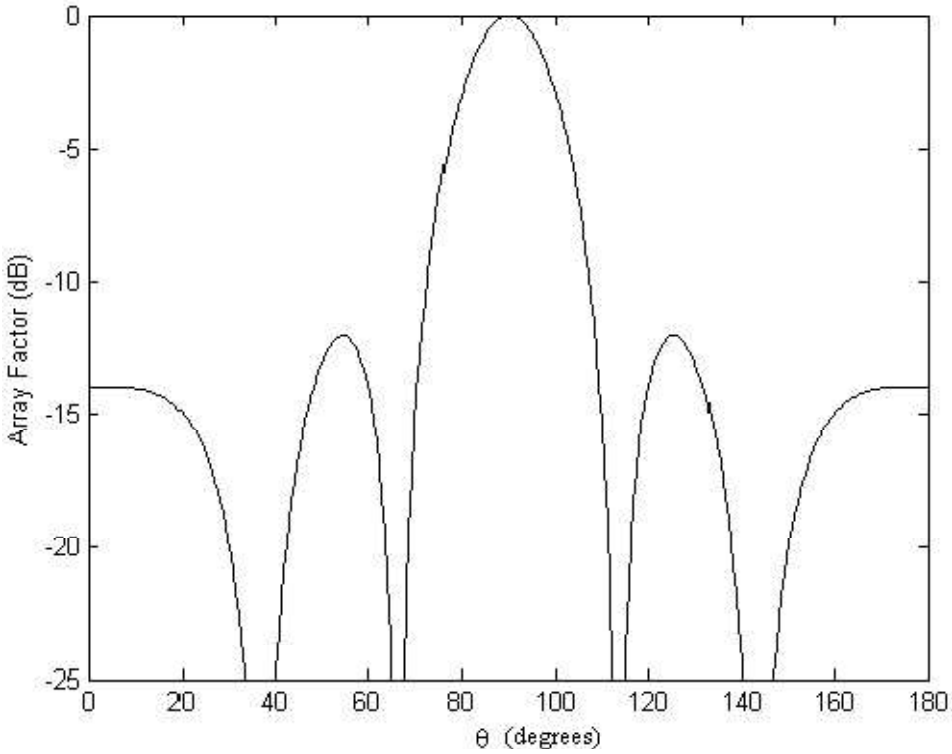


Figure 4. Magnitude of array factor for N=5 elements.

Directivity can be calculated for an array factor in the same manner as that of an antenna. In addition, important parameters of array factors include beamwidth and sidelobe level. The beamwidth is commonly specified as null-to-null or half-power beamwidth. The null-to-null beamwidth is the distance in degrees between the first nulls around the mainbeam. The half-power beamwidth is the distance in degrees between the half-power points (or 3 dB down on the array factor) around the mainbeam. The sidelobe level is commonly specified as the peak value of the array factor outside of the mainbeam.

As an example, the array factor for a 3x3 rectangular array is examined. The weights will again be uniform; i.e. $w_n = 1$ for all n. The positions for the N=9 element

array will be $\mathbf{d}_{ab} = (a\lambda/2,\ b\lambda/2,\ 0)$ for *a,b*=0,1,2. From (2.21), the array factor becomes

$$AF = \sum_{b=0}^{2} \sum_{a=0}^{2} e^{-j\pi \sin\theta(a\cos\phi + b\sin\phi)} . \tag{2.26}$$

Applying the sum formula (2.23) twice, (2.26) reduces to

$$AF = \left(\frac{1 - e^{-j3\pi \sin\theta\cos\phi}}{1 - e^{-j\pi \sin\theta\cos\phi}} \right) \left(\frac{1 - e^{-j3\pi \sin\theta\sin\phi}}{1 - e^{-j\pi \sin\theta\sin\phi}} \right). \tag{2.27}$$

By factoring, (2.27) can be written as

$$AF = \left(\frac{e^{-j3\pi \sin\theta\cos\phi/2}}{e^{-j\pi \sin\theta\cos\phi/2}} \right) \left(\frac{e^{-j3\pi \sin\theta\sin\phi/2}}{e^{-j\pi \sin\theta\sin\phi/2}} \right) *$$

$$\left(\frac{\sin(3\pi \sin\theta\cos\phi/2)}{\sin(\pi \sin\theta\cos\phi/2)} \right) \left(\frac{\sin(3\pi \sin\theta\sin\phi/2)}{\sin(\pi \sin\theta\sin\phi/2)} \right). \tag{2.28}$$

For ease in plotting, the following variables will be introduced:

$$u = \frac{k_x \lambda}{2\pi} = \sin\theta\cos\phi \tag{2.29}$$

$$v = \frac{k_y \lambda}{2\pi} = \sin\theta\sin\phi . \tag{2.30}$$

The magnitude of the array factor is plotted in Figure 5. The sidelobes are 9.54 dB down from the main lobe (which is normalized to 0 dB in the figure).

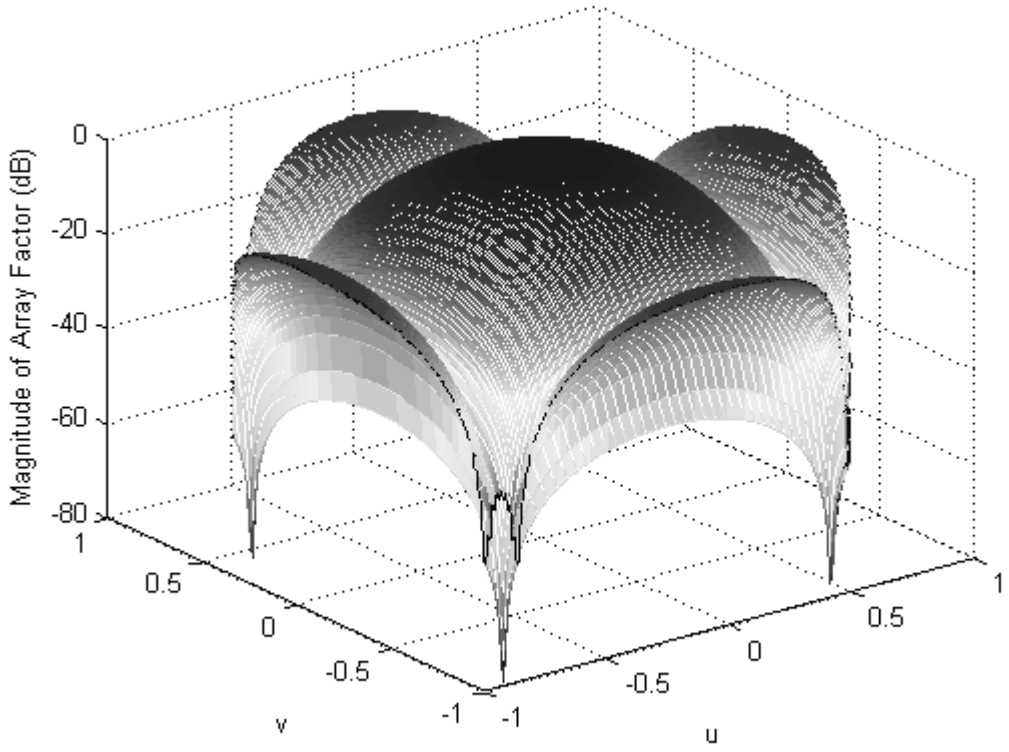


Figure 5. Magnitude of the array factor (dB) for 2-D array.

Beamwidths are more difficult to specify when the array factor is two dimensional. Commonly, beamwidths are specified in certain planes (for instance, elevation and azimuthal planes) and given in half-power or null-to-null form, as in the one-dimensional case. The sidelobe level is again the maximum value of the array factor outside of the main beam.

2.6. Aliasing

The steering vector (**v**) is the vector of propagation delays (or phase changes) across an array for a given wavevector, **k**. It can be written mathematically as

$$\mathbf{v}(\mathbf{k}) = \begin{bmatrix} e^{-j\mathbf{k}\cdot\mathbf{d}_1} \\ e^{-j\mathbf{k}\cdot\mathbf{d}_2} \\ \vdots \\ e^{-j\mathbf{k}\cdot\mathbf{d}_N} \end{bmatrix}. \tag{2.31}$$

Aliasing occurs when signals propagating in distinct directions produce the same steering vectors. In that case, the array's response towards the two directions will be identical, so that the array cannot distinguish the two directions. This is similar to the signal processing version of aliasing, where if the sampling rate is too small in time, then distinct frequencies cannot be resolved.

For uniformly spaced linear arrays, there will exist plane waves from distinct directions with identical steering vectors if the spacing between elements, Δ, is greater than $\lambda/2$. Similarly, for uniformly spaced rectangular (planar) arrays with elements on the x-y plane, there will exist distinct directions with identical steering vectors if the element spacing in the x- or y-directions is greater than $\lambda/2$. When aliasing exists, the main beam may be replicated elsewhere in the pattern. These replicated beams are referred to as *grating lobes*.

For arrays without a uniform structure, the distance between elements can be much larger than $\lambda/2$ without introducing aliasing. In this case, no two distinct angles of arrival will produce identical steering vectors. However, while aliasing technically does not occur, there may be steering vectors that are very similar so that grating lobes exist. Determining whether or not this occurs for an arbitrary array is very difficult. In general, if a non-uniform array is decided upon, the array factor can be checked to ensure that

grating lobes do not occur. Mathematical studies on the uniqueness of steering vectors can be found in [40-41].

III. WEIGHTING METHODS IN ANTENNA ARRAYS

3.1. Introduction

From (2.17), it is clear that the weights will have a significant impact on the output of the antenna array. Since the array factor is a linear function of the weights, weighting methods are well developed and can be selected to meet a wide range of objectives. These objectives include pattern steering, nulling energy from specific directions relative to an array, minimizing the Mean Squared Error (MSE) between a desired output and the actual output, or minimizing the sidelobe level outside a specified beamwidth in linear arrays. These techniques will be discussed in this chapter. In addition, adaptive signal processing methods applied to antenna arrays will be discussed. Most of the methods described here apply to arrays of arbitrary geometry. However, for simplicity, examples will be presented for uniform linear arrays with half-wavelength spacing. Hence, the element positions will be given by $\mathbf{d}_n = (0,\ 0,\ n\lambda/2)$ for $n = 0, 1, \ldots, N-1$.

3.2. Phased-Tapered Weights

The linear array of Section 2.4 had maximum response in the direction of $\theta = 90°$. The simplest method of altering the direction in which the array is steered is to apply a linear phase taper to the weights. The phase taper is such that it compensates for the phase delay associated with the propagation of the signal in the direction of interest. For example, if the array is to be steered in the direction θ_d, the weights would be given by

$$w_n = e^{jn\pi\cos\theta_d}. \tag{3.1}$$

For these weights, the array factor becomes

$$AF = \sum_{n=0}^{N-1} e^{jn\pi(\cos\theta_d - \cos\theta)}, \tag{3.2}$$

or

$$AF = \frac{1 - e^{j\pi N(\cos\theta_d - \cos\theta)}}{1 - e^{j\pi(\cos\theta_d - \cos\theta)}} \tag{3.3}$$

The magnitude of the array factor (normalized so that the peak is unity, or 0 dB) is plotted in Figure 6 for N=5 and $\theta_d = 45°$. The array factor has a maximum at the desired direction, and like the result in Figure 2.5 the sidelobes are 11.9 dB down from the mainlobe. This simple steering method can be used in two- or three-dimensional arrays as well as for arbitrary scan angles.

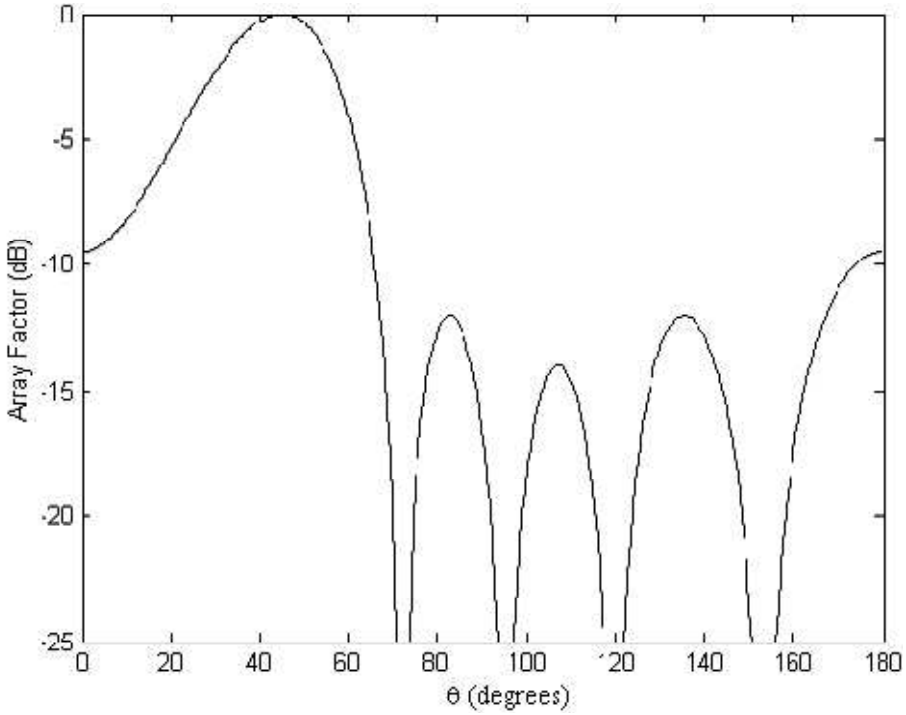


Figure 6. Array factor of steered linear array.

3.3. Schelkunoff Polynomial Method

A weighting scheme for placing nulls in specific directions of an array factor was developed by Schelkunoff [22, 42]. In general, an N-element array can null signals arriving from N-1 distinct directions.

To illustrate the method, the array factor

$$AF = \sum_{n=0}^{N-1} w_n e^{-j\pi n\cos\theta} \tag{3.4}$$

can be rewritten as a polynomial as

$$AF(z) = \sum_{n=0}^{N-1} w_n z^n \,, \tag{3.5}$$

where

$$z = e^{-j\pi\cos\theta} \,. \tag{3.6}$$

Since a polynomial can be written as the product of its own zeros, it follows that

$$AF(z) = w_{N-1} \prod_{n=0}^{N-2} (z - z_n) \,, \tag{3.7}$$

where the z_n are the zeros of the array factor. By selecting the desired zeros and setting (3.7) to (3.5), the weights can be found.

As an example, assume an N=3 element array with zeros to be placed at $45°$ and $120°$. In that case, the following values are calculated

$$z_0 = e^{-j\pi\cos 45°} \tag{3.8}$$

and

$$z_1 = e^{-j\pi\cos 120°} \,. \tag{3.9}$$

Arbitrarily letting $w_{N-1} = w_2 = 1$, (3.7) becomes

$$AF(z) = z^2 - z(z_0 + z_1) + z_0 z_1 \,. \tag{3.10}$$

Setting (3.10) equal to the original form of the array factor (3.5), the weights are easily found to be:

$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} z_0 z_1 \\ -(z_0 + z_1) \\ 1 \end{bmatrix}. \tag{3.11}$$

The normalized array factor for the specified weights is plotted in Figure 7. As desired, the pattern has nulls at $45°$ and $120°$.

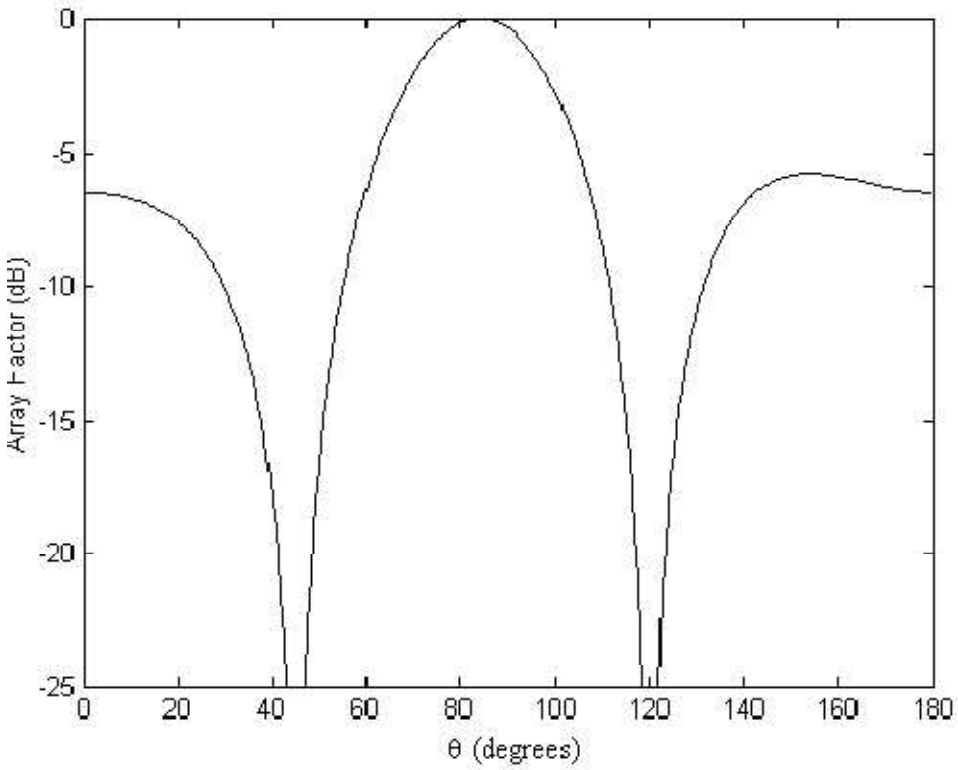


Figure 7. Array pattern with weights from Schelkunoff method.

3.4. Dolph-Chebyshev Method

Often in antenna arrays it is desirable to receive energy from a specific direction and reject signals from all other directions. In this case, for a specified main beamwidth the sidelobes should be as low as possible. For linear, uniformly spaced arrays of

isotropic sensors steered to broadside ($\theta_d = 90°$), the Dolph-Chebyshev method will return weights that achieve this. A weighting method for obtaining minimum sidelobes in arbitrarily spaced arrays of any dimension, steered to any scan angle and for any antenna type is derived in Chapter 6.

In observing array factors as in Figure 4, note that the sidelobes decrease in magnitude away from the mainbeam. To have the lowest overall sidelobe level, the sidelobe with the highest intensity should be decreased at the expense of raising the intensity of the lower sidelobes. The result will be that for the minimum overall sidelobe level, the sidelobes will all have the same peak value. Dolph observed this and employed Chebyshev polynomials, which have equal-magnitude peak variations (or ripples) over a certain range. By matching the array factor to a Chebyshev polynomial, the equal-ripple (or constant-sidelobe) weights can be obtained. The actual process is straightforward but cumbersome to write out; for details see [22]. Several articles have been written on efficient computation of the Dolph-Chebyshev weights [43-44].

As an example, a uniformly spaced linear array with half-wavelength spacing and N=6 is used. The Dolph-Chebyshev weights are calculated for a sidelobe level of -30 dB. The associated magnitude of the array factor is plotted in Figure 8. The null-to-null beamwidth is approximately $60°$. Note that all the sidelobes are equal in magnitude at -30 dB.

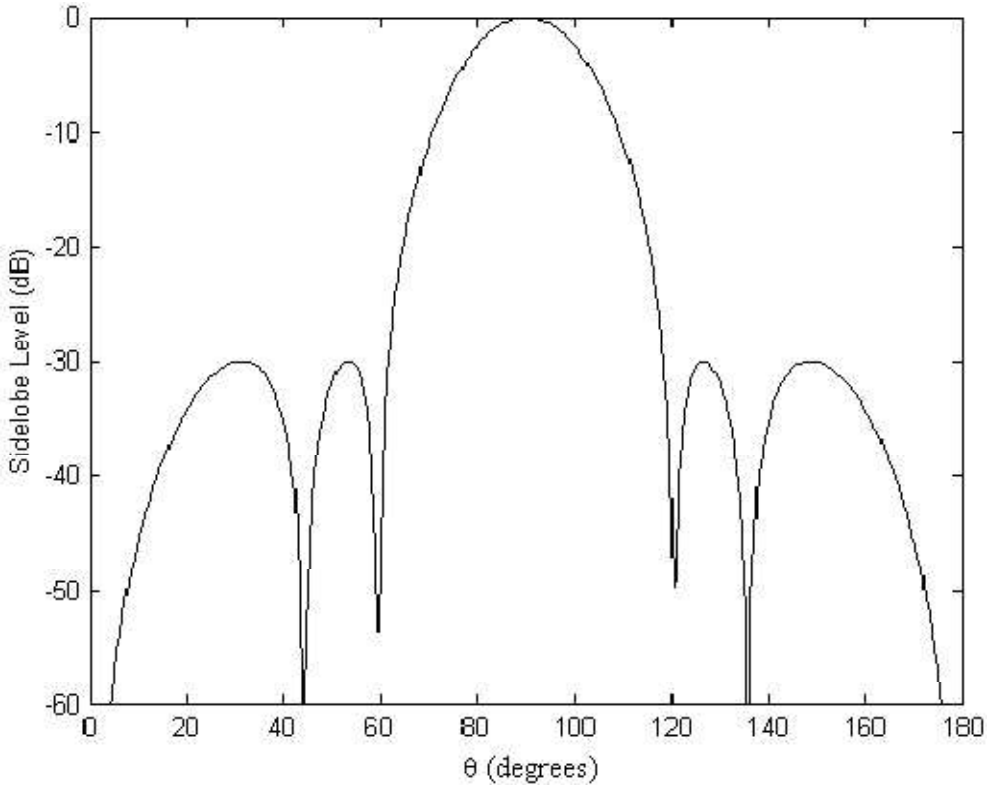


Figure 8. Dolph-Chebyshev array for N=6 with sidelobes at -30 dB.

3.5. Minimum Mean-Square Error (MMSE) Weighting

The weighting methods discussed previously have been deterministic; that is, they have not dealt with noise or statistical representations of the desired signals or interference. In this section, a more general beamforming technique is developed that takes into account the statistical behavior of the signal environment.

Assume now the input to the array consists of one desired signal, *s(t)*, with an associated wavevector $\mathbf{k}_s$. Assume there exists noise at each antenna, $n_i(t)$. The noise at each antenna can be written in vector form as

$$\mathbf{N}(t) = \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_N(t) \end{bmatrix}. \tag{3.12}$$

In addition, assume there are *G* interferers, each having narrowband signals given by $I_a(t)$ and wavevectors given by $\mathbf{k}_a$, $a = 1, 2, \ldots, G$. Using the steering vector notation for the phase delays as in (2.31), the input to the antenna array can then be written as

$$\mathbf{X}(t) = s(t)\mathbf{v}(\mathbf{k}_S) + \mathbf{N}(t) + \sum_{a=1}^{G} I_a(t)\mathbf{v}(\mathbf{k}_a). \tag{3.13}$$

The desired output from the antenna array (or spatial filter) is

$$Y_d(t) = s(t). \tag{3.14}$$

The actual output is

$$Y(t) = \mathbf{W}^H \mathbf{X}(t). \tag{3.15}$$

where *H* is the Hermitian operator (conjugate transpose). Equation (3.15) differs from (2.15) because the mathematics in the derivation will be simpler if the weights used are in the form of (3.15). The error can then be written as

$$e(t) = Y(t) - Y_d(t). \tag{3.16}$$

The minimum mean-squared error estimate (MMSE) seeks to minimize the expected value of the squared magnitude of *e(t)*. The mean-squared error (MSE) is

$$\text{MSE} = E[e(t)e^*(t)], \tag{3.17}$$

where * indicates complex conjugate and *E*[] is the expectation operator. Expanding (3.17) with (3.15), the MSE becomes

$$\text{MSE} = E[\left(\mathbf{W}^H\mathbf{X}(t) - s(t)\right)\left(\mathbf{X}^H(t)\mathbf{W} - s^*(t)\right)]. \tag{3.18}$$

Multiplying the terms above, the MSE becomes

$$\text{MSE} = E[\mathbf{W}^H\mathbf{X}(t)\mathbf{X}^H(t)\mathbf{W}] + E[s(t)s^*(t)] -$$

$$E[\mathbf{W}^H\mathbf{X}(t)s^*(t)] - E[s(t)\mathbf{X}^H(t)\mathbf{W}]. \tag{3.19}$$

The first term in (3.19) can be simplified to

$$E[\mathbf{W}^H\mathbf{X}(t)\mathbf{X}^H(t)\mathbf{W}] = \mathbf{W}^H E[\mathbf{X}(t)\mathbf{X}^H(t)]\mathbf{W}, \tag{3.20}$$

since the expectation is a linear operator and the weights are fixed. The autocorrelation matrix, $\mathbf{R}_{\mathbf{XX}}$, is defined to be

$$\mathbf{R}_{\mathbf{XX}} = E[\mathbf{X}(t)\mathbf{X}^H(t)]. \tag{3.21}$$

The second term in (3.19) is the signal power, σ_s^2:

$$\sigma_s^2 = E[s(t)s^*(t)]. \tag{3.22}$$

Defining

$$\mathbf{\Lambda} = E[\mathbf{X}(t)s^*(t)], \tag{3.23}$$

the third term in (3.19) becomes

$$E[\mathbf{W}^H\mathbf{X}(t)s^*(t)] = \mathbf{W}^H\mathbf{\Lambda}. \tag{3.24}$$

Finally, the fourth term in (3.19) is just the complex conjugate of the third term:

$$E[s(t)\mathbf{X}^H(t)\mathbf{W}] = \mathbf{\Lambda}^H\mathbf{W}. \tag{3.25}$$

Equation (3.19) can then be rewritten as

$$\text{MSE} = \mathbf{W}^H\mathbf{R}_{\mathbf{XX}}\mathbf{W} + \sigma_s^2 - \mathbf{W}^H\mathbf{\Lambda} - \mathbf{\Lambda}^H\mathbf{W}. \tag{3.26}$$

The goal is to find the **W** that produces the minimum MSE. The gradient of (3.26) with respect to **W** is

$$\nabla \text{MSE} = 2\mathbf{R}_{\mathbf{XX}}\mathbf{W} - 2\mathbf{\Lambda}. \tag{3.27}$$

Setting (3.27) equal to zero and solving gives the optimal weights, $\mathbf{W}_{opt}$:

$$\mathbf{W}_{opt} = \mathbf{R}_{\mathbf{XX}}^{-1}\mathbf{\Lambda}. \tag{3.28}$$

Equation (3.28) requires two pieces of information, the autocorrelation matrix and the vector $\mathbf{\Lambda}$. The inverse of the autocorrelation matrix is often estimated using the Sample Matrix Inverse (SMI) method. The estimate is denoted with the bar overhead, $\overline{\mathbf{R}_{\mathbf{XX}}^{-1}}$, and uses *K* snapshots of the input vector **X** to formulate the estimate.

$$\overline{\mathbf{R}_{\mathbf{XX}}^{-1}} = \left[\sum_{k=1}^{K}\mathbf{X}(k)\mathbf{X}^{H}(k)\right]^{-1}. \tag{3.29}$$

Assuming the signal of interest is uncorrelated in time with the noise and interference, (3.29) along with (3.13) yields

$$\mathbf{\Lambda} = E\left[\left(s(t)\mathbf{v}(\mathbf{k}_S) + \mathbf{N}(t) + \sum_{a=1}^{G} I_a(t)\mathbf{v}(\mathbf{k}_a)\right)s^*(t)\right] = \sigma_s^2\mathbf{v}(\mathbf{k}_s). \tag{3.30}$$

Hence, the vector $\mathbf{\Lambda}$ can be determined if the direction of the signal (given by $\mathbf{k}_s$) and the signal power (σ_s^2) are known. Often the incoming direction and power can be determined by using a training sequence to calibrate the array. The optimal weights can be rewritten using (3.30) as

$$\mathbf{W}_{opt} = \sigma_s^2\mathbf{R}_{\mathbf{XX}}^{-1}\mathbf{v}(\mathbf{k}_s). \tag{3.31}$$

Equation (3.31) represents the weights that minimize the MSE. The optimal MSE is found from substituting (3.31) into (3.26):

$$\mathrm{MSE}_{opt} = \sigma_s^2 - \sigma_s^4 \mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_{\mathbf{XX}}^{-1}\mathbf{v}(\mathbf{k}_s)\,. \tag{3.32}$$

Similar formulations can be used to formulate weights that maximize the signal to noise ratio (SNR) when the autocorrelation matrix of the interference and noise can be estimated [45].

As an example, consider the case of the desired signal arriving from $\theta_d = 110°$ with a signal power of σ_s^2=1. Two interferers, arriving from $\theta_1 = 40°$ and $\theta_2 = 90°$, each have $\sigma_I^2 = 10$. The array will have *N*=3 elements. Two cases will be considered, the first with noise power $\sigma_n^2 = 0.01$ (SNR=20 dB), and the second with $\sigma_n^2 = 1$ (SNR=0 dB). The optimal weights can then be calculated using (3.31).

The resulting array factor magnitudes are plotted in Figure 9. Observe that for the high SNR case, the pattern places nulls exactly in the directions of the interferers. For the low SNR case, the pattern puts less emphasis on nulling out the interferers. This is because the gain in combating independent noise sources is best obtained by combining the received signals with equal gain [45]. Note that neither array factor is maximum towards the signal of interest, $\theta_d = 110°$.

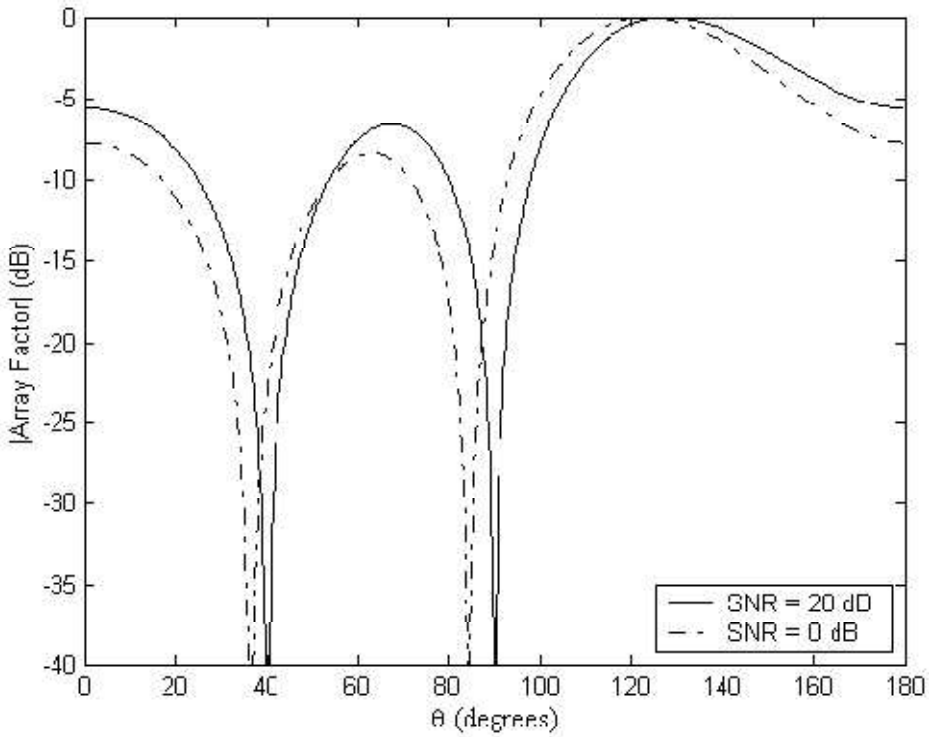


Figure 9. Array factor magnitudes for MMSE weights.

3.6. The LMS Algorithm

The weights discussed up until now have not been adaptable; that is, they do not attempt to change as the signal environment changes. A weight updating strategy that changes with its environment is known as an adaptive algorithm and adaptive signal processing has become a field in itself. In this section, the first and arguably most widely used adaptive algorithm is discussed, the Least Mean Square (LMS) algorithm. This algorithm was invented by Bernard Widrow along with M. E. Hoff, Jr. and published in a primitive form in 1960 [46]. The Applebaum algorithm [47] was developed independently in 1966 and largely uses the same ideas.

The algorithm assumes some *a priori* knowledge; in this version (the spatial LMS algorithm), the known information is assumed to be the desired signal power (σ_s^2) and

the signal direction, $\mathbf{k}_s$. The algorithm iteratively steps towards the MMSE weights. If the environment changes, then the algorithm will step towards the new MMSE weights. Samples of the input vector, **X**, will be ordered and written as **X**(k).

To accomplish the iterative minimization of the MSE, recall that the gradient of the MSE as a function of the weights (**W**) is given by (3.26). The LMS algorithm approximates the autocorrelation matrix at each time step by

$$\overline{\mathbf{R}_{\mathbf{XX}}}(k) = \mathbf{X}(k)\mathbf{X}^H(k). \tag{3.33}$$

Then the gradient of the MSE can be approximated at each time step as

$$\overline{\nabla \mathrm{MSE}}(k) = 2\mathbf{X}(k)\mathbf{X}^H(k)\mathbf{W}(k) - 2\sigma_s^2\mathbf{v}(\mathbf{k}_s). \tag{3.34}$$

To minimize the MSE, the LMS algorithm simply increments the weights in the direction of decreasing the MSE. The update algorithm for the weights can then be written as

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \frac{\lambda}{2}\overline{\nabla \mathrm{MSE}}(k), \tag{3.35}$$

where λ is a positive scalar that controls how large the steps are for the weights. Substituting (3.34) into (3.35) produces the LMS algorithm:

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \lambda\left\{\sigma_S^2\mathbf{v}(\mathbf{k}_S) - \mathbf{X}(k)\mathbf{X}^H(k)\mathbf{W}(k)\right\}. \tag{3.36}$$

Equation (3.36) actually represents one of the many forms of the LMS algorithm. The versions primarily differ in the *a priori* knowledge required.

The algorithm's simplicity is its primary reason for its widespread use. In addition, it has fairly decent convergence properties and has been extensively studied. In order to have stable results (the expected MSE will converge to a constant value), the parameter λ should be chosen according to

$$0 < \lambda < \frac{2}{\lambda_{MAX}(\mathbf{R_{XX}})}, \tag{3.37}$$

where $\lambda_{\max}(\mathbf{R_{XX}})$ is the largest eigenvalue of the autocorrelation matrix [48]. The speed of the convergence is governed by the condition number (ratio of largest to smallest eigenvalues) of the autocorrelation matrix [49].

As an example of the LMS algorithm, the interference and noise scenario of Section 3.3 is again considered, this time with a SNR=20 dB. The noise will be additive white Gaussian noise (AWGN) that is independent at each antenna. The array will be the linear array of N=5 elements with half-wavelength spacing. The algorithm is initiated with a weight of unity applied to all elements:

$$\mathbf{W}(1) = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}. \tag{3.38}$$

The parameter λ is chosen to be

$$\lambda = 0.015 = \frac{0.1}{\lambda_{\max}(\mathbf{R_{XX}})}. \tag{3.39}$$

An example run is conducted, and the resulting MSE is plotted at each iteration [from (3.26)], along with the optimal MSE [from (3.31)] in Figure 10. The LMS algorithm is fairly efficient in moving towards the optimal weights for this case. Since the algorithm uses a guess of the autocorrelation matrix at each time step, some of the steps actually increase the MSE. However, on average, the MSE decreases. This algorithm is also fairly robust to changing environments.

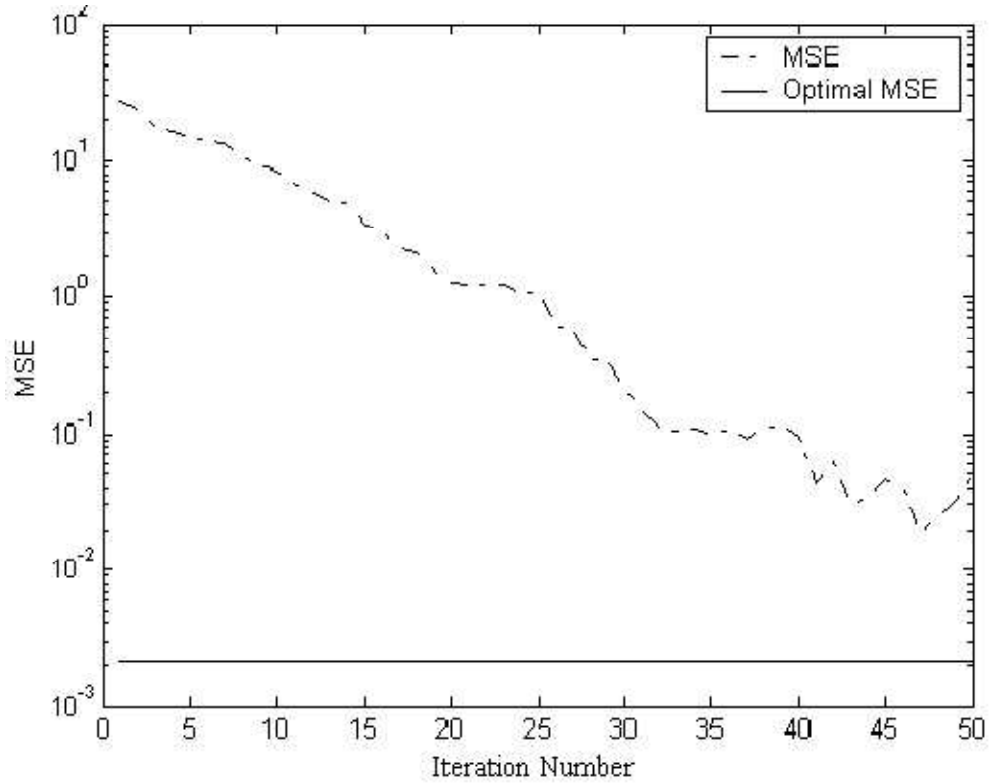


Figure 10. MSE at each iteration, along with the optimal MSE.

Several adaptive algorithms have expanded upon ideas used in the original LMS algorithm. Most of these algorithms seek to produce improved convergence properties at the expense of increased computational complexity. For instance, the recursive least-square (RLS) algorithm seeks to minimize the MSE just as in the LMS algorithm [48]. However, it uses a more sophisticated update to find the optimal weights that is based on the matrix inversion lemma [45]. Both of these algorithms (and all others based on the LMS algorithm) have the same optimal weights the algorithms attempt to converge to, given by (3.31).

IV. METHODS OF ANTENNA ARRAY GEOMETRY OPTIMIZATION

4.1. Introduction

The field of electromagnetics was unified into a coherent theory and set of four fundamental equations by James Clerk Maxwell in 1879 [50]. These equations are known as Maxwell's equations. The first is Gauss's law:

$$\nabla \cdot \mathbf{D} = \rho_V \,, \tag{4.1}$$

where **D** is the electric flux density and ρ_V is the volume charge density. The second equation states that "magnetic monopoles do not exist", and can be written in mathematical form:

$$\nabla \cdot \mathbf{B} = 0 \,, \tag{4.2}$$

where **B** is the magnetic flux density. The third equation is known as Ampere's law:

$$\nabla \times \mathbf{H} - \frac{\partial \mathbf{D}}{\partial t} = \mathbf{J} \,, \tag{4.3}$$

where **H** is the magnetic field and **J** is the impressed electric current density. The fourth is Faraday's law:

$$\nabla \times \mathbf{E} + \frac{\partial \mathbf{B}}{\partial t} = 0 \,, \tag{4.4}$$

where **E** is the electric field.

While there are only four equations in the set, they are complicated enough that they can only be solved in closed form for some basic canonical shapes. As a result, numerical methods for solving electromagnetic problems became necessary. A thorough introduction and survey of the methods can be found in [51].

Among the most popular of the numerical methods include the finite-difference time domain (FDTD) method developed in 1966 by Yee at Lawrence Livermore National

Laboratories [52]. This method discretizes space and time and computes the electric and magnetic fields using discretized forms of Ampere's law and Faraday's law. The algorithm initially computes the electric fields (assuming the magnetic fields are known) using Ampere's law. A small time step later, the algorithm computes the magnetic fields at that time using Faraday's law (along with the calculated electric field). This process is repeated as long as desired and has been widely successful in modeling numerous electromagnetic problems. Another popular method is the Integral Equation (IE) Method of Moments (MoM), which numerically solves complex integral equations by assuming a solution in the form of a sum of weighted basis functions along the structure being analyzed. The weights are then found by introducing boundary conditions and solving an associated matrix for the weights, thereby leading to the solution [53].

Because of the difficulty in obtaining solutions to electromagnetic problems, optimization is not simple. Antenna arrays, being a specific class of electromagnetic problems, are no exception. However, significant developments over the past 50 years in the field of mathematical optimization are now being applied to electromagnetic problems. The tremendous increase in computing power over the last few decades has enabled complex problems to be solved and led to large advances in the fields of numerical electromagnetics and in optimization. This chapter describes the optimization methods that have penetrated the electromagnetic field in the late 20th century.

The first set of methods, linear programming and convex optimization problems, are part of a class of optimization methods that are deterministic. The problems have a

unique solution that can be verified to be globally optimal. However, due to the complex nature of the problems, the solutions are obtained numerically and not analytically.

The second set of methods discussed in this chapter, Simulated Annealing (SA) and Particle Swarm Optimization (PSO), are part of a class of optimization methods that are stochastic in nature. These methods produce solutions to the most general optimization problems that have very little structure and cannot be solved via other methods. The resulting solutions from these methods are unfortunately not verifiable to be globally optimal. However, they have recently been receiving a lot of attention in the antenna field because they can be applied to a wide range of problems and can be used to obtain solutions that achieve a desired performance metric. In March 2007, the *IEEE Transactions on Antennas and Propagation* dedicated the entire issue to optimization techniques in electromagnetics and antenna system design. An overview of the methods and their applications to electromagnetics can be found in [54]. Many of these papers used techniques that were stochastic in nature, including the popular genetic algorithm (GA)[55].

These optimization techniques are often coupled with the numerical methods discussed previously. For instance, the PSO algorithm was used in conjunction with the FDTD method in [56]. The genetic algorithm was used along with the method of moments for the design of integrated antennas in [57].

4.2. Linear Programming

The most general form of a mathematical optimization problem can be expressed as

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{x} \in \chi \end{aligned} . \tag{4.5}$$

Here *f(***x***)* is the objective function to be minimized, and χ is known as the feasible set, or set of all possible solutions. In the following, 'subject to' will be abbreviated as '*s. t.*'. The solution ($\mathbf{x}_{opt}$) to (4.5) will have the property

$$f(\mathbf{x}_{opt}) \leq f(\mathbf{x}) \ \forall \ \mathbf{x} \in \chi, \tag{4.6}$$

where $\forall$ is commonly used in mathematics to state 'for all'. The solution is not necessarily unique but exists as long as χ is not the empty set.

A linear program (LP) is a widely studied optimization problem that has numerous practical applications, one of which is shown at the end of this section. The theory on this subject was developed by George Dantzig and John von Neumann in 1947 [58]. The variables in a linear program are written as an *N*-dimensional vector of real numbers:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}. \tag{4.7}$$

The objective function to be minimized is a linear function of the problem variables:

$$f(\mathbf{x}) = \mathbf{c}^T \mathbf{x}, \tag{4.8}$$

where **c** is an *N*-dimensional (real) vector.

The feasible set χ in a linear program is a set of *M* affine inequalities. Each inequality can be written in the form:

$$\mathbf{a}_i^T \mathbf{x} \leq b_i, \tag{4.9}$$

where $\mathbf{a}_i$ is an *N*-dimensional real vector, b_i is a real number, and $i = 1, 2, \ldots, M$. Without any constraints, the vector **x** can be any vector in $\Re^N$. Each constraint in the

form of (4.9) divides the space $\Re^N$ into two half-spaces. In two dimensions (N=2), the divider is a straight line; in three dimensions (N=3) the divider is a plane and so on. The resulting feasible region χ is the intersection of all of these half-spaces. The set of constraints

$$\begin{aligned} \mathbf{a}_1^T \mathbf{x} &\le b_1 \\ \mathbf{a}_2^T \mathbf{x} &\le b_2 \\ &\vdots \\ \mathbf{a}_M^T \mathbf{x} &\le b_M \end{aligned} \tag{4.10}$$

are often abbreviated as

$$\mathbf{A}\mathbf{x} \le \mathbf{b} \,. \tag{4.11}$$

In (4.11), **A** is an $M \times N$ matrix given by

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_M^T \end{bmatrix}, \tag{4.12}$$

and **b** is an M-dimensional vector given by

$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix}. \tag{4.13}$$

The inequality sign in (4.11) is understood to be component-wise (must be satisfied for all inequalities). The standard form of an LP can then be written as in (4.14).

$$\begin{aligned} &\min \ \mathbf{c}^T \mathbf{x} \\ &s.t. \ \ \mathbf{A}\mathbf{x} \le \mathbf{b} \end{aligned} \tag{4.14}$$

An equality constraint can be viewed as two inequality constraints, so LPs are often written in the form given in (4.15), where **C** is a matrix and **f** is a vector. If no vector **x** satisfies all the constraints, the problem is said to be infeasible.

$$\begin{aligned} &\min \ \mathbf{c}^T\mathbf{x} \\ &s.t. \ \ \mathbf{A}\mathbf{x} \le \mathbf{b} \\ &\qquad \mathbf{C}\mathbf{x} = \mathbf{f} \end{aligned} \tag{4.15}$$

Extensive work has gone into understanding the problem presented in (4.15). Solutions found to (4.15) must satisfy a set of optimality conditions, and they can therefore be verified to be globally optimal [59]. In addition, several numerical methods, such as the simplex algorithm [60] and the rapid interior point method [61], have been developed to efficiently solve the LP. As a result, if an optimization problem can be put into the form of an LP, an optimal vector (if one exists) can be found efficiently and verified to be globally optimal. Commonly used computational software programs, including Mathematica and Matlab, now have built in routines for solving linear programs.

To illustrate the utility of linear programs, a method of determining sidelobe minimizing weights for symmetric linear arrays with real weights steered to broadside will be presented. This follows the discussion in [27]. The results will be extended in Chapter 6 to work for arbitrarily spaced arrays with complex weights steered to any angle, with arbitrary antenna elements and an arbitrary beammwidth.

A symmetric linear array is an array with elements spaced symmetrically about the origin, as shown in Figure 11.

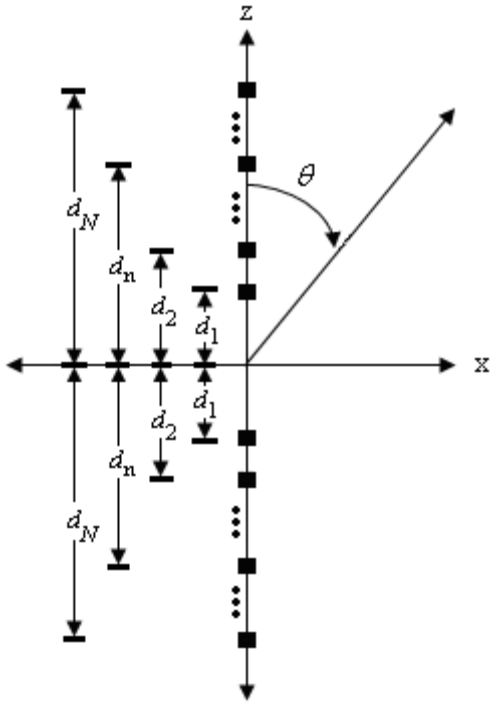


Figure 11. Symmetric linear array.

An array of this type with real weights and 2*N* elements will have an array factor given by

$$AF(\theta) = \sum_{n=1}^{N} w_n \cos(2\pi d_n \cos\theta) . \tag{4.16}$$

where d_n is the position of the n^{th} element along the z-axis. The objective is to determine the weights that produce the lowest possible sidelobe level. The sidelobe level will be defined as the maximum value of the magnitude of the array factor outside of a specified beamwidth. The set of all angles in which the array factor is to be suppressed will be written as Θ. The sidelobe level (*SLL*) can be written mathematically as

$$SLL = \max_{\theta \in \Theta} | AF(\theta) | . \tag{4.17}$$

Since the array factor is to be maximum at broadside, the following constraint is imposed:

$$AF(90°) = 1. \tag{4.18}$$

The problem of minimizing the sidelobe level can then be written as an optimization problem:

$$\begin{aligned} &\min \quad SLL \\ &s.t. \quad AF(90°) = 1 \end{aligned}. \tag{4.19}$$

This problem can be written as an LP in standard form. First, let *t* represent the maximum sidelobe level. Sample the region Θ into *R* sample points ($\theta_1, \theta_2, \ldots, \theta_R$). The sidelobes will be suppressed at the sample points; following the optimization procedure, it can be verified that the sidelobes are also suppressed between the samples. Equation (4.19) can be rewritten into the form given in (4.20).

$$\begin{aligned} &\min \quad t \\ &s.t. \quad AF(90°) = 1 \\ &\qquad |AF(\theta_i)| \le t, \quad i = 1, 2, \ldots, R \end{aligned} \tag{4.20}$$

Each one of the constraints in (4.20) can be written as an affine constraint as in (4.10). To see this, define the problem variables to be

$$\mathbf{X} = \begin{bmatrix} t \\ w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix}. \tag{4.21}$$

The objective function in (4.20) can be rewritten as

$$t = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \mathbf{X} = \mathbf{c}^T \mathbf{X}. \tag{4.22}$$

The equality constraint in (4.20) can be rewritten using (4.16) along with the vector **X** as

$$\begin{bmatrix} 0 & 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} t \\ w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = 1, \tag{4.23}$$

or

$$\mathbf{a}_0^T \mathbf{X} = 1. \tag{4.24}$$

Finally, the inequality constraints in (4.19) can be rewritten as

$$-t \le AF(\theta_i) \le t, \tag{4.25}$$

for $i = 1, 2, \ldots, R$. Using (4.16), the inequality on the right becomes

$$\begin{bmatrix} -1 & \cos(2\pi d_1 \cos\theta_i) & \cos(2\pi d_2 \cos\theta_i) & \cdots & \cos(2\pi d_N \cos\theta_i) \end{bmatrix} \begin{bmatrix} t \\ w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} \le 0, \tag{4.26}$$

or

$$\mathbf{a}_i^T \mathbf{X} \le 0. \tag{4.27}$$

Similarly, the inequality on the left in (4.25) becomes

$$-\begin{bmatrix} 1 & \cos(2\pi d_1 \cos\theta_i) & \cos(2\pi d_2 \cos\theta_i) & \cdots & \cos(2\pi d_N \cos\theta_i) \end{bmatrix} \begin{bmatrix} t \\ w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} \le 0, \tag{4.28}$$

or

$$\mathbf{f}_i^T \mathbf{X} \le 0 . \tag{4.29}$$

Using (4.22), (4.24), (4.27) and (4.29), the optimization problem of (4.20) can be rewritten as in (4.30).

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{X} \\ s.t. \quad & \mathbf{a}_0^T \mathbf{X} = 1 \\ & \mathbf{a}_i^T \mathbf{X} \le 0, \\ & \mathbf{f}_i^T \mathbf{X} \le 0, \ i = 1, 2, \ldots, R \end{aligned} \tag{4.30}$$

Equation (4.30) is in the same form as the standard LP in (4.15). Hence, solutions can be rapidly found to this problem using numerical computational software and guaranteed to be globally optimal.

As an example, consider the following 6-element symmetric linear array with positions

$$\mathbf{d}^T = [\pm 0.2\lambda \quad \pm 0.5\lambda \quad \pm 0.85\lambda] . \tag{4.31}$$

Finding weights that minimize the sidelobe level while directing the maximum to broadside cannot be found via the Dolph-Chebyshev method, because the array does not have uniform spacing. The beamwidth will be $40°$; hence, the region of sidelobe suppression will be

$$\Theta = \{0° \le \theta \le 70° \} \cup \{110° \le \theta \le 180°\} . \tag{4.32}$$

Using the linear programming method described in this section, the optimal weights can be found to be

$$\begin{aligned} w_1 &= 0.3724 \\ w_2 &= 0.1309 , \\ w_3 &= 0.4967 \end{aligned} \tag{4.33}$$

where w_1 is the weight associated with the first pair of positions in (4.31), while w_2 and w_3 are associated with the second and third pairs of positions, respectively. The resulting array factor is plotted in Figure 12. The dashed vertical lines in Figure 12 define the boundary of the main beam and specify the region in which the sidelobes are suppressed. The maximum sidelobe level outside the main beam is -11.21 dB. Note that the sidelobes are equal in magnitude, which is expected for sidelobe-minimizing weights.

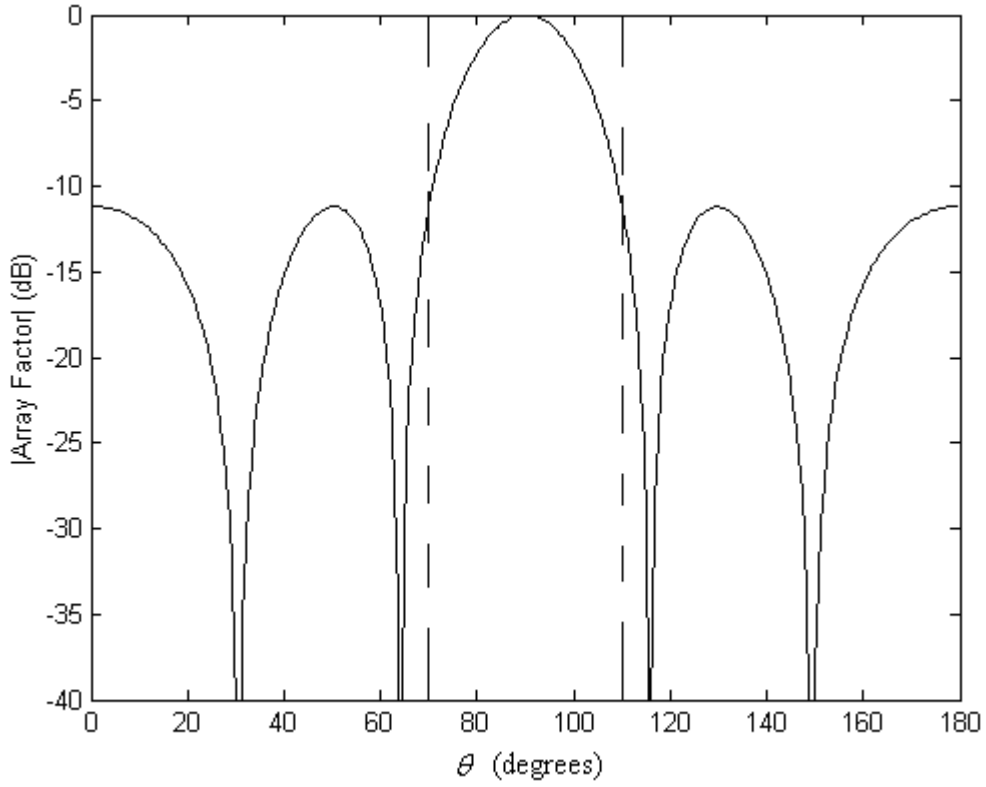


Figure 12. Array factor for optimal weights found via linear programming.

4.3. Convex Optimization

Convex optimization problems are a subclass of the general optimization problem given by (4.5). They have recently received a lot of attention in the engineering community because of their wide applicability. These applications include robotics [62],

signal processing [63], image processing [64] and information theory [65]. An excellent text for the engineering community on convex optimization has been written [66].

A convex optimization problem is defined by two fundamental characteristics: the feasible set χ must be convex and the objective function $f(\mathbf{X})$ is a convex function. A convex set is defined such that for every $\mathbf{x}_1$ and $\mathbf{x}_2$ in the set χ, then all points along a straight line between $\mathbf{x}_1$ and $\mathbf{x}_2$ are in χ. Mathematically, any point between $\mathbf{x}_1$ and $\mathbf{x}_2$ can be written as

$$\mathbf{z} = \alpha\mathbf{x}_1 + (1-\alpha)\mathbf{x}_2, \tag{4.34}$$

where α is a scalar between 0 and 1; for all α in this range $\mathbf{z}$ must be in the set. Examples of convex sets are shown in Figure 13. Convex sets are convenient to work with because search algorithms can always move between the current feasible point and the optimal point without running into the boundary of the set. Examples of non-convex sets are shown in Figure 14; each set contains points $\mathbf{x}_1$ and $\mathbf{x}_2$ such that not all points $\mathbf{z}$ between them, as in (4.34), are in the set.

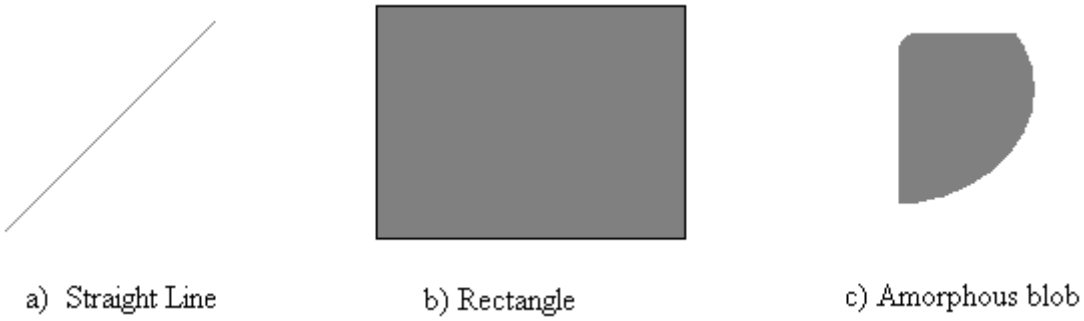


Figure 13. Examples of convex sets.

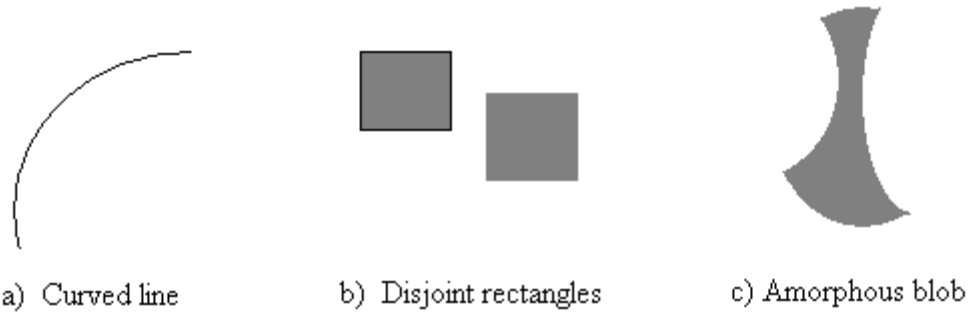


Figure 14. Examples of non-convex sets.

A function is said to be convex on a set χ if for any two points **X** and **Y** in χ, it satisfies the following inequality:

$$f(\alpha\mathbf{X} + (1-\alpha)\mathbf{Y}) \le \alpha f(\mathbf{X}) + (1-\alpha) f(\mathbf{Y}). \tag{4.35}$$

This means that the curve of the function *f* will always lie below a straight line connecting the two points *f*(**X**) and *f*(**Y**). This is illustrated for a one-dimensional function *f(t)* shown in Figure 15. The secant line between two points *x* and *y* is also drawn; note that *f(t)* lies below this line everywhere between *x* and *y*.

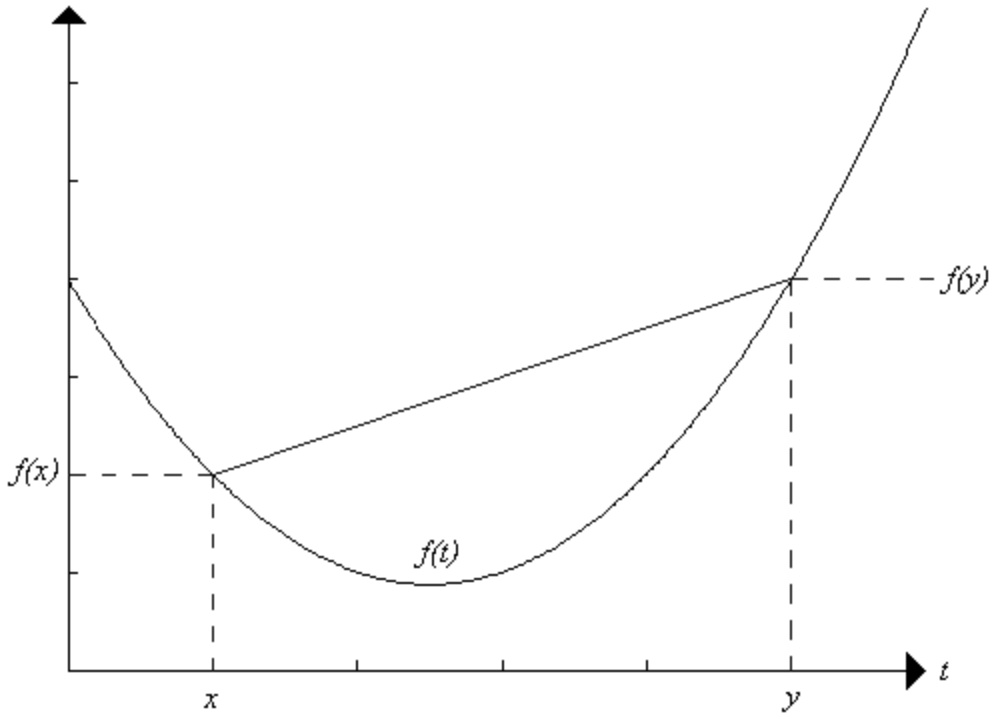


Figure 15. Illustration of a convex function.

A function is said to be strictly convex if the inequality ($\leq$) in (4.35) is replaced with a strict inequality (<). For convex functions, local minimums are always global minimums. For a strictly convex function, the global minimum is unique. This property makes convex functions convenient to work with in optimization.

As an example of proving a function is convex, consider *M* convex functions given by $f_1(\mathbf{X}),\ f_2(\mathbf{X}),\ldots,f_M(\mathbf{X})$. Define the function *F* to be the pointwise maximum of the set:

$$F(\mathbf{X}) = \max_i f_i(\mathbf{X}). \tag{4.36}$$

The goal is to show that *F* is also convex. To accomplish this, rewrite (4.36) as

$$F(\alpha\mathbf{X} + (1-\alpha)\mathbf{Y}) = \max_i f_i(\alpha\mathbf{X} + (1-\alpha)\mathbf{Y}). \tag{4.37}$$

Using the convexity of each function f_i, it follows that

$$\max_i f_i(\alpha\mathbf{X}+(1-\alpha)\mathbf{Y}) \le \max_i\left[\alpha f_i(\mathbf{X})+(1-\alpha)f_i(\mathbf{Y})\right]. \tag{4.38}$$

Since maximizing a sum of functions over one index must be less than maximizing each function individually,

$$\max_i\left[\alpha f_i(\mathbf{X})+(1-\alpha)f_i(\mathbf{Y})\right] \le \max_i \alpha f_i(\mathbf{X}) + \max_j (1-\alpha) f_j(\mathbf{Y}). \tag{4.39}$$

Equations (4.37)-(4.39) show that

$$F(\alpha\mathbf{X}+(1-\alpha)\mathbf{Y}) \le \alpha \max_i f_i(\mathbf{X}) + (1-\alpha)\max_j f_j(\mathbf{Y}), \tag{4.40}$$

which proves that *F* is convex. Hence, the pointwise maximum of convex functions is convex; this property will be used in Chapter 6.

Convex optimization problems are rapidly solvable with computers; the interior point methods developed for linear programs have been efficiently extended to convex problems [67]. Since these problems have a very general structure, they can be applied to a wide range of practical problems. In addition, since the optimal points found can be mathematically proven to be globally optimal, putting a problem into convex form is very desirable. Free convex optimization packages have been written for use with Matlab; examples packages include CVX and YALMIP and are available online. A convex optimization problem will be derived and solved in Chapter 6 which greatly extends the minimum sidelobe weighting vector of Section 4.2.

4.4. Simulated Annealing

The discussion now turns to stochastic optimization algorithms. These algorithms work on the most general type of optimization problems; however, they tend to use

random searches and the results are not guaranteed to be globally optimal. However, they have recently been employed extensively in the engineering community.

The simulated annealing (SA) algorithm attempts to mimic the physical process of annealing of solids. This process involves heating a solid material up to a high temperature and then allowing it to cool at a very slow rate. The result is that the particles in the solid arrange themselves in the lowest energy state configuration, usually an ordered lattice of some sort [68]. The SA algorithm attempts to optimize via the same procedure. The algorithm was originally introduced in 1983 by Kirkpatrick in the journal *Science* as a generalization of the Monte Carlo method for examining the equations of state of n-body systems [69].

The SA algorithm requires a cost function f [also known as the objective function in (4.5)], an initial feasible point ($\mathbf{x}_1$), and a perturbation mechanism for obtaining new points around the current point. The algorithm evaluates the cost function at the start point, perturbs the point to a new point, evaluates the cost function at this point and repeats. The following discussion describes finding a minimum.

From the current point $\mathbf{x}_i$, a candidate new point ($\hat{\mathbf{x}}_{i+1}$) in the feasible set is chosen using the perturbation mechanism. If the new point decreases the cost function, then

$$\Delta f_i = f(\hat{\mathbf{x}}_{i+1}) - f(\mathbf{x}_i) < 0 \tag{4.41}$$

and the current solution is updated according to

$$\mathbf{x}_{i+1} = \hat{\mathbf{x}}_{i+1}. \tag{4.42}$$

The algorithm does not want to only accept points that decrease the cost function; this would cause the algorithm to find a local minimum about the initial point. If the next

candidate point increases the objective function, then the probability that the algorithm updates the current solution to the candidate solution is given by

$$P\{\mathbf{x}_{i+1} = \widehat{\mathbf{x}}_{i+1}\} = \exp\left(\frac{-\Delta f_i}{T}\right), \tag{4.43}$$

where T represents the current "temperature" of the system. If T is very large, then almost all transitions occur and the result is a random walk through the space of points independent of the cost function. When T becomes small, only transitions that increase the value of the cost function occur; the result is that the algorithm converges to the local minimum of the neighborhood of points that the current point resides in. If the algorithm does not accept the next candidate point, then it simply remains at the previous point,

$$\mathbf{x}_{i+1} = \mathbf{x}_i . \tag{4.44}$$

The simulated annealing algorithm starts the optimization procedure at a high temperature (sufficiently high such that most transitions occur), and lowers the temperature slowly enough so that a satisfactory solution is found. There are many methods of choosing this 'cooling schedule'; a collection of these is described in [70] and a specific method is utilized in Chapter 5. The algorithm is stopped once transitions to new candidate points do not occur over a large number of attempts; the algorithm is then said to have converged.

The SA algorithm, while being relatively simple to implement, requires a good deal of care in choosing the initial temperature and an appropriate cooling schedule so that a globally optimum point is likely to be found. Increased confidence that the proposed solutions are globally optimum can be obtained by running the algorithm multiple times from various initial points.

4.5. Particle Swarm Optimization (PSO)

The PSO algorithm is a stochastic, evolutionary algorithm capable of effectively optimizing difficult multidimensional optimization problems. Examples of the successful application of the PSO algorithm in the electromagnetics community include antenna design [71] and array geometry selection [72]. Originally introduced by Kennedy and Eberhart in 1995 [73], the PSO algorithm has been gaining popularity over the genetic algorithm and other evolutionary algorithms because of its simplicity in implementation and efficient optimization. In addition, the algorithm lends itself well to parallel processing, which is an added bonus.

The PSO algorithm attempts to mimic the behavior of birds or bees in obtaining a food source. Initially, a flock of birds may start out in random directions searching for food. As each individual bird travels on its path, it may find food in various locations. The bird remembers its own 'personal best' location of where it had found food. In addition, the bird may periodically fly up and survey the progress of the other birds in the flock. In this manner, each individual bird will be aware of the 'global best' position, or location found with the most food by any bird in the flock. Using this general procedure, a flock of birds will descend on the region in the area that has a relatively high amount of food available.

The PSO algorithm translates this behavior into a mathematical algorithm for optimization. The PSO algorithm consists of a set of particles (the 'swarm'), which are analogous to the birds. The algorithm also has a cost or fitness function, which evaluates

the current position of each bird; this is analogous to a bird evaluating how much food is in a certain location.

It will be assumed in the following discussion that there is a feasible set χ to be optimized over in which each element can be represented as an *N*-dimensional real vector, and a fitness function $f : \chi \to \Re$ which can evaluate each position to a real number.

The algorithm starts with *M* particles selected at random positions within the feasible set. The number *M* depends on the dimension and difficulty of the problem, and is one of the parameters left to the algorithm implementer. The algorithm is iterative and the locations will change at each time step. The i^{th} particle at time *t* will be at the location given by $\mathbf{x}_i^t$, where *i* is an integer between 1 and *M*, and *t* is an integer specifying the current time step. Each particle will also have a randomly selected initial velocity vector. The i^{th} particle at time *t* will have a velocity written as $\mathbf{v}_i^t$.

In addition, each particle will record the location of its 'personal best position'. This is the location that the current particle has found to be the fittest (minimum) so far along its trajectory. The personal best positions will be written for the i^{th} particle as $\mathbf{p}_i$, and the corresponding fitness value for each of these positions will be written as P_i. Each particle will also be aware of the 'global best position', which is the position that has been found to be the fittest so far from among all the particles and will be written as the vector **g.** The global best value will also be recorded, and it will be written as

$$G = \min_i P_i. \tag{4.45}$$

Once the random initial positions and velocities have been chosen for each particle, the fitness value for each of the positions is evaluated, giving the personal and global best positions and values. The algorithm then updates the velocity and position of each particle at every time step until the simulation is stopped.

To perform the updates, first define matrices $\mathbf{U}_{1i}^{t}$ and $\mathbf{U}_{2i}^{t}$ according to

$$\mathbf{U}_{ai}^{t} = \begin{bmatrix} u_{a1}^{t} & 0 & 0 & \cdots & 0 \\ 0 & u_{a2}^{t} & 0 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & \cdots & u_{aN}^{t} \end{bmatrix}, \tag{4.46}$$

where a is equal to 1 or 2, and each u_{ai}^{t} is an independent uniformly distributed real variable on [0,1]. The velocity is then updated at each time step according to

$$\mathbf{v}_i^t = w_V \mathbf{v}_i^{t-1} + c_1 \mathbf{U}_{1i}^{t}(\mathbf{p}_i - \mathbf{x}_i^{t-1}) + c_2 \mathbf{U}_{2i}^{t}(\mathbf{g} - \mathbf{x}_i^{t-1}), \tag{4.47}$$

where w_V is a real number called the 'inertial weight', c_1 is a real number that accelerates the particle towards its personal best position, and c_2 is a real number that accelerates the particle towards its global best position. In this manner, each particle moves in a random fashion around the solution space, but is stochastically drawn towards the particle's previous best location and the swarm's global best location. The inertial weight w is a real number in the range [0,1] that controls how much the updated velocity depends on the previous velocity. Studies on PSO have shown that 2.0 is a good choice for both parameters c_1 and c_2 [74].

The position is then updated according to

$$\mathbf{x}_i^t = \mathbf{x}_i^{t-1} + \mathbf{v}_i^{t-1}, \tag{4.48}$$

and the fitness function is evaluated at each of the new locations. Finally, the personal best and global best positions and values are updated if possible, and then the process repeats again.

This algorithm is relatively simple to implement but performs well on general optimization problems in comparison to other evolutionary algorithms. The biggest drawback to the method is that the resulting solutions cannot be verified to be globally optimal.

V. ARRAY GEOMETRY OPTIMIZATION FOR INTERFERENCE SUPPRESSION

5.1. Introduction

In this chapter, the influence of array geometry on the performance of adaptive antenna arrays is examined by solving a specific wireless communication problem. The problem of interference is addressed, which can occur intentionally (as in jamming) or unintentionally (as in wireless devices sharing a frequency band).

The steady-state weights many adaptive algorithms (for instance, LMS and RLS) converge to are the MMSE weights in (3.31). The optimal MSE is rewritten from (3.32):

$$\mathrm{MSE}_{opt} = \sigma_s^2 - \sigma_s^4 \mathbf{v}^H(\mathbf{k}_s)\mathbf{R}_{\mathbf{XX}}^{-1}\mathbf{v}(\mathbf{k}_s), \tag{5.1}$$

which is a fair measure of the performance of an adaptive array. In (5.1) the weights are absent because the optimal weights have already been substituted into the expression. The signal power, σ_S^2, and the signal direction, given by $\mathbf{k}_s$, are part of the wireless environment and cannot be changed. The terms that remain in (5.1) are the steering vector $\mathbf{v}(\mathbf{k}_s)$ and the autocorrelation matrix $\mathbf{R}_{\mathbf{XX}}$. The steering vector can be rewritten from (2.31) as

$$\mathbf{v}(\mathbf{k}_s) = \begin{bmatrix} e^{-j\mathbf{k}_s\cdot\mathbf{d}_1} \\ e^{-j\mathbf{k}_s\cdot\mathbf{d}_2} \\ \vdots \\ e^{-j\mathbf{k}_s\cdot\mathbf{d}_N} \end{bmatrix}, \tag{5.2}$$

which is a nonlinear function of the array element positions. The autocorrelation matrix, defined in (3.21), is a function of the inputs (**X**) to the antenna array. The input is a summation of noise, the desired signal and the interference from various directions. The input to the array then depends on the positions of the array, the noise power, the signal

power and the powers of the interferers. Since an antenna array cannot control the power incident upon it, the autocorrelation matrix can only be altered by changing the array geometry, **D**. Hence, the MSE in (5.1) is actually only a complicated function of the geometry of the array. Naturally, the question of determining an optimum geometry for an adaptive array arises, which is the subject of this chapter.

5.2. Interference Environment

Military communication systems will potentially be used in environments with a large amount of co-channel interference from sources intending to impede communication. In this situation, an antenna array is suitable for blocking interference spatially separated from the desired signal direction.

The arrays are not operating in a unique situation in which the interference is from known directions. As a result, it would not be prudent to optimize the array geometry for a specific interference situation (example: 3 interferers from 3 distinct angles). Instead, the concept of an interference environment will be introduced as a statistical characterization of the expected directions and relative power of the interference. For example, a cell phone tower would expect the interference to be confined to a fixed range of elevation angles (directed towards the ground) and would not be concerned with blocking interference from the sky. Other arrays used in more dynamic environments may expect interference from all directions with equal probability. By optimizing an array geometry with respect to an interference environment, it is possible to minimize the expected (or average) interference power that is not rejected by the array. A specific

example of an interference environment will be detailed in Section 5.4. The task is now to derive an optimization problem whose solution yields an optimal array geometry.

Recall the definition of the autocorrelation matrix:

$$\mathbf{R}_{\mathbf{XX}} = E[\mathbf{X}(t)\mathbf{X}^{H}(t)], \tag{5.3}$$

where the expectation is over time. Each unique interference situation in which the array operates will have a unique autocorrelation matrix. Going one step further, the expected autocorrelation matrix, $\mathbf{S}_{\mathbf{X}}$, is now defined as

$$\mathbf{S}_{\mathbf{X}} = E_{I}[\mathbf{R}_{\mathbf{XX}}], \tag{5.4}$$

where the expectation operator is now over the interference situations (which defines the interference environment). A noteworthy observation is that if all the antenna elements have the same physical orientation, then $\mathbf{S}_{\mathbf{X}}$ can alternatively be found by treating the elements as isotropic sensors while simply adjusting the power levels in the interference environment. As a simple example, suppose an array was operating in an environment in which interference occurred from one of two distinct angles of arrival with equal probability. Each situation would have an autocorrelation matrix associated with it, which is written as $\mathbf{R}_{\mathbf{XX1}}$ and $\mathbf{R}_{\mathbf{XX2}}$. Then the expected autocorrelation matrix is

$$\mathbf{S}_{\mathbf{X}} = 0.5(\mathbf{R}_{\mathbf{XX1}} + \mathbf{R}_{\mathbf{XX2}}). \tag{5.5}$$

5.3. Optimization for Interference Suppression

If it is assumed that the interference has a larger power than the signal of interest or that there are many interferers, then the array's primary goal is to minimize the output power while restricting one of the weights in the array to be unity. This is similar to a sidelobe cancellation system [47] and is also the method used in a 7-element adaptive

array developed by Raytheon for combating interference in GPS systems [75]. By using the power minimization technique, the array can greatly reduce the amount of interference power that makes it into the next stage of processing (usually a temporal filter). Note that power minimization does not attempt to place the maximum of the array factor towards the signal of interest. This is a suboptimal technique in regards to the MSE; however, when the interference power is much stronger than the power of the desired signal, this technique produces weights close to those produced using the MMSE weights. The advantage of this technique is its simplicity, as it does not require estimating the direction of arrival of the signal of interest or its power.

The output power from the array at any time is

$$y^* y = \mathbf{w}^H \mathbf{X}\mathbf{X}^H \mathbf{w} . \tag{5.6}$$

For a fixed interference situation, the average output power $\overline{P}$ is then

$$\overline{P} = \mathbf{w}^H \mathbf{R}_{\mathbf{XX}} w . \tag{5.7}$$

A measure of the average output power for a given interference environment P is then

$$P = \mathbf{w}^H \mathbf{S}_{\mathbf{XX}} \mathbf{w} . \tag{5.8}$$

One of the weights is restricted to be unity so that the power minimization algorithm does not set all of the weights to zero. In addition, for practical reasons such as minimizing the effects of mutual coupling, it will be required that the separation between elements be at least $\lambda / 4$. Let $\mathbf{r}_{ij}$ be the separation between elements i and j. The problem of finding an optimal array for interference suppression can be written as in optimization problem,

given in (5.9). Note that $\mathbf{e}_1^T = [1\ 0\ 0\ \cdots 0]$. The minimization variables are the complex weights and the values of $\mathbf{r}_{ij}$.

$$\begin{aligned} \min \quad & \mathbf{w}^H \mathbf{S}_\mathbf{X} \mathbf{w} \\ s.t. \quad & \mathbf{w}^H \mathbf{e}_1 = 1 \\ & \mathbf{r}_{ij} \geq \lambda / 4, \quad \text{for } i \neq j \end{aligned} \tag{5.9}$$

Assuming the locations of the antenna elements are known (or held fixed), the optimal weight vector for this problem can be found by using Lagrange multipliers. The Lagrangian can be expressed as

$$L(\mathbf{w}, \Lambda) = \mathbf{w}^H \mathbf{S}_\mathbf{X} \mathbf{w} + \Lambda(\mathbf{w}^H \mathbf{e}_1 - 1). \tag{5.10}$$

Taking the gradient with respect to **w** of the Lagrangian and setting the result to zero, (5.10) becomes

$$\nabla L = 2\mathbf{S}_\mathbf{X} \mathbf{w}_{opt} + \Lambda \mathbf{e}_1 = 0, \tag{5.11}$$

where $\mathbf{w}_{opt}$ are the power-minimizing weights. Assuming that $\mathbf{S}_\mathbf{X}$ is invertible, the weights can be solved from (5.11) as

$$\mathbf{w}_{opt} = \frac{-\Lambda \mathbf{S}_\mathbf{X}^{-1} \mathbf{e}_1}{2}. \tag{5.12}$$

The parameter Λ can be determined by invoking the equality constraint of (5.9)

$$\mathbf{w}_{opt}^H \mathbf{e}_1 = -\frac{\Lambda}{2} \mathbf{e}_1^T \mathbf{S}_\mathbf{X}^{-1} \mathbf{e}_1 = 1, \tag{5.13}$$

and the property $(\mathbf{S}_\mathbf{X}^{-1})^* = \mathbf{S}_\mathbf{X}^{-1}$ was used, which follows from the definition of an autocorrelation matrix. The solution to (5.13) can be substituted into (5.12) yielding the power-minimizing weights,

$$\mathbf{w}_{opt} = \frac{\mathbf{S}_\mathbf{X}^{-1}\mathbf{e}_1}{\mathbf{e}_1^T\mathbf{S}_\mathbf{X}^{-1}\mathbf{e}_1}. \tag{5.14}$$

Substituting (5.14) into the objective function of (5.9), the minimum value of the objective function for a fixed geometry becomes

$$\min_{\mathbf{w}}\{\mathbf{w}^H\mathbf{S}_\mathbf{X}\mathbf{w}\} = \frac{1}{\mathbf{e}_1^T\mathbf{S}_\mathbf{X}^{-1}\mathbf{e}_1}. \tag{5.15}$$

The goal is now to minimize (5.15) over all array geometries that meet the constraints in (5.9). Minimization of (5.15) is equivalent to maximizing the reciprocal; this is true whenever a function is strictly nonnegative. Equation (5.15) is always nonnegative because an autocorrelation matrix is always positive semi-definite [76], positive semi-definite matrices always have nonnegative quadratic forms [77], and if a matrix is positive semi-definite then its inverse will be as well [77]. Hence, the minimization problem in (5.10) can be rewritten as a maximization problem as in (5.16). The notation $[\mathbf{Z}]_{mn}$ will be used to represent the element of the matrix $\mathbf{Z}$ from the m^{th} row and n^{th} column, so that the optimization problem can be written as

$$\begin{aligned} \max \quad & \mathbf{e}_1^T\mathbf{S}_\mathbf{X}^{-1}\mathbf{e}_1 = [\mathbf{S}_\mathbf{X}^{-1}]_{11} \\ s.t. \quad & \mathbf{r}_{ij} \geq \lambda/4, \ \text{for } i \neq j \end{aligned}. \tag{5.16}$$

The objective function in (5.16) is only a function of the antenna locations and the interference environment. Since the interference environment cannot be controlled, the performance of the array using the optimal weights of (5.14) is only a function of the antenna locations. The optimal element locations are those that maximize the objective function of (5.16) subject to the specified constraints. The solution to (5.16) will not be unique, because $\mathbf{S}_\mathbf{X}$ is invariant to translation (shifting the elements uniformly). The

optimization problem in (5.16) is what needs to be solved in order to determine an optimum array geometry for a given interference environment.

5.4. Planar Array with Uniform Interference at Constant Elevation

As an example of the solution to (5.16), a planar array of *N* elements with the same physical orientation is considered. This example assumes *M* interferers and takes the interference to be mutually independent and arriving from a uniform distribution in the azimuth direction ($\phi \in [0,2\pi]$ measured counterclockwise from the x-axis), but from a fixed elevation angle ($\theta \in [0,\pi]$ measured down from the z-axis towards the plane of the array). The elements will be located at positions in the x-y plane given by

$$\mathbf{r}_i = (x_i, y_i, 0), \quad i = 1, 2, \ldots, N . \tag{5.17}$$

With *M* interferers, the input to the array becomes

$$\mathbf{X}(t) = \sum_{n=1}^{M} f(\theta,\phi) s_n(t) \mathbf{v}_n(\mathbf{k}_n), \tag{5.18}$$

where $f(\theta,\phi)$ is the element pattern for each antenna, while $s_n(t)$ and $\mathbf{v}_n(\mathbf{k}_n)$ are the signal and the steering vector for the n^{th} interferer, respectively. For simplicity, it will be assumed that the antenna elements do not have a pattern that varies much in the azimuth direction. This assumption allows the element factor to be eliminated because the interferers are assumed to come from a fixed elevation angle; hence the response of the antenna can be lumped into the received power. Finally, the steering vectors will be rewritten as a function of the azimuth angle (ϕ_n) only, so that (5.18) simplifies to

$$\mathbf{X}(t) = \sum_{n=1}^{M} s_n(t) \mathbf{v}_n(\phi_n) . \tag{5.19}$$

The autocorrelation matrix becomes

$$\mathbf{R}_{\mathbf{XX}} = E[\mathbf{X}(t)\mathbf{X}^H(t)]$$

$$= E\left[\left(\sum_{n=1}^{M} s_n(t)\mathbf{v}_n(\phi_n)\right)\left(\sum_{n=1}^{M} s_n(t)\mathbf{v}_n(\phi_n)\right)^H\right]. \tag{5.20}$$

Assuming the interference to be independent, it follows that

$$E[(s_n(t)\mathbf{v}_n(\phi_n))(s_m(t)\mathbf{v}_m(\phi_m))^H] = 0 \tag{5.21}$$

because $E[s_n(t)s_m^*(t)] = 0$ for $m \neq n$. For $m=n$, it follows that the components of the autocorrelation matrix become

$$E[(s_n(t)\mathbf{v}_n(\phi_n))(s_n(t)\mathbf{v}_n(\phi_n))^H]_{ab}$$

$$= E\left[s_n(t)s_n^*(t)e^{-j(k_x x_a + k_y y_a)}e^{j(k_x x_b + k_y y_b)}\right]$$

$$= \sigma_n^2 e^{-j\frac{2\pi}{\lambda}\sin\theta(\cos\phi_n(x_a - x_b) + \sin\phi_n(y_a - y_b))}. \tag{5.22}$$

Equation (5.20) along with (5.22) gives the components of the autocorrelation matrix,

$$[\mathbf{R}_{\mathbf{XX}}]_{ab} = \sum_{n=1}^{M} \sigma_n^2 e^{-j\frac{2\pi}{\lambda}\sin\theta(\cos\phi_n(x_a - x_b) + \sin\phi_n(y_a - y_b))}. \tag{5.23}$$

The expected autocorrelation matrix can now be calculated from (5.23) and the fact that the interference is uniformly distributed in the azimuth direction. The expected value of (5.23) is taken, resulting in

$$E[\mathbf{R}_{\mathbf{xx}}]_{ab} = \int_0^{2\pi} \sum_{n=1}^{M} \sigma_n^2 e^{-j\frac{2\pi}{\lambda}\sin\theta(\cos\phi_n(x_a - x_b)+\sin\phi_n(y_a - y_b))} \frac{d\phi_n}{2\pi}$$

$$= \sum_{n=1}^{M} \sigma_n^2 \int_0^{2\pi} e^{-j\frac{2\pi}{\lambda}\sin\theta(\cos\phi_n(x_a - x_b)+\sin\phi_n(y_a - y_b))} \frac{d\phi_n}{2\pi}. \quad (5.24)$$

In order to evaluate the integral in (5.24), the following variable substitutions are made:

$$(x_a - x_b) = R_{ab}\cos\phi_{ab} \quad (5.25)$$

$$(y_a - y_b) = R_{ab}\sin\phi_{ab}. \quad (5.26)$$

In (5.25) and (5.26), R_{ab} is the distance between elements *a* and *b*. Using the trigonometric identity

$$\cos(u - v) = \cos(u)\cos(v) + \sin(u)\sin(v), \quad (5.27)$$

and substituting (5.25)-(5.27) into (5.24), the integral in (5.24) becomes

$$\int_0^{2\pi} e^{-j\frac{2\pi}{\lambda}R_{ab}\sin\theta\cos(\phi_n - \phi_{ab})} \frac{d\phi_n}{2\pi}. \quad (5.28)$$

Since the integral in (5.28) is over a complete cycle for ϕ_n, the term ϕ_{ab} will not contribute to the integral and can be arbitrarily set to zero without influencing the result. The Bessel function of the first kind of order *n* can be written in integral form as [78]

$$J_n(x) = \frac{j^{-n}}{2\pi}\int_0^{2\pi} e^{j(x\cos\phi + n\phi)} d\phi. \quad (5.29)$$

Hence, (5.24) can be rewritten using (5.28) and (5.29), as

$$[\mathbf{S}_{\mathbf{X}}]_{ab} = E[\mathbf{R}_{\mathbf{XX}}]_{ab} = \left(\sum_{n=1}^{M} \sigma_n^2 \right) J_0\left(\frac{2\pi}{\lambda} R_{ab} \sin\theta \right). \tag{5.30}$$

Equation (5.30) shows that the expected autocorrelation matrix depends on the total interference power incident on the array, and not on the number of interferers. Equation (5.30) along with (5.16) defines the optimization problem used to determine an array for suppressing interference. A method of determining an optimal array is the subject of the following section.

5.5. Using Simulated Annealing to Find an Optimal Array

The Simulated Annealing optimization algorithm described in Chapter 4 was found to be suitable for the problem at hand. For simplicity, an elevation angle of $\theta = 90°$ is chosen for the interferers. The candidate arrays (or points in the feasible space, as discussed in Section 4.4) at every time step are represented by a real vector in $\Re^{2N}$, which represents the x- and y- positions of the *N*-elements.

In using the SA algorithm, a circular array is chosen as the initial array. Ideally, the initial array chosen will have no effect on the optimization result. The perturbation mechanism is implemented by choosing a random vector in $\Re^{2N}$ that has an Euclidean norm that is zero-mean and with a small variance. The variance is chosen such that the average perturbation for each element is on the order of 0.01λ; a large variance will lead to an imprecise search of the solution space, while a small variance will lead to a long simulation time. The 2*N* components of this vector are added to the x-y coordinates of the current array. If the perturbation moves the elements too close to each other ($< 0.25\lambda$), then the perturbation is discarded and a new perturbation selected. Another constraint

is imposed such that all elements stay within 0.75λ of the origin (center of the initial circular array) to keep the search space finite. This number was chosen to be large enough such that the resulting optimal arrays were not altered by this constraint.

The initial temperature T_0 was chosen such that virtually all (>99%) of perturbations are accepted. The temperature is held constant for a fixed number (P) of perturbations. The temperature is then multiplied by a factor u<1. The solution array is then again perturbed P times. This process is performed until T is small enough that no perturbations that decrease the objective function are accepted (recall the optimization problem is one of maximization); once this happens the solutions has converged upon a local maximum. If P is sufficiently large and the temperature decreased sufficiently slowly, this method will converge to the global optimum [68]. In the solution for N=6, the parameters used were u=0.99, P=50 000, and T_0=12. The method of determining these numbers was to use small values of u, P and T_0, and increase them until the simulations consistently returned the same solution starting from various initial arrays. As u, P, and T_0 are increased, the probability of a correct (globally optimal) solution increases; if they are decreased, the solution is less likely to be optimal. However, the tradeoff lies in the computational time needed. The simulation for the 6-element array described below was performed using MATLAB on a computer with a 2.9 GHz processor, and the solution time was approximately 8 hours.

The optimum array configurations for the N=4, 5 and 6 element arrays are found using the above optimization procedure and plotted in Figures 16, 17 and 18, respectively. The dotted circles in these figures are of radius 0.25λ. The results suggest

that the interference suppression capabilities are best for arrays spaced as closely as possible. They all have a center element and are surrounded by a circular array of radius 0.25λ (the minimum distance allowed). This suggests a trade-off between interference suppression and largely spaced arrays used for diversity or to minimize mutual coupling.

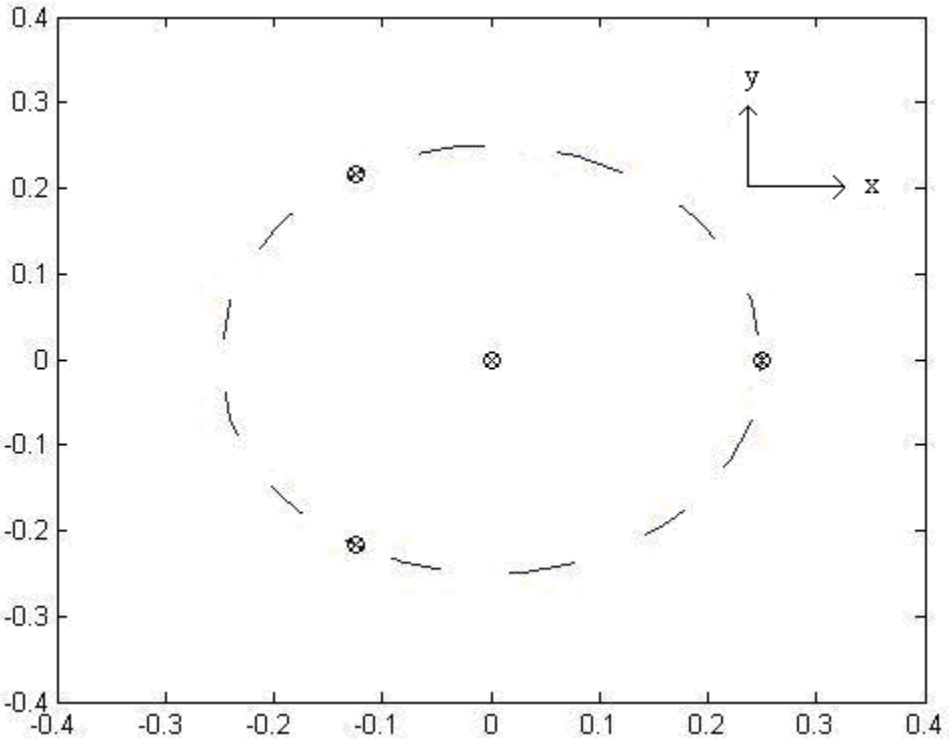


Figure 16. Optimum N=4 element array (measured in units of λ).

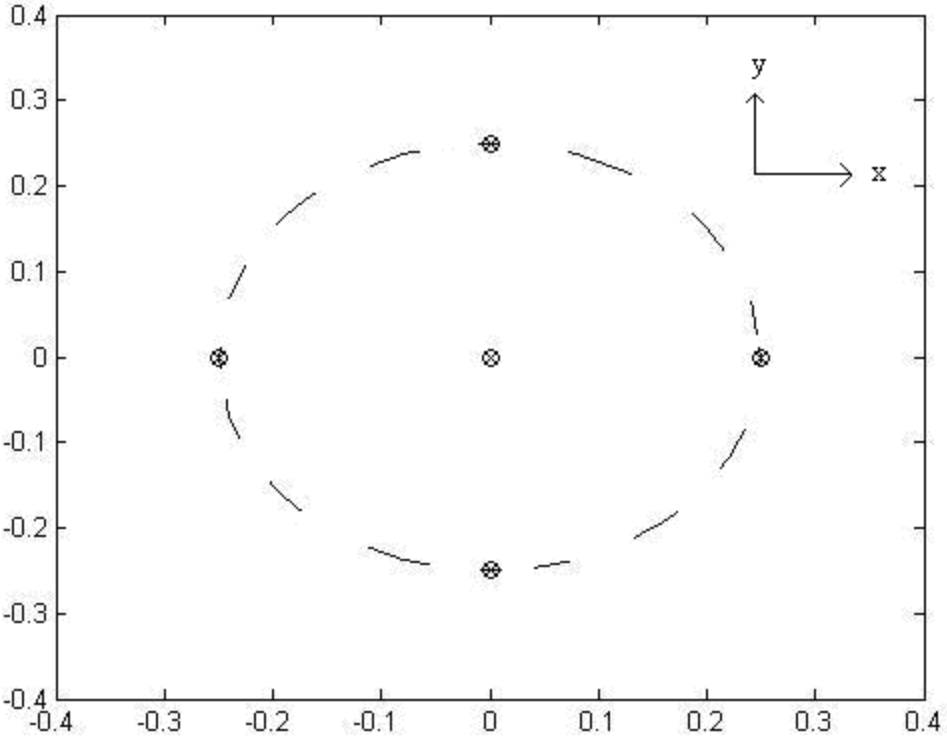


Figure 17. Optimum N=5 element array (measured in units of λ).

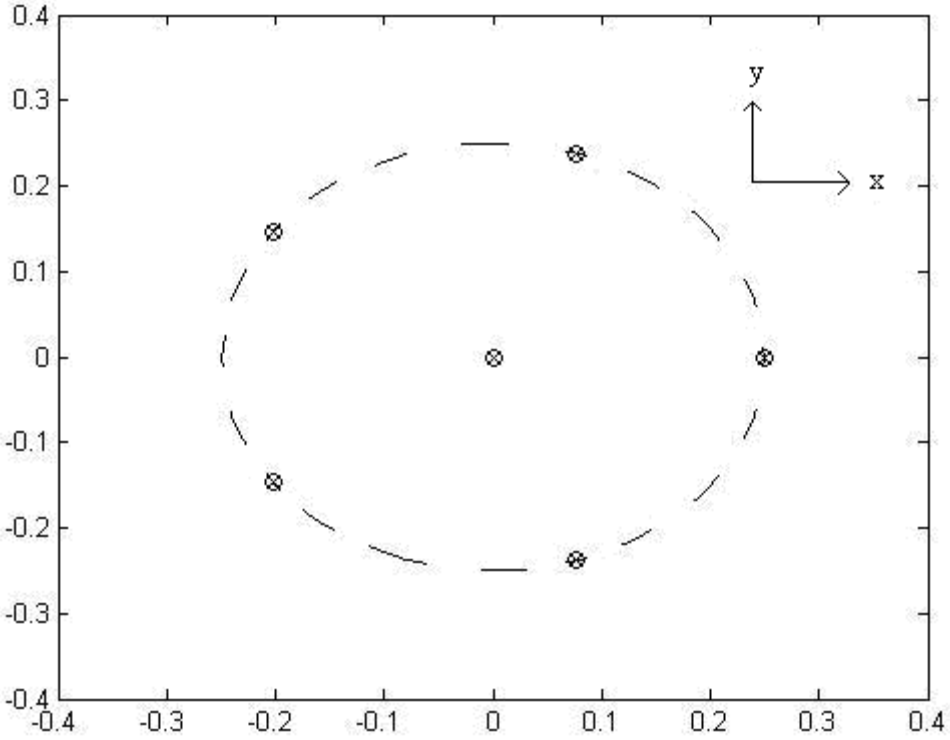


Figure 18. Optimum N=6 element array (measured in units of λ).

5.6. Evaluating the Performance of the Optimal Arrays

In order to illustrate the performance of the optimum array, it will be compared to three other standard arrays: a circular array with radius chosen such that the spacing along the circle between elements is 0.5λ as suggested in [79], a linear array with interelement spacing 0.5λ oriented along the z-axis, and a rectangular array with interelement spacing 0.5λ.

Interferers from six different angles are chosen, each randomly selected from a uniform distribution (on $[0, 360°]$) and all at the same elevation angle ($90°$). The output power ($\mathbf{w}^H \mathbf{R}_{\mathbf{xx}} \mathbf{w}$) is calculated when the weights are given by the optimal weight vector for this specific instance, given in (5.31). This is the steady-state solution the adaptive

power-minimization algorithm would converge to in practice, if the first weight is fixed at unity.

$$\mathbf{w}_{opt} = \frac{\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{e}_1}{\mathbf{e}_1^T\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{e}_1} \tag{5.31}$$

This process is repeated 100,000 times to form an average output power for this type of interference environment. The results are listed in Table I, where the average output power is given relative to the power allowed by the optimal array.

TABLE I
OUTPUT POWER COMPARISON AMONG DIFFERENT ARRAYS

Array	Relative Power (N=4)	Relative Power (N=5)	Relative Power (N=6)
Optimal	0 dB	0 dB	0 dB
Circular	11.5 dB	16.9 dB	32.2 dB
Rectangular	7.1 dB	20.5 dB	32.4 dB
Linear	12.2 dB	24.4 dB	37.8 dB

Table I illustrates the dramatic effect that array geometry can have on the interference-suppression capabilities of the array. The optimum arrays performed significantly better on average than the standard arrays used in practice, much more than reasonably expected. The output powers for the standard arrays are much higher than those for the optimal arrays, clearly showing their superior interference-suppression capabilities.

After viewing Figures 16-18, one may easily conjecture what the optimal 7-element array would be. The optimization procedure is applied and confirms the solution to be that as given in Figure 19. The interesting thing about the optimal 7-element array is that it is a hexagonally sampled planar array. In multidimensional digital signal processing, it is well known that the optimal sampling strategy to avoid aliasing for circularly

bandlimited signals is a hexagonally sampled lattice [80]. Hence, while sampling for reconstruction and sampling for interference suppression are fundamentally different, the problems are strongly related and the optimal solutions come out the same (for the case of a circular interference environment). This parallel strengthens the methods and procedures applied in determining optimal arrays.

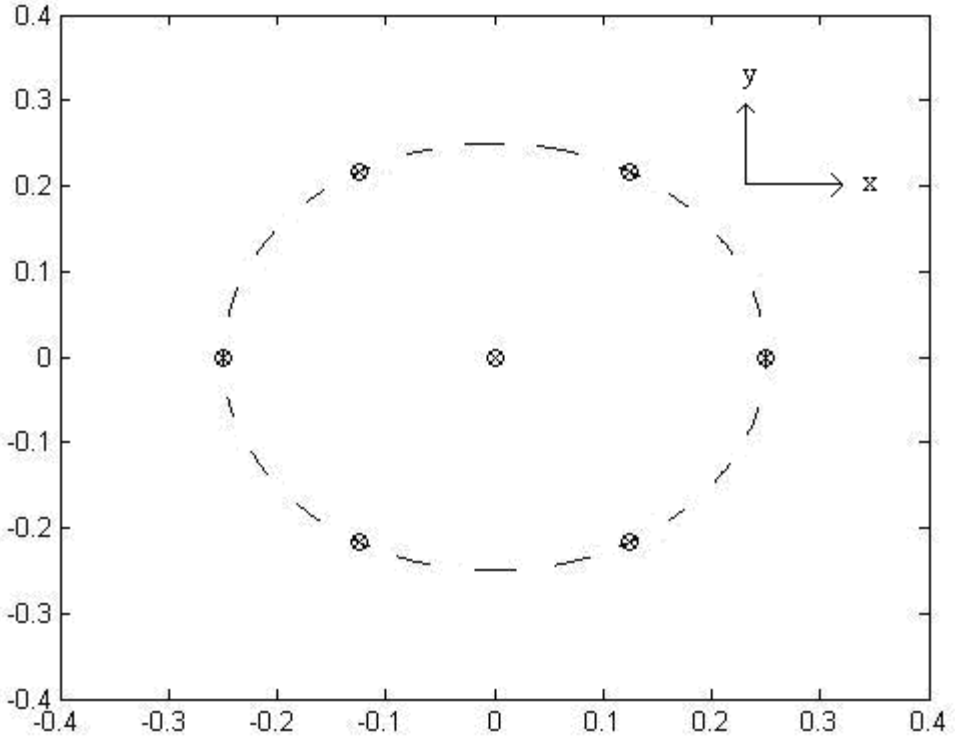


Figure 19. Optimum N=7 element array (measured in units of λ).

The array geometry in Figure 19 is the layout of the 7-element GAS-1 (GPS Antenna System) array developed by Raytheon whose primary function is to suppress interferers or jamming [75]. The method of this paper confirms that the GAS-1 geometry used is optimum for the case of a planar array in a circular interference environment. The elements of the GAS-1 array are circular patches each operating at the dual frequencies of L1 (1.575 GHz) and L2 (1.227 GHz).

The method derived above seeks to minimize output power. While this has its advantages, the primary disadvantage is that the desired signal may be muted along with the interferers. To get an idea of the signal to interference ratio (SIR) at the output of the array, a few test cases are considered. In Case 1, the desired signal arrives from $\theta_d = 45°$ and $\phi_d = 0°$. Twelve interferers are selected from a fixed elevation angle ($\theta_I = 90°$) and a random azimuth angle and 30 dB Interference-to-Signal Ratio (ISR). The weight vector used for each case is the vector that minimizes the MSE, given in (3-30). The process will be repeated (random interference directions selected) 100,000 times to form an expected SIR, given by

$$\overline{\text{SIR}} = \frac{\sum_n S_n}{\sum_n I_n}, \tag{5.32}$$

where S_n and I_n are the output signal power and the output interference power for the n^{th} situation, respectively. The resulting SIRs for N=5, 6, and 7 elements are determined for the optimal array along with a circular, linear, and rectangular array as before. The results for Case 1 are given in Table II. Case 2 will be the same as Case 1 except the signal arrives from $\theta_d = 0°$. The results for Case 2 are presented in Table III. Case 3 will be the same as Case 1 except the signal arrives from $\theta_d = 90°$. The results for Case 3 are presented in Table IV.

TABLE II
RELATIVE SIR FOR CASE 1

Array	Relative SIR (N=5)	Relative SIR (N=6)	Relative SIR (N=7)
Optimal	0 dB	0 dB	0 dB
Circular	-6.45 dB	-11.4 dB	-26.9 dB
Rectangular	-6.41 dB	-15.85 dB	-25.13 dB
Linear	2.98 dB	-5.82 dB	-19.15 dB

TABLE III
RELATIVE SIR FOR CASE 2

Array	Relative SIR (N=5)	Relative SIR (N=6)	Relative SIR (N=7)
Optimal	0 dB	0 Db	0 dB
Circular	-3.9 dB	-10.15 Db	-35.9 dB
Rectangular	-5.8 dB	-19.25 dB	-27.2 dB
Linear	5 dB	-10.24 dB	-21 dB

TABLE IV
RELATIVE SIR FOR CASE 3

Array	Relative SIR (N=5)	Relative SIR (N=6)	Relative SIR (N=7)
Optimal	0 dB	0 dB	0 dB
Circular	-11.3 dB	2.5 dB	-4.3 dB
Rectangular	-11.6 dB	-0.45 dB	-6.64 dB
Linear	-15.1 dB	-7.6 dB	-4.29 dB

The results given in Tables II-IV show that on average, the optimal array boosts the SIR compared to the other arrays. In some situations, like the N=7 arrays for Cases 1 and 2, the optimal array produces significant SIR gains compared to the standard arrays. The linear array has some advantage in blocking interference in that it has a smaller field of view than the other 2-D arrays. This is exhibited by the slightly superior results seen for Case 1 and 2 when N=5. However, when the signal is in the same plane as the interferers

(Case 3) the linear array performs poorly. Therefore, while the optimization problem was set up to minimize output power and thereby reduce interference, it does indeed raise the output SIR as desired.

5.7. Summary

To briefly summarize the chapter, an optimization problem (5.16) has been derived whose solution yields an optimal array for suppressing interference. Optimizing an adaptive antenna array's geometry can be done by defining an interference environment, or expected directions and level of interference. In this manner, the array is not optimized for a specific situation, but rather optimized to maximize the performance on average based on the expected environment the array is to operate in. A specific problem of a circular interference environment was studied, and a method of solution was demonstrated using the Simulated Annealing optimization algorithm. In addition, the results implicitly show that the array geometry used has a significant effect on the array's performance.

VI. MINIMUM SIDELOBE LEVELS FOR LINEAR ARRAYS

6.1. Introduction

One-dimensional arrays have been extensively analyzed, dating back to the early part of the 20th century. Their ubiquity in textbooks and actual applications is partly due to the relative ease with which they are analyzed. However, the question of determining the minimum possible sidelobe level for an *N*-element linear array has yet to be determined. Determining this for a linear array of arbitrary elements, steered to an arbitrary angle is the goal of this chapter.

Methods of weight selection were discussed in Chapter 3. The most important for the discussion of this chapter is the Dolph-Chebyshev weighting method. This method can determine minimum-sidelobe weights for uniformly spaced linear arrays of omni-directional antennas. Optimizing geometry for sidelobe minimization has also been examined via a range of techniques, as discussed in Chapter 1. Recently, [17] used the Particle Swarm Optimization (PSO) method to determine optimum sidelobe-minimizing positions for linear arrays assuming the weights were constant.

In this chapter, the weights and positions of a linear array will be optimized to lower sidelobes. In [13],[17] the authors force the arrays to have symmetry about the center to keep the array factor real; the work in this chapter does not require this restriction. In addition, the arrays will have no bounds on minimum or maximum element separation, except in Section 6.5 where a minimum element separation is needed.

For a given linear array, a method of finding the optimum weights for minimizing the sidelobe level is derived given:

- a beamwidth

- the array element's positions
- the individual antenna's radiation pattern
- a desired direction (θ_d) for the array to be scanned.

This problem will be posed in convex form; thus it can be solved without searching through the space of weights as in [28]. The element positions will be unrestricted and the space will be extensively searched via PSO in order to find optimum positions in conjunction with the corresponding optimum weights. The positions found via PSO are likely to be globally optimal as discussed in Section 6.4. Consequently, the results presented here likely represent global bounds on the minimum-possible sidelobe levels achievable for a given beamwidth. This information can be used by array designers in determining how well their arrays perform compared to the best design possible, to determine if altering the weights or element positions could potentially return a significant improvement in performance.

6.2. Problem Setup

The basic geometry of a one-dimensional linear array is shown in Figure 20. The positions of a one-dimensional *N*-element linear array can be written as a vector **d**= $(d_1, d_2, \ldots d_N)$, where d_n is the position of the n^{th} element measured from the origin along the *z*-axis. Incoming plane waves are characterized by an angle θ (measured from the *z*-axis) that specifies their direction of arrival.

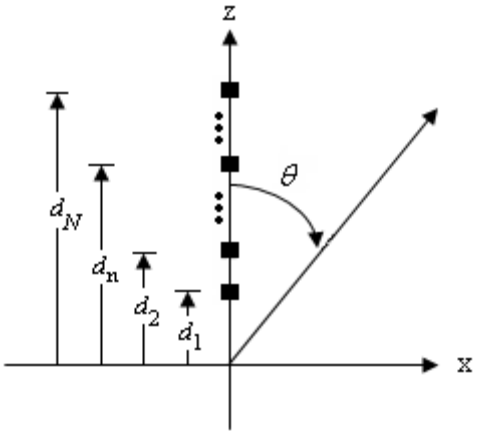


Figure 20. Basic setup of a linear *N*-element array.

Assuming a vector **w**=($w_1, w_2, \ldots, w_N$) of complex excitation weights, the array factor (*AF*) can be rewritten from (2.20) as

$$AF(\mathbf{w}, \mathbf{d}, \theta) = \sum_{n=1}^{N} w_n e^{jkd_n \cos\theta} \tag{6.1}$$

where $k = 2\pi / \lambda$. For a given angle θ , the array factor is a function of the weight vector and the positions of the elements. It will be assumed that the elements are identical and oriented in the same direction. The problem can be extended to arrays with distinct antenna elements in a straightforward manner. The total radiation pattern $T(\mathbf{w}, \mathbf{d}, \theta)$ is then the product of the array factor and the element pattern, given by

$$T(\mathbf{w}, \mathbf{d}, \theta) = f(\theta) AF(\mathbf{w}, \mathbf{d}, \theta), \tag{6.2}$$

where $f(\theta)$ is each elements' radiation pattern. This chapter addresses determining the minimum possible sidelobe level of an *N*-element array for a given beamwidth.

The sidelobe level of an array intrinsically depends on the element positions (**d**) and the weights (**w**). Determining the minimum possible sidelobe level of an *N*-element

linear array consists of finding the optimum combination of weights and element positions that minimize the sidelobe level. To accomplish this, Section 6.3 determines optimum weights for a linear array with arbitrary element positions. Since the optimum weights can be determined for every linear array, the problem reduces to finding the optimum positions that (along with the corresponding optimum weights) yield a sidelobe level that no other combination of weights or element positions can improve upon.

6.3. Determination of Optimum Weights for an Arbitrary Linear Array

For non-uniformly spaced linear arrays, a new method must be developed to determine the optimal sidelobe-minimizing weights. In [27], an optimization procedure using linear programming (LP) was developed for sidelobe-level minimizing weights; however, their results only apply for symmetric arrays and real-valued weights.

Let the positions of an arbitrarily spaced *N*-element linear array be described by the vector **d**. The sidelobe level will be defined as the maximum value of the total radiation pattern outside of the main beam. The beamwidth is then the angular range in which the radiation pattern is not to be minimized. Letting Θ represent the angles in which the radiation pattern is to be suppressed, the sidelobe level (*SLL)* can be written mathematically as

$$SLL = \max_{\theta \in \Theta} |T(\mathbf{w}, \mathbf{d}, \theta)|. \tag{6.3}$$

The normalized radiation pattern is constrained to be unity towards the desired direction (θ_d). The optimum sidelobe-minimizing weights are therefore the solution to the optimization problem given in (6.4).

$$\begin{aligned} &\min_{\mathbf{w} \in C^N} \left\{ \max_{\theta \in \Theta} |T(\mathbf{w}, \mathbf{d}, \theta)| \right\} \\ &s.t. \quad T(\mathbf{w}, \mathbf{d}, \theta_d) = 1 \end{aligned} \tag{6.4}$$

In (6.4), C^N is the set of all *N*-element vectors with complex components, and the complex weight vector **w** is the variable in the problem. This problem can be put into a fairly simple convex optimization form, which is rapidly solvable and solutions are guaranteed to be globally optimum.

To accomplish this, the weights are first expressed in terms of their real and imaginary parts,

$$w_n = w_n^{RE} + j w_n^{IM} . \tag{6.5}$$

The real part of the total radiation pattern can then be written as

$$\mathrm{Re}\{T(\mathbf{w}, \mathbf{d}, \theta)\} = f(\theta) \sum_{n=1}^{N} \{ w_n^{RE} \cos(k d_n \cos\theta) - w_n^{IM} \sin(k d_n \cos\theta) \} , \tag{6.6}$$

and the imaginary part as

$$\mathrm{Im}\{T(\mathbf{w}, \mathbf{d}, \theta)\} = f(\theta) \sum_{n=1}^{N} \{ w_n^{IM} \cos(k d_n \cos\theta) + w_n^{RE} \sin(k d_n \cos\theta) \} . \tag{6.7}$$

Next, Θ is partitioned into *M* discrete sample points $\theta_1, \theta_2, \ldots, \theta_M$. Selection of the sample points is discussed at the end of the section. Minimizing the magnitude of the total radiation pattern at a fixed position θ_i, while the beam is maximum in direction θ_d, can be written as an optimization problem (with variables $w_1^{RE}, w_1^{IM}, \ldots, w_N^{RE}, w_N^{IM}$, t_i and s_i), given in (6.8).

$$\begin{aligned} &\min \ t_i^2 + s_i^2 \\ \text{s.t.} \quad &-t_i \le \mathrm{Re}\{T(\mathbf{w},\mathbf{d},\theta_i)\} \le t_i \\ &-s_i \le \mathrm{Im}\{T(\mathbf{w},\mathbf{d},\theta_i)\} \le s_i \\ &\mathrm{Re}\{T(\mathbf{w},\mathbf{d},\theta_d)\} = 1 \\ &\mathrm{Im}\{T(\mathbf{w},\mathbf{d},\theta_d)\} = 0 \end{aligned} \tag{6.8}$$

In the above optimization problem, t_i and s_i are dummy variables. The inequality constraints in (6.8) on the real part of the total radiation pattern can be written as a linear inequality (for notational simplicity, let $p_n^i = kd_n \cos\theta_i$),

$$f(\theta_i)\begin{bmatrix} \cos p_1^i & -\sin p_1^i & \cdots & \cos p_N^i & -\sin p_N^i & -1 & 0 \\ -\cos p_1^i & \sin p_1^i & \cdots & -\cos p_N^i & \sin p_N^i & -1 & 0 \end{bmatrix} \begin{bmatrix} w_1^{RE} \\ w_1^{IM} \\ \vdots \\ w_N^{RE} \\ w_N^{IM} \\ t_i \\ s_i \end{bmatrix} \le \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{6.9}$$

Similarly, the constraints on the imaginary part of the total radiation pattern can also be expressed in this form. Hence, (6.8) can be rewritten into the simpler form given in (6.10).

$$\begin{aligned} &\min t_i^2 + s_i^2 \\ &s.t. \ \ \mathbf{A}_i\mathbf{Z} \le \mathbf{0} \\ &\mathbf{BZ} = \mathbf{0} \end{aligned} \tag{6.10}$$

In (6.10), $\mathbf{A}_i$ is the matrix that describes the inequality constraints, **B** is a matrix that describes the equality constraints, **Z** is a vector of the problem variables ($w_1^{RE}, w_1^{IM}, w_2^{RE}, \ldots, w_N^{IM}, t_i, s_i$), and **0** is a vector of zeros. The optimization problem of (6.10) is a simple quadratic program, which is easily solved numerically [81].

This procedure can be accomplished at every location θ_i for which the total radiation pattern is to be suppressed. Extend the vector **Z** to include the weights and all the dummy variables:

$$\mathbf{Z}^T = [w_1^{RE}\ w_1^{IM} \cdots w_N^{RE}\ w_N^{IM}\ t_1\ t_2 \cdots t_M\ s_1\ s_2 \cdots s_M]. \tag{6.11}$$

Adding the constraints for all *M* positions to be included in the matrix **A**, the problem in (6.10) can be extended into the form given in (6.12).

$$\begin{aligned} &\min \max_{i=1,\cdots,M} t_i^2 + s_i^2 \\ &s.t.\ \ \mathbf{AZ} \le \mathbf{0} \\ &\quad\ \ \mathbf{BZ} = \mathbf{0} \end{aligned} \tag{6.12}$$

This problem will minimize the array factor at all desired locations. Since $t_i^2 + s_i^2$ is a convex function for all i, the pointwise maximum in (6.12) is also a convex function (as derived in Chapter 4).

Finally, (6.12) does not guarantee that the magnitude of total array radiation pattern is a maximum at θ_d (the maximum could be anywhere outside the region Θ). To have a maximum at θ_d, a necessary condition is for the derivative of the squared magnitude of the radiation pattern to be zero. Writing $T(\theta) = F(\theta) + jI(\theta)$ in terms of its real and imaginary parts, it follows that

$$\frac{dT \cdot T^*}{d\theta} = T\frac{dT^*}{d\theta} + T^*\frac{dT}{d\theta}$$

$$= (F + jI)(F' - jI') + (F - jI)(F' + jI') \,. \tag{6.13}$$

At θ_d, the equality constraints force F=1 and I=0, which implies that the squared magnitude of the total radiation pattern has zero derivative if

$$F' = \frac{d}{d\theta}\mathrm{Re}\{T(\mathbf{w}, \mathbf{d}, \theta_d)\} = 0 \,. \tag{6.14}$$

This constraint is also a linear constraint on **w**, and it can be added to the matrix **B**. When this is implemented, the method is sufficient to have the maximum of the radiation pattern in the direction θ_d (at least for all cases considered in this chapter).

In summary, the result is that (6.12) represents a convex optimization problem and is therefore solvable, and the solutions represent global optima [66]. This problem can be solved via a standard numerical optimization routine, or via commercial software such as MATLAB (for example, using the function *fminimax*).

The only question left then is the selection of the number of sample points, *M*. The only two values of θ that need to be selected are those that define the boundary of the main beam; the remaining values can be selected fairly sparsely. Usually, a spacing of $5°$ between sample points is sufficient. Once the weights are determined, the total radiation pattern can be plotted to show that the sidelobes are indeed suppressed as desired, even in between the sampled points. In theory, it is desirable to choose sample points as closely spaced as possible to guarantee the radiation pattern is suppressed. In

practice, the method works with sparse spacing, and it is advantageous to choose a sparse sampling to speed up the computation time.

The method developed in this section will return weights almost identical to that of the Dolph-Chebyshev method for uniformly spaced linear arrays of omnidirectional antenna elements. The discrepancy results from the Dolph-Chebyshev method using null to null beamwidth and suppressing sidelobes outside of that region; this method suppresses sidelobes outside a specified beamwidth, which isn't necessarily null to null. However, as will be seen in the Section 6.5, the difference is extremely small and the weights are the same to at least three significant digits for the arrays in question.

6.4. Broadside Linear Array

In this section, broadside ($\theta_d = 90°$) *N*-element linear arrays of omnidirectional [$f(\theta) = 1$] antennas are considered. The goal is to determine the optimum element positions for minimizing sidelobes, and consequently, the global bound on sidelobe level for linear arrays. Suppose the desired beamwidth is $2\Delta°$; then the region Θ in which the radiation pattern is to be minimized can be written as

$$\Theta = \{\theta : 0° \le \theta \le \theta_d - \Delta \text{ or } \theta_d + \Delta \le \theta \le 180°\}. \tag{6.15}$$

The sidelobe level (*SLL*) for a given weight vector **w** can be expressed in dB as

$$SLL(\mathbf{w}) = \max_{\theta \in \Theta} 20\log_{10} |T(\mathbf{w}, \mathbf{d}, \theta)|. \tag{6.16}$$

Two cases will be analyzed in this chapter. *Case 1* will have a large beamwidth $2\Delta° = 60°$, and *Case 2* will have a relatively small beamwidth, $2\Delta° = 30°$. A minimum inter-element separation of 0.25λ can be enforced. However, the results are the same

whether or not this constraint is applied since the elements will tend to separate largely from each other to achieve low sidelobes.

Since for any array configuration **d** the optimum weights can be found, the problem now becomes one of finding the best antenna element positions to minimize the *SLL*. For an array with two elements (N=2), the problem is a function of a single variable ($d_2 - d_1$ = element separation and hence a global optimum can easily be found). For linear arrays with more than two elements, a search method needs to be employed that is rapid (without excessive computation time) and accurate (solutions are consistent and no other method leads to a better solution). The Particle Swarm Optimization (PSO) method was found to be suitable for this task. For a detailed discussion of the details of PSO that goes beyond the elementary discussion in Chapter 4, the reader is referred to [17, 82].

While no method short of an exhaustive search can guarantee a global optimum for problems such as these (many local minima in the objective function), intelligent use of PSO will give high confidence that the solutions are indeed globally optimum. For the results presented in this work, the PSO is used by initially choosing a set of P random arrays (these are the particles used in PSO). The arrays are described by a vector of element positions, **d**. For each position vector, the optimum weights are calculated and the sidelobe level is determined. The element positions are updated via the PSO technique, and the process is repeated. The algorithm is run using the PSO parameters set to $w_V = 0.5$, $c_1 = 2.0$, and $c_2 = 2.0$. Each run of the algorithm is executed for approximately 20-200 iterations, or long enough such that several iterations no longer

decrease the objective function. If repeated use of the algorithm does not produce identical results, *P* is increased until consistency is achieved.

The number of particles *P* required for regular convergence is given in Table V. The number of particles required appears to grow roughly as *N!* which indicates PSO cannot consistently return optimum solutions for large arrays. Because the arrays (or particles) start from random positions every time, and because the number of antenna elements is small, this method is fairly certain to return the globally optimum array if the number of particles is large and repeated application of the algorithm returns identical arrays. Table V also gives the approximate time per simulation on a single 3.0 GHz processor running MATLAB (speedup by approximately a factor of *S* can be obtained if the code is written for *S*-processors in parallel).

TABLE V
NUMBER OF PARTICLES REQUIRED FOR CONVERGENCE FOR VARYING ARRAY SIZE WITH SIMULATION TIME

Array Size (*N*)	***P***	**Time (Hours)**
2	2	0.05
3	4	0.2
4	16	1.3
5	100	7
6	700	50
7	3000	300

Results for broadside arrays of size *N*=2-7 elements are presented in Tables VI-IX (note that $d_1 = 0$ for all arrays in this chapter). For both cases, the optimum arrays are very regular, with either uniform or nearly uniform spacing. The weights for this case are allowed to be complex, but are found to be real valued. The weights given in Table VII are identical to the Dolph-Chebyshev weights (to at least 3 significant digits). *Case 2* has

a larger array length, consistent with results developed in [83], in which the beamwidth is reported to decrease with increased array length. The magnitude of the array factor for both cases is shown in Figure 21 for N=6, and the results for N=7 are shown in Figure 22. Because the results for *Case 1* have uniform spacing, the method in [15] correctly determines the optimum sidelobe level for this problem. However, this will not hold in future scenarios.

TABLE VI
OPTIMUM ELEMENT POSITIONS (IN λ) FOR *CASE 1* (BW=60°, $\theta_d = 90°$)

N	d_2	d_3	**Error! Objects cannot be created from editing field codes.**	d_5	d_6	d_7	SLL (dB)
2	0.667						-6.02
3	0.667	1.333					-16.90
4	0.667	1.333	2.000				-27.05
5	0.667	1.333	2.000	2.667			-39.45
6	0.667	1.333	2.000	2.667	3.333		-51.21
7	0.667	1.333	2.000	2.667	3.333	4.000	-62.62

TABLE VII
OPTIMUM WEIGHTS FOR *CASE 1* (BW=60°, $\theta_d = 90°$)

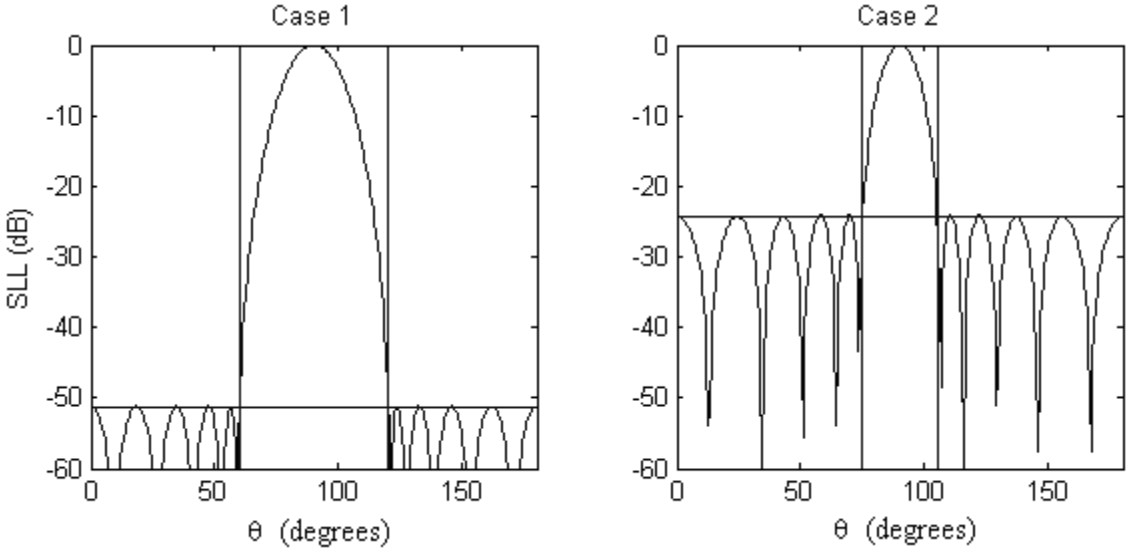
N	w_1	w_2	w_3	w_4	w_5	w_6	w_7
2	0.500	0.500					
3	0.286	0.429	0.286				
4	0.154	0.346	0.346	0.154			
5	0.083	0.247	0.340	0.247	0.083		
6	0.044	0.166	0.290	0.290	0.166	0.044	
7	0.024	0.107	0.227	0.286	0.227	0.107	0.024

TABLE VIII
OPTIMUM ELEMENT POSITIONS (IN λ) FOR *CASE 2* (BW=30°, $\theta_d = 90°$)

N	d_2	d_3	**Error! Objects cannot be created from editing field codes.**	d_5	d_6	d_7	SLL (dB)
2	0.794						-1.95
3	0.794	1.589					-6.59
4	0.794	1.589	2.383				-12.24
5	0.794	1.589	2.383	3.178			-18.18
6	0.794	1.589	2.383	3.178	3.97		-24.15
7	0.794	1.589	2.383	3.178	3.97	4.762	-30.07

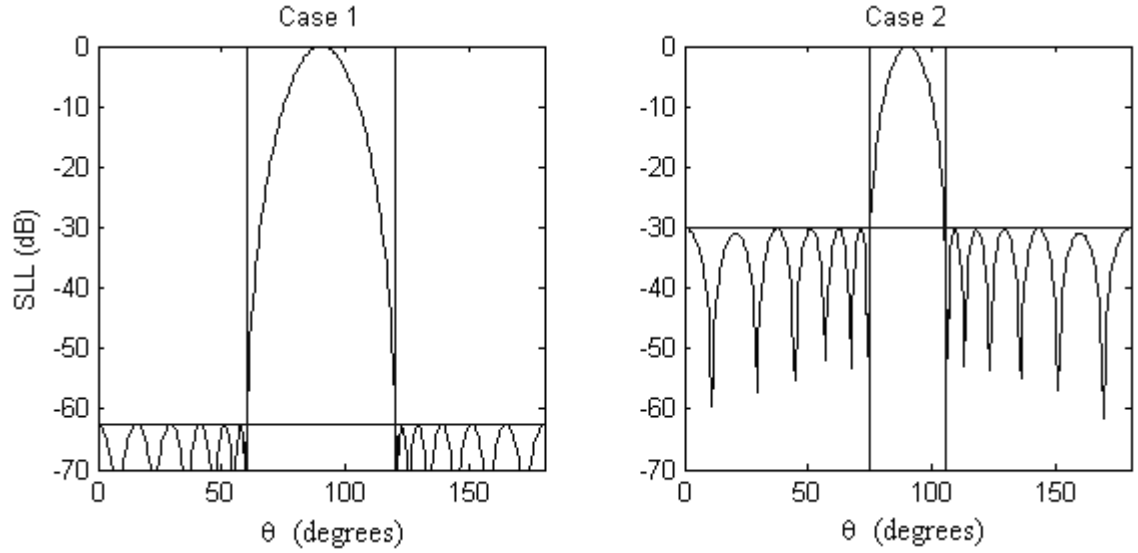
TABLE IX
OPTIMUM WEIGHTS FOR *CASE 2* (BW=30°, $\theta_d = 90°$)

N	w_1	w_2	w_3	w_4	w_5	w_6	w_7
2	0.500	0.500					
3	0.286	0.429	0.286				
4	0.154	0.346	0.346	0.154			
5	0.083	0.247	0.340	0.247	0.083		
6	0.044	0.166	0.290	0.290	0.166	0.044	
7	0.059	0.127	0.198	0.227	0.199	0.130	0.060



(a) ($BW = 60°,\ \theta_d = 90°$) (b) ($BW = 30°,\ \theta_d = 90°$)

Figure 21. Magnitude of array factor for optimal arrays (N=6).



(a) ($BW = 60°,\ \theta_d = 90°$) (b) ($BW = 30°,\ \theta_d = 90°$)

Figure 22. Magnitude of array factor for optimal arrays (N=7).

6.5. Array Scanned to 45 Degrees

Suppose now the goal is to find the optimum array pattern for an arbitrarily spaced *N*-element linear array scanned to $45°$ from broadside ($\theta_d = 45°$). This problem is solved in an identical manner to that in Section 6.4. However, for this problem, the arrays tend to favor closely spaced elements. Consequently, a minimum separation of 0.25λ between elements was enforced, as in [30]. The optimum element positions and the weight vectors are given in Tables X-XIII. The magnitude of the array factor for both cases is shown in Figure 23 for *N*=6, and Figure 24 plots the results for *N*=7.

The weights found in this section are complex. For clarity, $X\angle Y$ is equal to $X\cos(Y) + jX\sin(Y)$. The positions found are very irregular. The elements favor having at least one element separation being the minimum allowable (0.25λ) for many of the cases. Comparing these results to Section 6.4, it is clear that it is more difficult for a linear array to have low sidelobes when it is scanned away from broadside.

TABLE X
OPTIMUM ELEMENT POSITIONS (IN λ) FOR *CASE 1* (BW=$60°$, $\theta_d = 45°$).

N	d_2	d_3	**Error! Objects cannot be created from editing field codes.**	d_5	d_6	d_7	SLL (dB)
2	0.508						-0.76
3	0.250	1.432					-5.44
4	0.250	1.512	1.762				-8.74
5	0.250	1.475	1.725	1.975			-12.64
6	0.250	0.707	1.442	1.912	2.162		-19.60

7	0.250	0.734	1.369	1.850	2.108	3.935	-23.00

TABLE XI
OPTIMUM WEIGHTS FOR *CASE 1* (BW=60°, $\theta_d = 45°$)

(a) Results for *N*=2-4.

	N=2	*N*=3	*N*=4
w_1	0.499	$0.425\angle 43°$	$0.304\angle 45°$
w_2	$0.501\angle 231°$	$0.465\angle 248°$	$0.415\angle 250°$
w_3		$0.386\angle 5°$	$0.408\angle 21°$
w_4			$0.297\angle 227°$

(b) Results for *N*=5-7.

	N=5	*N*=6	*N*=7
w_1	$0.287\angle 30°$	$0.231\angle 51°$	$0.146\angle 44°$
w_2	$0.345\angle 246°$	$0.293\angle 260°$	$0.258\angle 257°$
w_3	$0.359\angle 63°$	$0.172\angle 132°$	$0.195\angle 143°$
w_4	$0.427\angle 279°$	$0.167\angle 37°$	$0.195\angle 48°$
w_5	$0.22\angle 135°$	$0.292\angle 272°$	$0.275\angle 292°$
w_6		$0.237\angle 121°$	$0.169\angle 145°$
w_7			$0.057\angle 25°$

TABLE XII
OPTIMUM ELEMENT POSITIONS (IN λ) FOR *CASE 2* (BW=30°, $\theta_d = 45°$)

N	d_2	d_3	**Error! Objects cannot be created from editing**	d_5	d_6	d_7	SLL (dB)

			field codes.				
2	0.535						-0.31
3	0.250	2.030					-3.77
4	0.302	1.933	2.976				-6.92
5	0.250	2.130	2.380	3.617			-8.85
6	0.250	2.123	2.373	3.653	4.657		-10.39
7	0.250	0.926	2.112	2.362	3.806	4.056	-12.36

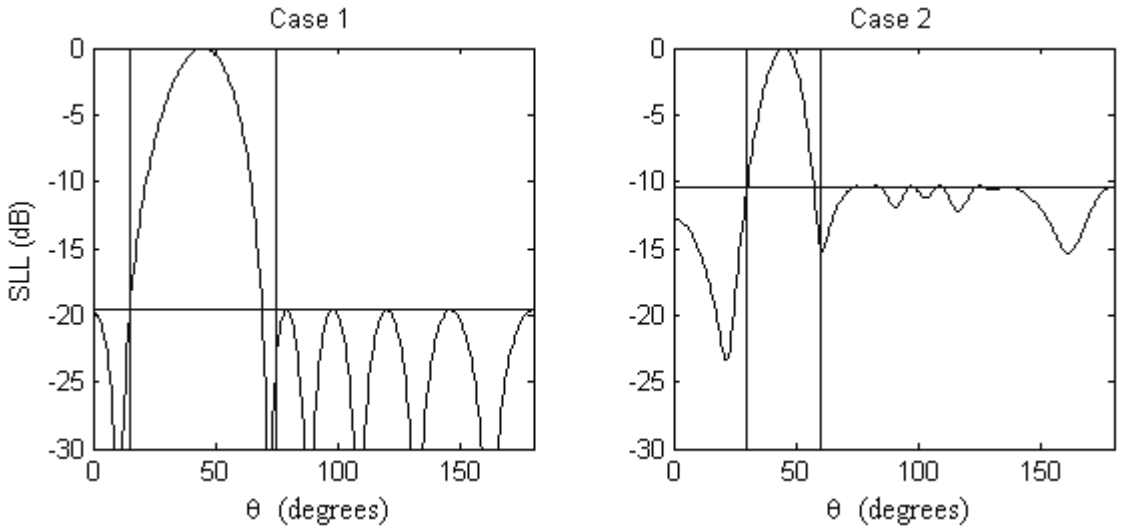
TABLE XIII
OPTIMUM WEIGHTS FOR *CASE 2* (BW=60°, $\theta_d = 45°$)

(a) Results for *N*=2-4.

	N=2	*N*=3	*N*=4
w_1	0.50	0.469∠47°	0.298∠37°
w_2	0.5∠224°	0.524∠248°	0.335∠248°
w_3		0.34∠211°	0.298∠226°
w_4			0.191∠330°

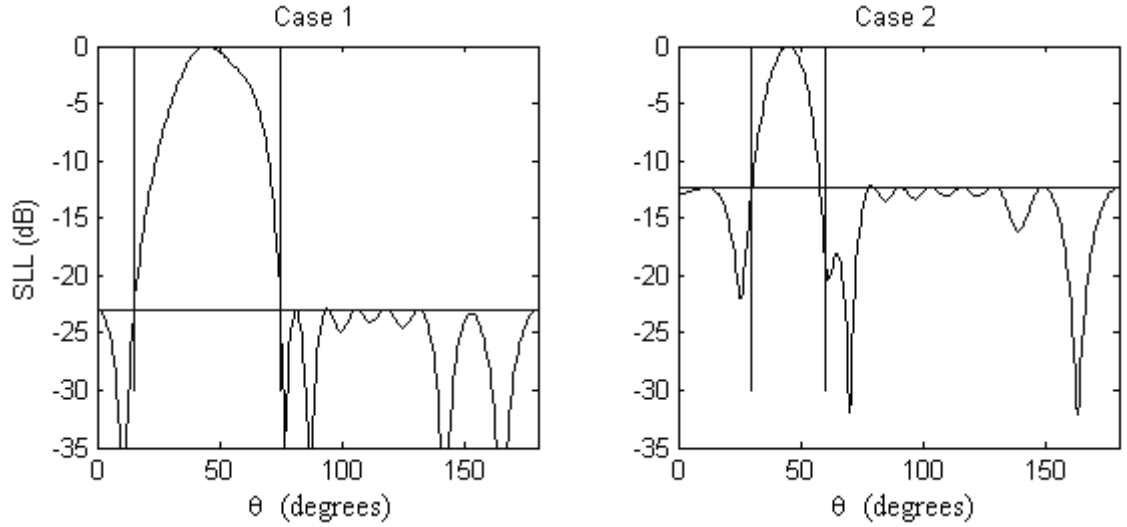
(b) Results for *N*=5-7.

	N=5	*N*=6	*N*=7
w_1	0.23∠49°	0.196∠49°	0.21∠42°
w_2	0.252∠247°	0.218∠248°	0.18∠238°
w_3	0.322∠224°	0.355∠223°	0.165∠111°
w_4	0.385∠70°	0.361∠64°	0.293∠228°
w_5	0.195∠175°	0.186∠169°	0.245∠69°
w_6		0.068∠251°	0.163∠164°
w_7			0.162∠7°



(a) $(BW = 60°,\ \theta_d = 45°)$ (b) $(BW = 30°,\ \theta_d = 45°)$

Figure 23. Magnitude of array factor for optimal arrays (N=6).



(a) $(BW = 60°,\ \theta_d = 45°)$ (b) $(BW = 30°,\ \theta_d = 45°)$

Figure 24. Magnitude of array factor for optimal arrays (N=7).

6.6. Array of Dipoles Scanned to Broadside

The broadside case is considered again, this time assuming the antennas are short or ideal dipoles having a normalized radiation pattern

$$f(\theta) = \sin\theta \,. \tag{6.17}$$

For this case, no minimum separation is required since the elements tend to spread out as in Section 6.4. The problem is solved again as in Section 6.4. The resulting optimum element positions and weights are given in Tables XIV-XVII. The magnitude of the total radiation pattern for both cases is shown in Figure 25 for N=6, and Figure 26 gives the results for N=7.

TABLE XIV
OPTIMUM ELEMENT POSITIONS (IN λ) FOR *CASE 1* WITH DIPOLES (BW= $60°, \theta_d = 90°$)

N	d_2	d_3	**Error! Objects cannot be created from editing field codes.**	d_5	d_6	d_7	SLL (dB)
2	0.793						-11.15
3	0.726	1.452					-22.38
4	0.703	1.403	2.116				-34.00
5	0.696	1.391	2.087	2.788			-45.40
6	0.693	1.382	2.069	2.755	3.441		-56.50
7	0.681	1.362	2.043	2.724	3.405	4.086	-66.90

TABLE XV
OPTIMUM WEIGHTS FOR *CASE 1* WITH DIPOLES (BW= $60°, \theta_d = 90°$)

N	w_1	w_2	w_3	w_4	w_5	w_6	w_7
2	0.500	0.500					
3	0.276	0.448	0.276				

4	0.150	0.351	0.351	0.148			
5	0.082	0.25	0.343	0.246	0.079		
6	0.043	0.165	0.292	0.292	0.165	0.043	
7	0.023	0.106	0.227	0.288	0.227	0.106	0.023

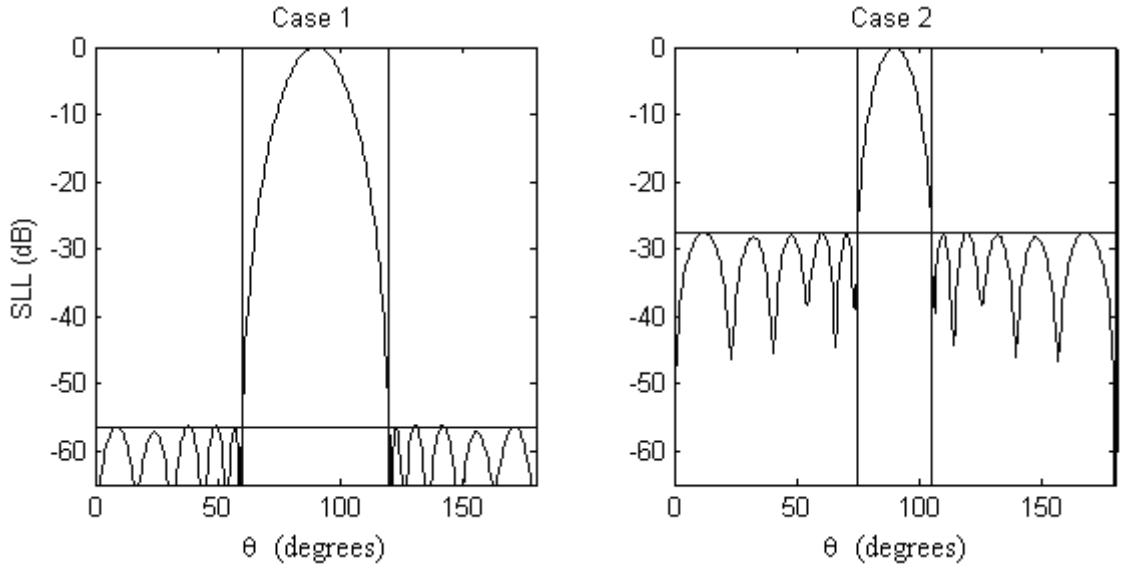
TABLE XVI
OPTIMUM ELEMENT POSITIONS (IN λ) AND SLL FOR *CASE 2* WITH DIPOLES (BW=30°, $\theta_d = 90°$)

N	d_2	d_3	**Error! Objects cannot be created from editing field codes.**	d_5	d_6	d_7	SLL (dB)
2	1.130						-4.64
3	0.951	1.900					-9.93
4	0.915	1.779	2.675				-15.70
5	0.876	1.723	2.565	3.463			-21.70
6	0.842	1.681	2.516	3.358	4.233		-27.50
7	0.860	1.706	2.528	3.362	4.192	5.024	-33.58

TABLE XVII
OPTIMUM WEIGHTS FOR *CASE 2* WITH DIPOLES (BW=30°, $\theta_d = 90°$)

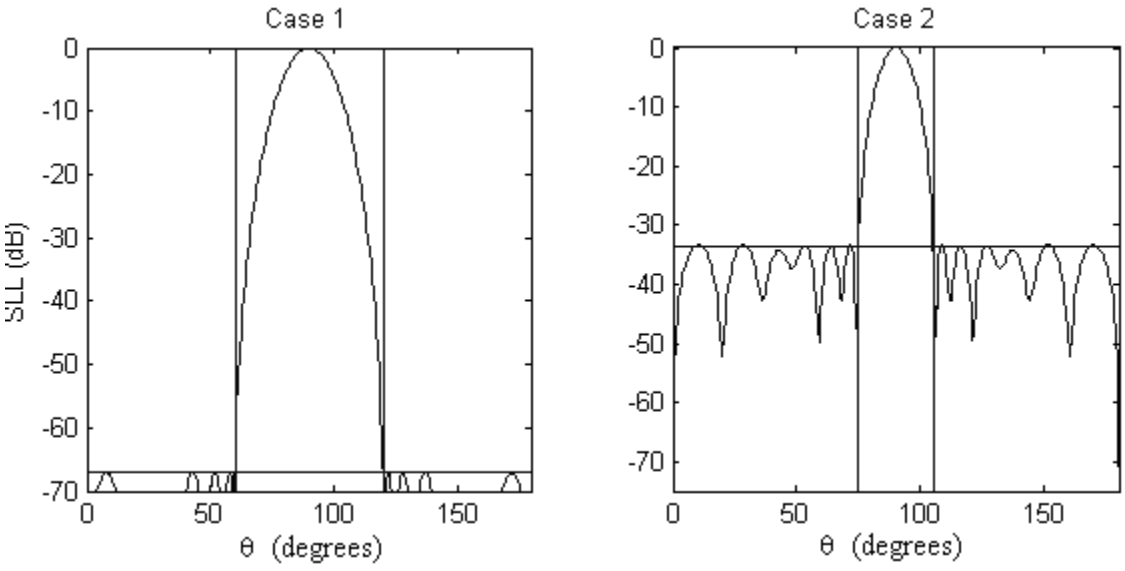
N	w_1	w_2	w_3	w_4	w_5	w_6	w_7
2	0.500	0.500					
3	0.341	0.313	0.347				
4	0.224	0.283	0.273	0.220			
5	0.139	0.224	0.270	0.224	0.143		
6	0.087	0.168	0.239	0.238	0.178	0.090	

7	0.047	0.122	0.196	0.231	0.208	0.134	0.062



(a) $(BW = 60°,\ \theta_d = 90°)$ (b) $(BW = 30°,\ \theta_d = 90°)$

Figure 25. Magnitude of the total radiation pattern for optimal arrays of dipoles (N=6).



(a) $(BW = 60°,\ \theta_d = 90°)$ (b) $(BW = 30°,\ \theta_d = 90°)$

Figure 26. Magnitude of the total radiation pattern for optimal arrays of dipoles (N=7).

The results of this section, compared with the results of Section 6.4, show that the optimum linear array element positions (and associated weights) will be different depending on the type of antenna elements used in the array. The elements in this section have a larger spacing than the broadside array considered in Section 6.4. The individual dipole's radiation pattern works to lower the total radiation pattern away from broadside. Consequently, this helps to lower the overall sidelobe level, so that an array of dipoles has lower sidelobes than an array of omnidirectional radiators. This is evident by comparing the results of this section and Section 6.4.

6.7. Mutual Coupling

Mutual coupling is present in all antenna arrays to some degree. This coupling affects the radiation pattern of the elements which can degrade the overall radiation pattern [84]. In this section, the extent to which the above results vary due to mutual coupling is considered.

Without mutual coupling, the output of the array will be written as $\mathbf{X}_{ideal}$. When mutual coupling is present, the output of the array will be written as $\mathbf{X}_{actual}$. Because of the linearity in Maxwell's equations, it is reasonable to model the coupling as a linear system. Hence the relationship between the ideal and actual array outputs can be written as

$$\mathbf{X}_{actual} = \mathbf{C}\mathbf{X}_{ideal}, \tag{6.18}$$

where **C** is a square matrix known as the *mutual coupling matrix*. This matrix can be modeled as [85]

$$\mathbf{C} = Z_L(\mathbf{Z} + Z_L\mathbf{I})^{-1}, \tag{6.19}$$

where Z_L is the load impedance in each element. In (6-19), **Z** is the impedance matrix, which relates the current into each antenna to the voltage,

$$\mathbf{V=ZI.} \tag{6.20}$$

If the mutual coupling matrix is known, then the array input can be pre-multiplied by the inverse of the coupling matrix to obtain the decoupled weights, as in (6.21).

$$\mathbf{X} = \mathbf{C}^{-1}\mathbf{X}_{actual} \tag{6.21}$$

Since the output of the array is given by

$$y = \mathbf{w}^H\mathbf{X} = \mathbf{w}^H\mathbf{C}^{-1}\mathbf{X}_{actual}\ , \tag{6.22}$$

the optimal weights derived in section 6.3 can be replaced by

$$\mathbf{w}'_{opt} = \mathbf{C}^{-1}\mathbf{w}_{opt}\ . \tag{6.23}$$

The resulting total radiation patterns will then be the same as presented here, to the extent that the mutual coupling matrix model is correct. Experimental results by Huang *et al.* [85] suggest that the model performs fairly well. A circular array of dipoles was considered in that work, which will necessarily have a strong degree of mutual coupling because the elements are each in line with the other element's direction of maximum radiation. Using (6.23), the compensated radiation patterns for the arrays were compared to the ideal or non-coupled case and the results were found to be in agreement. Hence, it is expected that arrays without such a strong degree of coupling (as commonly used in practice), can also be accurately modeled using (6.18) and (6.19).

6.8. Conclusions

This chapter determined the limits of performance on linear arrays of size N=2-7. A method of determining globally optimum weights for minimizing sidelobes for a given

linear array was presented. The elements were then varied in position until it was certain that a global optimum was found. Consequently, it is very likely that no other weight strategy or element placement scheme will lead to sidelobes lower than those presented in this paper. These results can be used as a benchmark in comparing existing array performance to determine if it is worth updating the array placement or weighting strategy.

VII. MINIMIZING SIDELOBES IN PLANAR ARRAYS

7.1. Introduction

The natural next step in studying sidelobe-minimization in antenna arrays is to look at two-dimensional or planar arrays. In this chapter, many of the ideas from Chapter 6 are extended to two-dimensional arrays, which are mathematically similar but the total radiation pattern is more complex.

Sidelobe minimization has received renewed interest due to the difficult nature of the wireless channel. To block interference, it is best to place nulls in the direction of the interference. However, this often does not work well in practice. For example, the European standard for 3rd generation of mobile communication is known as Wideband Code Division Multiple Access (WCDMA). In this scheme, the same frequency spectrum is shared simultaneously by all users; for an in-depth description, see [86]. Consequently, interference is a major problem. These systems are designed to work with a large number of users, and since an N-element antenna array can only place N-1 nulls, an impractically large number of antennas would be needed to null out signals from all directions not of interest. In addition, due to multipath effects, each signal will be arriving from several distinct angles, which further reduces the performance of a nulling based approach. In [87], the performance of arrays with different weighting methods used in WCDMA systems are compared. It was found that for a large number of interferers, a low sidelobe method will outperform a nulling method. The low sidelobe method is also preferred because no processing needs to be performed to determine direction-of-arrivals for varying signals. Hence, as the capacity requirements of wireless

communication systems increase, methods for reducing sidelobe levels will become increasingly important.

In addition, the WCDMA systems should not be modeled using a single-frequency or narrowband total radiation pattern. To address this, wideband arrays will be studied in the latter half of this chapter. The sidelobe-minimizing weighting methods will be extended to work for wideband arrays. Previous work has been performed in an attempt to develop wideband sidelobe-minimizing weights. In [88], the author develops a wideband weighting method that works for 2D rectangular arrays. This method does incorporate the antenna patterns into determining the weight vector, which increases the utility of the method. However, the results of that work assume all signals arrive from a fixed elevation angle, which is a major restriction. In addition, the results are clearly suboptimal in viewing the resulting sidelobes.

Other wideband weighting methods use antenna coefficients that vary with frequency in order to improve the radiation pattern over a range of frequencies. This is done using a tapped delay line filter in [89], and with a recursive filter in [90]. In this chapter, the weights will continue to be constant (not a function of frequency) so that the weights derived are easily implemented in a real system.

The chapter is organized as follows. In Section 7.2, the two-dimensional sidelobe minimization problem is addressed. In Section 7.3, sidelobe-minimizing weights are developed for two-dimensional arrays of arbitrary elements. In Section 7.4, optimal arrays and sidelobe levels are obtained for arrays of omnidirectional antennas of size N=4-7 for two distinct beamwidths. In Section 7.5, optimal arrays and sidelobe levels

are obtained for arrays of patch (or microstrip) antennas. A method of determining weight-minimizing sidelobes over a range of frequencies is developed in Section 7.6. Optimal arrays and sidelobe levels are obtained for wideband arrays of omnidirectional antennas in Section 7.7. Finally, in Section 7.8 optimal arrays and sidelobe levels are obtained for wideband arrays of patch antennas, and conclusions presented in Section 7.9.

7.2. Two-Dimensional Symmetric Arrays

The elements of the array are assumed to lie in the x-y plane, at z=0. The position of the n^{th} element is $d_n = (x_n, y_n, 0)$ as shown in Figure 27.

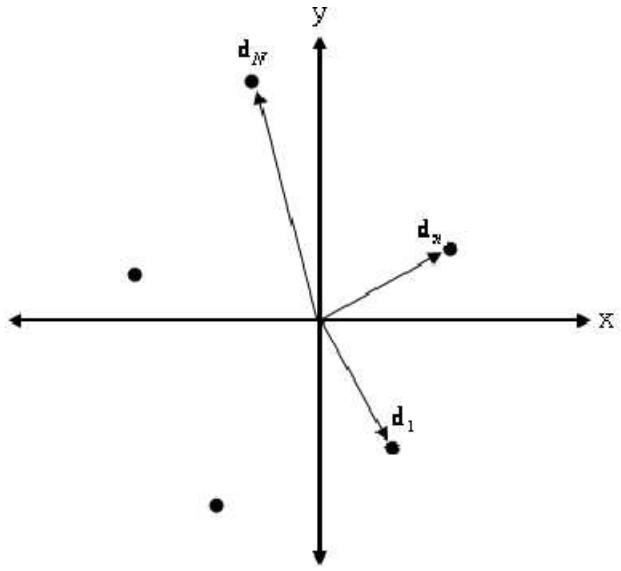


Figure 27. Arbitrary planar array.

The output or radiation pattern of an antenna array (or spatial filter) is given by

$$T(\mathbf{w}, \mathbf{D}, k_x, k_y) = \sum_{n=1}^{N} w_n f_n(k_x, k_y) e^{j(k_x x_n + k_y y_n)} \tag{7.1}$$

where w_n is the weight multiplying the signal of the n^{th} element, and $f_n(k_x, k_y)$ is the antenna gain for the n^{th} element in the direction determined by (k_x, k_y). It is desired for the array pattern to have a maximum at the desired direction, denoted (k_{xd}, k_{yd}).

This chapter deals with two-dimensional arrays with symmetry about the origin. That is, if an element is located at $d_n = (x_n, y_n, 0)$ and not at the origin, there exists another element in the array with the same weighting coefficient located at $-d_n = (-x_n, -y_n, 0)$. This constraint keeps the array factor real when the weights are real, allowing efficient computation of the results.

When the antennas in the array are identical (have the same individual radiation pattern), the radiation pattern for the entire antenna array takes the form given in (7.2) if there is an element at the origin (odd number of elements). If there is an even number of elements, the radiation pattern will have the form given in (7.3).

$$T(\mathbf{w}, \mathbf{D}, k_x, k_y) = w_1 f(k_x, k_y) + f(k_x, k_y) \sum_{n=2}^{(N+1)/2} 2w_n \cos(k_x x_n + k_y y_n) \tag{7.2}$$

$$T(\mathbf{w}, \mathbf{D}, k_x, k_y) = f(k_x, k_y) \sum_{n=1}^{N/2} 2w_n \cos(k_x x_n + k_y y_n) \tag{7.3}$$

7.3. Sidelobe-Minimizing Weights for Two-Dimensional Arrays

A method of sidelobe minimization for symmetric linear arrays with real weights was given as an example in Section 4.2. The results are now extended for two-dimensional arrays of non-isotropic elements. It will be assumed that the array positions **D** are known, the array is to be steered toward $\theta_d = 0$ or $(k_{xd}, k_{yd}) = (0, 0)$, and to have a specified beamwidth for the main beam.

The discussion will follow the development in Section 4.2. The transition region is defined as the region in (k_x, k_y) space in which the sidelobes are not to be suppressed. The suppression region Θ is now two-dimensional, and a circular transition region will be assumed. For a circular transition region, the cutoff region occurs when $\theta \geq \theta_c$. The suppression region can be specified as

$$\Theta = \left\{ (k_x, k_y) : k_c^2 \leq k_x^2 + k_y^2 \leq \left(\frac{2\pi}{\lambda} \right)^2 \right\}, \tag{7.4}$$

where a cutoff value ($k_c = 2\pi \sin\theta_c / \lambda$) is specified. The cutoff value dictates how wide or narrow the array's mainbeam is to be. The suppression region is illustrated in Figure 28 in (k_x, k_y) space. The region in the (k_x, k_y) plane with magnitude less than $2\pi / \lambda$ is commonly referred to as the *visible region*. Values of (k_x, k_y) outside of this region do not correspond to any value of (θ, ϕ) at the frequency of interest.

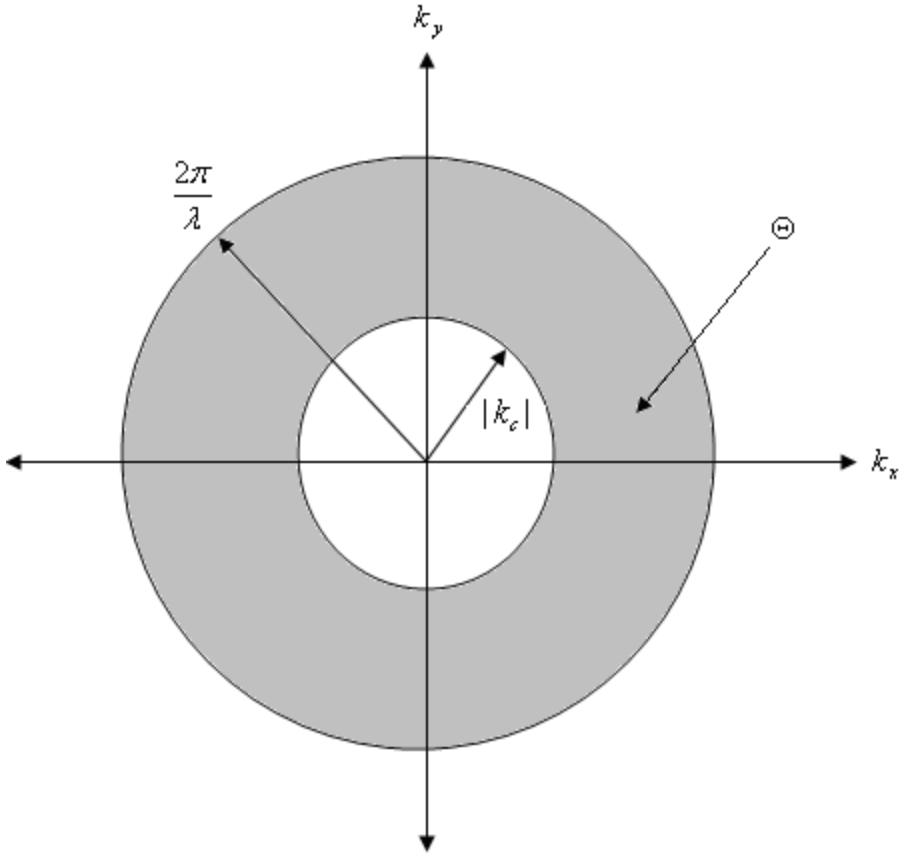


Figure 28. Suppression region for two-dimensional arrays.

The suppression region can be sampled at R places as in Section 4.2. Each sample point is denoted by (k_{xi}, k_{yi}) for $i = 1, 2, \ldots, R$. The parameter R is chosen sufficiently large such that when the resulting radiation pattern is plotted, it is suppressed even between sample points. The LP problem of (4-20) is rewritten to include the non-isotropic element pattern in (7.5).

$$\begin{aligned} &\min \quad t \\ &s.t. \quad T(\mathbf{w}, \mathbf{D}, 0, 0) = 1 \\ &\qquad |T(\mathbf{w}, \mathbf{D}, k_{xi}, k_{yi})| \le t, \quad i = 1, 2, \ldots, R \end{aligned} \tag{7.5}$$

The constraints in (7.5) are again linear functions of the weights, as seen in (7.2-7.3). Hence, the constraints in (7.5) can be rewritten as affine inequalities exactly as done in

Section 4.2. The result is that the problem of finding the optimal sidelobe-suppressing weights for two-dimensional array of non-isotropic elements is again a linear program, and therefore rapidly solvable.

As an example, consider the 7-element hexagonal array of Figure 19, but with a radius of 0.77λ. A weighting method with a linear phase-taper (from Section 3.2) will be used for comparison. When the array is steered to broadside, the sidelobe level for the linear phase-tapered array is -10.9 dB. The beamwidth is $30°$. Using this beamwidth to define the suppression region as in Figure 28, the optimal weights can be determined. The sidelobe level for the array factor with the optimal weights is -13.9 dB. The optimal array factor is plotted, along with the array factor using the phase-tapered weights, in Figure 29 (elevation plot for $\phi = 0°$) and Figure 30 (elevation plot for $\phi = 45°$). The positions and optimal weights are listed in Table XVIII.

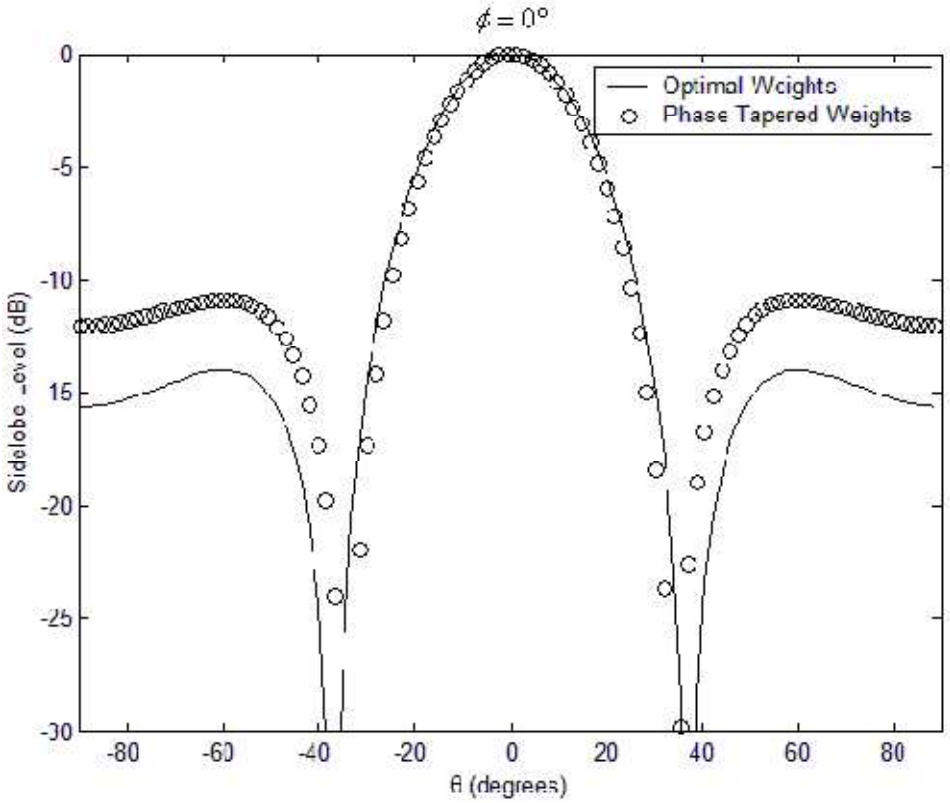


Figure 29. Array factors for optimal weighted and phase-tapered array ($\phi = 0°$).

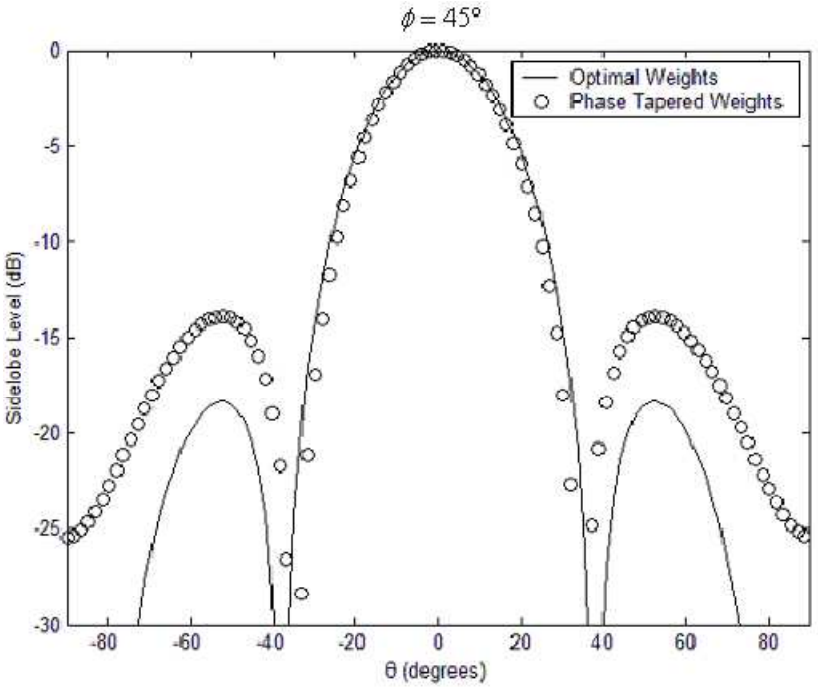


Figure 30. Array factors for optimal weighted and phase-tapered array ($\phi = 45°$).

TABLE XVIII
OPTIMAL WEIGHTS FOR 7-ELEMENT HEXAGONAL ARRAY

Position	Weight
(0, 0)	0.200
± (0.77, 0)	0.133
± (0.385, 0.667)	0.133
± (-0.385, 0.667)	0.133

7.4 Sidelobe-Minimizing Weights for Scanned Two-Dimensional Arrays

In this section, the procedure of Section 7.3 is extended for two-dimensional arrays scanned from broadside. It is assumed that the array is scanned towards (θ_d, ϕ_d), so that the wavevector components in the desired direction are

$$k_{xd} = \frac{2\pi}{\lambda} \sin\theta_d \cos\phi_d \tag{7.6}$$

$$k_{yd} = \frac{2\pi}{\lambda} \sin\theta_d \sin\phi_d \tag{7.7}$$

Two beamwidths are specified. The first is the polar (elevation) beamwidth $\Delta\theta$, which is the beamwidth when the azimuth angle is fixed at ϕ_d. The second is the azimuth beamwidth $\Delta\phi$, which is the beamwidth when the polar (elevation) angle is fixed at θ_d.

The method of Section 7.3, in which the weights are real, will not produce optimal weights when the array is to be steered from broadside. As a result, the complex method of Section 6.3 must be used. Once the suppression region has been specified as in Figure 28, the method can be directly implemented for the two-dimensional case.

An example will now be presented using this method. A 5x5 rectangular array with uniform spacing of $\lambda/4$ is used. The results are compared to the performance of an array with a linear phase taper, as discussed in Section 3.2. The array is scanned to $(\theta_d, \phi_d) = (90°, 0°)$. The magnitude of the array factor is plotted in Figure 31. The maximum sidelobe level is -12.04 dB. The beamwidth selected for determining the optimal weights is identical to the result for the linear phase taper array for comparison. Hence, $\Delta\theta/2 = 68.7°$ and $\Delta\phi/2 = 38.2°$.

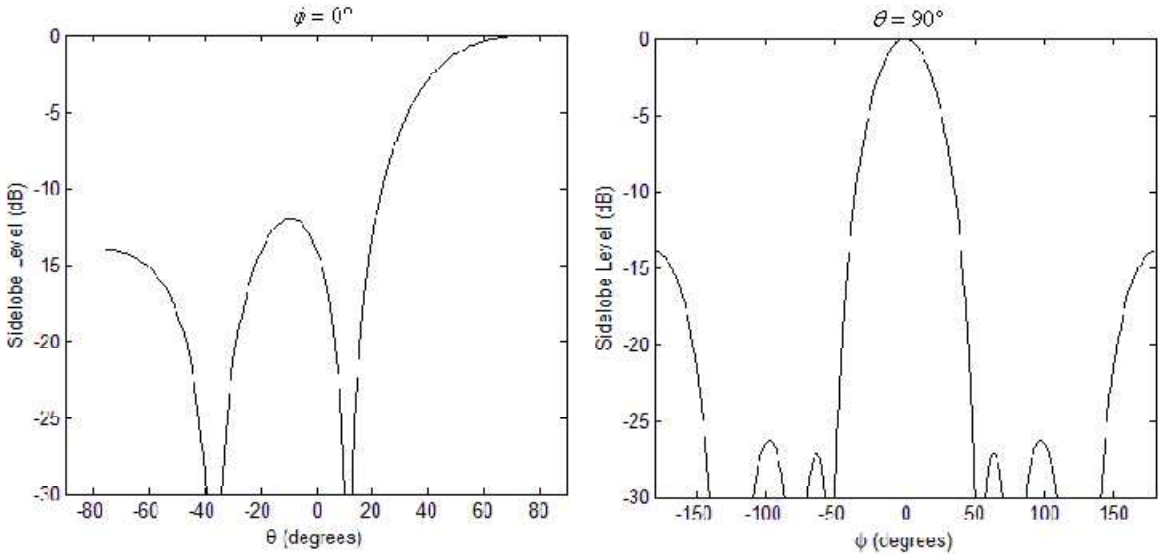


Figure 31. $|AF|$ for phase-tapered weights; (a) elevation plot, (b) azimuth plot.

The following parameters are defined that indicate the boundary of the suppression region in the (k_x, k_y) plane:

$$k_{x\phi} = \frac{2\pi}{\lambda} \sin\theta_d \cos\left(\phi_d + \frac{\Delta\phi}{2}\right) \tag{7.8}$$

$$k_{y\phi} = \frac{2\pi}{\lambda} \sin\theta_d \sin\left(\phi_d + \frac{\Delta\phi}{2}\right) \tag{7.9}$$

$$k_{x\theta} = \frac{2\pi}{\lambda} \sin\left(\theta_d - \frac{\Delta\theta}{2}\right) \cos\phi_d \tag{7.10}$$

$$k_{y\theta} = \frac{2\pi}{\lambda} \sin\left(\theta_d - \frac{\Delta\theta}{2}\right) \sin\phi_d \tag{7.11}$$

Increasing θ for a fixed azimuth angle is equivalent to moving outward in the radial direction in the (k_x, k_y) plane. Increasing ϕ for a fixed elevation angle is equivalent to moving in a circle (at a fixed distance from the origin) in the (k_x, k_y) plane.

Consequently, the suppression region Θ has the form plotted in Figure 32 for this example.

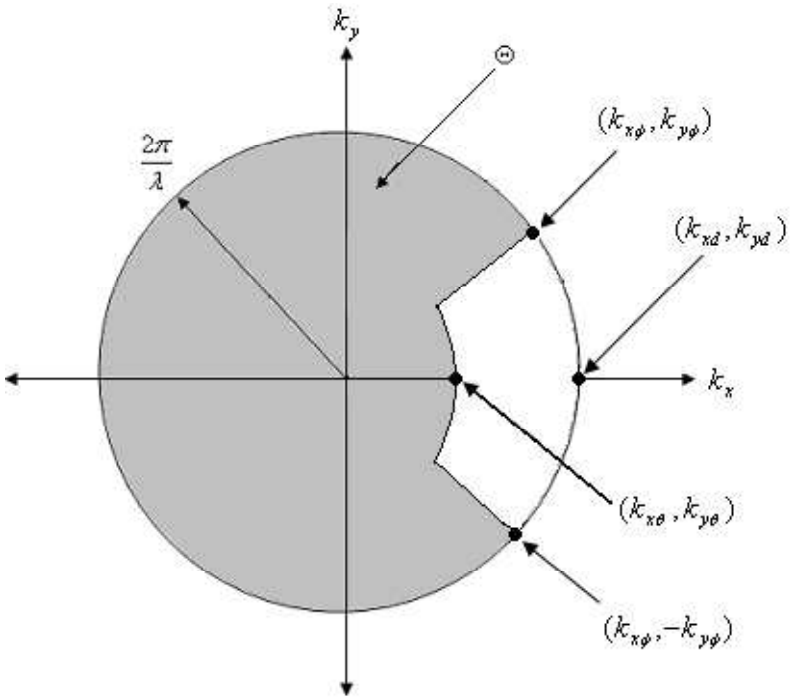


Figure 32. Suppression region for an array scanned away from broadside.

Employing the method of Section 6.3 with the suppression region in Figure 32, the optimal weights can be determined, and they are listed along with their respective positions in Table XIX. The sidelobe level is reduced to -31.2 dB, showing the superiority of this method over the linear phase-taper method. The array factor using the optimal weights is plotted, along with the array factor using the phase-tapered weights, in Figure 33 for an azimuth scan with a fixed elevation angle ($\theta = 90°$). The elevation scan is plotted in Figure 34 with a fixed azimuth angle ($\phi = 0°$).

TABLE XIX
OPTIMAL WEIGHTS WITH ASSOCIATED POSITIONS

Position	**Weight**	**Position**	**Weight**
(-0.5, 0.5)	0.103∠246°	(0, -0.25)	0.020∠288°
(-0.5, 0.25)	0.192∠31°	(0, -0.5)	0.166∠355°
(-0.5, 0)	0.227∠227°	(0.25, 0.5)	0.126∠235°
(-0.5, -0.25)	0.204∠22°	(0.25, 0.25)	0.179∠133°
(-0.5, -0.5)	0.090∠234°	(0.25, 0)	0.291∠256°
(-0.25, 0.5)	0.184∠118°	(0.25, -0.25)	0.211∠142°
(-0.25, 0.25)	0.182∠246°	(0.25, -0.5)	0.108∠244°
(-0.25, 0)	0.182∠246°	(0.5, 0.5)	0.066∠126°
(-0.25, -0.25)	0.180∠221°	(0.5, 0.25)	0.174∠346°
(-0.25, -0.5)	0.137∠113°	(0.5, 0)	0.196∠143°
(0, 0.5)	0.210∠350°	(0.5, -0.25)	0.185∠350°
(0, 0.25)	0.067∠168°	(0.5, -0.5)	0.060∠137°
(0, 0)	0.418∠4°		

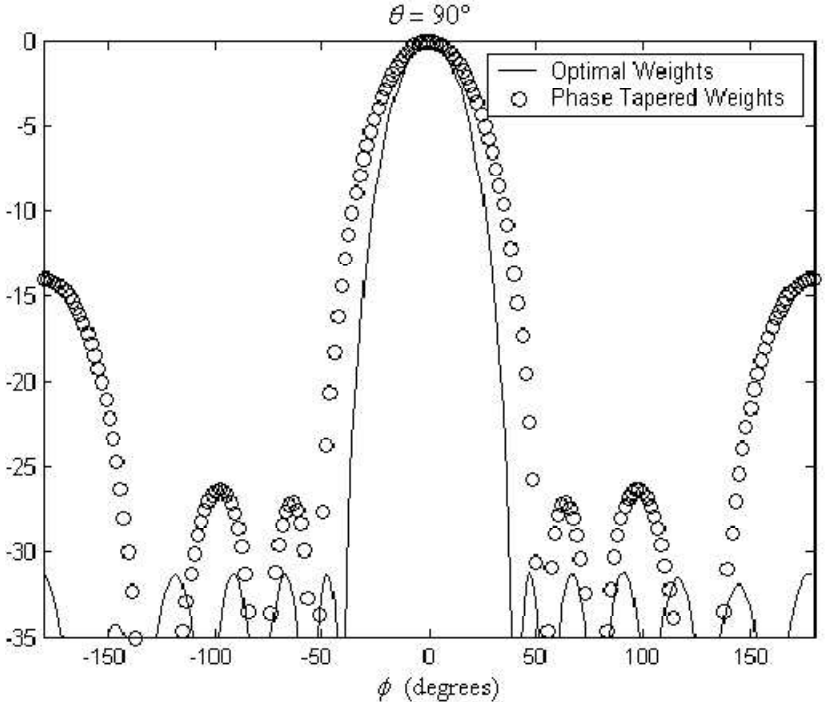


Figure 33. Azimuth plot of array factors with optimal and phase-tapered weights.

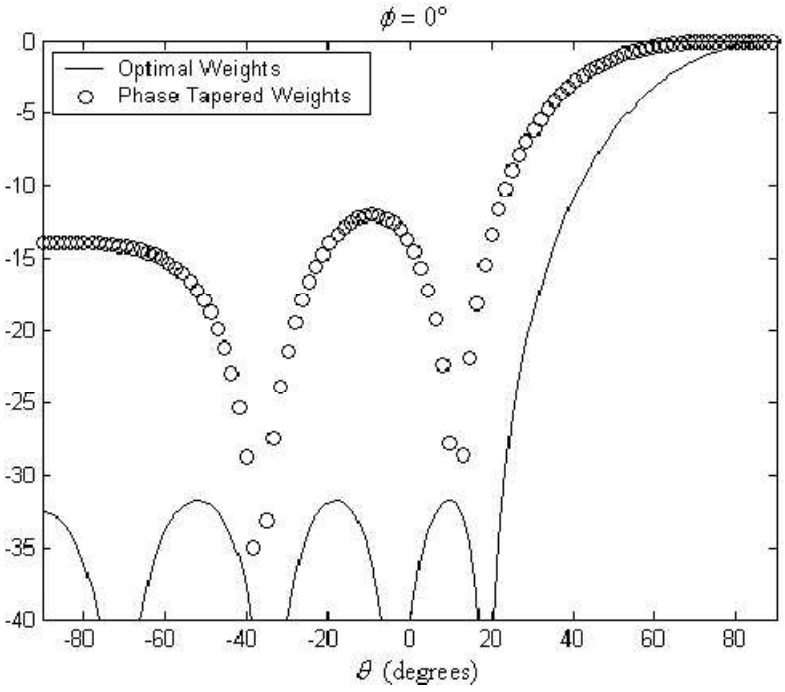


Figure 34. Elevation plot of array factors with optimal and phase-tapered weights.

7.5. Symmetric Arrays of Omnidirectional Elements

In this section, the antennas are omnidirectional, so that

$$f(k_x, k_y) = 1. \tag{7.12}$$

The array elements are allowed to assume an arbitrary geometry, subject to the arrays being symmetric, as in (7.2) and (7.3). The goal of this section is to determine the geometry that, accompanied with the optimum weights of Section 7.2, yield the minimum sidelobe level. The positions will be varied using the PSO algorithm as in Chapter 6 in order to determine an optimal geometry for minimum sidelobes. The parameters and method of implementation used are the same as in Chapter 6.

The algorithm is again run with P particles. The particles move around and interact via the PSO algorithm. As the element positions are varied, the optimum weights are

calculated for each particle (or array) at every iteration. Consequently, the weights and the array geometry are simultaneously optimized. The number of particles is increased until successive runs of the algorithm return identical results. Because the algorithm starts with random and independent particles (or arrays) every time, and because the algorithm consistently returns identical solutions, it is likely that the results are globally optimal.

Two cases are considered in this section. In *Case 1*, the beamwidth will be $60°$; this indicates the sidelobes are to be suppressed when $\theta \geq 30°$. The cutoff value can be calculated to be

$$k_{c1} = \frac{2\pi}{\lambda}\sin(30°) = \frac{\pi}{\lambda}. \tag{7.13}$$

For *Case 2*, a smaller beamwidth of $30°$ is considered. The sidelobes will therefore be suppressed when $\theta \geq 15°$. The cutoff value is then

$$k_{c2} = \frac{2\pi}{\lambda}\sin(15°) = \frac{0.518\pi}{\lambda}. \tag{7.14}$$

For symmetric arrays, there exists no optimal 2 or 3 element symmetric arrays, as the symmetry forces the arrays to be linear. A linear array cannot suppress sidelobes in two dimensions because the pattern of a linear array is only a function of one variable. Results will be presented for arrays of size N=4-7.

The number of particles required for regular convergence is given in Table XX, along with the average simulation time. The simulations were performed on a 3.0 GHz processor running MATLAB.

TABLE XX
NUMBER OF REQUIRED PARTICLES FOR PSO AND COMPUTATION TIME FOR N=4-7

N	P	Time (hours)
4	290	3
5	300	3.5
6	800	7
7	1000	10

The optimal arrays for *Case 1* are plotted in Figure 35. For N=4, 5, 7, the optimal arrays are close to being circular with an increasingly large radius, and a center element if the array has an odd number of elements. The result for N=6 is distinct, as it takes a cross shape. The optimal array when N=7 is also a hexagonal array, as discussed in Section 5.6.

The optimal positions are listed with the sidelobe levels in Table XXI. The corresponding optimal weights are given in Table XXII. In Figure 36, the magnitude of the array factor is plotted as a function of the elevation angle θ for several azimuth angles. The mainbeam is almost identical within the transition region ($\theta \leq 30°$) for distinct azimuth angles. This indicates a circularly symmetric mainbeam, as expected with a circular suppression region.

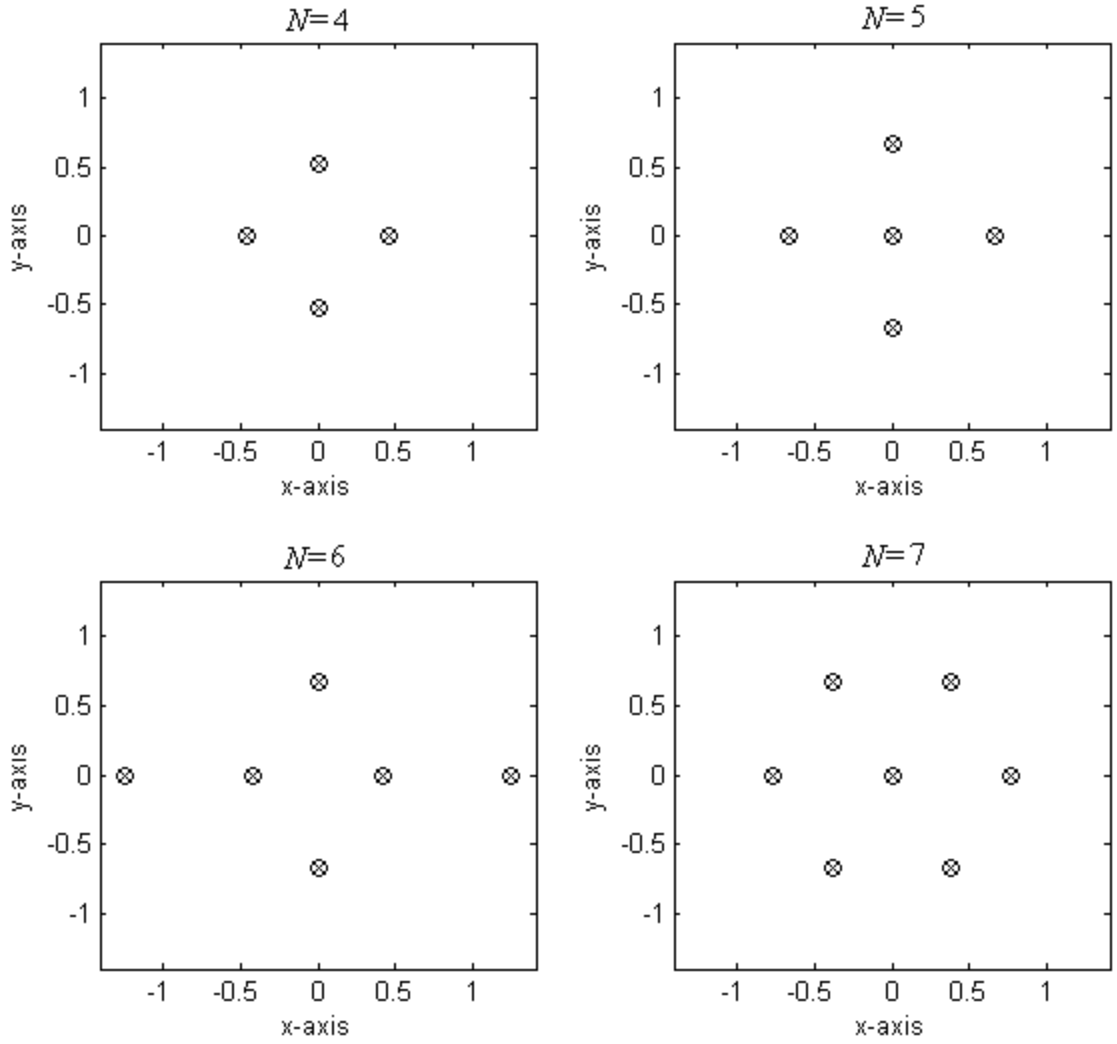


Figure 35. Optimal symmetric array locations for *Case 1* (dimensions in λ).

TABLE XXI
OPTIMAL SLL AND POSITIONS FOR *CASE 1* (DIMENSIONS IN λ)

	(x_1, y_1)	(x_2, y_2)	(x_3, y_3)	(x_4, y_4)	SLL (dB)
N=4	(0.45, 0)	(0, 0.52)			-5.5
N=5	(0, 0)	(0.67, 0)	(0, 0.67)		-6.9
N=6	(0.41, 0)	(1.24, 0)	(0, 0.67)		-7.9
N=7	(0, 0)	(0.77, 0)	(0.39, 0.67)	(-0.39, 0.67)	-13.9

TABLE XXII
OPTIMAL WEIGHTS FOR *CASE 1*

	w_1	w_2	w_3	w_4
N=4	0.556	0.444		
N=5	0.268	0.183	0.183	
N=6	0.442	0.159	0.400	
N=7	0.200	0.133	0.133	0.133

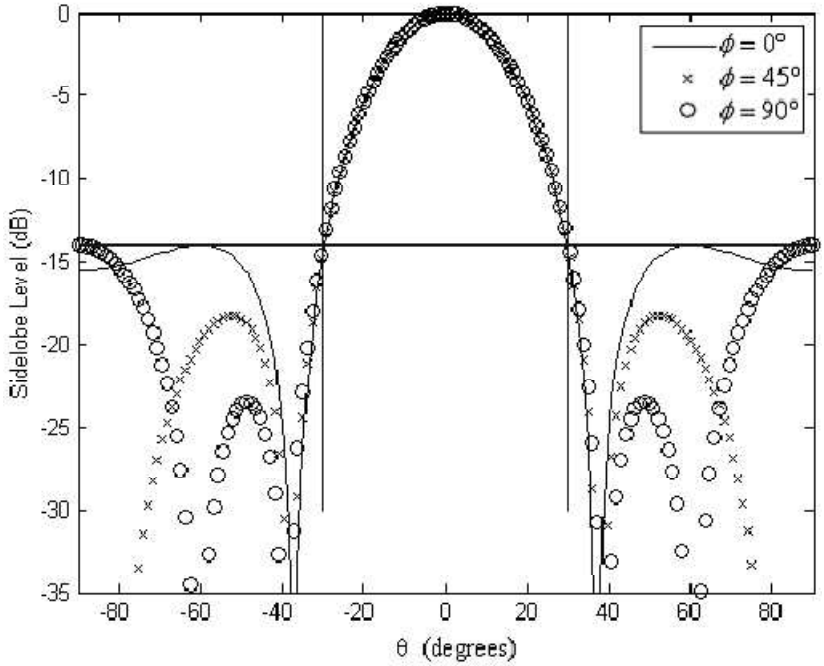


Figure 36. Magnitude of $T(\theta)$ at distinct azimuthal angles (*Case 1*), N=7.

The optimal arrays for *Case 2* are plotted in Figure 37. The positions and sidelobe levels for this case are presented in Table XXIII. The optimal weights are listed in Table XXIV. The arrays for this case are similar to the results for *Case 1* except they are spread out farther, which is expected for a narrower mainbeam. The results for N=6 differ significantly between the two cases, indicating that the results can have significant variance depending on the beamwidth. The results for *Case 2* for N=6 and N=7 are almost identical, the difference being the center element. Note that the addition of this

element only lowers the sidelobe level by 0.1 dB. This information would be advantageous to an array designer in determining the number of elements needed to achieve a sidelobe level. The extra complexity introduced by adding a seventh element in this case would not be very beneficial. The magnitude of the array factor at distinct azimuthal angles is plotted in Figure 38 for the N=7 array. The mainbeam is again identical when $\theta \leq 15°$ for distinct azimuth angles, indicating a circular mainbeam.

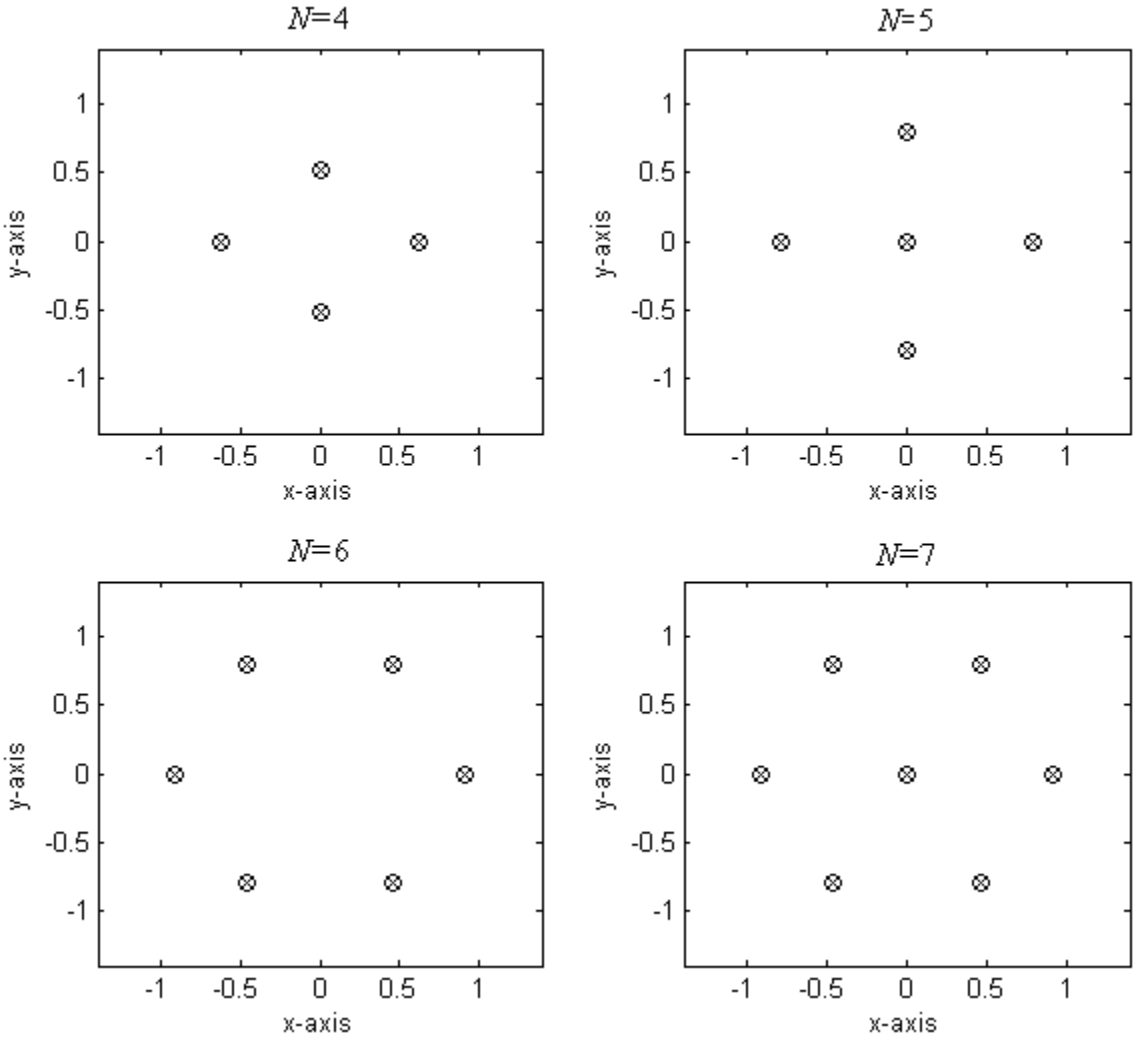


Figure 37. Optimal symmetric array locations for *Case 2* (dimensions in λ).

TABLE XXIII
OPTIMAL SLL AND POSITIONS FOR *CASE 2* (DIMENSIONS IN λ)

	(x_1, y_1)	(x_2, y_2)	(x_3, y_3)	(x_4, y_4)	SLL (dB)
N=4	(0.49, 0)	(0, 0.70)			-1.9
N=5	(0, 0)	(0.79, 0)	(0, 0.79)		-3.2
N=6	(0.92, 0)	(0.46, 0.80)	(-0.46, 0.80)		-5.7
N=7	(0, 0)	(0.92, 0)	(0.46, 0.80)	(-0.46, 0.80)	-5.8

TABLE XXIV
OPTIMAL WEIGHTS FOR *CASE 2*

	w_1	w_2	w_3	w_4
N=4	0.661	0.339		
N=5	0.268	0.183	0.183	
N=6	0.442	0.159	0.400	
N=7	0.200	0.133	0.133	0.133

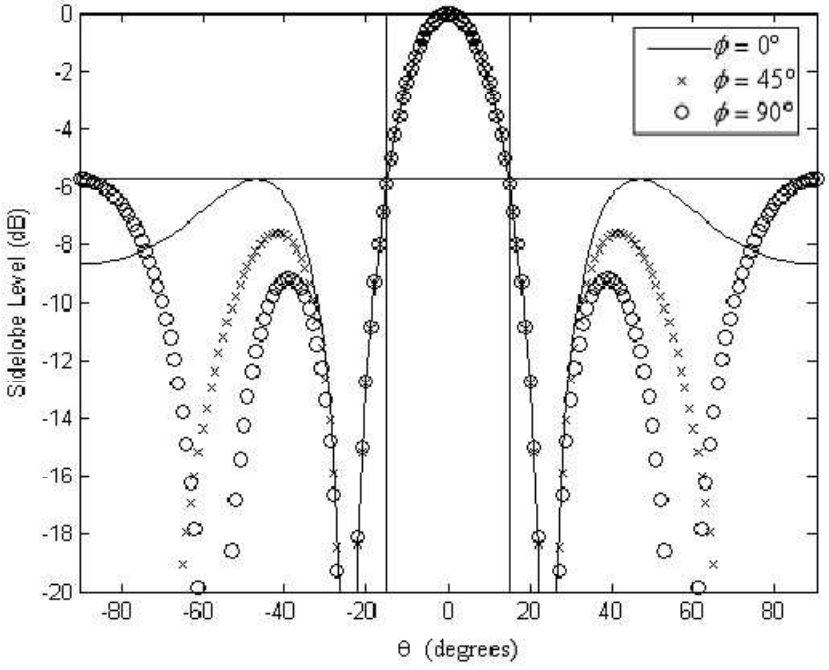


Figure 38. Magnitude of $T(\theta)$ at distinct azimuthal angles (*Case 2*), *N*=7.

7.6. Symmetric Arrays of Patch Antennas

In this section, symmetric arrays of patch antennas steered to $\theta_d = 0$ are considered. The method of solution is identical to that in Section 7.3. When a microstrip or patch antenna has a thin dielectric, the far field components of the electric field are approximately given by (7.15) and (7.16), when the polar angle $\theta \le \pi/2$ [91]. For $\theta > \pi/2$, which is the region below the patch, the radiated fields will assumed to be zero. In (7.9-7.10), k is the free-space wavenumber, W is the width of the patch, and L is the length of the patch.

$$E_\theta = E_0 \frac{\sin\left(\frac{kW\sin\theta\sin\phi}{2}\right)}{\frac{kW\sin\theta\sin\phi}{2}} \cos\left(\frac{kL}{2}\sin\theta\cos\phi\right)\cos\phi \tag{7.15}$$

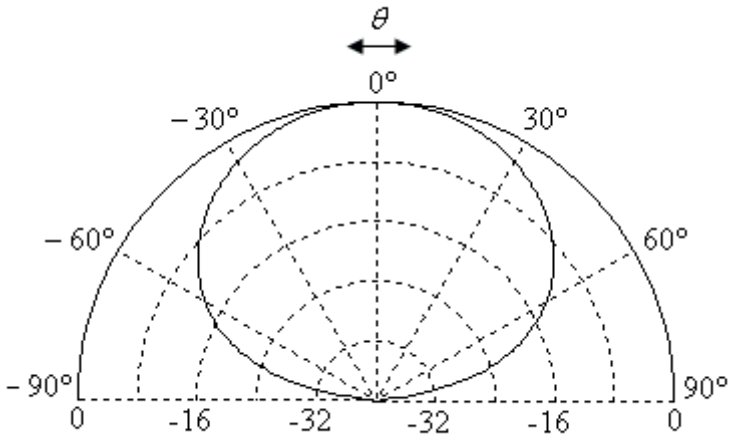
$$E_\varphi = -E_0 \frac{\sin\left(\frac{kW\sin\theta\sin\phi}{2}\right)}{\frac{kW\sin\theta\sin\phi}{2}} \cos\left(\frac{kL}{2}\sin\theta\cos\phi\right)\cos\theta\sin\phi \tag{7.16}$$

The normalized pattern to be used for $f(\theta,\phi)$ as in (7.1) will be

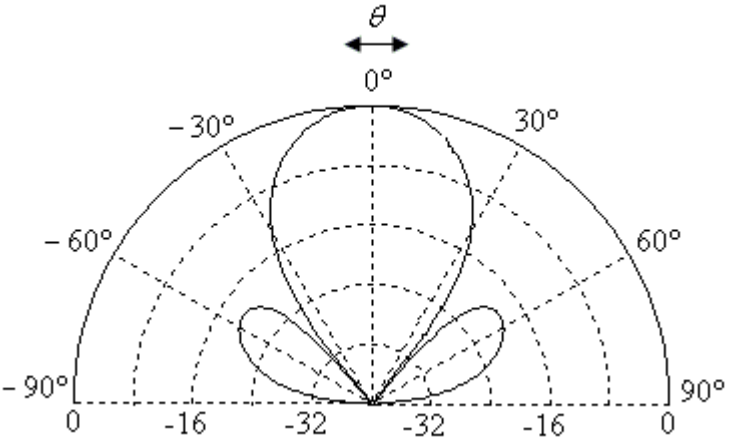
$$f(\theta,\phi) = \sqrt{\left(\frac{E_\theta}{E_0}\right)^2 + \left(\frac{E_\phi}{E_0}\right)^2} \tag{7.17}$$

In this section, the patch dimensions are chosen to be W=L=0.5 λ. The directivity for this antenna can be numerically calculated to be 9.34 dB. The patch pattern is plotted in Figure 39 for two fixed azimuth angles. The pattern here is complicated enough that it is highly unlikely that an analytical weighting method can be developed that minimizes the sidelobes in an patch array. However, using the development of Section 7.2, adding

the pattern does not significantly increase the difficulty of determining the optimal weights. Using a realistic, complicated antenna pattern helps to highlight the utility of the LP method of weight selection.



(a) $\phi = 0°$



(b) $\phi = 90°$

Figure 39. Magnitude of patch pattern (in dB).

The two cases discussed in Section 7.3 are again considered here. Using the solution method discussed previously, the optimal arrays for *Case 1* (beamwidth equal to $60°$) are presented in Figure 40. The optimal arrays of patch antennas are skewed

somewhat owing to the non-isotropy of the antenna pattern. The results for N=4, 5 and 7 are fairly similar to the *Case 1* results of omnidirectional elements; however they are slightly rotated and more spread out. The arrays are more spread out (or elongated) because the array factor effectively has a narrower main-beamwidth (due to the patch pattern which decreases in magnitude for $\theta > 0°$). The narrower mainbeam leads to more spread out arrays, as seen in *Case 2* of Section 7.3. The result for N=6 is a significant departure from the omnidirectional case, which indicates that the antenna pattern must be taken into account in determining an optimal geometry.

The positions and optimal sidelobe levels for *Case 1* of patch elements are listed in Table XXV. The corresponding optimal weights are listed in Table XXVI. The magnitude of the radiation pattern is plotted in Figure 41 as a function of θ for three distinct azimuth angles for the N=7 array. The directivity of this array is evaluated numerically to be 17.67 dB.

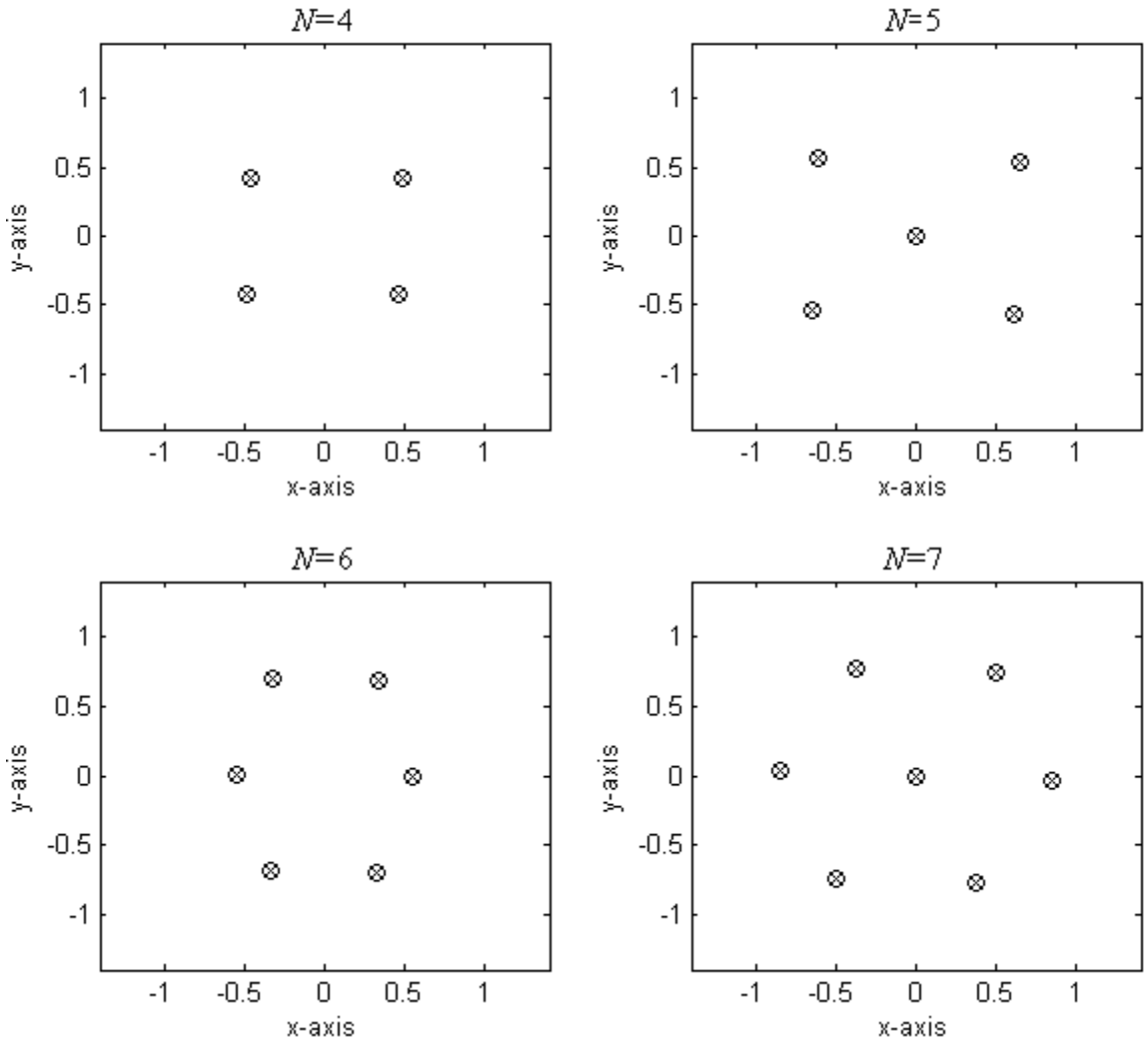


Figure 40. Optimal symmetric patch array locations for *Case 1* (units of λ).

TABLE XXV
OPTIMAL SLL AND POSITIONS FOR *CASE 1* OF PATCH ELEMENTS (UNITS OF λ)

	(x_1, y_1)	(x_2, y_2)	(x_3, y_3)	(x_4, y_4)	SLL (dB)
N=4	(0.46,-0.42)	(0.49, 0.42)			-12.5
N=5	(0.00, 0.00)	(0.61, -0.57)	(0.65, .53)		-13.1
N=6	(0.34, 0.68)	(-0.32, .69)	(.54, -.01)		-16.6
N=7	(0.00, 0.00)	(0.50, 0.74)	(0.85, -.04)	(.37, -.77)	-21.5

TABLE XXVI
OPTIMAL WEIGHTS FOR *CASE 1* WITH PATCH ELEMENTS

	w_1	w_2	w_3	w_4
N=4	0.256	0.244		
N=5	0.233	0.193	0.191	
N=6	0.133	0.133	0.234	
N=7	0.218	0.124	0.132	0.135

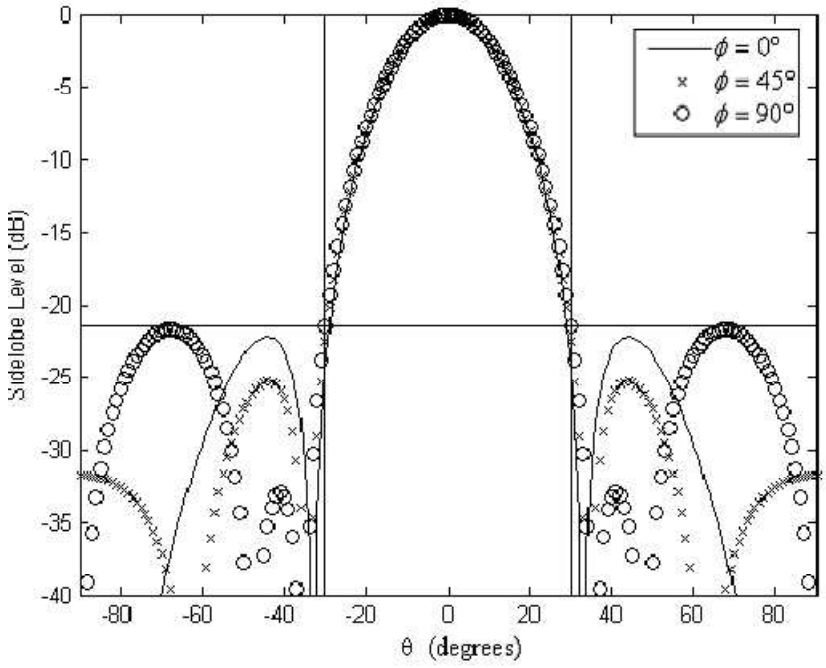


Figure 41. Magnitude of $T(\theta)$ at distinct azimuth angles (*Case 1*), N=7 (patch).

Next, *Case 2* (beamwidth equal to $30°$) is considered with patch elements. The optimal arrays are plotted in Figure 42. The arrays for this case are fairly similar to the results of *Case 1*, except for again being spread out farther. The result for N=6 is more symmetric and less football shaped, while the result for N=7 is again a hexagonally sampled array.

The optimal positions along with the sidelobe levels are listed in Table XXVII. The sidelobe level increased on average by 7.5 dB when compared to *Case 1* of Section 7.4. The optimal weights are given in Table XXVIII. The magnitude of the radiation pattern is plotted as a function of θ for distinct azimuth angles in Figure 43 for the N=7 array. The directivity of this array is evaluated numerically to be 17.72 dB.

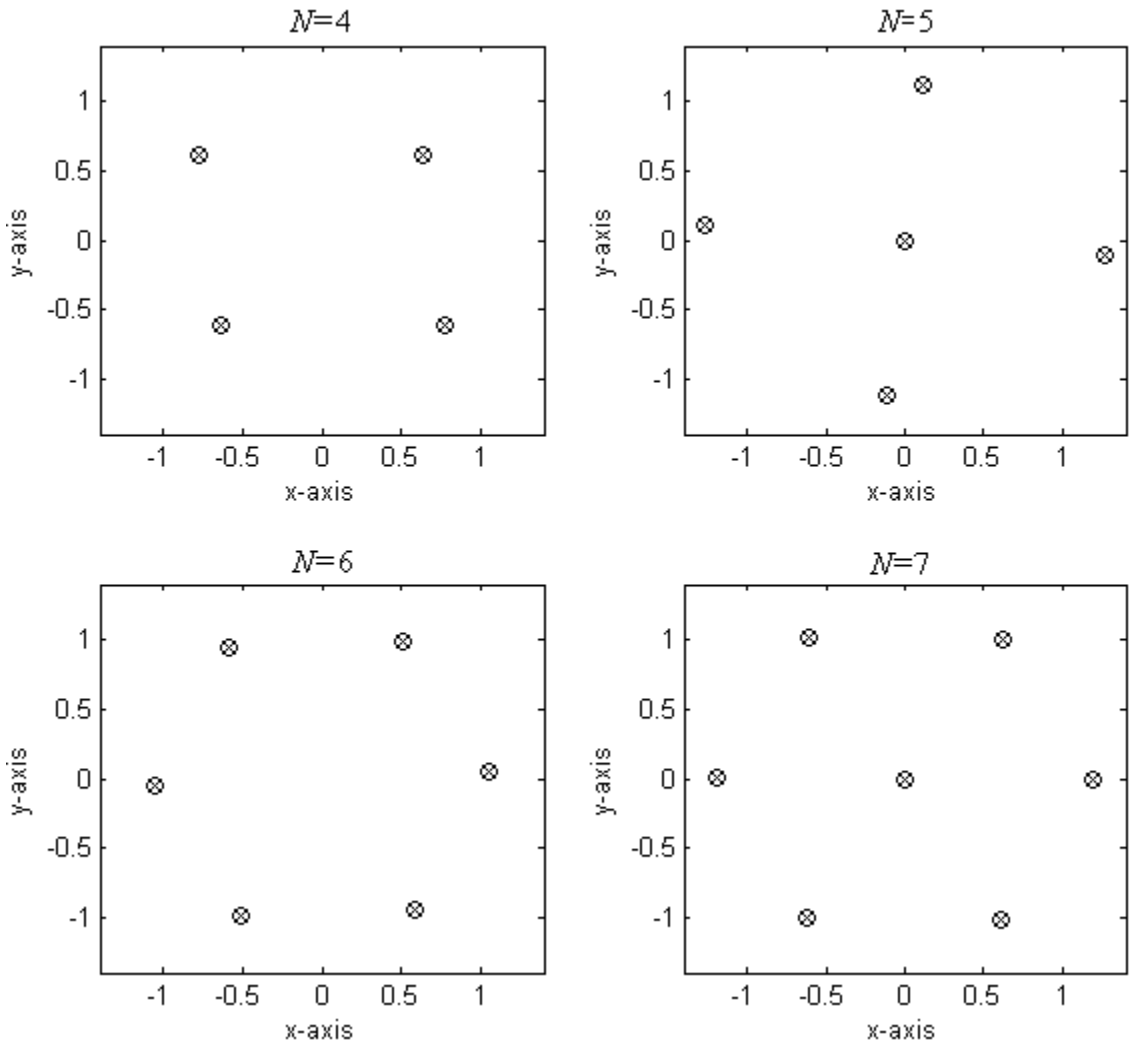


Figure 42. Optimal symmetric patch array locations for *Case 2* (units of λ).

TABLE XXVII
OPTIMAL SLL AND POSITIONS FOR *CASE 2* OF PATCH ELEMENTS (UNITS OF λ)

	(x_1, y_1)	(x_2, y_2)	(x_3, y_3)	(x_4, y_4)	SLL (dB)
N=4	(0.64, 0.61)	(0.77, -0.61)			-5.7
N=5	(0.00, 0.00)	(1.27, -0.11)	(0.11, 1.12)		-7.5
N=6	(1.05, 0.05)	(0.58, -0.94)	(0.51, 0.99)		-9.6
N=7	(0.00, 0.00)	(0.60, -1.02)	(0.63, 1.01)	(1.20, -0.01)	-10.7

TABLE XXVIII
OPTIMAL WEIGHTS FOR *CASE 2* WITH PATCH ELEMENTS

	w_1	w_2	w_3	w_4
N=4	0.272	0.228		
N=5	0.194	0.184	0.219	
N=6	0.178	0.163	0.159	
N=7	0.061	0.163	0.161	0.145

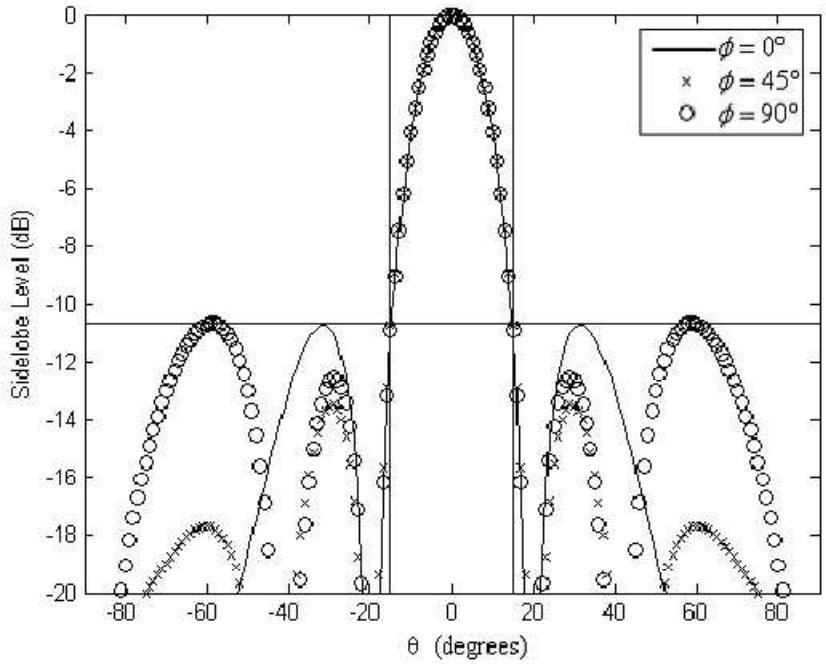


Figure 43. Magnitude of $T(\theta)$ at distinct azimuth angles (*Case 2*), *N*=7 (patch).

7.7. Wideband Weighting Method

The previous efforts have been focused on optimizing an antenna array for minimizing sidelobes at a single frequency. In practice, arrays transmit and receive information over a range of frequencies, generally centered about some carrier frequency. An array that has low sidelobes at a given frequency is not guaranteed to have low sidelobes at other frequencies within the band the array is operating in. In this section, a method of choosing weights that yield the minimum sidelobes over a range of frequencies is developed.

The minimum sidelobe level in a wideband array is written as

$$SLL = \max_{(k_x, k_y, \lambda) \in \Theta} |T(\mathbf{w}, \mathbf{d}, k_x, k_y, \lambda)|, \tag{7.18}$$

where again Θ is the region in which the sidelobes are to be suppressed. It will be assumed that the sidelobes are to be suppressed within a continuous frequency range, and in the region given by $\theta \geq \theta_c$ for all frequencies. This interval will be written as

$$[f_L, f_U] = \left[\frac{c}{\lambda_L}, \frac{c}{\lambda_U}\right], \tag{7.19}$$

where $f_L < f_U$.

The weighting method used in Section 7.3 could be extended such that every constraint is duplicated at every frequency; this was the method proposed in [92]. However, to do so, particularly for 2D arrays or for very wideband arrays, would add an intractable number of constraints. While this would be technically correct, the optimization would often be computationally intractable. Hence, a more efficient method is developed.

Defining

$$\Theta(\lambda_i) = \left\{ (k_x, k_y) : \left(\frac{2\pi \sin\theta_c}{\lambda_i} \right)^2 \le k_x^2 + k_y^2 \le \left(\frac{2\pi}{\lambda_i} \right)^2 \right\}, \tag{7.20}$$

it follows that

$$\Theta = \bigcup_{\lambda_i} \Theta(\lambda_i), \tag{7.21}$$

where the union is over all wavelengths within $[\lambda_U, \lambda_L]$.

For the wideband case, the total radiation pattern is written as the product of the element pattern and the array factor,

$$T(\mathbf{w}, \mathbf{d}, k_x, k_y, \lambda) = f(k_x, k_y, \lambda) AF(\mathbf{w}, \mathbf{d}, k_x, k_y, \lambda). \tag{7.22}$$

The two-dimensional array factor is rewritten from (2.21) with the wavevector definitions in (2.7) as

$$AF(\mathbf{w}, \mathbf{d}, k_x, k_y) = \sum_{n=1}^{N} w_n e^{-j(k_x x_n + k_y y_n)}. \tag{7.23}$$

Equation (7.23) indicates that the array factor does not depend on λ when k_x and k_y are specified.

This is not true for the element pattern, $f(k_x, k_y, \lambda)$, which in general will not be independent of λ when k_x and k_y are specified. This can be seen from the patch element pattern of Section 7.3 given in (7.15-7.17), which cannot be written as only a function of two variables.

Most antennas will exhibit a notable change in radiation pattern over the band of operation. In this case, the variation of the antenna pattern with frequency should be

taken into account. To accomplish this, an auxiliary antenna pattern, $H(k_x, k_y)$ is defined as

$$H(k_x, k_y) = \max_{\lambda \in [\lambda_U, \lambda_L]} \left| f(k_x, k_y, \lambda) \right|. \tag{7.24}$$

This auxiliary antenna pattern is the maximum value of the antenna pattern evaluated at (k_x, k_y) over the frequency range of interest. Note that the maximum in (7.24) is taken only over the frequency range for which (k_x, k_y) is in the visible region. For instance, at the value $(k_x, k_y) = (2\pi / \lambda_U, 0)$, the only frequency that has this value in the visible region is $f = f_U$. For the narrowband or single frequency case, the auxiliary antenna pattern reduces to the antenna pattern at the frequency of interest.

Using (7.22) and (7.24), it follows that

$$| T(\mathbf{w}, \mathbf{d}, k_x, k_y, \lambda) | \leq H(k_x, k_y) \, | AF(\mathbf{w}, \mathbf{d}, k_x, k_y) | . \tag{7.25}$$

Hence, the total radiation pattern as a function of frequency can be minimized by minimizing the right hand side of (7.19), which is only a function of k_x and k_y. The minimization is performed over the region specified in (7.21). Letting

$$k_{cL} = \frac{2\pi \sin\theta_c}{\lambda_L}, \tag{7.26}$$

the suppression region for the wideband case can be illustrated graphically, as shown in Figure 44. The wideband case is equivalent to minimizing the sidelobes in the (k_x, k_y) plane beginning at the cutoff value for the lowest frequency (k_{cL}), and extending the region to the largest wavenumber in the visible region at the highest frequency $(2\pi / \lambda_U)$.

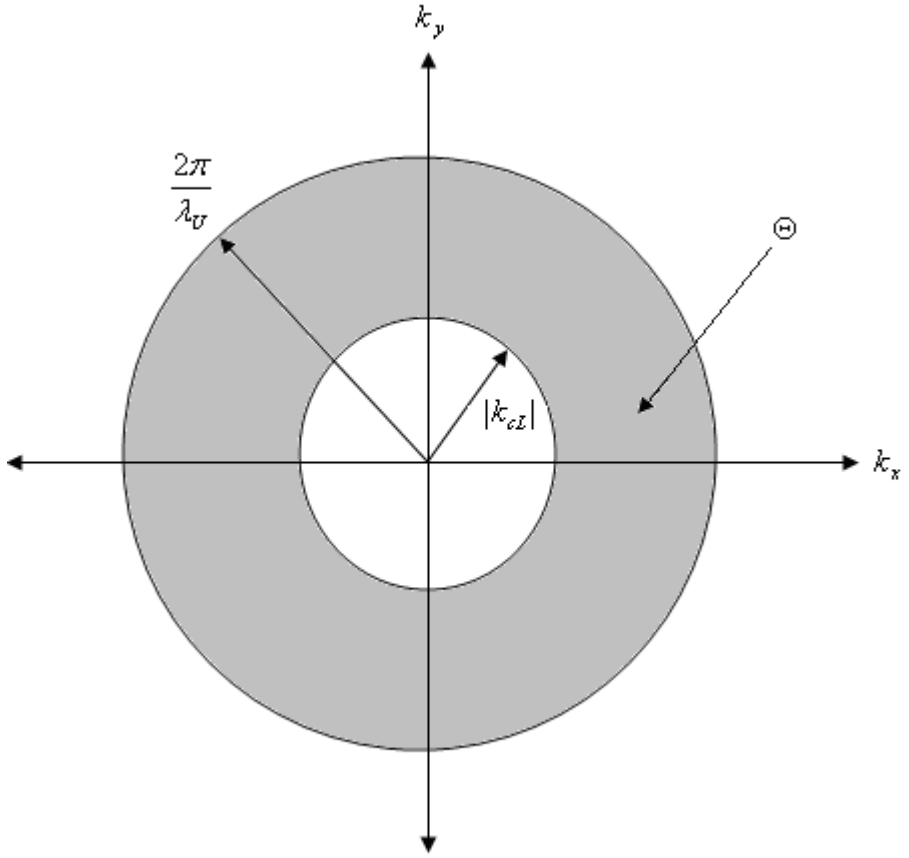


Figure 44. Suppression region for two-dimensional arrays over a frequency band.

Writing

$$H(k_x, k_y)AF(\mathbf{w}, \mathbf{d}, k_x, k_y) = G(\mathbf{w}, \mathbf{d}, k_x, k_y), \tag{7.27}$$

the single-frequency sidelobe minimizing optimization problem of (7.5) is rewritten for the wideband case in (7.28).

$$\begin{aligned} &\min \quad t \\ &s.t. \quad G(\mathbf{w}, \mathbf{D}, 0, 0) = 1 \\ &\qquad |G(\mathbf{w}, \mathbf{D}, k_{xi}, k_{yi})| \le t, \quad i = 1, 2, \ldots, R \end{aligned} \tag{7.28}$$

In (7.22), the R samples are sampled over the suppression region illustrated in Figure 44. The solution to (7.22) yields weights that produce the minimum sidelobe level over the frequency band of interest.

When the antenna elements do not have significantly different radiation patterns over the frequency band of interest, the radiation pattern can be approximated,

$$f(\theta,\phi,\lambda_i) \approx f(\theta,\phi,\lambda_j), \tag{7.29}$$

for all λ_i, λ_j within the frequency band. For this case, determining the wideband weights is as simple as extending the suppression region as in Figure 37 and using the procedure of Section 7.2.

7.8. Optimal Wideband Arrays of Omnidirectional Antennas

In this section, the arrays are optimized to determine the minimum sidelobe level over a range of frequencies. The frequency range will be specified by the fractional bandwidth (*FBW*),

$$FBW = \frac{\Delta f}{f_c} = \frac{f_U - f_L}{f_c} = \frac{f_U - f_L}{(f_U + f_L)/2}, \tag{7.30}$$

where f_c is the center frequency. Fractional bandwidths are considered wideband when $0.2<FBW<0.5$, and are considered ultra-wideband when $FBW \geq 0.5$ [93]. In this section, an ultra-wideband case is considered in which $FBW=0.50$. The antenna elements are omnidirectional and independent of frequency over the frequency range of interest, so that

$$f(\theta,\phi,\lambda) = 1. \tag{7.31}$$

When the antenna's radiation pattern is independent of frequency over the bandwidth of interest, the antennas are referred to as *frequency independent*. Hence, the optimization in this section will focus on the array factor, which is equivalent to the total radiation

pattern for this case. For comparison with the results of Section 7.4, the beamwidth will be $60°$, so that the sidelobes will be suppressed when $\theta \geq 30°$ for all frequencies.

The optimization procedures that were applied in the previous sections of this chapter are again sufficient for the problem at hand. The resulting optimal arrays are found for *N*=4-7 and are presented in Figure 45. Note that the results are now given in units of $\lambda_c = c / f_c$. The optimal positions are also tabulated in Table XXIX. The corresponding optimal weights are listed in Table XXX.

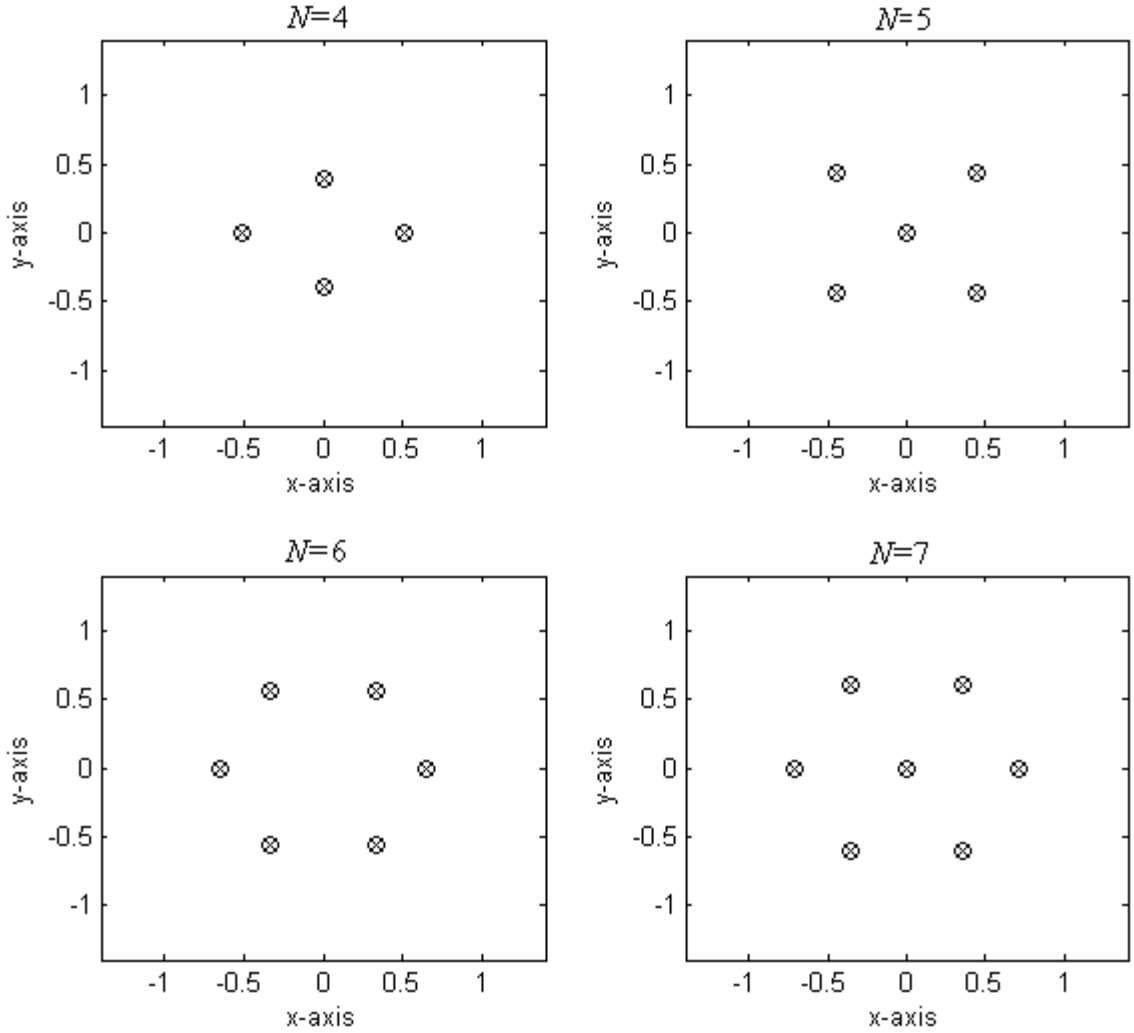


Figure 45. Optimal symmetric array locations for *FBW*=0.5 (units of λ_c).

TABLE XXIX
OPTIMAL SLL AND POSITIONS FOR OMNIDIRECTIONAL ELEMENTS (UNITS OF λ_c, FBW=0.5)

	(x_1, y_1)	(x_2, y_2)	(x_3, y_3)	(x_4, y_4)	SLL (dB)
N=4	(0.51, 0)	(0, 0.39)			-2.4
N=5	(0.0, 0.0)	(0.44, -.44)	(0.44, 0.44)		-3.9
N=6	(0.65, 0.0)	(0.33, 0.56)	(-0.33, 0.56)		-6.0
N=7	(0.0, 0.0)	(0.71, 0)	(0.36, .61)	(-.36, 0.61)	-7.2

TABLE XXX
OPTIMAL WEIGHTS FOR OMNIDIRECTIONAL ELEMENTS (FBW=0.5)

	w_1	w_2	w_3	w_4
N=4	0.189	0.311		
N=5	0.183	0.204	0.204	
N=6	0.169	0.169	0.169	
N=7	0.043	0.160	0.160	0.160

The results are interesting when compared with the narrowband results of Section 7.4. The *SLL* increased on average of 3.6 dB when the array is designed to perform in this ultra-wideband situation. The arrays are slightly less spread out as in the narrowband case. The result for N=6 is a circular array in the wideband case, whereas it was a cross shape for the narrowband case. In addition, in extending the array from narrowband to ultra-wideband, the *SLL* increased by only 1.9 dB for N=6. This is not a large penalty in *SLL* for greatly extending the bandwidth. However, the *SLL* increase was 6.7 dB for N=7, which is relatively large.

The total radiation pattern for the optimal array of size N=7 is now presented. Since it is now a function of frequency, the pattern will be plotted as a function of θ for distinct azimuth angles at the lower frequency (f_L, given in Figure 46), the center

frequency (f_c , given in Figure 47), and the upper frequency (f_U , given in Figure 48). Note that the beamwidth varies depending on the frequency. However, for all frequencies in the range of interest, the beamwidth is less than $60°$ and the sidelobes never rise above the *SLL* (-7.2 dB) in the suppression region, as desired.

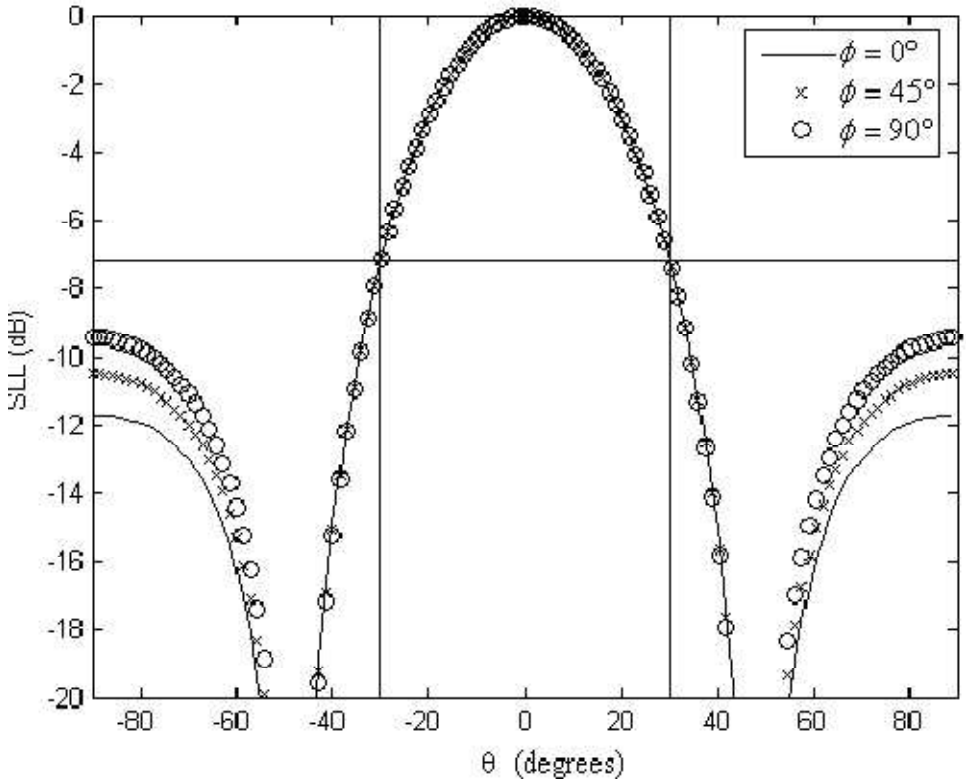


Figure 46. Magnitude of $T(\theta)$ at distinct azimuth angles (N=7) for $f = f_L$.

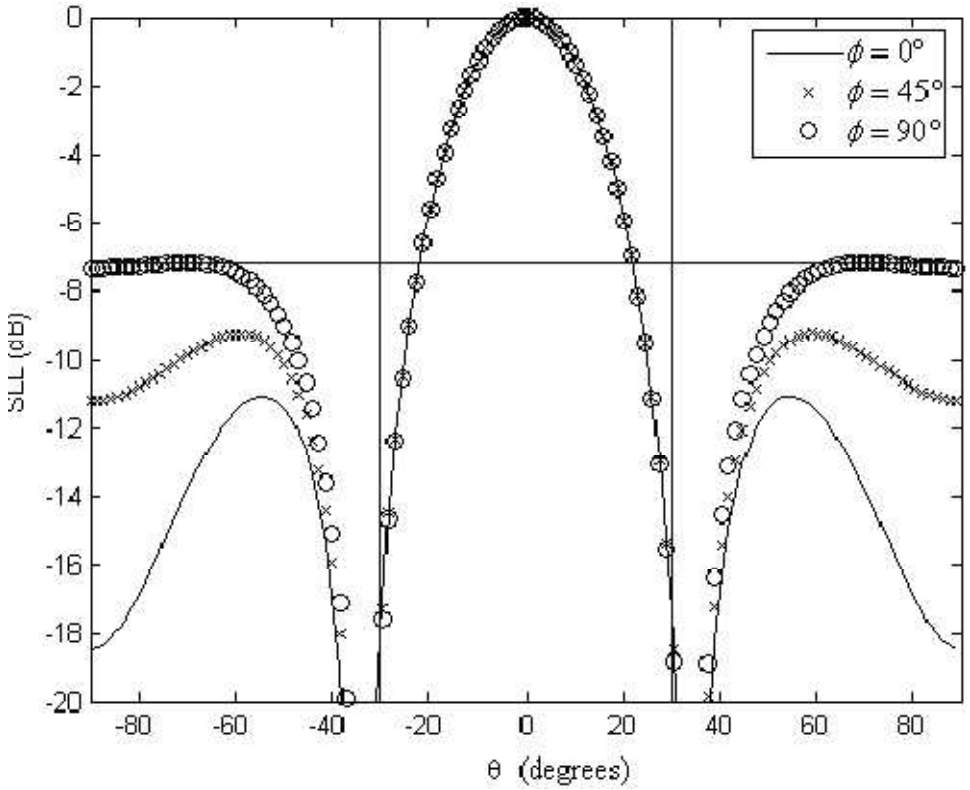


Figure 47. Magnitude of $T(\theta)$ at distinct azimuth angles (N=7) for $f = f_c$.

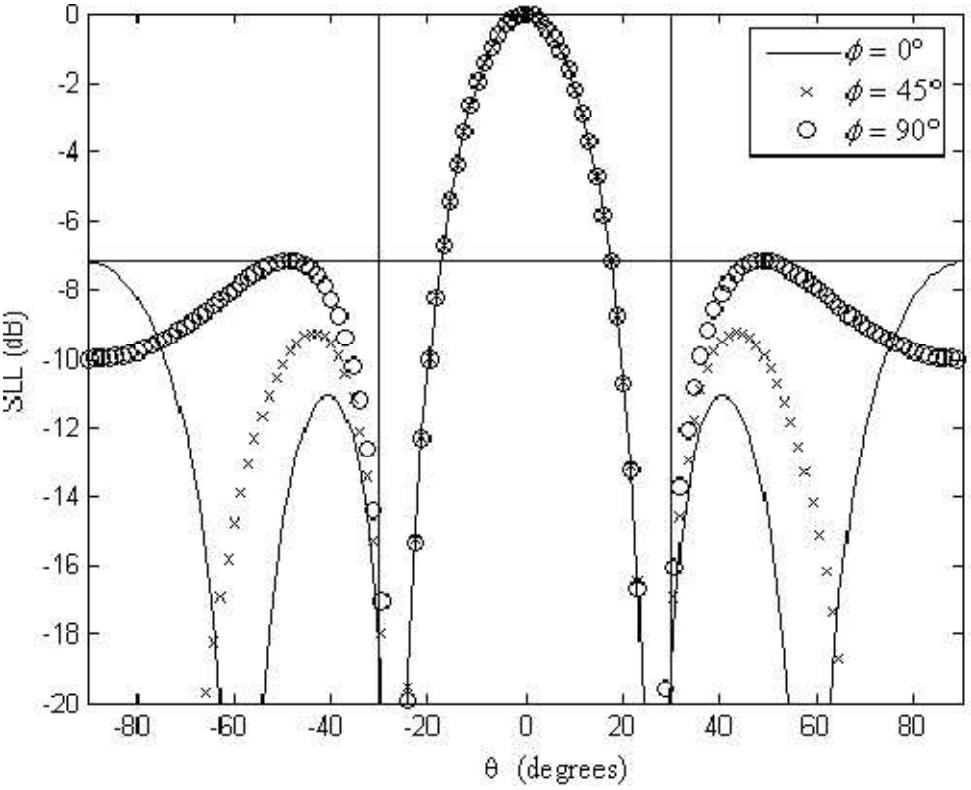


Figure 48. Magnitude of $T(\theta)$ at distinct azimuth angles (N=7) for $f = f_U$.

7.9. Optimal Wideband Arrays of Patch Antennas

In this section, wideband arrays of patch antennas are examined. The bandwidth is selected to be FBW=0.2, which is much smaller than the ultra-wideband case of Section 7.7 but still wideband enough that the narrowband assumption is not valid. Patch antennas have radiation patterns that vary significantly with frequency, but can be made to have a wider bandwidth using various methods including adding slits [94] or adding a U-slot to the patch [95]. The beamwidth will again be $60°$ for comparison with the results of Section 7.5.

The normalized field patterns for the patch, given in (7.9-7.11) will now be rewritten as a function of frequency in (7.32-7.33).

$$E_\theta(\lambda) = E_0 \frac{\sin\left(\frac{\pi W \sin\theta \sin\phi}{\lambda}\right)}{\frac{\pi W \sin\theta \sin\phi}{\lambda}} \cos\left(\frac{\pi L}{\lambda}\sin\theta\cos\phi\right)\cos\phi \tag{7.32}$$

$$E_\varphi(\lambda) = -E_0 \frac{\sin\left(\frac{\pi W \sin\theta \sin\phi}{\lambda}\right)}{\frac{\pi W \sin\theta \sin\phi}{\lambda}} \cos\left(\frac{\pi L}{\lambda}\sin\theta\cos\phi\right)\cos\theta\sin\phi \tag{7.33}$$

The normalized pattern to be used for $f(\theta,\phi)$ as in (7.1) will be

$$f(\theta,\phi,\lambda) = \sqrt{\left(\frac{E_\theta(\lambda)}{E_0}\right)^2 + \left(\frac{E_\phi(\lambda)}{E_0}\right)^2}\ . \tag{7.34}$$

The implicit assumption in (7.32-7.34) is that E_0 is approximately constant over the frequency range of interest.

The PSO algorithm is again applied to determine the optimal positions. The resulting optimal arrays are found for N=4-7 and are presented in Figure 49. The results are again given in units of $\lambda_c = c / f_c$. The optimal positions are also tabulated in Table XXXI. The corresponding optimal weights are listed in Table XXXII.

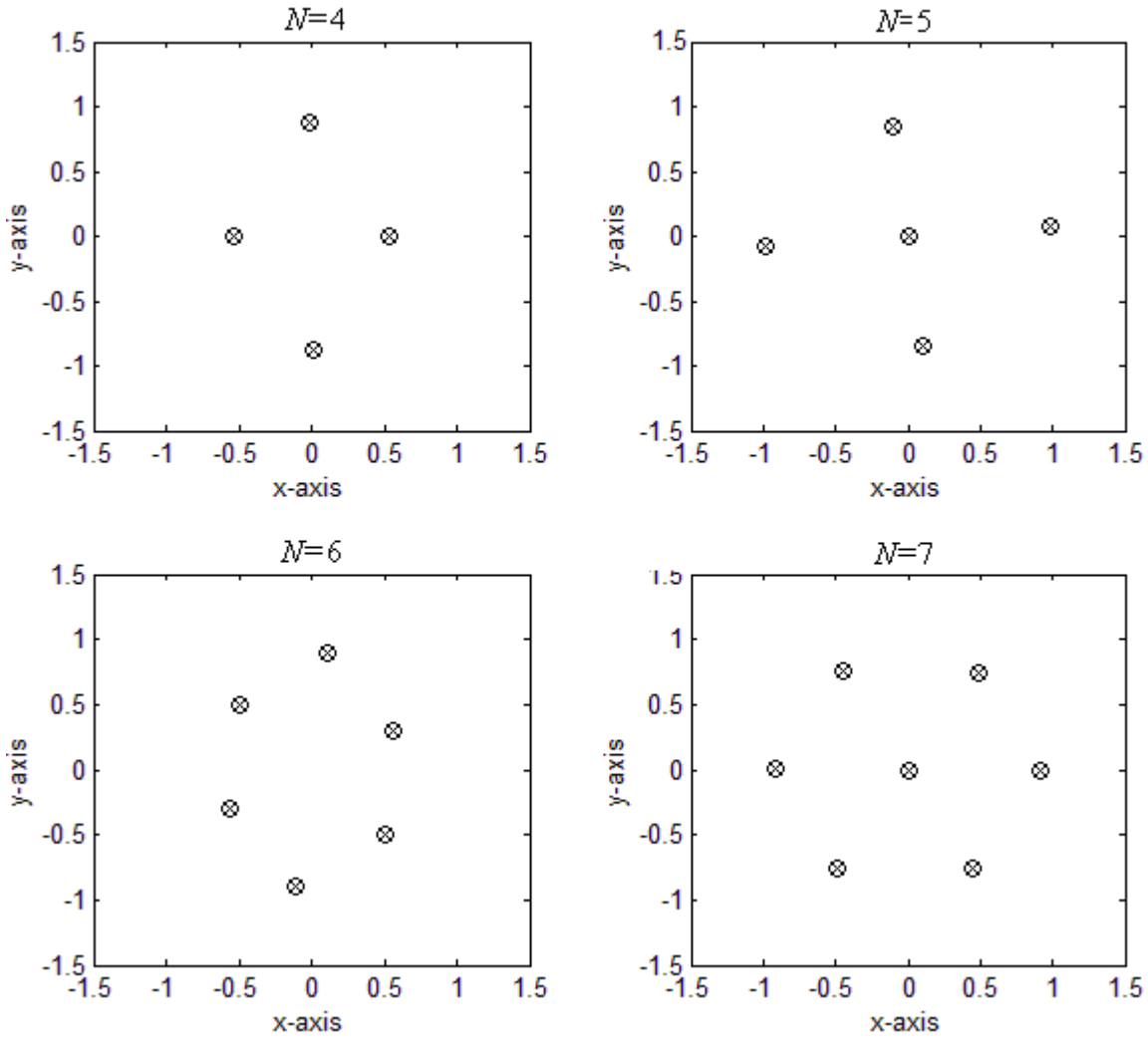


Figure 49. Optimal symmetric patch array locations for FBW=0.2 (units of λ_C).

The arrays are similar to the narrowband case of patch elements with the same bandwidth given in Figure 33. The seven element array is again approximately

hexagonal, which has been a recurring theme throughout this work. The arrays for this case appear to be spread out further than the narrowband case, when measured in units of the center wavelength.

On average, the *SLL* increased by 1.9 dB in order to guarantee the sidelobe level over the frequency range of operation. The *N*=5 element array exhibited the lowest rise in sidelobes (only 1.1 dB) by extending the bandwidth of the array. The *N*=7 element array exhibited the highest rise in sidelobes (3.0 dB) in order to extend the bandwidth.

TABLE XXXI
OPTIMAL SLL AND POSITIONS FOR PATCH ELMENTS (UNITS OF λ_c, *FBW*=0.2)

	(x_1, y_1)	(x_2, y_2)	(x_3, y_3)	(x_4, y_4)	SLL (dB)
N=4	(0.53, 0.00)	(.01, -0.87)			-10.8
N=5	(0.0, 0.0)	(0.99, 0.07)	(-0.10, 0.84)		-12.0
N=6	(0.56, 0.30)	(0.50, -0.49)	(0.11, 0.90)		-14.7
N=7	(0.0, 0.0)	(0.45, -0.76)	(0.49, 0.75)	(0.91, -0.01)	-18.5

TABLE XXXII
OPTIMAL WEIGHTS FOR PATCH ELEMENTS (*FBW*=0.2)

	w_1	w_2	w_3	w_4
N=4	0.320	0.180		
N=5	0.268	0.169	0.197	
N=6	0.215	0.185	0.100	
N=7	0.188	0.142	0.136	0.128

Finally, the total radiation patterns for the optimal *N*=7 arrays are again plotted at the lower, center and upper frequencies for fixed elevation angles. The radiation pattern at the lower frequency ($f = f_L$) is plotted in Figure 50, the center frequency ($f = f_c$) radiation pattern is given in Figure 51, and the upper frequency ($f = f_U$) radiation

pattern is plotted in Figure 52. As seen in the ultra-wideband case, the beamwidth again varies depending on the frequency. However, the variance is less pronounced in this case because of the lower fractional bandwidth considered.

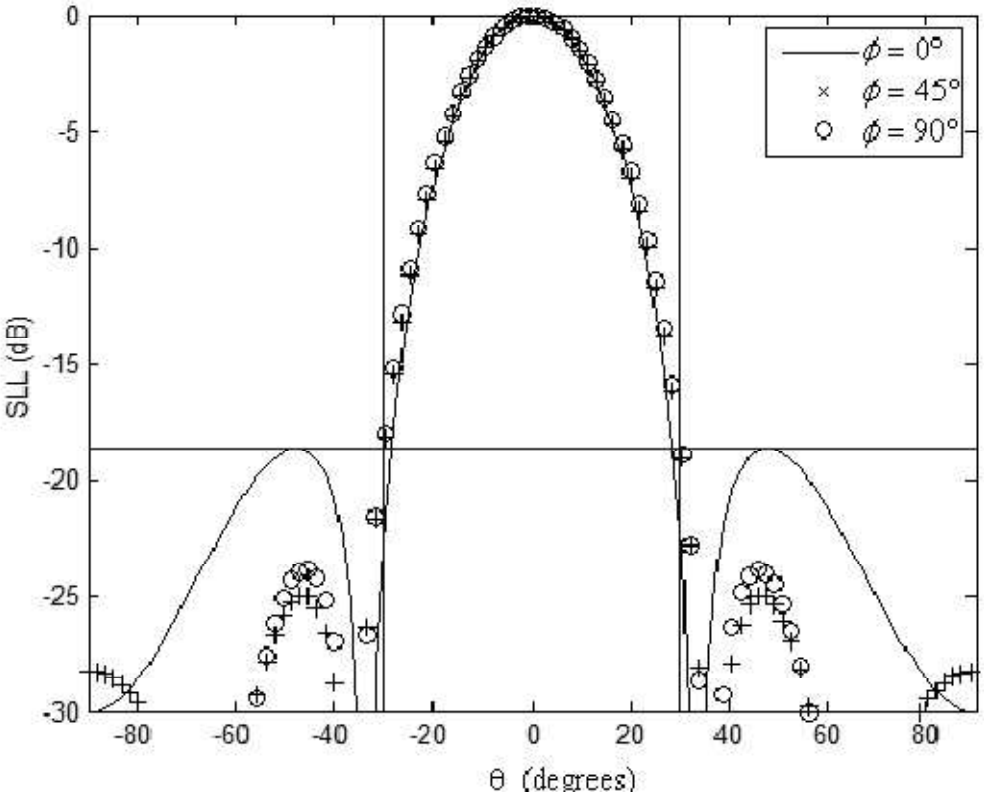


Figure 50. Magnitude of $T(\theta)$ at distinct azimuth angles (N=7) for $f = f_L$.

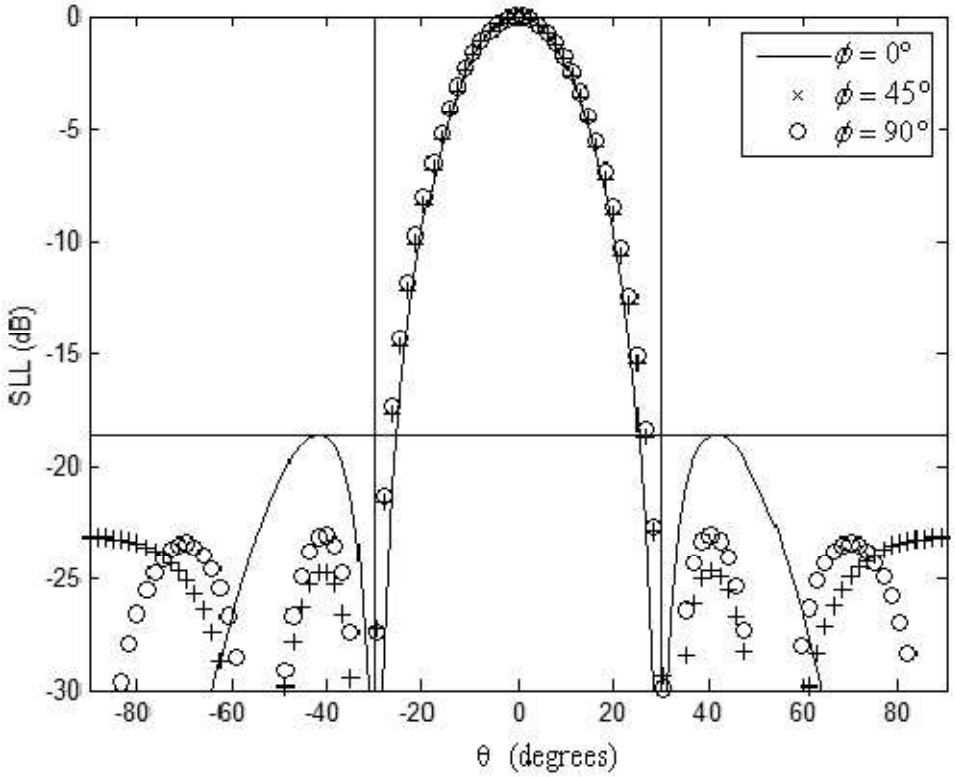


Figure 51. Magnitude of $T(\theta)$ at distinct azimuth angles (N=7) for $f = f_c$.

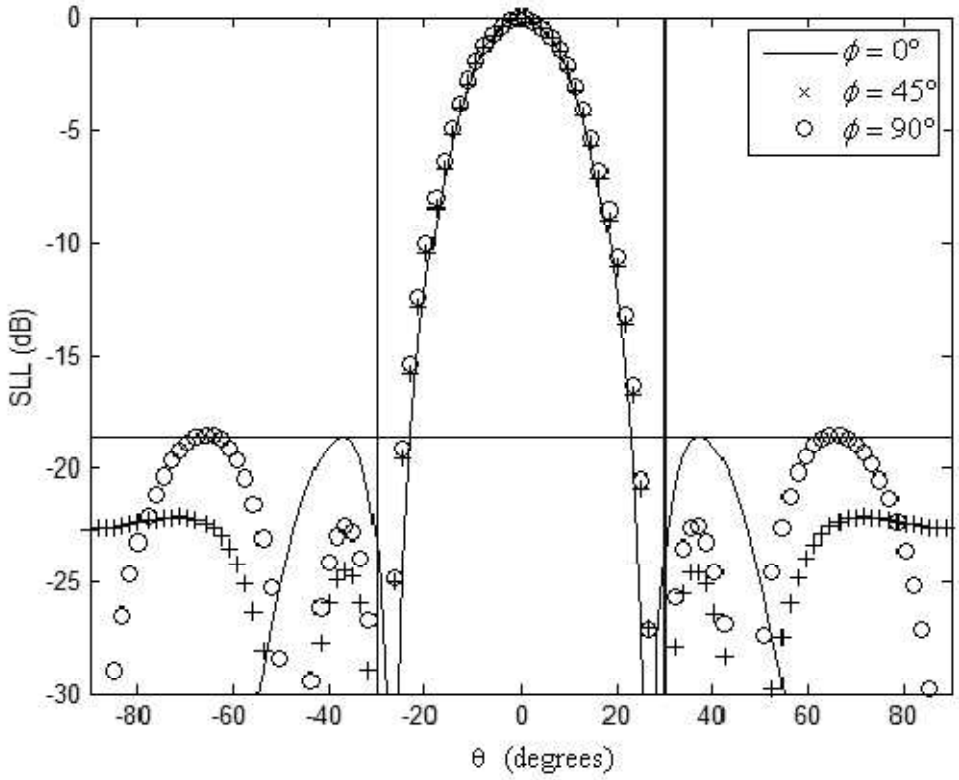


Figure 52. Magnitude of $T(\theta)$ at distinct azimuth angles (N=7) for $f = f_U$.

7.10. Conclusions

The sidelobe minimization technique was extended from 1D to 2D in this chapter. It was seen that the optimal geometry varies depending on the beamwidth and antenna elements used in the array.

The arrays were then studied over a wide frequency range. Optimal weights were derived that minimize the sidelobe level over a range of frequencies. The optimal geometries and weights were found to vary with the fractional bandwidth used by the array. Hence, if an array is used over a frequency range, the narrowband optimization technique will not be optimal.

The PSO method was again effective in optimizing the array geometry. This optimization technique has proven to work well for a wide variety of problems. In

addition, the method lends itself well to be employed using parallel processing, which can significantly speed up computation time.

VIII. SUMMARY, CONCLUSIONS, AND FUTURE WORK

8.1. Summary and Conclusions

The primary goal of this dissertation has been to show the effect of an array's geometry on metrics of interest. These metrics include sidelobe level, interference-rejection and SINR. Because of the difficulty in analyzing an array's geometry, the geometry is often chosen to be a standard geometry in practice. However, this work has shown that gain in performance can be achieved via suitable optimization of the array geometry.

The secondary goal has been to improve upon existing weight-selection strategies via convex optimization. The relatively new field of convex optimization will likely be a tremendously effective tool as it makes its way into the antenna field.

In Chapter 5, improving the performance of an adaptive array was considered. Because of the wide range of environments in which these arrays operate, the concept of an interference environment was introduced. In this manner, the performance can be optimized on average over the likely scenarios in which the array is to perform. This was a necessary development, as optimizing over a specific situation would not have been extremely useful for an adaptive array. The interference-suppression capability of adaptive arrays was shown to vary with geometry, and consequently, the optimization of the geometry was of interest. It was shown that the interference allowed through the spatial-filter (that is the antenna array) can be lowered by varying the geometry. In addition, this lowering of interference power was then shown to often translate into gains in overall SINR, a critical parameter in wireless communication.

Sidelobe level was shown to be critical in WCDMA systems in [87]. In Chapter 6, the process of determining the minimum possible *SLL* in a linear array was developed. Sidelobe-minimizing weights were derived that can be efficiently computed for any linear geometry, beamwidth, scan angle, and antenna type. The Dolph-Chebyshev sidelobe-minimizing weighting method was derived in 1946 [21] and has been used extensively since its publication. The derivation of the weights presented in Chapter 6 was a significant expansion of the capabilities of that method. The total radiation pattern depends on both the weights, positions and elements in the array. Since the optimal weights can be found for any array geometry and any antenna type, the only variables remaining were the array positions. The PSO algorithm was employed to determine optimal positions, that along with the optimal weights, determined the optimal sidelobe levels for linear arrays of size N=2-7. Results were presented for linear arrays steered to broadside and $45°$, and for two different beamwidths. In addition, arrays of omnidirectional elements and short dipoles were examined to show the effect of the antenna's radiation pattern on the optimal geometry, weights and sidelobe level.

In Chapter 7, the optimal sidelobe level for 2D or planar arrays was considered. The methods for minimizing sidelobes in linear arrays were extended to the planar case. Optimal symmetric planar arrays of size N=4-7 were found for two beamwidths, and for arrays with either omnidirectional or patch antenna elements. In addition, a method of minimizing sidelobes in wideband planar arrays was developed. Sidelobe minimizing weights were derived that suppress the sidelobes over a range of frequencies, instead of at a single frequency as is done in the narrowband case. This weighting method is valid for

arbitrary bandwidths, beamwidths, antenna types, and planar array geometries. The positions were optimized simultaneously with the weights to determine optimal sidelobe levels for wideband arrays. Results were presented for an ultra-wideband case of omnidirectional elements, and for a wideband case of patch antenna elements.

Throughout this work, the hexagonal array has been a recurring optimal two-dimensional array. For interference suppression, the optimal 7-element array was a closely spaced hexagonal array. For sidelobe suppression in the planar case, the hexagonal array arose as the solution for distinct element types and beamwidths. Hence, when using an array with a number of elements that fits well with the hexagonal structure, it is likely that this geometry would be a good starting point.

As the traffic in wireless communication increases, every variable that can be exploited to improve performance will be optimized. Since the demand for higher data rates and reliability for a given bandwidth continues to grow exponentially, it is likely array geometry optimization will be employed in real systems.

8.2. Future Work

There is no shortage of applications in which array geometry optimization would prove useful. The obvious next steps would be to continue the work of this dissertation for arrays with a larger number of elements. The minimum sidelobe-producing antenna arrays for one and two dimensions could be studied for increasing number of elements to determine the characteristics of the optimal arrays as the number of elements becomes large. The same extensions could be done to the interference-suppressing adaptive arrays.

Another topic of interest would be to optimize the weights and geometries of antennas consisting of non-identical elements. Antenna array analysis is almost exclusively performed with identical elements, and it would be interesting to observe if gains could be made by exploiting elements with different radiation patterns.

Another interesting practical problem would be to minimize cross-polarization in antenna arrays while holding a certain criteria constant (*SLL*, MSE, etc.). This problem could likely be solved in a similar manner to the solution methods of Chapters 6 and 7. Implementing precise weights can sometimes be difficult in actual systems. Hence, deriving an optimization problem that returns weights from a discrete set of allowable weights would be advantageous. Then optimizing over the positions to determine an optimal geometry for the discretized weights could be performed.

The geometry-optimization in this work has focused on translating the elements. For non-omnidirectional elements, the array could be optimized by allowing the elements to rotate or be put at an angle relative to the other elements. This would add new degrees of freedom to each element, which could translate to potentially large gains in performance.

On the theoretical side, optimization methods for proving an array's geometry is globally optimal would be of value. This has not been done due to the mathematical intractability of the problem (many locally optimal points). However, as the field of optimization expands, it is possible that a clever technique could be developed to verify that an array is globally optimal.

Finally, in digital communications, the bit error rate (BER) for a given data rate is the definitive measure of performance for a wireless communication system. Hence, more general modeling methods that ultimately minimize the BER would be valuable. However, because of the large and complex nature of wireless communication systems, this would not be an easy task.

REFERENCES

[1] L. Coe, *Wireless Radio: A History*. New York: McFarland & Company, 2006.

[2] L. W. Alvarez, *Alvarez: Adventures of a Physicist*. New York: Basic Books, 1987.

[3] H. Unz, "Linear arrays with arbitrarily distributed elements," *IEEE Trans. Antennas Propag.*, vol. 8, pp. 222-223, Mar. 1960.

[4] D. King, R. Packard and R. Thomas, "Unequally spaced, broad-band antenna arrays," *IEEE Trans. Antennas Propag.*, vol. 8, pp. 380-384, Jul. 1960.

[5] R. Harrington, "Sidelobe Reduction by Nonuniform Element Spacing," *IEEE Trans. Antennas and Propag.*, vol. 9, pp. 187-192, Mar. 1961.

[6] M. Skolnik, G. Nemhauser, and J. Sherman, "Dynamic programming applied to unequally spaced arrays," *IEEE Trans. Antennas and Propag.*, vol. 12, pp. 35-43, Jan. 1964.

[7] M. Skolnik, J. Sherman and F. Ogg, Jr., "Statistically designed density-tapered arrays," *IEEE Trans. Antennas and Propag.*, vol. 12, pp. 408-417, July 1964.

[8] W. L. Stutzman, "Shaped-beam synthesis of nonuniformly spaced linear arrays," *IEEE Trans. Antennas Propagat.*, vol. AP-20, pp. 499-501, July 1972.

[9] S. U. Pillai, Y. Bar-Ness, and F. Haber, "A new approach to array geometry for improved spatial spectrum estimation," *IEEE Proc.*, vol. 73, pp. 1522-1524, Oct. 1985.

[10] M. Gavish and A. J. Weiss, "Array geometry for ambiguity resolution in direction finding," *IEEE Trans. Antennas and Propag.*, vol. 44, pp. 889-895, June 1996.

[11] C. W. Ang, C. M. See, and A. C. Kot, "Optimization of array geometry for identifiable high resolution parameter estimation in sensor array signal processing," in *Proc. Int. Conf. Inf., Commun. Signal Process.*, Singapore, Sep. 1997, pp. 1613-1617.

[12] A. Goldsmith, *Wireless Communications*. New York: Cambridge University Press, 2005.

[13] B. P. Kumar and G. R. Branner, "Design of unequally spaced arrays for performance improvement," *IEEE Trans. Antennas Propag.*, vol. 47, pp. 511-523, Mar. 1999.

[14] B. P. Kumar and G. R. Branner, "Generalized Analytical Technique for the synthesis of unequally spaced arrays with linear, planar, cylindrical or spherical geometry," *IEEE Trans. Antennas Propag.*, vol. 53, pp. 621-633, Feb. 2005.

[15] P. Jarske, T. Sramaki, S. K. Mitra, and Y. Neuvo, "On the properties and design of nonuniformly spaced linear arrays," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 372-380, Mar. 1988.

[16] T. H. Ismail and M. M. Dawoud, "Null steering in phased arrays by controlling the element positions," *IEEE Trans. Antennas Propag.*, vol. 39, pp. 1561-1566, Nov. 1991.

[17] M. M. Khodier and C. G. Christodoulou, "Sidelobe Level and Null Control Using Particle Swarm Optimization," *IEEE Trans. Antennas Propag.*, vol. 53, pp. 2674-2679, Aug. 2005.

[18] N. Petrella, *et. al.*, "Planar array synthesis with minimum sidelobe level and null control using particle swarm optimization," *Int. Conf. Microwaves, Radar, Wireless Comm.*, pp. 1087-1090, May, 2006.

[19] A. Tennant, M. M. Dawoud, and A. P. Anderson, "Array pattern nulling by element position perturbations using a genetic algorithm," *Electron. Lett.*, vol. 30, no. 3, pp. 174-176, Feb. 1994.

[20] O. Quevedo-Teruel and E. Rajo-Iglesias, "Ant colony optimization in thinned array synthesis with minimum sidelobe level," *IEEE Trans. Antennas Propag. Letters*, vol. 5, pp. 349-352, 2006.

[21] C. L. Dolph, "A current distribution for broadside arrays which optimizes the relationship between beamwidth and side-lobe level," *Proc. IRE*, vol. 34, pp. 335-348, June 1946.

[22] C. A. Balanis, *Antenna Theory: Analysis and Design, 3rd ed.* New York: Wiley, 2005.

[23] H. J. Riblet, "Discussion of Dolph's paper," *Proc. IRE*, vol. 35, pp. 489-492, May 1947.

[24] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing, 3rd Ed.* New Jersey: Prentice Hall, 1996.

[25] R. H. DuHamel, "Optimum patterns for endfire arrays," *Proc. IRE*, vol. 41, pp. 652-659.

[26] G. Sinclair and F. V. Cairns, "Optimum patterns for arrays of non-isotropic sources," *Trans. IRE*, PGAP-1, pp. 50-61, Feb. 1952.

[27] S. Holm and B. Elgetun, "Optimization of the beampattern of 2D sparse arrays by weighting," *Proc. IEEE Ultrasonics Symp.*, Cannes, France, 1995.

[28] V. Murino, A. Trucco and C. S. Regazzoni, "Synthesis of unequally spaced arrays by simulated annealing," *IEEE Trans. Signal Processing*, vol. 44, pp. 119-123, Jan. 1996.

[29] B. Widrow, P. E. Mantey, and L. J. Griffiths, "Adaptive antenna systems," *Proc. IEEE*, vol. 55, pp. 2143-2159, Dec. 1967.

[30] P. Bevelacqua and C. A. Balanis, "Optimizing antenna array geometry for interference suppression", *IEEE Trans. Antennas Propag.*, vol. 53, pp. 637-641, Mar. 2007.

[31] A. A. Abouda, H. M. El-Sallabi and S. G. Haggman, "Effect of antenna array geometry and ULA azimuthal orientation on MIMO channel properties in urban city street grid," *Progress in Electromagnetics Research*, PIER 64, pp. 257-278, 2006.

[32] X. Li and Z. Nie, "Effect of array orientation on performance of MIMO wireless channels," *IEEE Antennas Propag. Letters*, vol. 3, pp. 368-372, 2004.

[33] F. T. Ulaby, *Fundamentals of Applied Electromagnetics*. New Jersey: Prentice Hall, 2001.

[34] C. A. Balanis, *Advanced Engineering Electromagnetics*. New York: Wiley, 1989.

[35] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design, 2nd Ed*. New York: Wiley, 1998.

[36] J. D. Kraus and R. J. Marhefka, *Antennas, 2nd Ed*. New York: McGraw-Hill, 2001.

[37] M. Skolnik, *Introduction to Radar Systems*. New York: McGraw-Hill, 2001.

[38] B. Sklar, *Digital Communications: Fundamentals and Applications, 2nd Ed.* New Jersey: Prentice Hall, 2001.

[39] F. Alam, "Space time processing for third generation CDMA systems," Ph. D. dissertation, Virginia Inst. Technol., Blacksburg, Nov. 2002.

[40] L. C. Godara and A. Cantoni, "Uniqueness and linear independence of steering vectors in array space," *J. Acoust. Soc. Amer.*, vol. 70, no. 2, pp. 467-475, Aug. 1981.

[41] K. Tan, K. Ho and A. Nehorai, "Uniqueness study of measurements obtainable with arrays of electromagnetic vector sensors," *IEEE Trans. Signal Proc.*, vol. 44, pp. 1036-1039, Apr. 1996.

[42] S. A. Schelkunoff, "A mathematical theory of linear arrays," *Bell System Technical Journal*, vol. 22, pp. 80-107, 1943.

[43] A. D. Bresler, "A new algorithm for calculating the current distributions of Dolph-Chebyshev arrays," *IEEE Trans. Antennas Propag*., vol. AP-28, pp. 951-952, Nov. 1980.

[44] N. Balakrishnan and R. Sethuraman, "Easy generation of Dolph-Chebyshev excitation coefficients," *Proc. IEEE*, vol. 69, pp. 1508-1509, Nov. 1981.

[45] H. L. Van Trees, *Optimum Array Processing (Detection, Estimation and Modulation Theory, Part IV)*. New York: Wiley, 2002.

[46] B. Widrow and M. E. Hoff, Jr., "Adaptive switching circuits," *Proc. IRE WESCON Conf. Rec.*, part 4, 1960, pp. 96-104.

[47] S. Applebaum, "Adaptive Arrays," *IEEE Trans. Antennas Propag.*, vol. 24, pp. 585-598, Sep. 1976.

[48] S. Haykin, *Adaptive Filter Theory*. New Jersey: Prentice Hall, 1996.

[49] O. Macchi, *Adaptive Processing: The LMS Approach with Applications in Transmission*. New York: Wiley, 1995.

[50] J. C. Maxwell, *A Treatise on Electricity and Magnetism, Vol. 1 and 2*. Oxford: Clarendon Press, 1873.

[51] A. F. Peterson, S. L. Ray and R. Mittra, *Computational Methods for Electromagnetics.* New York: Wiley-IEEE Press, 1997.

[52] K. S. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media," *IEEE Trans. Antennas Propag. Soc. Int. Symposium,* vol. 14, no. 3, pp. 302-307, 1966.

[53] R. F. Harrington, *Field Computation by Moment Methods*. New York: Wiley-IEEE Press, 1993.

[54] A. Hoorfar, "Evolutionary programming in electromagnetic optimization: a review," *IEEE Trans. Antennas Propag.,* vol. 55, pp. 523-537, Mar. 2007.

[55] R. L. Haupt, "Antenna Design with a mixed integer genetic algorithm," *IEEE Trans. Antennas Propag.*, vol. 55, pp. 577-582, Mar. 2007.

[56] N. Jin and Y. Rahmat-Samii, "Parallel particle swarm optimization and finite-difference time-domain (PSO/FDTD) algorithm for multiband and wide-band patch antenna designs," *IEEE Trans. Antennas Propag.*, vol. 53, pp. 3459-3468, Nov. 2005.

[57] J. M. Johnson and Y. Rahmat-Samii, "Genetic algorithms and method of moments (GA/MOM) for the design of integrated antennas," *IEEE Trans. Antennas Propag.*, vol. 47, pp. 1606-1614, Oct. 1999.

[58] A. Schriver, *Theory of Linear and Integer Programming*. New York: Wiley, 1998.

[59] G. B. Dantzig and M. N. Thapa, *Linear Programming 1: Introduction*. New York: Springer, 1997.

[60] S. G. Nash and A. Sofer, *Linear and Nonlinear Programming*. New York: McGraw-Hill, 1996.

[61] N. K. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica*, vol. 4, pp. 373-395, 1984.

[62] H. Park and F. Park, "Convex optimization algorithms for active balancing of humanoid robots," *IEEE Trans. Robotics*, vol. 23, pp. 817-822, Aug. 2007.

[63] Z. Q. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE J. Sel. Areas Commun.*, vol. 24, pp. 1426-1438, Aug. 2006.

[64] K. S. Ni and T. Q. Nguyen, "Image superresolution using support vector regression," *IEEE Trans. Image Process.*, vol. 16, pp. 1596-1610, June 2007.

[65] T. Zhang, "Sequential greedy approximation for certain convex optimization problems," *IEEE Trans. Infor. Theory,* vol. 49, pp. 682-691, Mar. 2003.

[66] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U. K.: Cambridge University Press, 2004.

[67] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*. Philadelphia, PA: SIAM, 1994.

[68] P. J. M. van Laarhoven and E. H. L. Aarts, *Simulated Annealing: Theory and Applications*. Dordrecht, Holland: D. Reidel, 1987.

[69] S. Kirkpatrick, C. D. Gelatt, Jr. and M. P. Vecchi, "Optimization by Simulated Annealing", *Science*, vol. 220, pp. 671-680, May 1983.

[70] B. Hajeck, "Cooling schedules for optimal annealing," *Math. Oper. Res.,* vol. 13, no. 2, pp. 311-329, 1988.

[71] J. Robinson, S. Sinton, and Y. Rahmat-Samii, "Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna," *IEEE International Symposium on Antennas & Propagation.* San Antonio, Texas. June, 2002.

[72] T. B. Chen, Y. B. Chen, Y. C. Jiao and F. S. Zhang, "Synthesis of antenna array using particle swarm optimization," *Asia-Pacific Microwave Conference Proceedings*, vol. 3, 2005.

[73] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE Conf. Neural Networks IV*, Piscataway, NJ, 1995.

[74] R. C. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proc. 2001 Congr. Evolutionary Computation*, vol. 1, 2001.

[75] S. C. Clark, A. W. Morrison, and M. D. Guyse, "Practice safer GPS navigation – get protection," *Raytheon Syst. Limited Internal Rep.*, 2005.

[76] R. M. Gray and L. M. Robinson, *Statistical Signal Processing*. New York: Cambridge University Press, 2004.

[77] G. E. Shilov, *Linear Algebra*. New York: Dover, 1977.

[78] F. Bowman, *Introduction to Bessel Functions.* New York, Dover, 1958.

[79] C. P. Mathews and M. D. Zoltowski, "Eigenstructure techniques for 2-D angle estimation with uniform circular arrays," *IEEE Trans. Signal Process.*, vol. 42, pp. 2395-2407, 1994.

[80] D. E. Dudgeon and R. M. Mersereau, *Multidimensional Digital Signal Processing.* London, U. K.: Prentice-Hall, 1984.

[81] G. M. Lee, N. N. Tam and N. D. Yen, *Quadratic Programming and Affine Variational Inequalities: A Qualitative Study.* Springer, 2005.

[82] J. Robinson and Y. Rahmat-Samii, "Particle swarm optimization in electromagnetics," *IEEE Trans. Antennas Propag.*, vol. 52, no. 2, pp. 397-407, Feb. 2004.

[83] M. G. Andreasan, "Linear arrays with variable interelement spacing," *IEEE Trans. Antennas Propagat.*, vol. AP-10, pp. 137-143, Mar. 1962.

[84] I. J. Gupta and A. K. Ksienski, "Effect of mutual coupling on the performance of adaptive arrays," *IEEE Trans. Antennas Propag.*, vol. AP-31, pp. 785-791, May 1983.

[85] Z. Huang and C. A. Balanis, "Mutual coupling compensation in UCAs: simulations and experiement," *IEEE Trans. Antennas Propag.*, vol. 54, pp. 3082-3086, Nov. 2006.

[86] R. Tanner and J. Woodard, *WCDMA: Requirements and Practical Design*. New York: Wiley, 2004.

[87] R. Khanna and R. Saxena, "Performance improvement in array processing architectures of WCDMA systems by low side lobe beamforming," *IEEE Int. Conf. Personal Wireless Comm.*, pp. 324-328, Jan. 2005.

[88] M. Ghavami, "Wideband Smart Antenna Theory Using Rectangular Array Structures," *IEEE Trans. Signal Process.*, vol. 50, pp. 2143-2151, Sep. 2002.

[89] K. Nishikawa *et al*., "Wideband beamforming using fan filter," in *Proc. ISCAS*, 1992, pp. 533-536.

[90] M. Ghavami and R. Kohno, "Recursive fan filters for broadband partially adaptive antenna," *IEEE Trans. Commun.*, vol. 48, pp. 185-188, Feb. 2000.

[91] K. R. Carver and J. W. Mink, "Microstrip antenna technology," *IEEE Trans. Antennas Propag.*, vol. AP-22, pp. 2-24, Jan. 1981.

[92] H. Lebret and S. Boyd, "Antenna array pattern synthesis via convex optimization," *IEEE Trans. Signal Processing*, vol. 45, pp. 526-532, Mar. 1997.

[93] D. B. Ward, Z. Ding and R. Kennedy, "Broadband DOA estimation using frequency invariant beamforming," *IEEE Trans. Signal Processing*, vol. 46, pp. 1463-1469, May 1998.

[94] K. Wong and W. Hsu, "A broad-band rectangular patch antenna with a pair of wide slits," *IEEE Trans. Antennas Propag.*, vol. 49, pp. 1345-1347, Sep. 2001.

[95] K. Tong, K. Luk, K. Lee and R. Q. Lee, "A broad-band U-slot rectangular patch antenna on a microwave substrate," *IEEE Trans. Antennas Propag.*, vol. 48, pp. 954-960, June 2000.