Reducing the Number of Ultrasound Array Elements with the Matrix Pencil Method

By

Kirk L. Sales

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Electrical Engineering

2012

ABSTRACT

Reducing the Number of Ultrasound Array Elements
with the Matrix Pencil Method

By

Kirk L. Sales

Phased arrays are diversely applied with some specific areas including biomedical imaging and therapy, non-destructive testing, radar and sonar. In this thesis, the matrix pencil method is employed to reduce the number of elements in a linear ultrasound phased array. The non-iterative, linear method begins with a specified pressure beam pattern, reduces the dimensionality of the problem, then calculates the element locations and apodization of a reduced array. Computer simulations demonstrate a close comparison between the initial array beam pattern and the reduced array beam pattern for four different linear arrays. The number of elements in a broadside-steered linear array is shown to decrease by approximately 50% with the reduced array beam pattern closely approximating the initial array beam pattern in the far-field. While the method returns a slightly tapered spacing between elements, for the arrays considered, replacing the tapered spacing with a suitably-selected uniform spacing provides very little change in the main beam and low-angle side lobes.

ACKNOWLEDGMENT

I would like to thank Dr. Robert McGough for his support and guidance during my master's degree, and for the opportunity to be part of the Biomedical Ultrasonics and Electromagnetics Lab. His experience and insight into biomedical ultrasound, phased arrays, research, and publication was an invaluable resource during my research and while writing my thesis.

I would like to thank Dr. Hayder Radha for sharing his extensive experience in signal processing, signal compression and information theory, and for the opportunity to be part of the Waves Lab. Through Dr. Radha's insights I've deepened my interest in signal processing and its applications.

I would also like to thank Dr. Seline Aviyente for serving on my committee, for sharing her knowledge of time-frequency analysis, and for helping me understand the various applications of the techniques.

Table of Contents

List of Tables

List of Figures

Chapter 1 – Introduction

1.1- Motivation:

Thermal therapy elevates tissue temperature throughout a spatial target-region while minimizing the temperature increase in adjacent regions. This can increase blood flow, increasing cell response to chemical and radiation treatment, and can prove lethal to cell function with sufficient temperature and duration [1]. An ultrasound phased array can deliver spatially localized energy as part of a thermal therapy treatment plan. The beam pattern can be steered and, in the near-field, the beam can be focused to generate a specified heating pattern over an internal or external target-region.

The cost and complexity of an ultrasound phased array is proportional to the number of elements. This thesis reduces the number of elements in a linear ultrasound phased array with the matrix pencil method. The approach provides a comparable pressure beam pattern between the initial array and the reduced array.

1.2- Background:

1.2.1- The Ultrasound Linear Array:

A linear array refers to a linear configuration of array elements. The target-region describes a region in three-dimensional space where a specified pressure beam pattern is generated. Examples of target-regions include a line or a volume, and in some applications, the volume can approximate the shape of a tumor. This thesis will consider a semi-circular target-region in the far-field of the array.

The Fourier transform of the array aperture determines the shape of the pressure beam pattern in the far-field. For a linear array, the aperture can be represented as a rectangular function which yields a sinc-shaped beam pattern. Consistent with Fourier theory, as the aperture width increases, the main lobe width decreases. Reducing the excitation amplitude for elements toward the ends of the array reduces the side lobe levels at the expense of widening the main lobe. The distribution of element excitation amplitudes, sometimes called the element weights, is referred to as the array apodization.

The main lobe may be steered and/or focused by defining a phase distribution across the array. Broadside-steering refers to positioning the main lobe in front of the center of the array. This thesis considers linear, broadside-steered arrays in the far-field.

Spatial aliasing occurs when the array elements are spaced too far apart leading to larger-amplitude side lobes called grating lobes. In thermal therapy, grating lobes heat tissue in unfavorable spatial regions. One way to avoid grating lobes is to select an array element spacing 0.5 wavelengths or less across the entire array. In this thesis, all initial arrays are configured with a 0.5 wavelength element spacing. Reducing the number of array elements increases the element spacing and may lead to grating lobes if the spacing becomes too large.

1.2.2- Thinning a Linear Array:

Thinning an array refers to reducing the number of active or electrically-excited elements. Some methods employed to synthesize thinned arrays include genetic algorithms [2], the particle swarm optimization algorithm [3], and simulated annealing [4]. The prony method [5], a polynomial method, and the matrix pencil method [6] represent linear, non-iterative

methods. The matrix pencil method has been shown to be less sensitive to noise than the prony method [7] and has been applied to reduce the number of elements in a linear array while maintaining comparable beam patterns between the initial array and the reduced array [8]. The matrix pencil method analyzes each data snapshot independently and operates well in non-stationary environments [9].

1.2.3- The Matrix Pencil Method with Applications:

The matrix pencil method is based on the notion that a function may be decomposed into a sum of complex exponentials. The method admits samples of the initial function and generates the parameters for a set of complex exponentials.   Summing the exponentials returns an approximation to the initial signal.

The matrix pencil method is widely applied, particularly in the study of radar and signal processing. Two radar applications, an antenna radiation pattern example, and a music spectral analysis application are introduced next.

<u>Detecting Objects on the Tracks of a Passenger Railway System</u>

A passenger railway system is configured with rail tracks in front of a passenger platform. Objects such as luggage and even humans may be present on the tracks. The radar-based application senses the presence of one or more objects on the tracks and can identify the type of object from a set of known objects [10].

A radiating transmission line is routed along the rail platform with a pulse generator at one end and a receiver at the opposite end. Discontinuities in the transmission line radiate some of the

energy and any nearby object reflects the energy back to the transmission line. The receiver collects the directly transmitted signal first, followed by the reflected signal from the object.

The total least-squares matrix pencil method is employed to extract the complex natural resonances (CNR) of the object. The CNR of the object under consideration, and the CNR of objects which may be found on railway platforms, are presented to the discrimination process proposed in [10] to classify the object. The method is shown to reliably classify humans and other objects on or near railway platforms and is capable of distinguishing one object from another, such as an item of luggage from a person.

Detecting Maneuvering Objects in a Maritime Environment

Maneuvering targets in a maritime environment may be detected by radar [11]. In addition to noise, the application is characterized by sea clutter associated with reflections from the ocean surface. The effects become intensified under high wind conditions along with the high-resolution and low grazing angle nature of the application. The resulting coherent reflected signals can appear similar to the returns from fishing boats and other targets-of-interest. The total least-squares matrix pencil method and the singularity expansion method are employed to detect the presence of one or more targets. The target is further identified as either a trawler, a sardine boat or a longline boat.

Quantifying an Antenna Radiation Pattern

The radiation pattern of an antenna is quantified in a semi-anechoic environment. Signal leakage along with reflections and diffractions can degrade the measurement quality,

particularly when using low-directivity probes [12]. The total least-squares matrix pencil method can be employed to distinguish the direct-path radiation from reflections and lead to a more accurate representation of the radiation pattern.

One application considers two antennas facing each other in a semi-anechoic chamber [12]. The problem is considered to have a directly propagating component and a component reflected from the chamber floor. The system transmission coefficient is uniformly-sampled and the samples populate a complex-valued matrix which undergoes a singular value decomposition. The largest-magnitude singular value corresponds to the directly-propagating wave. The remaining singular values correspond to the reflected wave, noise, or both. In this way, the transmission coefficient due to the directly-propagating wave is reconstructed.

Performing Spectral Analysis on a Musical Signal

The natural-sounding character of music results from closely-spaced frequencies which lead to amplitude beat frequencies [13]. An analysis tool must exhibit good spectral resolution in order to identify the beat frequencies, and here, the matrix pencil method is employed to analyze the second-harmonic of a piano note.

The piano note is sampled at 22.5 kHz and the sampled signal is band-pass filtered to extract the second harmonic. The filtered signal contains too many samples for an effective application of the matrix pencil method and is reduced by demodulation and down-sampling prior to implementing the matrix pencil method. The synthesized signal is comparable to the initial signal.

1.3-   Thesis Organization:

This thesis applies the matrix pencil method to reduce the number of elements in a linear, broadside-steered ultrasound phased array in the far-field. Chapter 2 introduces four forms of the matrix pencil method and describes how one of the forms can be implemented to reduce the number of elements in an ultrasound array. Chapter 3 replicates the 20 element array reduction in [8]. Next, 128, 64, and 32 element linear arrays are reduced. All of the arrays are steered broadside and the pressure beam pattern is calculated in the far-field.

The array elements in [8] are modeled as point-sources and the target-region pressure is calculated with the far-field approximation. In contrast, the 128, 64, and 32 element arrays are configured with non-zero area elements and the pressure is calculated without the far-field approximation. Chapter 2 considers the pressure calculations in more detail.

It is shown that a significant reduction in the number of array elements can be achieved while maintaining a comparable beam pattern between the initial array and the reduced array. The simulations further reveal that while the reduced array spacing is slightly tapered, replacing the tapered spacing with a suitably-selected uniform spacing provides very little change in the main beam and low-angle side lobes.

Chapter 4 discusses the simulation results in more detail and considers some topics for future work. Chapter 5 presents the conclusions, and the Appendix includes Matlab code which reduces the number of elements in a linear, broadside-steered array.

Chapter 2 – Theory

2.1- The Matrix Pencil Method:

2.1.1- The Sampled Exponential Sequence:

The problem of approximating a function with a sum of complex exponentials has attracted attention for many years. Some applications include multiple transient signal processing, radio direction finding, and the synthesis of an antenna pattern [7].

A sampled exponential data sequence can be represented by

$$y(kT_S) = x(kT_S) + n(kT_S) \sim \sum_{i=1}^{M} R_i z_i^k + n(kT_S) \quad \text{for k = 0, 1, ..., (N-1)} \quad (2.1)$$

and
$$z_i = e^{(-\alpha_i + j\omega_i)T_S} \quad \text{for i = 1, 2, ..., M,} \quad (2.2)$$

where T_S is the sampling period, M is the number of sinusoids, R_i are the residues, $n(kT_S)$ is the system noise and z_i are the poles with damping factors α_i and angular frequencies ω_i.

Estimates are sought for the number of sinusoids, the poles, and the residues. Solving for all three parameters simultaneously is a nonlinear problem. Here, the linear problem will be solved. Two popular approaches to solving the linear problem are the polynomial method, for example the prony method, and the matrix pencil method. The matrix pencil method provides greater computational efficiency and increased noise robustness relative to the polynomial method [7]. The matrix pencil method also performs well when analyzing highly-damped signals, short data records, or closely-spaced frequencies [13]. This thesis considers the matrix pencil method.

The objective of the matrix pencil method is to construct two matrices such that the generalized eigenvalues of a matrix pencil formed from the two matrices provides the desired parameters, the poles [6]. The matrix pencil method is sometimes referred to as the generalized pencil-of-function and takes on different forms. Section 2.1.2 outlines the noiseless case while Sections 2.1.3, 2.1.4 and 2.1.5 describe the total least-squares, the band-pass, and the total forward-backward matrix pencil method, respectively.

2.1.2- The Noiseless Case:

For noiseless data, matrices $[Y_1]$ and $[Y_2]$ are defined as

$$[Y_1] = \begin{bmatrix} x(0) & x(1) & \cdots & x(L-1) \\ x(1) & x(2) & \cdots & x(L) \\ \vdots & \vdots & \vdots & \vdots \\ x(N-L-1) & x(N-L) & \cdots & x(N-2) \end{bmatrix}_{(N-L)xL} \qquad (2.3)$$

and

$$[Y_2] = \begin{bmatrix} x(1) & x(2) & \cdots & x(L) \\ x(2) & x(3) & \cdots & x(L+1) \\ \vdots & \vdots & \vdots & \vdots \\ x(N-L) & x(N-L+1) & \cdots & x(N-1) \end{bmatrix}_{(N-L)xL} \qquad (2.4)$$

where the values x(0), x(1), ..., x(N-1) represent N samples of the noiseless exponential sequence $x(kT_S)$ and L represents the pencil parameter which establishes a balance between accuracy and computational efficiency. $[Y_1]$ and $[Y_2]$ may also be expressed as

$$[Y_1] = [Z_1][R][Z_2] \qquad (2.5)$$

and

$$[Y_2] = [Z_1][R][Z_0][Z_2], \qquad (2.6)$$

where

$$[Z_1] = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_M \\ \vdots & \vdots & \vdots & \vdots \\ z_1^{(N-L-1)} & z_2^{(N-L-1)} & \cdots & z_M^{(N-L-1)} \end{bmatrix}_{(N-L) \times M} \qquad (2.7)$$

$$[Z_2] = \begin{bmatrix} 1 & z_1 & \cdots & z_1^{(L-1)} \\ 1 & z_2 & \cdots & z_2^{(L-1)} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & z_M & \cdots & z_M^{(L-1)} \end{bmatrix}_{M \times L} \qquad (2.8)$$

$$[Z_0] = diag[z_1, z_2, \cdots, z_M]_{M \times M} \qquad (2.9)$$

and

$$[R] = diag[R_1, R_2, \cdots, R_M]_{M \times M}. \qquad (2.10)$$

If

$$M \leq L \leq N - M, \qquad (2.11)$$

the poles z_i become the generalized eigenvalues of the matrix pencil

$$[Y_2] - \xi[Y_1] = [Z_1][R]\{[Z_0] - \xi[I]\}[Z_2] \qquad (2.12)$$

which may be determined by solving the eigenvalue problem

$$\{[Y_1]^+[Y_2] - \xi[I]\} = 0, \qquad (2.13)$$

where

$$[Y_1]^+ = \{[Y_1]^H[Y_1]\}^{-1}[Y_1]^H \qquad (2.14)$$

is the Moore-Penrose pseudo inverse of $[Y_1]$, "I" represents the identity matrix and "H" designates the conjugate transpose.

2.1.3- The Total Least-Squares Matrix Pencil Method:

When the signal contains noise, the total least-squares matrix pencil method provides improved performance. Here, the matrix [Y] is defined as

$$[Y] = \begin{bmatrix} y(0) & y(1) & \cdots & y(L) \\ y(1) & y(2) & \cdots & y(L+1) \\ \vdots & \vdots & \vdots & \vdots \\ y(N-L-1) & y(N-L) & \cdots & y(N-1) \end{bmatrix}_{(N-L)x(L+1)}, \qquad (2.15)$$

where $[Y_1]$ and $[Y_2]$ in Section 2.1.2 are formed from $[Y]$ by deleting the last and first column of $[Y]$, respectively. Setting the pencil parameter L between N/3 and N/2 returns a minimum variance in the z_i parameters due to noise [7].

A singular value decomposition is performed on [Y] according to

$$[Y] = [U][\Sigma][V]^H, \qquad (2.16)$$

where the unitary matrices [U] and [V] represent the eigenvectors of $[Y][Y]^H$ and $[Y]^H[Y]$, respectively, and the singular values of [Y] populate the diagonal matrix [Σ]. Generally, matrix [Y] contains Q principal singular values corresponding to the number of poles in the signal.

If the singular value magnitudes decline slowly from the principal to the non-principal singular values, it may become difficult to determine the number of poles. Here, filtering the signal prior to invoking the matrix pencil method can lead to a greater distinction between the principal and non-principal singular values. The pre-filtering approach is known as the band-pass matrix pencil method and is described further in Section 2.1.4. In general, if the magnitude of the singular values does not decrease with increasing singular value index, then representing the input function as a sum of complex exponentials is not the correct approach [7].

The number of principal singular values is determined by considering the ratio of each

singular value to the largest-magnitude singular value. A threshold of 10^{-p} is set where p represents the number of significant digits in the initial data. The principal singular values are those singular values with a ratio greater than 10^{-p}.

The matrix [V'] is defined as the leftmost Q columns of [V] resulting in

$$[Y_1'] = [U][\Sigma'][V_1']^H \qquad (2.17)$$

and
$$[Y_2'] = [U][\Sigma'][V_2']^H, \qquad (2.18)$$

where [Σ'] represents the Q largest-magnitude singular values and V_1' and V_2' result from removing the last and first rows of [V'], respectively. The eigenvalues are calculated from

$$\{[V_1']^H\}^+ \{[V_2']^H\}^+ - \xi[I] = 0. \qquad (2.19)$$

When the number of sinusoids Q and the poles z_i are known, the residue values R_i are determined from the least-squares problem

$$\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_Q \\ \vdots & \vdots & \vdots & \vdots \\ z_1^{(N-1)} & z_2^{(N-1)} & \cdots & z_Q^{(N-1)} \end{bmatrix} \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_Q \end{bmatrix}. \qquad (2.20)$$

An estimate of the initial signal is reconstructed from the poles and residues.

This thesis will employ the total least-squares matrix pencil method to reduce the number of elements in a linear ultrasound phased array. Section 2.2 describes the theoretical details and Chapter 3 applies the approach to 20, 32, 64, and 128 element linear arrays.

11

2.1.4- The Band-Pass Matrix Pencil Method:

The band-pass matrix pencil method offers increased noise robustness by filtering the input signal before applying the matrix pencil method. The method requires knowing the location and spectral-width of the input signal in the frequency domain and applies appropriately-designed band-pass filters before invoking the matrix pencil method.

The approach is implemented in [14] by cascading multiple second-order band-pass filters. The filter transfer function poles are located near the input signal poles and a filter zero is placed at z = 1 and z = -1 to suppress the low and high-frequency components. The filter transfer function becomes

$$H(z) = K \frac{(z+1)(z-1)}{(z - re^{j\omega_0})(z - re^{-j\omega_0})}, \tag{2.21}$$

where K sets the filter gain and ω_0 represents the input signal spectral peak. The variable r is defined as

$$r = \frac{1}{1 - BW}, \tag{2.22}$$

where BW reflects the input signal spectral bandwidth and (0 < BW < 1). After filtering, the matrix pencil method determines the signal poles.

2.1.5- The Total Forward-Backward Matrix Pencil Method:

If the input exponential sequence is undamped, the estimation accuracy may be improved by employing the total forward-backward matrix pencil method [15]. Here, the input signal samples populate the matrix

$$Y_{fb} = \begin{bmatrix} y_0 & y_1 & \cdots & y_{L-1} & y_L \\ y_L^* & y_{L-1}^* & \cdots & y_1^* & y_0^* \end{bmatrix}_{2(N-L) \times (L+1)}, \tag{2.23}$$

where
$$y_j^T = [y_j, y_{j+1}, \cdots, y_{N-L+j-1}] \quad j = 0,1,2,\cdots,L, \tag{2.24}$$

and * denotes the complex conjugate. The matrix $[Y_{fb}]$ is sometimes referred to as the all-data matrix. The matrices $[Y_{0fb}]$ and $[Y_{1fb}]$ are defined as

$$Y_{0fb} = \begin{bmatrix} y_0 & y_1 & \cdots & y_{L-2} & y_{L-1} \\ y_L^* & y_{L-1}^* & \cdots & y_2^* & y_1^* \end{bmatrix}_{2(N-L) \times L} \tag{2.25}$$

and
$$Y_{1fb} = \begin{bmatrix} y_1 & y_2 & \cdots & y_{L-1} & y_L \\ y_{L-1}^* & y_{L-2}^* & \cdots & y_1^* & y_0^* \end{bmatrix}_{2(N-L) \times L} \tag{2.26}$$

by removing the last and first columns of $[Y_{fb}]$, respectively. In the noiseless case, the undamped poles (frequencies) may be obtained by solving for the generalized eigenvalues of the matrix pencil

$$[Y_{1fb}] - \xi[Y_{0fb}] = 0, \tag{2.27}$$

where ξ is a complex-valued scalar.

In the presence of noise, it is beneficial to perform a singular value decomposition on $[Y_{fb}]$, defined as

$$[Y_{fb}] = [U][\Sigma][V]^H, \tag{2.28}$$

where $[U]$ and $[V]$ represent the left and right singular vectors, respectively and $[\Sigma]$ is a diagonal matrix containing the singular values of $[Y_{fb}]$. The Q principal singular values are

13

retained and the non-principal singular values are considered to be related to noise and are discarded. This leads to

$$[Y'_{fb}] = [U'][\Sigma'][V']^H, \qquad (2.29)$$

where (') designates the reduced-dimension version of the matrix. Matrices $[Y_{0fb}]$ and $[Y_{1fb}]$ become

$$[Y'_{0fb}] = [U'][\Sigma'][V'_0]^H \qquad (2.30)$$

and $\qquad [Y'_{1fb}] = [U'][\Sigma'][V'_1]^H, \qquad (2.31)$

where $[V'_0]$ and $[V'_1]$ are obtained by deleting the last and first columns of [V'], respectively. The matrix pencil becomes

$$[Y'_{1fb}] - \xi[Y'_{0fb}] = 0 \qquad (2.32)$$

and solving the generalized eigenvalue problem

$$q^H(V'^H_1 V'_0 - \xi V'^H_0 V'_0) = 0^H \qquad (2.33)$$

provides the estimated frequencies [15].

2.1.6- Additional forms of the Matrix Pencil Method:

Some additional forms of the matrix pencil method are listed below and summarized in the following paragraphs.

- Coupled Matrix Pencil Method
- Matrix Enhancement and Matrix Pencil Algorithm
- Multiple Invariance Beamspace Matrix Pencil Method
- Single Invariance Beamspace Matrix Pencil Method

- Unitary Matrix Pencil Method
- Two-Dimensional Enhanced Matrix Pencil Method
- Modified Matrix pencil Method 3
- Modified Total Least-Squares Matrix Pencil Method

Coupled Matrix Pencil Method

The coupled matrix pencil method is introduced in [16] to extrapolate electromagnetic solutions at intermediate frequencies given a known low-frequency dataset and a known high-frequency dataset. The matrix pencil method is applied to each dataset separately which results in two sets of poles. The two sets of poles are combined into a single set and along with the input datasets lead to a set of coupled residuals. An updated signal model is constructed from the poles and coupled residuals and the model is applied to extrapolate solutions at intermediate frequencies [16].

Matrix Enhancement and Matrix Pencil Algorithm

Hua presents the matrix enhancement and matrix pencil method (MEMP) for estimating two-dimensional frequencies [17]. The MEMP begins by populating an enhanced Hankel matrix with the two-dimensional (2D) data samples according to the partitioning and stacking process presented in [17]. The enhanced matrix undergoes a singular value decomposition and only the principal singular values are retained. The principal singular values lead to two matrix pencils, one for each dimension of the problem. The generalized eigenvalues of each matrix pencil provide two sets of poles. The poles are paired such that a given pole in one set is associated with a specific pole in the other set. The pole pairing is carried out according to the maximization method described in [17]. Each of the 2D pole pairs generates a unique 2D frequency through a set of algebraic equations.

<u>Multiple Invariance Beamspace Matrix Pencil Method, Single Invariance Beamspace Matrix Pencil Method, and Unitary Matrix Pencil Method</u>

The unitary matrix pencil method (UMP), single invariance beamspace matrix pencil method (SBMP), and the multiple invariance beamspace matrix pencil method (MBMP) provide similar average root mean square errors while estimating the direction of arrival of a signal impinging on a uniform linear array. The computational complexity of the methods differ significantly [9]. The UMP applies a unitary matrix transformation to reduce the complex-valued calculations to real-valued calculations and decreases the computational burden by a factor of four while offering a comparable estimation accuracy relative to the matrix pencil method (MPM). The SBMP and MBMP methods offer additional computational burden reduction relative to the UMP when priori information is available. The priori information enables projecting the original data into a lower-dimension subspace prior to estimating the unknown parameters. The accuracy of the MBMP is comparable to the MPM while the SBMP offers marginally less accuracy relative to the MPM [9].

<u>Two-Dimensional Enhanced Matrix Pencil Method</u>

The authors in [18] apply the two-dimensional enhanced matrix pencil method (2D-EMPM) to obtain the azimuth and elevation angles of arrival from a signal impinging on a uniform rectangular array. An array residing in the XY plane provides one sample of the signal in the X direction and another sample in the Y direction. The samples populate an enhanced Hankel matrix which undergoes a singular value decomposition. The result is a set of left and right singular vectors which are separated into the signal space and the noise space according to the criteria presented in Section 2.1.3. A matrix is constructed from the left eigenvectors and the generalized eigenvalues of the matrix lead to a set of X-axis parameters. Similarly, a matrix

populated with the right eigenvectors leads to a set of Y-axis parameters. The X and Y-axis parameters are paired according to the process described in [18] and a set of equations yield the azimuth and elevation angles. For a reduced number of sensors and low signal-to-noise ratio (SNR), the 2D-EMPM offers improved performance relative to the 2D-MPM and the EMPM. The X and Y axis pairing of the 2D-EMPM reduces the computational complexity compared to the shuffling matrix process of the EMPM.

<u>Modified Matrix pencil Method 3 and Modified Total Least-Squares Matrix Pencil Method</u>

The traditional matrix pencil method does not consider the Hankel structure of the matrix pencil and provides reduced estimation accuracy in a low SNR environment. The modified matrix pencil method 3 (MMP3) in [19] delivers improved estimation accuracy by performing a preprocessing step prior to applying the matrix pencil method. The preprocessing step employs the reduced-rank Hankel approximation operation which attempts to retain both the rank-deficient and the Hankel properties of the initial data matrix. In a high SNR environment the MMP3 offers comparable performance to the traditional matrix pencil method while at a low SNR the MMP3 significantly outperforms the traditional matrix pencil method [19]. The MMP3 may be combined with the total least-squares matrix pencil method presented in Section 2.1.3 yielding the modified total least-squares matrix pencil method [20].

2.2- Reducing the Number of Ultrasound Phased Array Elements with the Matrix Pencil Method:

2.2.1- Overview:

The number of elements in a linear, broadside-steered ultrasound phased array will be reduced by applying the total least-squares matrix pencil method described in Section 2.1.3. The array

reduction algorithm begins by considering a linear, broadside-steered ultrasound array [8]. The far-field pressure beam pattern is uniformly-sampled across the target-region and the samples are arranged into a Hankel matrix which undergoes a singular value decomposition. The principle singular values represent the number of elements in the reduced array. Solving a generalized eigenvalue problem yields the poles and solving an algebraic problem provides the reduced array element locations. The number of elements in the reduced array and the poles lead to the formulation of a least-squares problem and the residues of the problem become the reduced array element weights. The reduced array pressure beam pattern is comparable to the initial array beam pattern in the far-field.

2.2.2- Beam Pattern Calculations:

Figure 2.1 illustrates a linear array and semi-circular target-region. The semi-circular target-region shape ensures that the beam-space is uniformly-sampled with respect to the angle Θ.

This thesis employs two different approaches to calculate the far-field pressure beam pattern. The array factor and far-field approximation approach described first is applied to the 20 element initial array and the FOCUS software approach calculates pressures for the 128, 64, and 32 element initial arrays.

The array factor of an M element linear array is given as

$$F_M(\theta) = \sum_{i=1}^{M} R_i e^{jkd_i \cos(\theta)}, \qquad (2.34)$$

18
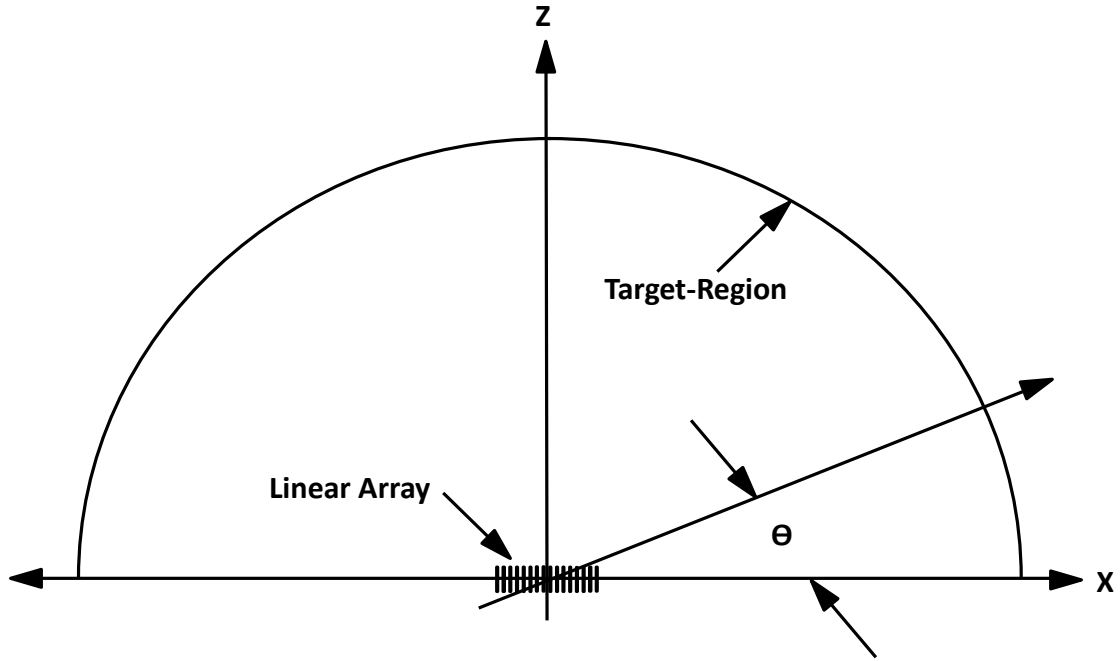
Figure 2.1: Linear Array and Semi-Circular Target-Region
(Not to Scale)
The semi-circular target-region shape ensures that the beam-space is uniformly-sampled with respect to the angle θ.

$$p(x,y,z,t) = \frac{\rho_0 c U_0 e^{j\omega t}}{2\pi} \left[(x-a) \int_{y-b}^{y+b} \frac{e^{-jk\sqrt{z^2+(y')^2+(x-a)^2}} - e^{-jkz}}{(y')^2 + (x-a)^2} dy' \right.$$

$$-(x+a) \int_{y-b}^{y+b} \frac{e^{-jk\sqrt{z^2+(y')^2+(x+a)^2}} - e^{-jkz}}{(y')^2 + (x+a)^2} dy'$$

$$+(y-b) \int_{x-a}^{x+a} \frac{e^{-jk\sqrt{z^2+(x')^2+(y-b)^2}} - e^{-jkz}}{(x')^2 + (y-b)^2} dx'$$

$$\left. -(y+b) \int_{x-a}^{x+a} \frac{e^{-jk\sqrt{z^2+(x')^2+(y+b)^2}} - e^{-jkz}}{(x')^2 + (y+b)^2} dx' \right]$$

(2.35)

where M represents the number of initial array elements, R_i is the weight of the i_{th} array element, k is the wave number, d_i is the location of the i_{th} array element, and Θ is the angle between the array axis and the target-region point under consideration. The array factor models each array element as a zero-area point-source and applies the far-field approximation.

In contrast to the array factor pressure calculation approach, the FOCUS software includes the non-zero element areas and calculates distances from each array element to each target-region point without applying the far-field approximation. More specifically, the time-harmonic pressure generated by a rectangular piston in rectangular coordinates is determined by numerically integrating the expression shown in Equation 2.35 where the piston width and height are 2a and 2b, respectively [21]. The array element is centered at (x',y',z') and the target-region point under consideration is centered at (x,y,z). The FOCUS ultrasound simulation software is available at (http://www.egr.msu.edu/~fultras-web/).

2.2.3- The Hankel Matrix and Singular Value Decomposition:

The far-field pressure beam pattern across a semi-circular target-region is calculated for a broadside-steered linear array. The variable M represents the number of elements in the initial array and the target-region is sampled at N points that are uniformly-spaced from {-1 ≤ cos(Θ) ≤ 1}. The samples populate a Hankel matrix according to

$$[Y] = \begin{bmatrix} y(0) & y(1) & \cdots & y(L) \\ y(1) & y(2) & \cdots & y(L+1) \\ \vdots & \vdots & \vdots & \vdots \\ y(N-L-1) & y(N-L) & \cdots & y(N-1) \end{bmatrix}_{(N-L)x(L+1)}, \quad (2.36)$$

where y(0) and y(N-1) represent the target-region pressure at cos(θ) = -1 and cos(θ) = 1, respectively. The pencil parameter L establishes a balance between accuracy and computational efficiency [22], and setting the pencil parameter to between N/3 and N/2 can improve performance in noisy applications [7]. The parameters N and L follow from the equations

$$(N - L - 1) \geq M, \tag{2.37}$$

$$(L + 1) \geq M \tag{2.38}$$

and
$$N \geq \frac{4d_{max}}{\lambda} + 1, \tag{2.39}$$

where M is the number of elements in the initial array, λ is the wavelength and $d_{max} = \max(d_i)$.

The singular value decomposition of the Hankel matrix is

$$[Y] = [U][\Sigma][V]^H, \tag{2.40}$$

where [U] and [V] are complex-valued matrices containing the left and right singular vectors, respectively. The diagonal matrix [Σ] contains the singular values in order of decreasing magnitude.

2.2.4- Selecting the Number of Elements in the Reduced Array:

The number of principal singular values corresponds to the number of elements in the reduced array. In Section 2.1.3, a ratio is defined between each singular value and the singular value with the largest-magnitude. A threshold is set based upon the number of significant digits in the initial data, and singular values with ratios below the threshold are considered related to noise and are discarded.

Here, the mean square error (MSE) is calculated between the initial array beam pattern and the reduced array beam pattern for reduced arrays with a number of elements ranging from four to the number of elements in the initial array. The MSE is calculated according to

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(PI_i - PR_i)^2, \tag{2.41}$$

where N is the number of uniformly-spaced target-region points and PI_i and PR_i represent the initial array pressure and the reduced array pressure at the target-region location (i). The number of elements in the reduced array which yield the minimum MSE becomes the number of principal singular values. Setting the non-principal singular values equal to zero results in the reduced matrix

$$[Y_Q] = [U][\Sigma_Q][V]^H, \tag{2.42}$$

where Q represents the number of elements in the reduced array.

Solving the generalized eigenvalue problem

$$([Y_{Q,f}] - z'[Y_{Q,l}])\bar{v}' = 0 \tag{2.43}$$

results in the non-zero eigenvalues z_i'. The matrices $[Y_{Q,f}]$ and $[Y_{Q,l}]$ result from deleting the first and last column of $[Y_Q]$, respectively.

An alternate generalized eigenvalue calculation method employed in this thesis is to calculate the non-zero eigenvalues of the matrix

$$([V_{Q,b}]^H[V_{Q,b}])^{-1}([V_{Q,t}]^H[V_{Q,b}]), \tag{2.44}$$

where the leftmost Q vectors in [V] become $[V_Q]$. The matrices $[V_{Q,t}]$ and $[V_{Q,b}]$ are obtained by removing the top and bottom rows of $[V_Q]$, respectively [8].

Reducing the number of elements in an array while holding the other array parameters constant reduces the power output of the array. Here, the reduced array element width is increased such that the aperture area remains constant between the initial and reduced arrays.

2.2.5- Estimating the Reduced Array Element Locations and Weights:

The reduced array element locations are obtained from

$$\hat{d}_i' = \frac{(N-1)}{2jk} \ln(\hat{z}_i'),$$ (2.45)

where $\hat{z}_i'$ represents the non-zero eigenvalues. Normalizing the eigenvalues with

$$\hat{z}_i' = \frac{z_i'}{\left|z_i'\right|}$$ (2.46)

ensures that $\left|\hat{z}_i'\right| = 1$ and returns real-valued element locations [8]. The eigenvalue amplitudes before normalization are very close to 1.0 for all of the arrays considered in this thesis.

The reduced array apodization is computed as

$$\hat{R}_i' = ([\hat{Z}]^H[\hat{Z}])^{-1}[\hat{Z}]^H f_M,$$ (2.47)

where $$f_M = (y(0), y(1), \cdots, y(N-1))^T$$ (2.48)

23

and
$$[\hat{Z}] = \begin{bmatrix} (\hat{z}_1')^{-\left(\frac{N-1}{2}\right)} & (\hat{z}_2')^{-\left(\frac{N-1}{2}\right)} & \cdots & (\hat{z}_Q')^{-\left(\frac{N-1}{2}\right)} \\ (\hat{z}_1')^{-\left(\frac{N-3}{2}\right)} & (\hat{z}_2')^{-\left(\frac{N-3}{2}\right)} & \cdots & (\hat{z}_Q')^{-\left(\frac{N-3}{2}\right)} \\ \vdots & \vdots & \vdots & \vdots \\ (\hat{z}_1')^{\left(\frac{N-1}{2}\right)} & (\hat{z}_2')^{\left(\frac{N-1}{2}\right)} & \cdots & (\hat{z}_Q')^{\left(\frac{N-1}{2}\right)} \end{bmatrix}_{NxQ}.$$ (2.49)

The element locations and weights define a reduced array with a beam pattern that approximates the initial array beam pattern in the far-field.

Chapter 3 – Simulation Results

In Section 3.1, the 20 element array reduction in [8] is replicated. The array elements are modeled as point-sources and the far-field approximation is employed. Section 3.2 considers a normalized Taylor window as an alternative to the Chebyshev window for the initial array apodization function. Sections 3.3, 3.4, and 3.5 reduce 128, 64, and 32 element linear arrays, respectively. Here, the non-zero element areas are considered and the pressure at each target-region point is calculated without the far-field approximation. Section 3.6 considers the simulation time.

3.1- Replicating the 20 Element Array Reduction in [8]:

A 20 element linear array is reduced to an array of fewer elements with the total least-squares matrix pencil method presented in Section 2.1.3. The array is modeled with point-sources and the far-field approximation is applied.

The initial array contains 20 point-sources. The center-to-center spacing is 0.5 wavelengths with all elements located symmetric about the center of the array. The array is steered broadside and apodized by a Chebyshev window with a constant 30 dB side lobe amplitude. Columns 2-3 in Table 3.1 list the element locations and apodization. The element locations range from -4.75 wavelengths to +4.75 wavelengths, and the apodization decreases from 1.00 at the center of the array to 0.29 toward the ends of the array. The speed of sound in the medium is 1540 m/s and the array is operated at 1.0 MHz, which yields a wavelength of 0.0015 m. The array aperture is 0.0146 m wide.

i	20 Element Initial Array		12 Element Array in [8]		12 Element Replicated Array	
	d_i/λ	R_i	d_i/λ	R_i	d_i/λ	R_i
1	0.25	1.0000	0.4254	1.0000	0.4253	1.0000
2	0.75	0.97010	1.2755	0.91407	1.2753	0.9141
3	1.25	0.91243	2.1236	0.75974	2.1233	0.7598
4	1.75	0.83102	2.9671	0.56719	2.9667	0.5673
5	2.25	0.73147	3.8011	0.37122	3.8005	0.3713
6	2.75	0.62034	4.6371	0.26841	4.6367	0.2685
7	3.25	0.50461	--	--	--	--
8	3.75	0.39104	--	--	--	--
9	4.25	0.28558	--	--	--	--
10	4.75	0.32561	--	--	--	--

Table 3.1: Initial Array and Reduced Array Element Locations and Weights
The initial and reduced array element locations and weights are symmetric about the center of the array. Columns 2-3 present the initial array parameters, and columns 4-5 represent the reduced array parameters calculated in [8]. The parameters replicated here, and as listed in columns 6-7, compare to the corresponding parameters calculated in [8] to within three decimal places.

The variable M represents the number of elements in the initial array and L specifies the pencil parameter. Here, M = L = 20 and the number of uniformly-spaced target-region points, N, is set to 41. The target-region is located in the far-field of the array at a radius of 1000 times the aperture width (14.6 m) from the center of the array.

The pressures at each of the 41 target-region points populate a Hankel matrix and a singular value decomposition is computed. Figure 3.1 provides the singular values in order of decreasing magnitude. Consistent with [8], the 12 largest-magnitude singular values are
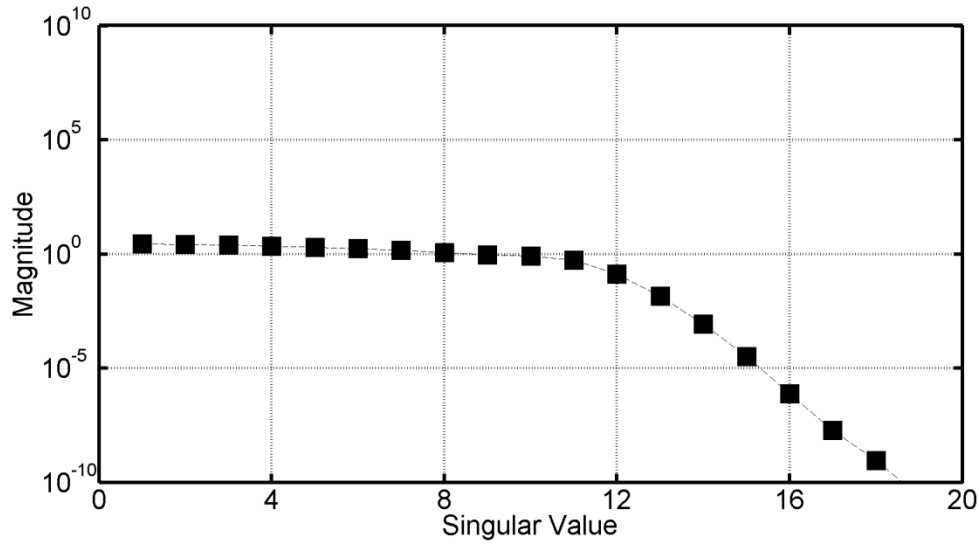
Figure 3.1: Hankel Matrix Singular Values

The 20 element initial array Hankel matrix singular values range from greater than 1 to less than 10^{-10}. Consistent with [8], the 12 largest-magnitude singular values are retained corresponding to 12 elements in the reduced array. The remaining singular values are set equal to zero.

retained which correspond to the number of elements in the reduced array. The remaining singular values are set equal to zero. Columns 6-7 in Table 3.1 provide the reduced array element locations and weights which compare to the corresponding values calculated in [8] to within 3 decimal places. Figure 3.2 illustrates the close comparison between the initial and reduced array beam patterns across the target-region. Grating lobes do not appear in the result.

Figure 3.3 shows the reduced array apodization which ranges from 1.00 at the center of the array to 0.27 at the ends of the array. Reducing the excitation amplitude toward the ends of the array relative to the center of the array reduces the side lobe amplitude at the expense of an increased main lobe width. Here, the apodization is symmetric about the center of the array and closely approximates a length 12 Chebyshev window with 30 dB constant-amplitude side
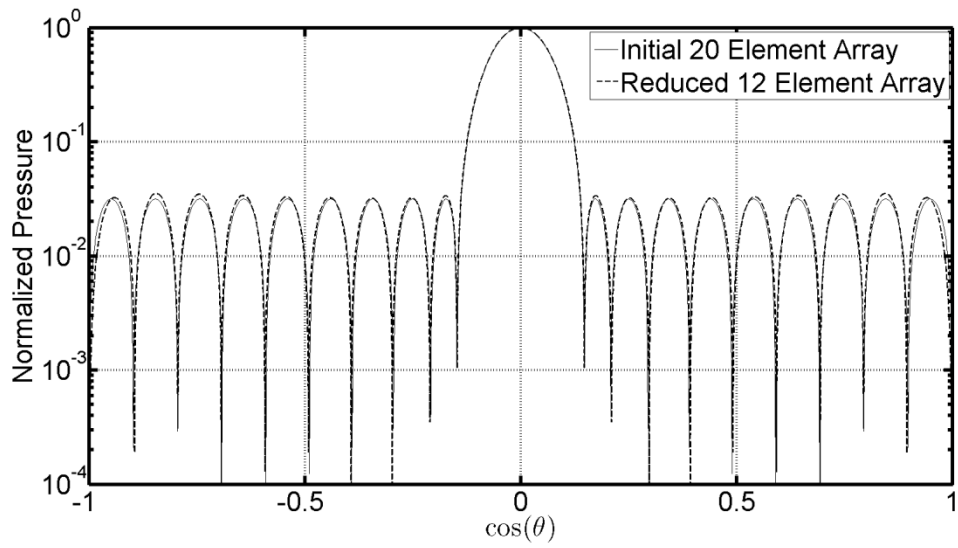
Figure 3.2: Initial Array and Reduced Array Pressure Beam Patterns
The simulated far-field pressure beam patterns for the 20 element initial array and the 12 element reduced array. The beam patterns nearly overlap across the target-region.
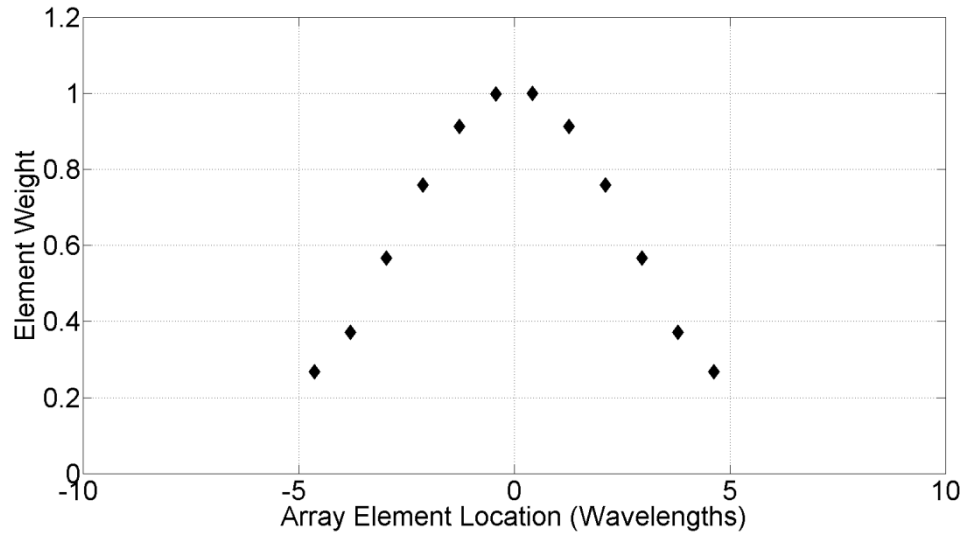

Figure 3.3: Reduced Array Apodization
The 12 element reduced array apodization ranges from 1.00 at the center of the array to 0.27 at the ends of the array. The apodization is symmetric about the center of the array.
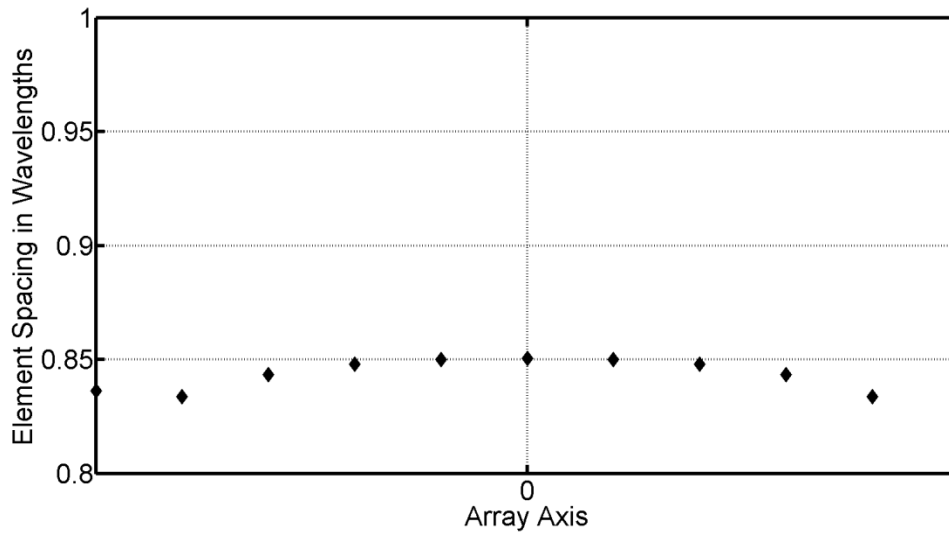
Figure 3.4: Reduced Array Element Spacing
The reduced array element spacing is slightly tapered and ranges from 0.85 wavelengths at the center of the array to 0.83 wavelengths toward the ends of the array. The spacing is symmetric about the center of the array.

lobes.

Figure 3.4 shows the slightly tapered reduced array element spacing which begins with 0.85 wavelengths at the center of the array and decreases to 0.83 wavelengths toward the ends of the array. The spacing is symmetric about the center of the array. The slightly tapered spacing is replaced with uniform spacing selected such that both configurations yield the same 0.0143 m aperture width. Here, the uniform element spacing is set to 0.84 wavelengths and Figure 3.5 shows the comparable beam patterns. The tapered spacing can be replaced with uniform spacing.

3.2- The Initial Array Apodization Function:

The 20 element array in Section 3.1 is apodized by a Chebyshev window of length 20 configured
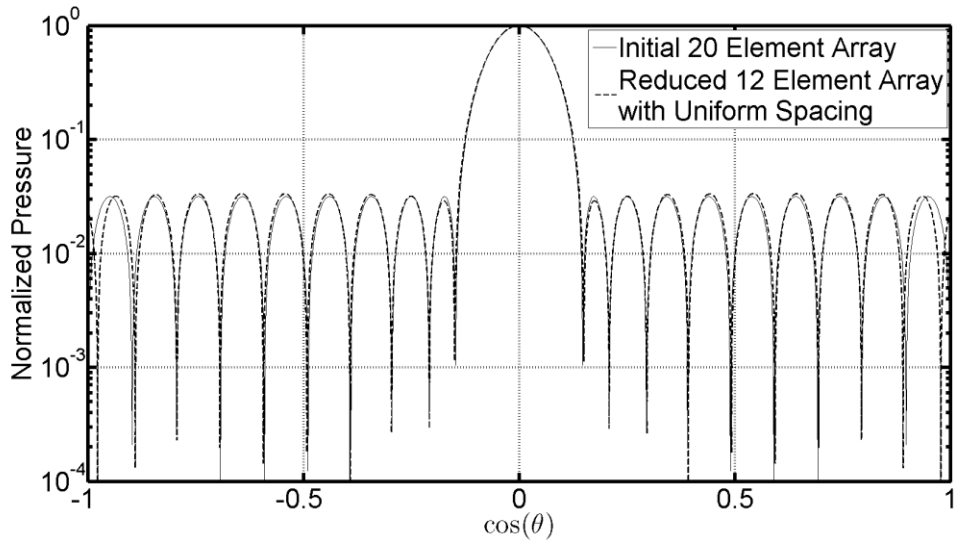
Figure 3.5: Initial Array and Uniformly-spaced Reduced Array
The reduced array is configured with 12 elements and a uniform center-to-center element spacing of 0.84 wavelengths. The uniformly-spaced reduced array beam pattern compares well to the initial array beam pattern across the target-region.

with 30 dB constant-amplitude side lobes. Figure 3.6 shows the length 20 window and a length 128 window. As the window size increases, delta functions appear at the ends of the window and the delta function amplitude increases with increasing window size until the delta functions achieve a maximum amplitude of 1.0. Window sizes larger than the minimum window size that achieves a delta function amplitude of 1.0 yield a reduced amplitude at the center of the window. Here, an increasing window size produces a decreasing amplitude at the center of the window. The bottom panel in Figure 3.6 illustrates the reduction in amplitude at the center of the length 128 window from 1.0 to 0.68 and the maximum delta function amplitude of 1.0. The delta functions result from the design method employed in the Matlab Chebyshev window function [20]. The presence of the delta functions and the reduction in amplitude at the center of the window generate undesirable artifacts in the reduced array element location and element weight parameters.
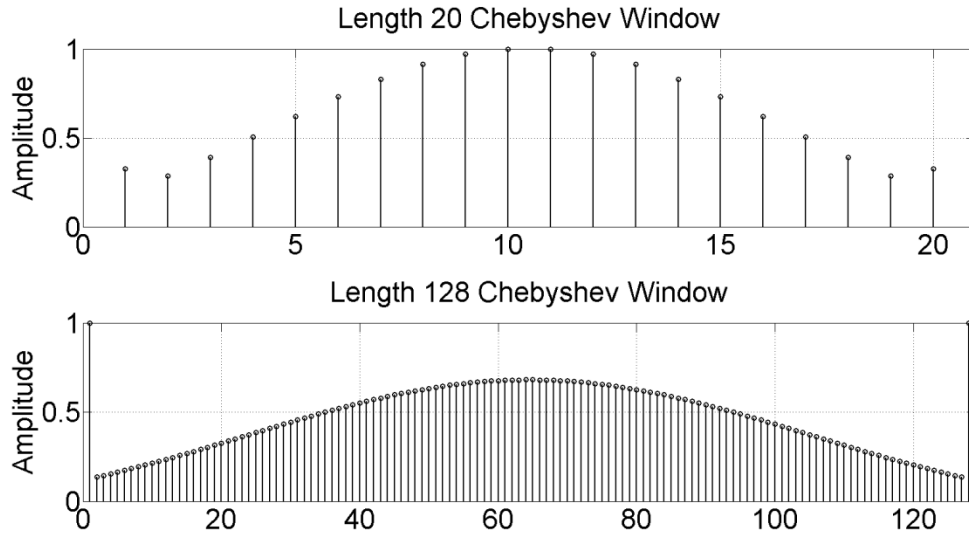
Figure 3.6: The Chebyshev Window
The length 20 Chebyshev window does not exhibit delta functions at the ends due to the shorter length of the window. The length 128 Chebyshev window shows delta functions at the ends and a reduction in amplitude to 0.68 at the center of the window.

A preferred apodization window would provide a comparable main lobe width to the Chebyshev 30 dB window without the presence of delta functions at the ends and without the reduction in amplitude at the center of the array. A normalized Taylor window approximates the desired behavior. Figure 3.7 shows a 128 element linear array pressure beam pattern resulting from a Chebyshev 30 dB apodization window and the beam pattern from the same array apodized by a normalized Taylor window. The Chebyshev and Taylor windows provide a comparable beam pattern for the main lobe and low-angle side lobe regions with the Chebyshev window providing a slightly lower amplitude at the main lobe peak. While the Taylor window beam pattern amplitude decreases with increasing distance from the main lobe, thermal therapy does not require a constant-amplitude side lobe.

Figure 3.8 compares the length 128 Chebyshev 30 dB apodization window and the length 128
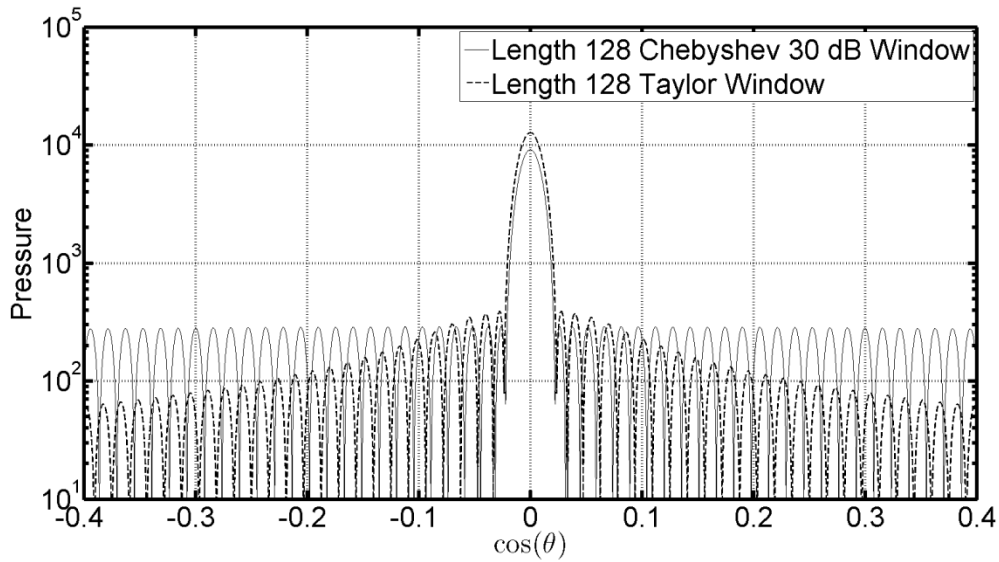
Figure 3.7: A 128 Element Linear Array Apodization Comparison
A 128 element linear array is apodized by a Chebyshev and a normalized Taylor window. The array is broadside-steered and driven by a 1.0 MHz continuous-wave excitation. The normalized Taylor window provides a comparable main lobe and low-angle side lobes to the Chebyshev window.

normalized Taylor apodization window. The Taylor window does not exhibit delta functions and is normalized such that the center of the window provides unity amplitude. The 128, 64, and 32 element initial arrays in Sections 3.3, 3.4, and 3.5, respectively, will be apodized with a normalized Taylor window prior to applying the matrix pencil method.

The delta functions at the ends of the window considered here apply to the constant side lobe amplitude Chebyshev window. Alternate window types like the Hanning, Hamming, and Flattop windows do not suffer from delta functions at the ends and can be implemented directly.

3.3- Reducing a 128 Element Linear Array:

A 128 element linear array is reduced to an array of fewer elements with the total least-squares
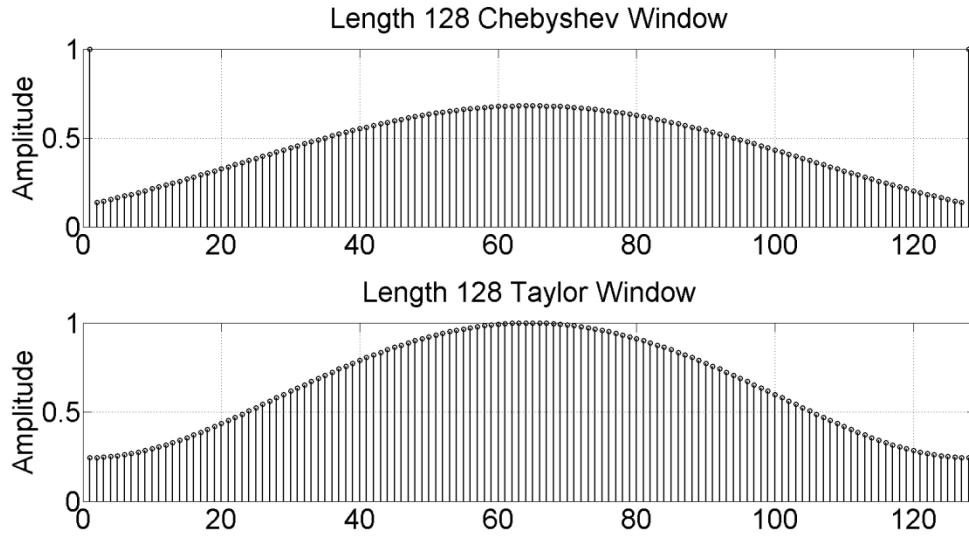
Figure 3.8: The Chebyshev and Taylor Windows

The length 128 Chebyshev window provides delta functions at the ends and a reduced amplitude at the center of the window, while the normalized Taylor window yields a unity amplitude at the center of the window without delta functions.

matrix pencil method presented in Section 2.1.3. In Section 3.1, the array elements are modeled as point-sources and the far-field approximation is applied. Here, and in Sections 3.4 and 3.5, the array is configured with non-zero-area elements and the target-region pressure is calculated without applying the far-field approximation.

The initial linear array is configured with 128 rectangular elements. The center-to-center element spacing is 0.5 wavelengths with all elements located symmetric about the center of the array. Each element is 16 wavelengths tall and 0.4 wavelengths wide with a 0.1 wavelength kerf. The array is steered broadside and apodized by a normalized Taylor window. Here, the normalized Taylor window approximates the main lobe and low-angle side lobes of a Chebyshev window as described in Section 3.2. The speed of sound in the medium is 1540 m/s and the array is operated at 1.0 MHz, yielding a wavelength of 0.0015 m. The array aperture is

0.0978 m. wide.

The variable M represents the number of elements in the initial array and L specifies the pencil parameter. Here, M = L = 128 and the number of uniformly-spaced target-region points, N, is set to 257. The target-region is located in the far-field of the array at a radius of 1000 times the aperture width (97.8 m) from the center of the array.

The FOCUS software calculates the pressure at each of the 257 target-region points. The pressure samples populate a Hankel matrix and a singular value decomposition is computed. Figure 3.9 provides the singular values in order of decreasing magnitude and Figure 3.10 shows the MSE calculated between the initial and reduced array beam patterns. The MSE is calculated across the entire beam pattern. The number of elements in the reduced array corresponding to the minimum MSE in Figure 3.10 becomes the number of principal singular values. Here, the minimum MSE of 182 occurs when the reduced array contains 66 elements resulting in 66 principal singular values. The reduced array aperture area is increased to match the initial array aperture area. Here, the initial array aperture area is 0.00190 m^2 which is preserved by increasing the reduced array element width to 0.78 wavelengths. Both arrays employ elements which are 16 wavelengths tall.

Figure 3.11 illustrates the close comparison between the initial array and the reduced array beam patterns. The continuous-wave pressure is calculated in the far-field for each of the 257 points in the semi-circular target-region. The initial array is configured with 128 elements which are 0.4 wavelengths wide with a 0.5 wavelength center-to-center spacing. The 66 reduced array elements are 0.78 wavelengths wide with the tapered spacing shown in Figure
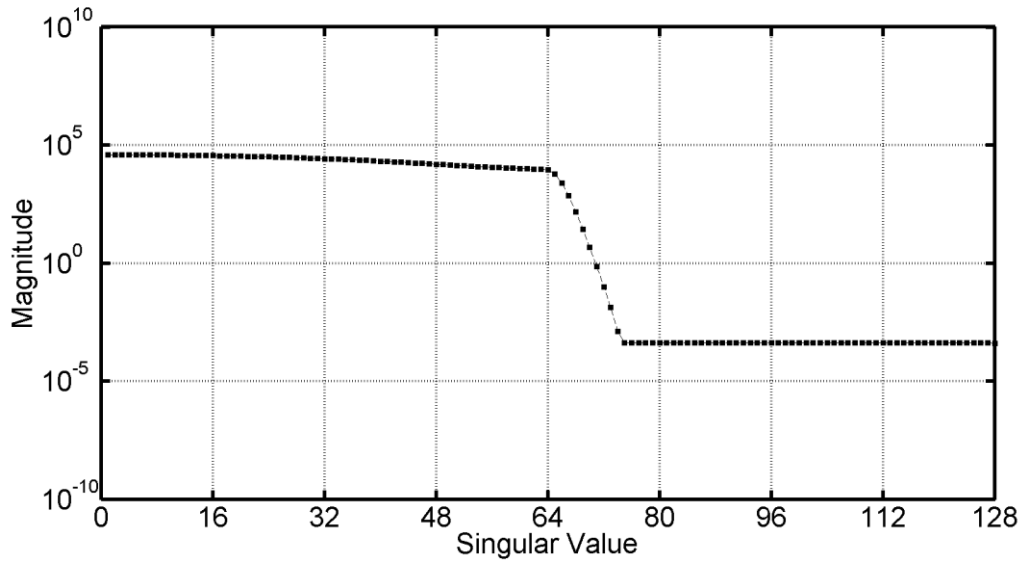
Figure 3.9: Hankel Matrix Singular Values

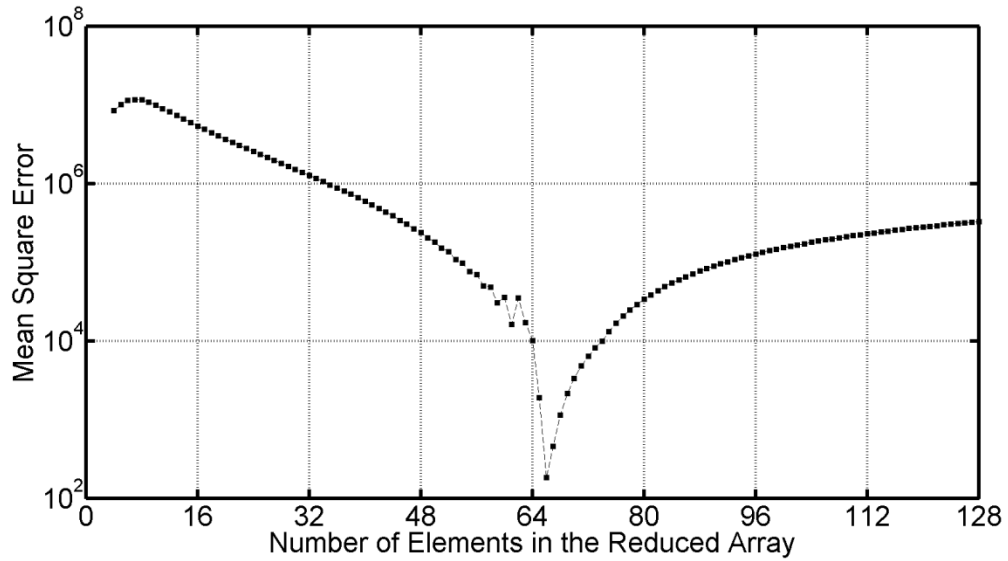*The 128 element initial array Hankel matrix singular values range from greater than 10^{4} to less than 10^{-3}.*

Figure 3.10: Mean square Error Between the Initial and Reduced Array
Beam Patterns

A reduced array with 66 elements provides the minimum mean square error of 182.
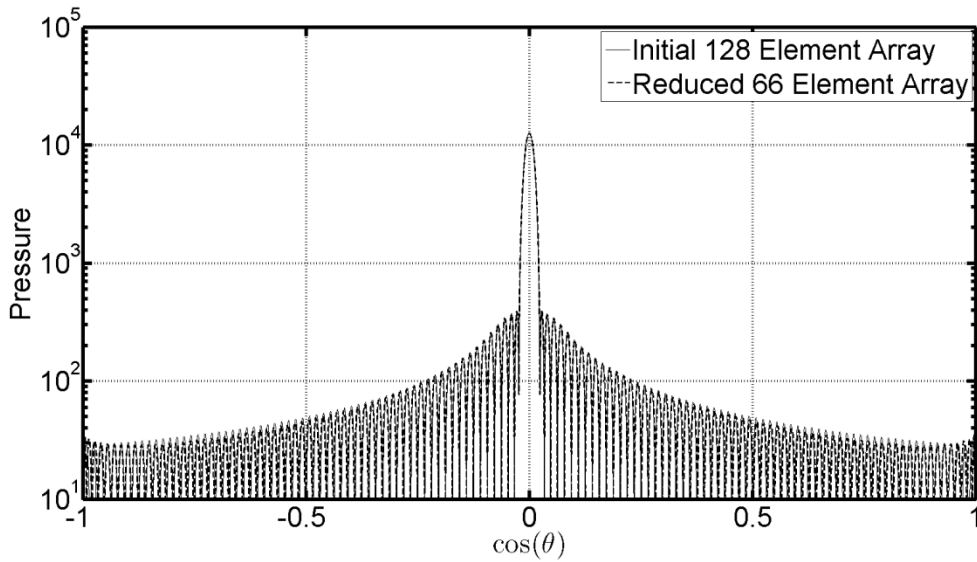
35

Figure 3.11: Initial Array and Reduced Array Pressure Beam Patterns
The simulated far-field pressure beam patterns for the 128 element initial array and the 66 element reduced array. The beam patterns are nearly indistinguishable across the target-region. Grating lobes do not appear.

3.14. Grating lobes do not appear. Figure 3.12 expands the $\cos(\Theta)$ axis to confirm the close comparison for the main lobe and the first 48 side lobes.

Figure 3.13 shows the reduced array apodization which ranges from 1.00 at the center of the array to 0.21 at the ends of the array. Reducing the excitation amplitude toward the ends of the array relative to the center of the array reduces the side lobe amplitude at the expense of an increased main lobe width. Here, the apodization is symmetric about the center of the array and closely approximates a length 66 normalized Taylor window.

Figure 3.14 provides the slightly tapered reduced array element spacing, which begins with 0.97 wavelengths at the center of the array and decreases to 0.89 wavelengths at the ends of the array. All array elements except four are spaced between 0.95 to 0.97 wavelengths apart. The spacing is symmetric about the center of the array.

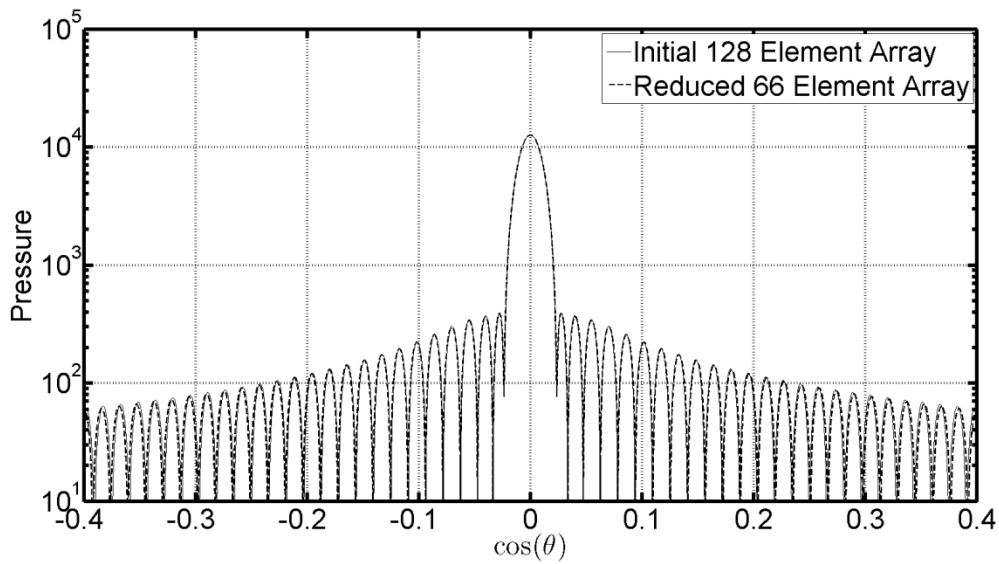**Figure 3.12: Expanded View of Figure 3.11**

The main lobe and 48 lowest-angle side lobes illustrate the close agreement between the initial array and the reduced array beam patterns.
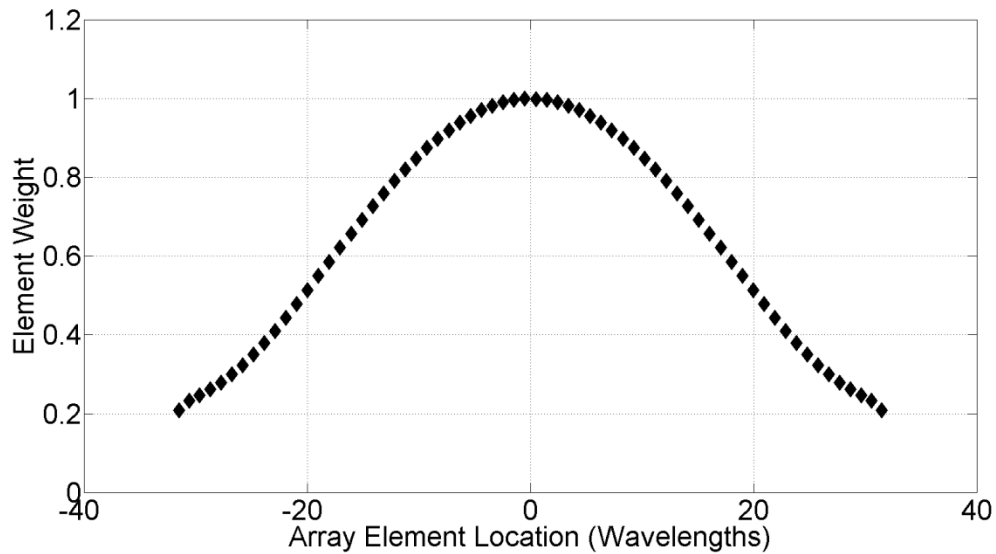

Figure 3.13: Reduced Array Apodization

The 66 element reduced array apodization decreases from 1.00 at the center of the array to 0.21 at the ends of the array and is symmetric about the center of the array.

Figure 3.15 shows the initial and reduced arrays in Cartesian coordinates. The center of both arrays is located at the center of the coordinate system, and all elements in each array are 16 wavelengths tall. The initial array elements are 0.40 wavelengths wide and uniformly-spaced at 0.5 wavelengths with a 0.1 wavelength kerf. The reduced array elements are 0.78 wavelengths wide with a slightly tapered spacing ranging from 0.97 wavelengths at the center of the array to 0.89 wavelengths at the ends of the array. The initial array and reduced array apertures are 0.0978 m and 0.0969 m wide, respectively.

Replacing the slightly tapered spacing with uniform spacing set to 0.97 wavelengths yields the same 0.0969 m aperture width. Figure 3.16 compares the uniformly-spaced reduced array to the initial array and illustrates a comparable beam pattern for the main lobe and low-angle side lobe regions. The expanded view in Figure 3.17 confirms the comparison at low-angles. At larger angles, the uniformly-spaced array provides increasing pressure amplitude which achieves a maximum pressure of approximately 1% of the main lobe peak. The increased pressure at larger angles is acceptable in thermal therapy.

3.4- Reducing a 64 Element Linear Array:

A 64 element linear array is reduced to an array of fewer elements with the total least-squares matrix pencil method presented in Section 2.1.3. The initial linear array is configured with 64 rectangular elements. The center-to-center element spacing is 0.5 wavelengths with all elements located symmetric about the center of the array. Each element is 16 wavelengths tall and 0.4 wavelengths wide with a 0.1 wavelength kerf. The array is steered broadside and apodized by a normalized Taylor window. The normalized Taylor window approximates the
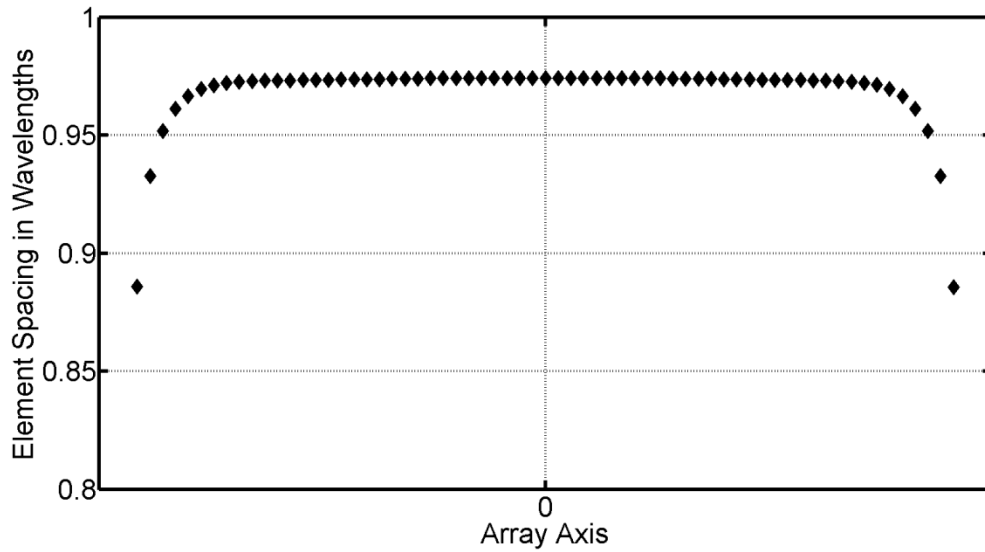
Figure 3.14: Reduced Array Element Spacing

The 66 element reduced array element spacing is slightly tapered and ranges from 0.97 wavelengths at the center of the array to 0.89 wavelengths at the ends of the array. The spacing is symmetric about the center of the array.
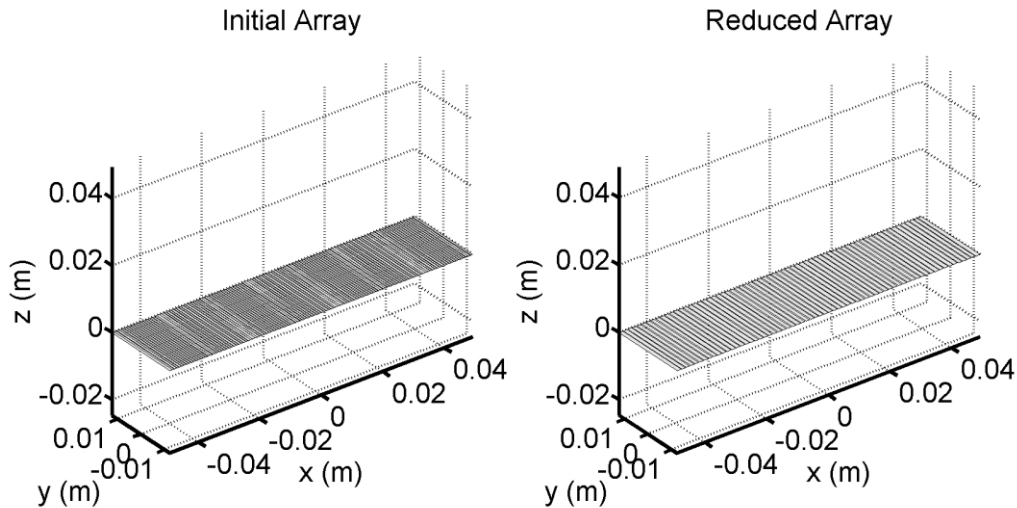
Figure 3.15: Initial 128 Element Array and Reduced 66 Element Array

The initial array elements are 0.40 wavelengths wide with 0.5 wavelength center-to-center spacing and a 0.1 wavelength kerf. The reduced array elements are 0.78 wavelengths wide with a slightly tapered spacing decreasing from 0.97 wavelengths at the center of the array to 0.89 wavelengths at the ends of the array. The initial and reduced array apertures are 0.0978 m and 0.0969 m wide, respectively. Both arrays employ 16 wavelengths tall elements.
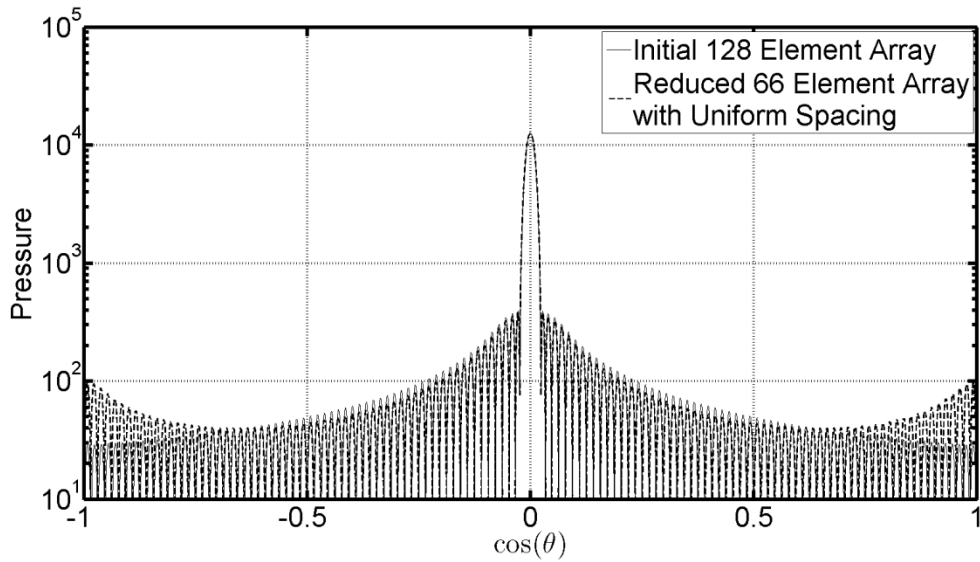
Figure 3.16: Initial Array and Uniformly-spaced Reduced Array

The 66 element reduced array tapered spacing is replaced with uniform spacing set to 0.97 wavelengths to preserve the 0.0969 m aperture width. The initial array and the uniformly-spaced reduced array provide comparable beam patterns for the main lobe and low-angle side lobe regions. The increased pressure at larger angles achieves approximately 1% of the main lobe peak pressure and is acceptable in thermal therapy.

Figure 3.17: Expanded View of Figure 3.16

Replacing the 66 element reduced array tapered element spacing with uniform spacing set to 0.97 wavelengths achieves a comparable beam pattern to the initial array for the main lobe and low-angle side lobe regions.

40

main lobe and low-angle side lobes of a Chebyshev window. Section 3.2 considers the Taylor window in more detail. The speed of sound i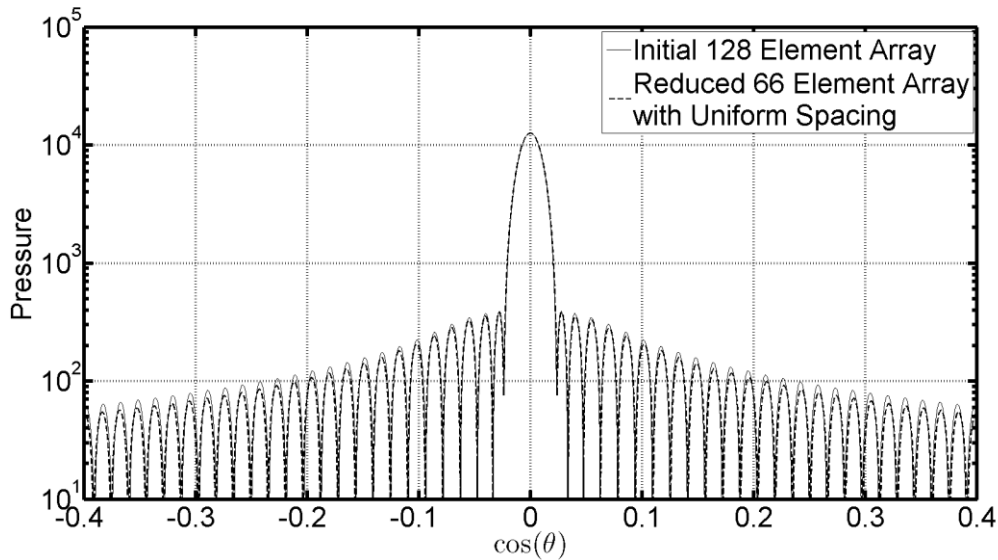n the medium is 1540 m/s and the array is operated at 1.0 MHz, yielding a wavelength of 0.0015 m. The array aperture is 0.0485 m wide.

The variable M represents the number of elements in the initial array and L specifies the pencil parameter. Here, M = L =64 and the number of uniformly-spaced target-region points, N, is set to 129. The target-region is located in the far-field of the array at a radius of 1000 times the aperture width (48.5 m) from the center of the array.

The FOCUS software calculates the pressure at each of the 129 target-region points. The pressure samples populate a Hankel matrix and a singular value decomposition is computed. Figure 3.18 provides the singular values in order of decreasing magnitude and Figure 3.19 shows the MSE calculated between the initial and reduced array beam patterns. The MSE is calculated across the entire beam pattern. The number of elements in the reduced array corresponding to the minimum MSE in Figure 3.19 becomes the number of principal singular values. Here, the minimum MSE of 498 occurs when the reduced array contains 34 elements resulting in 34 principal singular values.

The reduced array aperture area is increased to match the initial array aperture area. Here, the initial array aperture area is 0.00097 m^2 which is preserved by increasing the reduced array element width to 0.75 wavelengths. Both arrays employ elements which are 16 wavelengths tall.

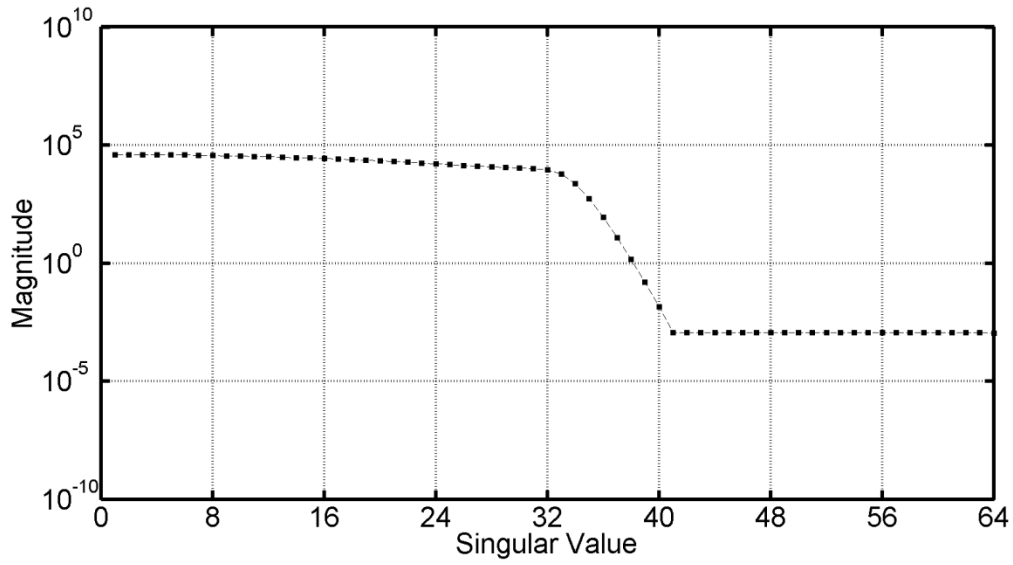Figure 3.20 illustrates the close comparison between the initial array and the reduced array

Figure 3.18: Hankel Matrix Singular Values

The 64 element initial array Hankel matrix singular values range from greater than 10^4 to approximately 10^{-3}.
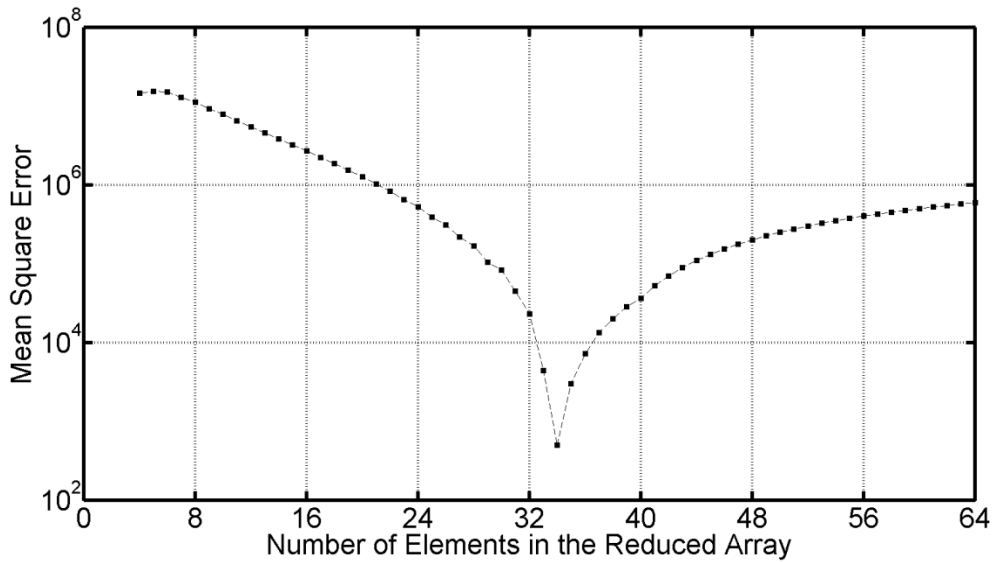

Figure 3.19: Mean square Error Between the Initial and Reduced Array

Beam Patterns

A reduced array with 34 elements provides the minimum mean square error of 498.

42

beam patterns. The continuous-wave pressure is calculated in the far-field for each of the 129 points in the semi-circular target-region. The initial array is configured with 64 elements which are 0.4 wavelengths wide with a 0.5 wavelength center-to-center spacing. The 34 reduced array elements are 0.75 wavelengths wide with the tapered spacing shown in Figure 3.22. Grating lobes do not appear.

Figure 3.21 shows the reduced array apodization which ranges from 1.00 at the center of the array to 0.21 at the ends of the array. Reducing the excitation amplitude toward the ends of the array relative to the center of the array reduces the side lobe amplitude at the expense of an increased main lobe width. Here, the apodization is symmetric about the center of the array and closely approximates a length 34 normalized Taylor window.

Figure 3.22 shows the slightly tapered reduced array element spacing which begins with 0.95 wavelengths at the center of the array and decreases to 0.87 wavelengths at the ends of the array. All array elements except four are spaced between 0.93 to 0.95 wavelengths apart. The spacing is symmetric about the center of the array.

Figure 3.23 shows the initial and reduced arrays in Cartesian coordinates. The center of both arrays is located at the center of the coordinate system and all elements in each array are 16 wavelengths tall. The initial array elements are 0.40 wavelengths wide and uniformly-spaced at 0.5 wavelengths with a 0.1 wavelength kerf. The reduced array elements are 0.75 wavelengths wide with a slightly tapered spacing ranging from 0.95 wavelengths at the center of the array to 0.87 wavelengths at the ends of the array. The initial array and reduced array apertures are 0.0485 m and 0.0478 m wide, respectively.
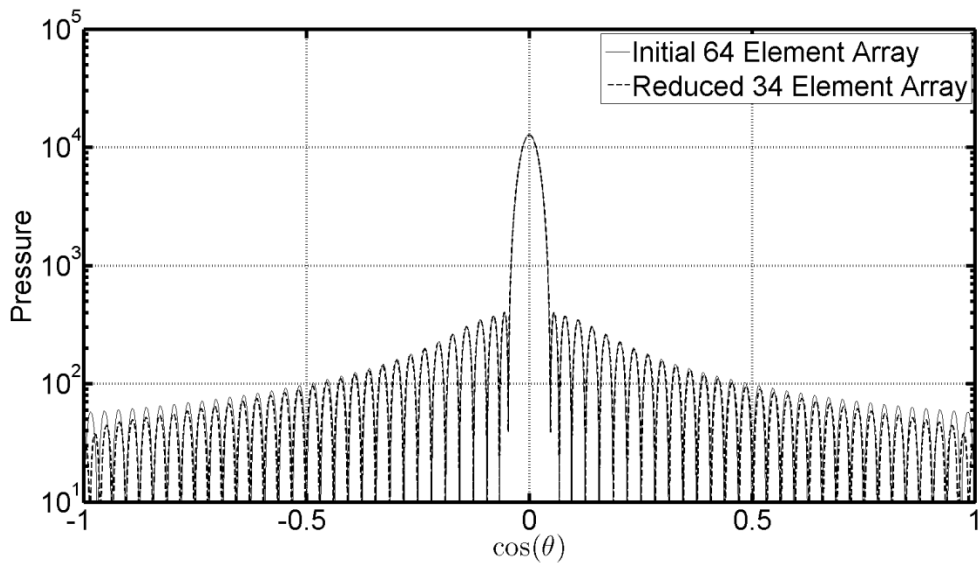
Figure 3.20: Initial Array and Reduced Array Pressure Beam Patterns

The simulated far-field pressure beam patterns for the 64 element initial array and the 34 element reduced array. The beam patterns are similar across the target-region and grating lobes do not appear.
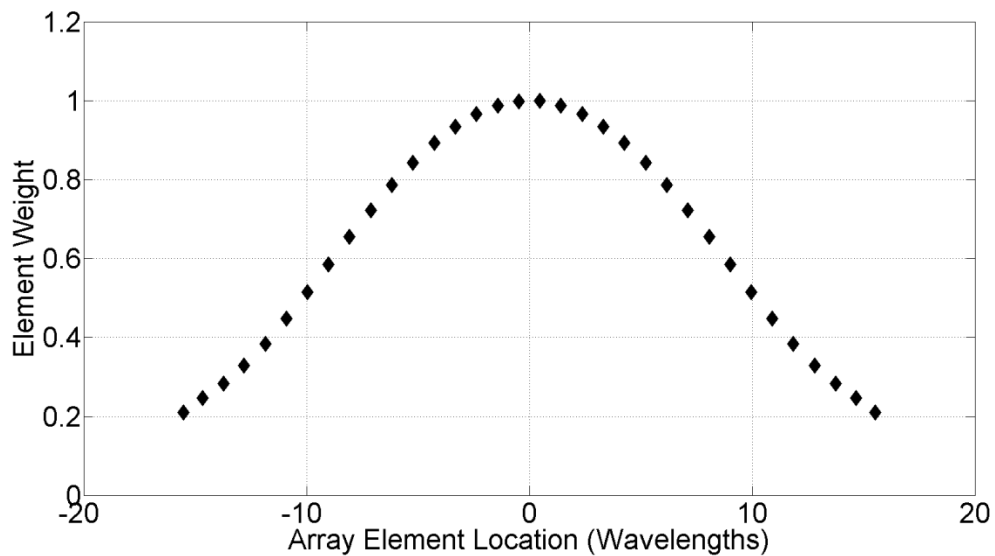
Figure 3.21: Reduced Array Apodization

The 34 element reduced array apodization decreases from 1.00 at the center of the array to 0.21 at the ends of the array and is symmetric about the center of the array.
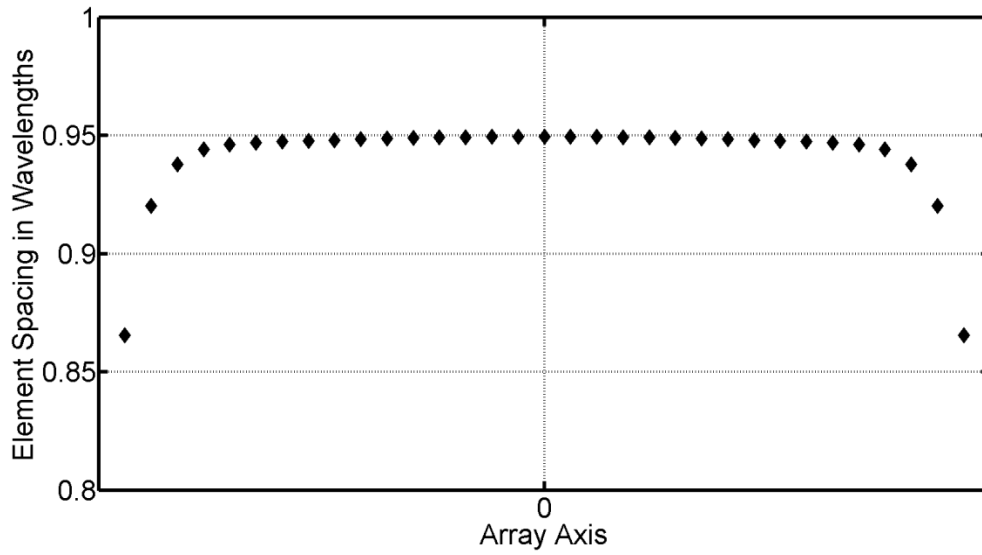
Figure 3.22: Reduced Array Element Spacing

The 34 element reduced array element spacing is slightly tapered and ranges from 0.95 wavelengths at the center of the array to 0.87 wavelengths at the ends of the array. The spacing is symmetric about the center of the array.
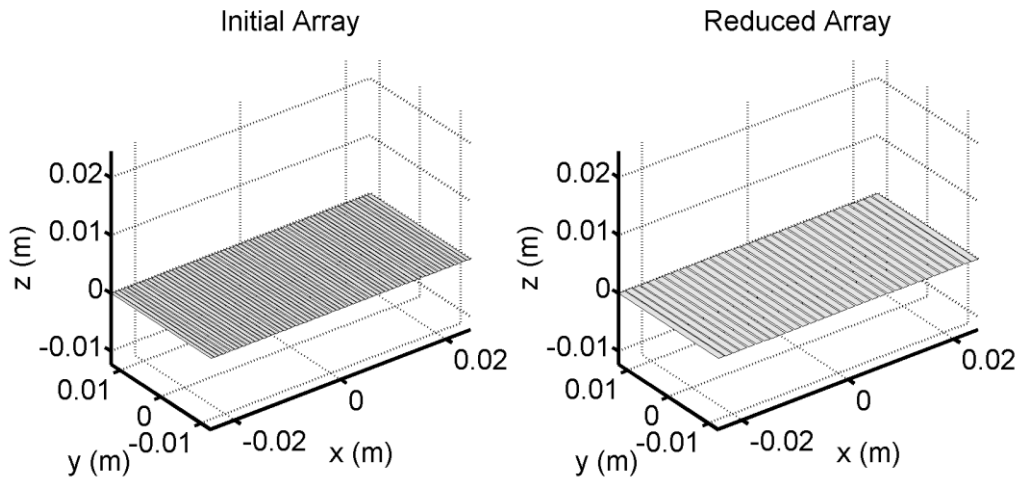
Figure 3.23 Initial 64 Element Array and Reduced 34 Element Array

The initial array elements are 0.40 wavelengths wide and are uniformly-spaced 0.5 wavelengths apart with a 0.1 wavelength kerf. The reduced array elements are 0.75 wavelengths wide with a slightly tapered spacing decreasing from 0.95 wavelengths at the center of the array to 0.87 wavelengths at the ends of the array. The initial and reduced array apertures are 0.0485 m and 0.0478 m wide, respectively. Both arrays employ 16 wavelengths tall elements.

45

Replacing the slightly tapered spacing with uniform spacing set to 0.94 wavelengths yields the same 0.0478 m aperture width. Figure 3.24 compares the uniformly-spaced reduced array to the initial array and illustrates a comparable beam pattern for the main lobe and low-angle side lobe regions.   At larger angles, the uniformly-spaced array provides increasing pressure amplitude which achieves a maximum pressure of approximately 1% of the main lobe peak. The increased pressure at larger angles is acceptable in thermal therapy.

3.5- Reducing a 32 Element Linear Array:

A 32 element linear array is reduced to an array of fewer elements with the total least-squares matrix pencil method presented in Section 2.1.3. The initial linear array is configured with 32 rectangular elements.   The center-to-center element spacing is 0.5 wavelengths with all elements located symmetric about the center of the array. Each element is 16 wavelengths tall and 0.4 wavelengths wide with a 0.1 wavelength kerf. The array is steered broadside and apodized by a normalized Taylor window. Here, the normalized Taylor window approximates the main lobe and low-angle side lobes of a Chebyshev window. The speed of sound in the medium is 1540 m/s and the array is operated at 1.0 MHz leading to a wavelength of 0.0015 m. The array aperture is 0.0239 m wide.

The variable M represents the number of elements in the initial array and L specifies the pencil parameter. Here, M = L = 32 and the number of uniformly-spaced target-region points, N, is set to 65. The target-region is located in the far-field of the array at a radius of 1000 times the aperture width (23.9 m) from the center of the array.

The FOCUS software calculates the pressure at each of the 65 target-region points. The
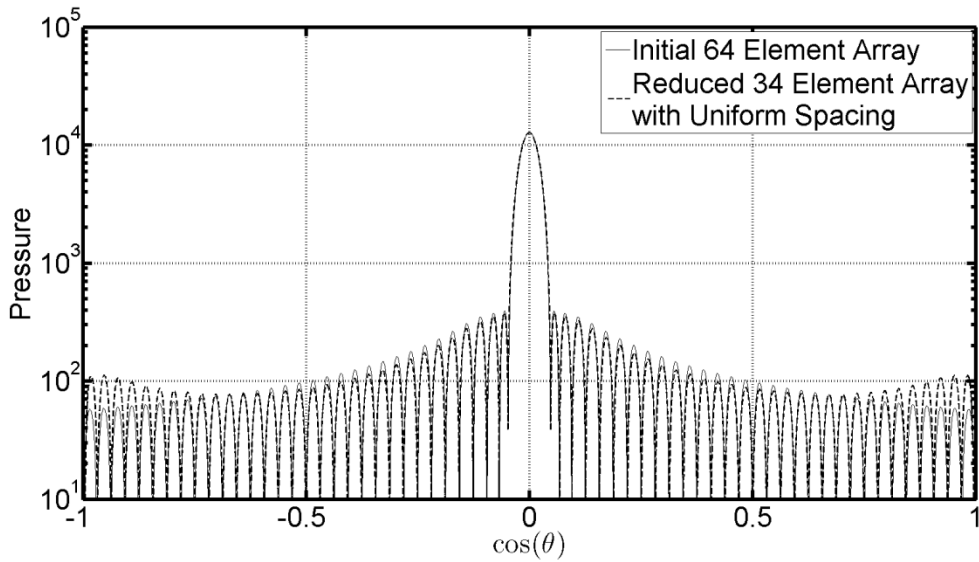
Figure 3.24: Initial Array and Uniformly-spaced Reduced Array
The 34 element reduced array with tapered spacing is replaced with an array with uniform
spacing set to 0.94 wavelengths. The initial array and the uniformly-spaced reduced array
provide comparable beam patterns for the main lobe and low-angle side lobes. The
increased pressure at larger angles achieves approximately 1% of the main lobe peak
pressure and is acceptable in thermal therapy.

pressure samples populate a Hankel matrix and a singular value decomposition is computed.

Figure 3.25 provides the singular values in order of decreasing magnitude and Figure 3.26

shows the MSE calculated between the initial and reduced array beam patterns. The MSE is

calculated across the entire beam pattern. The number of elements in the reduced array

corresponding to the minimum MSE in Figure 3.26 defines the number of principal singular

values. Here, the minimum MSE of 1646 occurs when the reduced array contains 18 elements,

resulting in 18 principal singular values.

The reduced array aperture area is increased such that the initial array and the reduced array

provide the same aperture area. Here, the initial array aperture area is 0.00049 m^2 which is

preserved by increasing the reduced array element width to 0.71 wavelengths. Both arrays

47
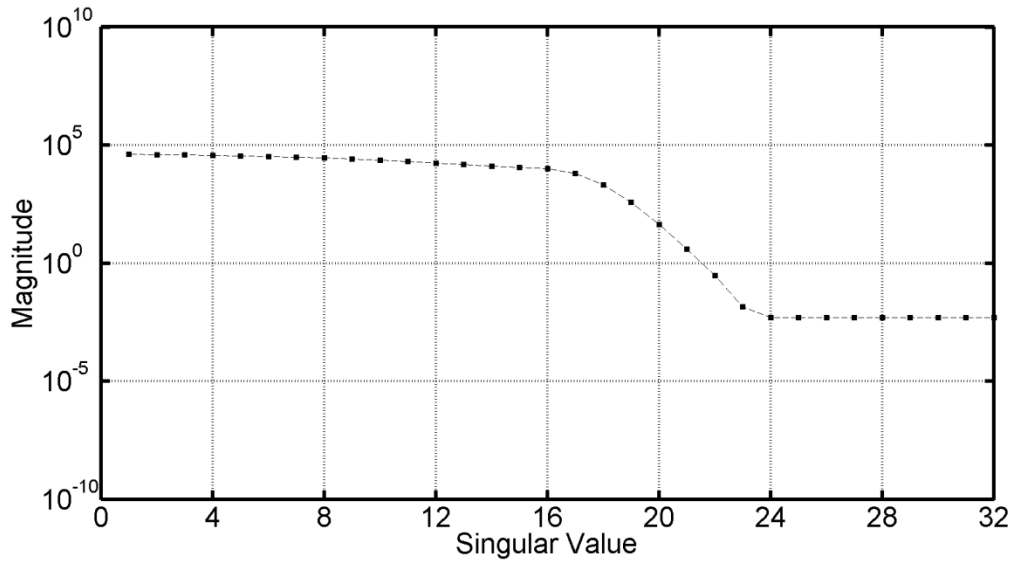
Figure 3.25: Hankel Matrix Singular Values

The 32 element initial array Hankel matrix singular values range from greater than 10^4 to less than 10^{-2}.
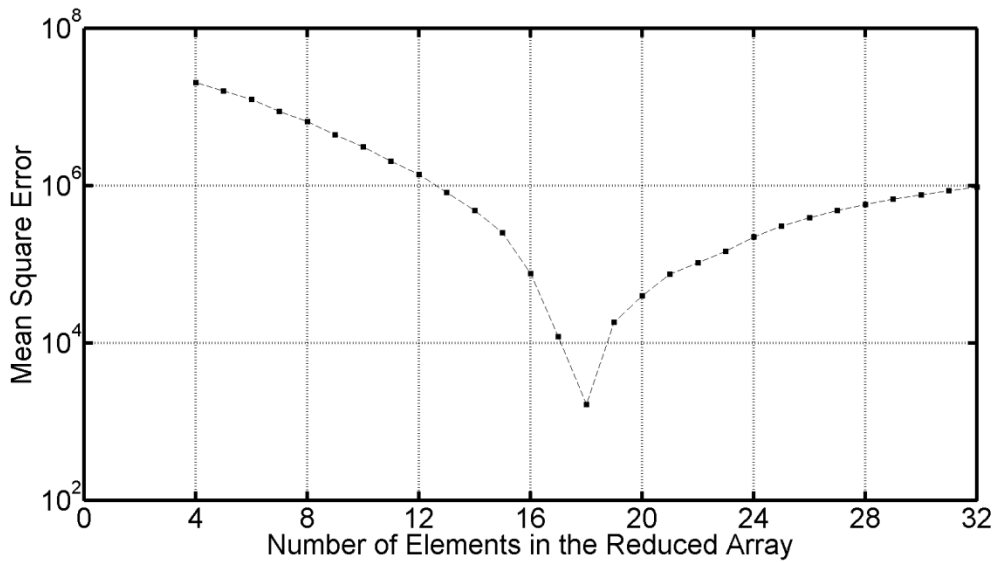
Figure 3.26: Mean square Error Between the Initial and Reduced Array Beam Patterns

A reduced array with 18 elements provides the minimum mean square error of 1646.

employ elements which are 16 wavelengths tall.

Figure 3.27 illustrates the close comparison between the initial array and the reduced array beam patterns. The continuous-wave pressure is calculated in the far-field for each of the 65 points in the semi-circular target-region. The initial array is configured with 32 elements which are 0.4 wavelengths wide with a 0.5 wavelength center-to-center spacing. The 18 reduced array elements are 0.71 wavelengths wide with the tapered spacing shown in Figure 3.29. Grating lobes do not appear.

Figure 3.28 shows the reduced array apodization which ranges from 1.00 at the center of the array to 0.22 at the ends of the array. Here, the apodization is symmetric about the center of the array and closely approximates a length 18 normalized Taylor window. Figure 3.29 shows the slightly tapered reduced array element spacing which begins with 0.90 wavelengths at the center of the array and decreases to 0.84 wavelengths at the ends of the array. All array elements except four are spaced between 0.89 to 0.90 wavelengths apart. The spacing is symmetric about the center of the array.

Figure 3.30 shows the initial and reduced arrays in Cartesian coordinates. The center of both arrays is located at the center of the coordinate system and all elements in each array are 16 wavelengths tall. The initial array elements are 0.40 wavelengths wide and uniformly-spaced at 0.5 wavelengths with a 0.1 wavelength kerf. The reduced array elements are 0.71 wavelengths wide with a slightly tapered spacing ranging from 0.90 wavelengths at the center of the array to 0.84 wavelengths at the ends of the array. The initial array and reduced array apertures are 0.0239 m and 0.0233 m wide, respectively.
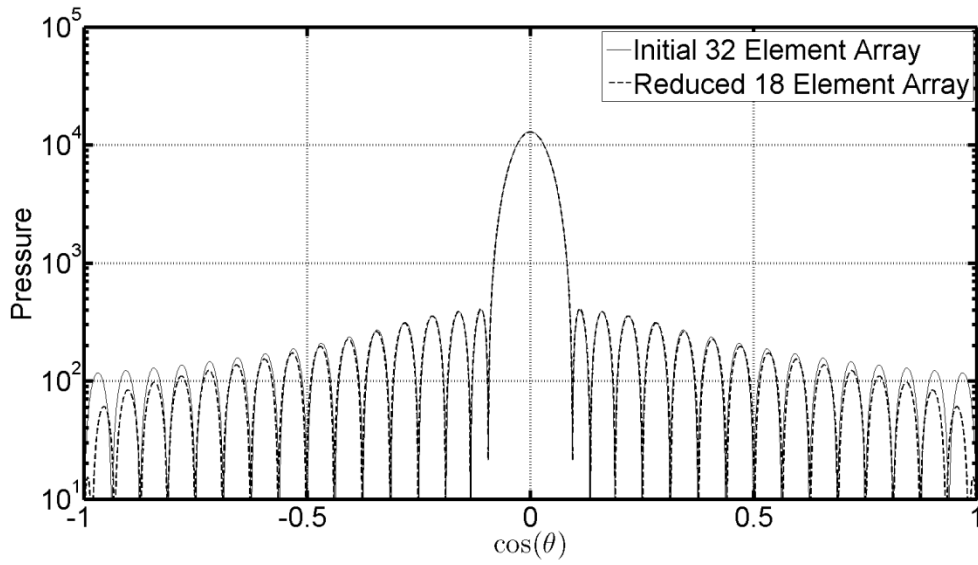
Figure 3.27: Initial Array and Reduced Array Pressure Beam Patterns
The simulated far-field pressure beam patterns for the 32 element initial array and the 18 element reduced array. The beam patterns are comparable across the target-region and grating lobes do not appear.
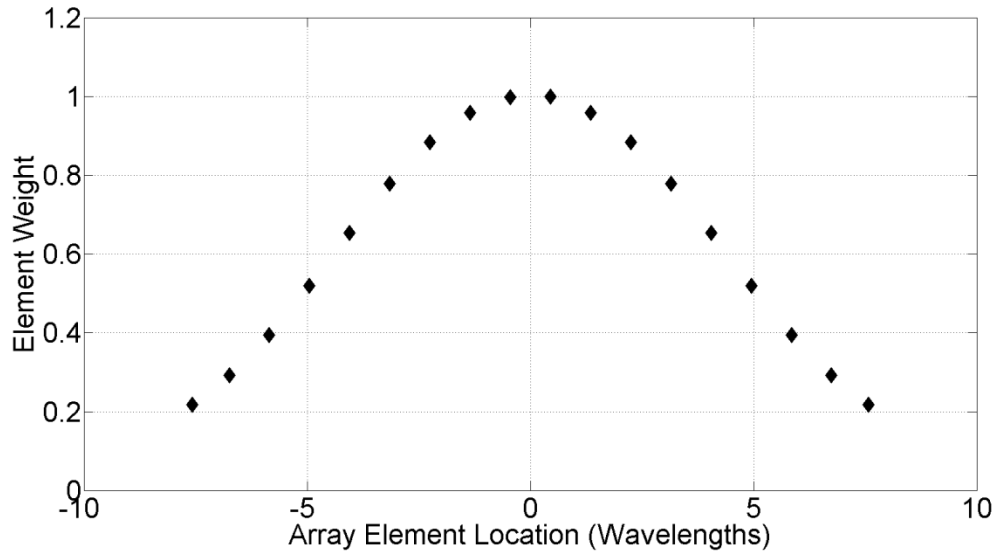
Figure 3.28: Reduced Array Apodization
The 18 element reduced array apodization decreases from 1.00 at the center of the array to 0.22 at the ends of the array and is symmetric about the center of the array.
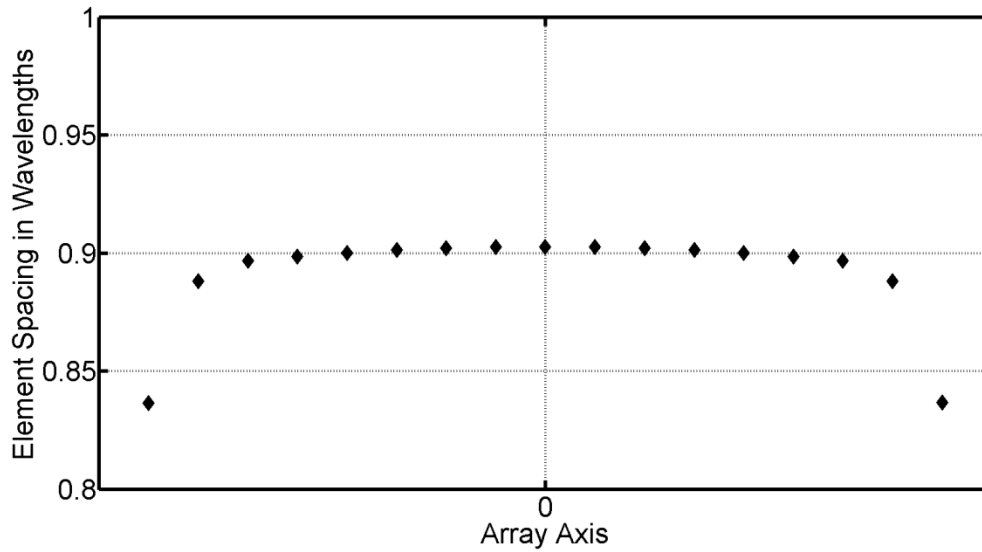
Figure 3.29: Reduced Array Element Spacing

The 18 element reduced array is slightly tapered. The element spacing ranges from 0.90 wavelengths at the center of the array to 0.84 wavelengths at the ends of the array. The spacing is symmetric about the center of the array.
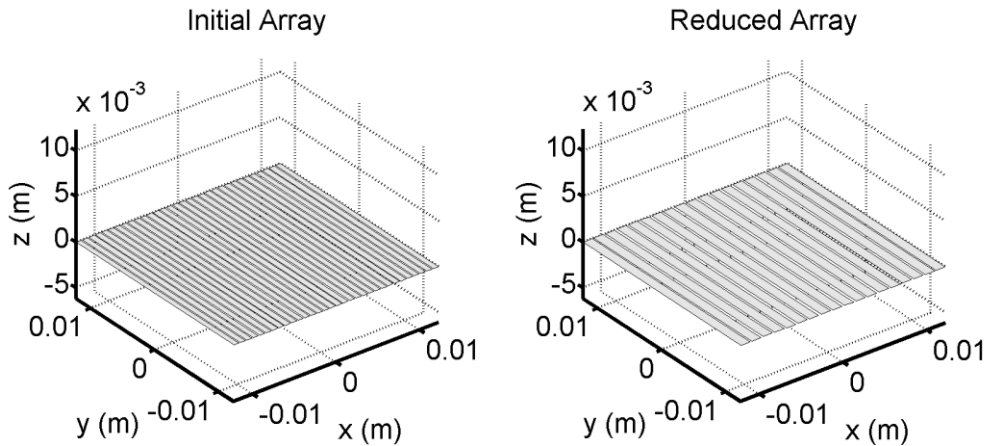
Figure 3.30: Initial 32 Element Array and Reduced 18 Element Array

The initial array elements are 0.40 wavelengths wide and are uniformly-spaced 0.5 wavelengths apart with a 0.1 wavelength kerf. The reduced array elements are 0.71 wavelengths wide with a slightly tapered spacing decreasing from 0.90 wavelengths at the center of the array to 0.84 wavelengths at the ends of the array. The initial and reduced array apertures are 0.0239 m and 0.0233 m wide, respectively. Both arrays employ 16 wavelengths tall elements.
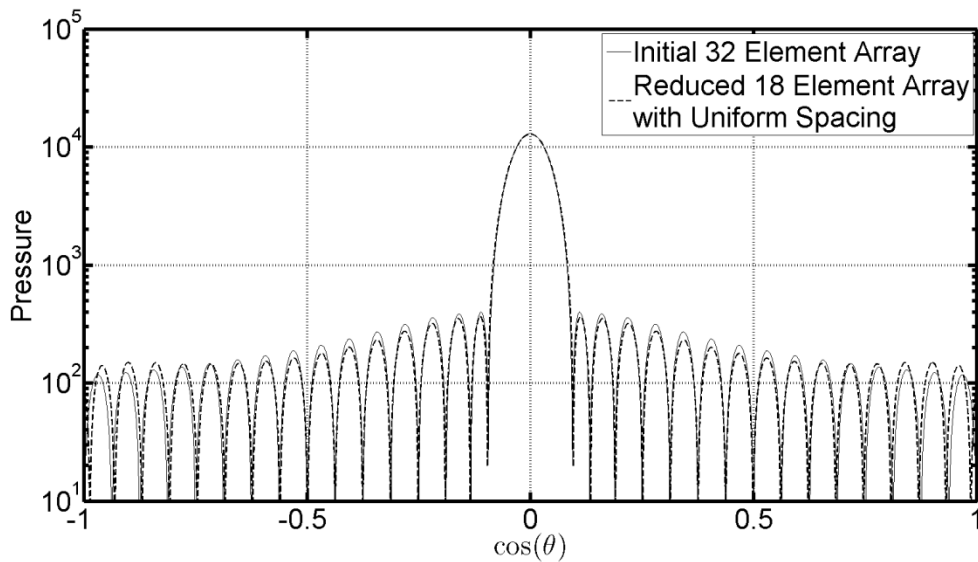
Figure 3.31: Initial Array and Uniformly-spaced Reduced Array
The 18 element reduced array with slightly tapered spacing is replaced with uniform spacing set to 0.89 wavelengths to preserve the 0.0233 m aperture width. The initial array and the uniformly-spaced reduced array provide comparable beam patterns for the main lobe and low-angle side lobe regions. The increased pressure at larger angles provided by the uniformly-spaced array is acceptable in thermal therapy.

Replacing the slightly tapered spacing with uniform spacing set to 0.89 wavelengths yields the same 0.0233 m aperture width. Figure 3.31 compares the uniformly-spaced reduced array to the initial array illustrating a comparable beam pattern for the main lobe and low-angle side lobe regions. At larger angles, the uniformly-spaced array provides increasing pressure amplitude which achieves a maximum pressure of approximately 1% of the main lobe peak. The increased pressure at larger angles is acceptable in thermal therapy.

3.6- Simulation Time:

The 128 element array reduces to a 66 element array in 0.21 seconds on an Antec computer equipped with an Intel Core 2 Quad CPU operating at 2.66 GHz, 8.00 GB of RAM and the Windows 7 Professional 64 bit operating system. Version 0.332 of the FOCUS software is

running in Matlab version 7.10.0.499 (R2010a).

Chapter 4 – Discussion

4.1- Reducing the Number of Ultrasound Phased Array Elements with the Matrix Pencil Method:

The matrix pencil method is a linear, non-iterative approach which can directly reduce the number of elements in a linear, broadside-steered phased array while maintaining a comparable far-field pressure beam pattern between the initial array and the reduced array. The matrix pencil method begins by populating a Hankel matrix with uniformly-spaced samples from a specified pressure beam pattern. A singular value decomposition along with criteria to determine the number of principal singular values determines the number of elements in the reduced array. The solution to a generalized eigenvalue problem provides the reduced array element locations and the residues of a least-squares problem yield the reduced array apodization.

4.2- The Initial Array Apodization:

In this thesis, the matrix pencil method is employed to approximate the initial array pressure beam pattern by a reduced array configured with fewer elements. If the initial array is uniformly-apodized with a very small kerf, the beam pattern resembles the beam pattern of a single element with a comparable aperture area. Here, the multi-element initial array can be reduced to a single element without employing the matrix pencil method. Simulation results for this special case are not included in Chapter 3. Instead, Chapter 3 presents results for initial arrays configured with Chebyshev and Taylor apodization functions, and in the case of the 128, 64, and 32 element initial arrays, the element spacing and width yield a 0.1 wavelength kerf.

4.3- The Pencil Parameter L:

The pencil parameter L influences the size of the Hankel matrix and consequently impacts the number of singular values. The parameter establishes a balance between accuracy and computational efficiency [22]. Setting L to values larger than approximately 500 can lead to long calculation times and potential numerical problems [13] while constraining L to between N/3 and N/2 returns a minimum variance in the pole parameters due to noise [7]. In particular, N/3 provides improved performance for noisy signals while N/2 is more suitable for signals with a larger signal-to-noise ratio [13].

4.4- The Singular Values and the Reduced Array Aperture Width:

Figure 3.9 shows the singular value magnitudes for the 128 element initial array in order of decreasing magnitude. As the number of elements in the reduced array increases from 4 to 64 elements, corresponding to singular values 4 to 64 in Figure 3.9, the reduced array aperture width increases almost linearly from much smaller than the initial array aperture width to approximately the same as the initial array aperture width, respectively. The increasing aperture width yields a narrowing main beam and a reducing mean square error between the initial and reduced array pressure beam patterns.

Reduced arrays configured with 65 to 74 elements correspond to the transition region between singular values 65 to 74. Here, the reduced array aperture width continues to increase and quasi-asymptotically approaches an aperture width slightly larger than the initial array aperture width. As shown in Figure 3.10, the MSE continues to decrease as the number of elements in the reduced array progresses from 65 to 66 elements, and then begins to increase with an

increasing number of elements for arrays configured with 67 to 74 elements.

Configuring a reduced array with 75 to 80 elements provides a reduced array aperture width which is approximately 1.5X the initial array aperture width. Increasing the number of elements in the reduced array from 75 to 80 yields a decreasing pressure across the entire target-region and an increasing MSE. Here and as described in Section 3.3, the initial and reduced array aperture areas and the array excitation are held constant.

Increasing the number of elements in the reduced array from 81 to 128 elements yields an aperture width which is approximately 2X the initial array aperture width. This provides a continued decrease in pressure across the target-region with a corresponding increase in the MSE. In general, since singular values 75 to 128 in Figure 3.9 are not the principal singular values, reduced arrays with 75 to 128 elements should not be selected when reducing the 128 element initial array. The reduced array aperture width and MSE behavior described here applies similarly to the 64 and 32 element initial arrays. For the one-half wavelength sampled, broadside-steered, linear arrays considered in this thesis, an alternate method to reduce the number of elements in the array is to first calculate the singular value magnitudes, list the singular values in order of decreasing magnitude, then calculate the MSE for reduced arrays with a number of elements corresponding to the singular value transition region only. For the 128 element initial array, the transition region corresponds to singular values 65 to 74 as shown in Figure 3.9. The reduced array corresponding to the minimum MSE is selected.

4.5- The Reduced Array Element Locations and the Poles:

The reduced array element locations are calculated by scaling the angle of the complex-valued

poles. As the number of elements in the reduced array increases to slightly more than one-half the number of elements in the initial array, the largest-angle poles approach an angle of $\pi/2$ radians. For the 128 element initial array, reduced arrays with fewer than 75 elements provide poles with angles less than $\pi/2$ radians. As the number of reduced array elements exceeds 75, some of the complex-valued poles achieve angles larger than $\pi/2$ radians which yield a reduced array with an aperture width larger than the initial array aperture width. As described in section 4.4, selecting a number of reduced array elements which corresponds to the number of principal singular values will ensure that the complex-valued pole angles remain less than $\pi/2$ radians, and will provide a reduced array aperture width less than or equal to the initial array aperture width.

4.6-   Replacing the Reduced Array Tapered Spacing with Uniform Spacing:

All reduced arrays synthesized in this thesis present a slightly tapered element spacing. In each case, replacing the tapered spacing with uniform spacing provides virtually the same main beam shape and width. The uniform spacing affords a slightly reduced low-angle side lobe amplitude and an increased high-angle side lobe amplitude. The maximum high-angle side lobe amplitude achieves approximately 1% of the main beam amplitude and is considered to be acceptable for thermal therapy. The uniform spacing increment is set such that the uniformly-spaced reduced array aperture width is equal to the tapered spaced reduced array aperture width.

4.7-   The Focused Near-Field Case:

The total least-squares matrix pencil method may be applied to reduce the number of elements

in a linear, focused ultrasound array in the near-field. Here, the array is focused a distance of approximately one aperture width in front of the array and the beam may be steered off-axis. The array reduction algorithm begins by uniformly-sampling the focused, steered, near-field beam pattern across the target-region and constructing a Hankel matrix from the samples. The Hankel matrix undergoes a singular value decomposition.

The complex-valued pressure samples contain phase terms which represent focusing and steering. The focusing and steering phase-effects carry through the singular value decomposition into the singular vectors and the effects must be separated prior to calculating the reduced array element locations and weights. The phase-effect separation is part of the future work presented Section 4.9. Synthesizing a focused, steered array in the near-field with the matrix pencil method is expected to provide a reduced array beam pattern which compares well to the initial array beam pattern.

4.8- Dimensionality Reduction:

The singular value decomposition reduces the dimensionality of the matrix pencil method approach by separating the signal space from the noise space as described in Section 2.1.3. Further dimensionality reduction might be obtained by reducing the dimension of the initial array pressure beam pattern prior to applying the matrix pencil method. One lossy compression approach might apply a wavelet decomposition, with the wavelet decomposition down-sampling operation corresponding to reducing the array spatial sampling rate by a factor of 2. The spatial sampling rate reduction yields grating lobes when the array element spacing exceeds 1 wavelength and increasing the spacing moves the grating lobes closer to the main

lobe [23]. If the target-region field-of-view can be restricted to less than $\pm\ \pi/2$ then wavelet decomposition might become feasible.

The general approach could begin with decomposing the initial array pressure beam pattern using a discrete wavelet family such as Haar wavelets. A subset of the wavelet coefficients would become the signal that populates the Hankel Matrix. A singular value decomposition of the Hankel matrix is calculated and followed by the remaining steps described in Section 2.2. The preferred outcome is a reduced array configured with fewer elements than that which would be possible by the application of the matrix pencil method alone, and a comparable reduced array pressure beam pattern to the initial array beam pattern over the restricted target-region field-of-view.

4.9- Future Research:

Future research could consider steering away from broadside and focused arrays in the near-field. Additional array geometries including planar and spherical arrays, fully-populated and sparsely-populated, with both regular and random element spacings could be investigated. The initial array uniformly-spaced target-region sampling could be relaxed to non-uniform sampling and alternate dimensionality-reduction strategies could be considered such as wavelet decomposition, either as a substitute for or in combination with the singular value decomposition. A more general approach involves directly determining the reduced array configuration from a specified pressure beam pattern, which is in contrast to calculating the beam pattern from an initial array, then reducing the array.

Chapter 5 – Conclusion

The 20 element broadside-steered, linear array of point-sources in [8] is reduced to a 12 element array with the replicated parameters comparing to the corresponding values in [8] to within three decimal places. The initial and reduced array beam patterns provide comparable far-field pressures across the target-region and replacing the slightly tapered reduced array spacing with uniform spacing provides very little change in the beam pattern. 128, 64, and 32 element linear arrays of non-zero area elements are reduced to 66, 34, and 18 element arrays, respectively. Here, the reduced array beam pattern is comparable to the initial array beam pattern, and the reduced array provides a slightly tapered element spacing. Replacing the tapered spacing with uniform spacing provides little change in the beam pattern for the main lobe and low-angle side lobe regions. Pressures are calculated without the far-field approximation.

Apodizing an initial array with a normalized Taylor window or a Chebyshev window with 30 dB constant-amplitude side lobes provides comparable beam patterns for the main lobe and low-angle side lobe regions. The Taylor window does not exhibit delta functions at the ends of the window.

APPENDICES

The attached Matlab code reduces the number of elements in a linear, broadside-steered, unfocused ultrasound array with the matrix pencil method. The script titled mpm_main configures the initial array and simulation parameters, synthesizes the reduced array and plots the results. The script titled mpm_aux defines the target-region and calculates the pressure beam pattern. The mpm_main script calls the mpm_aux script multiple times. Both scripts call additional Matlab scripts and multiple C programming language files which may be downloaded from (http://www.egr.msu.edu/~fultras-web/). Section 3.6 provides the software version information.

The beginning of the mpm_main script defines the initial array and the simulation configuration. The 20 element initial array from [8] and the 128, 64, and 32 element initial arrays are pre-configured. Selecting the preferred array for simulation entails enabling the respective code line. For example, to simulate the 128 element array enable the code line: replicate=0;M=128;result=zeros(19969,200);.

Setting the [replicate] variable equal to 0 implements non-zero-area array elements, employs the FOCUS software to calculate the pressures, and calculates a non-normalized pressure beam pattern. Setting the [replicate] variable equal to 1 implements point-sources, applies the array factor along with the far-field approximation to calculate the pressure, and normalizes the pressure beam pattern. M represents the number of elements in the initial array.

The initial array beam pattern is sampled at (2 * M + 1) points uniformly-spaced across the semi-circular target-region according to $\cos(\Theta)$. Figure 2.1 shows the target-region location relative to the array. The uniformly-spaced pressure samples populate the Hankel matrix which undergoes a singular value decomposition. A generalized eigenvalue problem is solved and the

reduced array element locations and apodization are calculated.

Multiple reduced arrays are synthesized with a number of elements from four to the number of elements in the initial array. The mean square error is calculated between each of the reduced array beam patterns and the initial array beam pattern. The poles [poles], reduced array apodization [Rih], reduced array element locations in wavelengths [dilh], pressure beam pattern [result] and mean square error [MSE_] results are captured in the respective variables designated by [ ]. The variable [result_index] specifies which reduced array will be compared to the initial array when plotting the results. The mpm_main script generates multiple plots including a plot comparing the initial and the selected reduced array beam patterns, and plots illustrating the reduced array apodization and element spacing. The Matlab workspace is saved in the file sim_results.mat.

```
%mpm_main.m
%Thesis rev f MPM code with non-uniformly apodized initial array (taylor window)
%and non-normalized pressures.  file: calc_orig_array.m

clear; clc; close all;

%if replicate=1 then implement point-sources, and perform calculations per initial MPM paper,
normalize pressure plots
%if replicate=0 then implement non-zero area elements and apply FOCUS to calculate
pressures, do not normalize pressure plots

%replicate=1;M=20;result=zeros(3121,200);  %3121 = 2*M*78+1
replicate=0;M=32;result=zeros(4993,200);
%replicate=0;M=64;result=zeros(9985,200);
%replicate=0;M=128;result=zeros(19969,200);

%simulate reduced arrays with multiple number of elements
red_elem_start=4;red_elem_end=M;red_elem_increment=1;

%these vars store results from sims at a different number of reduced array elements
%each reduced array's values are stored in a column with the column index
%representing the number of elements in the array
%The variable [result] holds the reduced array pressure beam pattern with col 200 holding
%the initial array beam pattern
%assumes will not have more than 199 elements n the initial array
dilh=result; %element locations
Rih=result; %element weights
square_error=result;
MSE_=zeros(1,200);
poles=zeros(1,200); %complex-valued poles (same as generalized eigenvalues)
array_struct_all(200,200)=struct('shape',[],'half_width',[],'half_height',[],'complex_weight',[],'ti
me_delay',[],'center',[],'euler',[]);

for red_elem_ctr=red_elem_start:red_elem_increment:red_elem_end %simulate for these
number of reduced array elements

%initialization
N=M;L=M;
nelex = M; %number of x-dimension elements
neley = 1; %number of y-dimension elements (linear array)
plot_N=78*N;  %increase plot resolution
font_size=35;legend_fontsize=35;

%make some tissue structs
```

```
define_media
lossy = lossless;
lossy.soundspeed = 1540;
lossy.attenuationdBcmMHz = 0;
f0 = 1e6;
lambda = lossy.soundspeed / f0;
k=2*pi/lambda;

aperture_multiplier=1000;
if replicate==0;Ri=(1/max(taylorwin(M)))*taylorwin(M);end %Apply normalized taylor window
to approximate chebwin(M,30) window that was used in initial paper 20 element array
if replicate==1;Ri=[.32561 .28558 .39104 .50461 .62034 .73147 .83102 .91243 .97010 1 1
.97010 .91243 .83102 .73147 .62034 .50461 .39104 .28558 .32561];end

%define element size based upon lambda
kerf=lambda/10;
xspacing=lambda/2;
halfwidth=(xspacing-kerf)/2;initial_array_halfwidth=halfwidth;
halfheight=8*lambda;
yspacing=2*halfheight+kerf;
initial_array_area=(2*halfwidth)*(2*halfheight)*M;

array_struct = create_rect_planar_array(nelex,neley,halfwidth,halfheight,xspacing,yspacing);
di_lambda=zeros(1,M);for ctr=1:M;di_lambda(1,ctr)=array_struct(ctr).center(1)/lambda;end
aperture=array_struct(nelex).center(1)-array_struct(1).center(1); %calculate aperture
r=aperture_multiplier*aperture; %use semicircular focal plane distance from center of array

%check N, L and M relationship
if 2*N-L<M
    'L M N incorrectly defined'
    pause
end
if L+1<M
    'L M N incorrectly defined'
    pause
end
P=min([2*N-L+1;L+1]);

%calculate sampled pressure profile for input into MPM algorithm
di_values=zeros(1,nelex);
for ctr=1:nelex
    di_values(1,ctr)=array_struct(ctr).center(1);
end
Ri_values=Ri;
```

```matlab
mpm_aux;
y=fpp';

%calculate higher resolution beam pattern for plotting
N_orig=N;N=plot_N;
mpm_aux; %this script began as run88
if replicate==0;y_orig_plot=abs(fpp);end
if replicate==1;y_orig_plot=abs(fpp)/max(abs(fpp));end %normalized
N=N_orig;
result(:,200)=y_orig_plot;
array_struct_orig=array_struct;  %save initial array structure

%calculate (sampled) x-axis cos(theta) values
for focal_ctr=1:(2*N+1)
   n=focal_ctr-(N+1);
   theta_var2(focal_ctr)=-abs(acos(n/N));
end

%calculate (plotted) x-axis cos(theta) values
for focal_ctr=1:(2*plot_N+1)
   n=focal_ctr-(plot_N+1);
   theta_var(focal_ctr)=-abs(acos(n/plot_N));
end

%confirm that have adequate sampling points
[junk,tempvar1]=size(y);
if tempvar1<(2*M-1)
   'Insufficient sampling points'
   pause;
end

%build hankel matrix
tempvar1=(2*N)-L+1; tempvar2=L+1;
hankel_matrix=zeros(tempvar1,tempvar2);
for ctr=1:tempvar1
   y_shifted=circshift(y,[0,-(ctr-1)]);
   y_truncated=y_shifted(1:tempvar2);
   hankel_matrix(ctr,:)=y_truncated;
end

%perform svd
[U,S,VH]=svd(hankel_matrix);  %hankel_matrix variable represents paper variable Y
s_diag=diag(S); %real-valued singular values
```

```matlab
%plot singular values
figure(3)
if M==20;semilogy(s_diag,'--ks','markeredgecolor','k','markerfacecolor','k','markersize',30);end
if M~=20;semilogy(s_diag,'--ks','markeredgecolor','k','markerfacecolor','k','markersize',7);end
h=gca;set(h,'fontsize',font_size);
set(gca,'LineWidth',5);
axis([0 M 10^-10 10^10]);grid;set(gca,'yminorgrid','off')
xlabel('Singular Value');ylabel('Magnitude');
if M==20;set(gca,'XTick',0:4:20);set(gca,'XTickLabel',{'0','4','8','12','16','20'});end
if M==32;set(gca,'XTick',0:4:32);set(gca,'XTickLabel',{'0','4','8','12','16','20','24','28','32'});end
if M==64;set(gca,'XTick',0:8:64);set(gca,'XTickLabel',{'0','8','16','24','32','40','48','56','64'});end
if
M==128;set(gca,'XTick',0:16:128);set(gca,'XTickLabel',{'0','16','32','48','64','80','96','112','128'});
end

Q=red_elem_ctr; %number of reduced array elements

SQ=S;
[tempvar1,junk]=size(s_diag);
VQ=VH(:,1:Q);
Vqb=VQ(1:L,:);Vqt=VQ(2:(L+1),:);

%ensure the following matrix can be inverted
if det(Vqb'*Vqb)==0
    'Warning:  Cannot invert matrix with zero determinant - script paused'
    pause;
end

%paper equation 12 eigenvalue/vector calculation approach
[V,D]=eig((inv(Vqb'*Vqb))*(Vqt'*Vqb));

z=sort(diag(D)); %Order eigenvalues in ascending order

%normalize pole magnitude to 1.0
%calculate matrix Z (note zhat and Zhat are different variables)
%below operation sets amplitude to 1, while leaving phase (angle) unchanged
zhat=zeros(Q,1);
for ctr=1:Q
    zhat(ctr)=z(ctr)/abs(z(ctr));
end
poles(1:length(z),red_elem_ctr)=zhat';

di_lambda_hat=zeros(Q,1);
for i=1:Q
```

```matlab
di_lambda_hat(i)=-(N/(2*pi))*imag(log(zhat(i))); %imag of log leads to real valued
di_lambda_hat values
end

Zhat=ones(2*N+1,Q);
for ctr=1:(2*N+1)
 Zhat(ctr,:)=zhat.^(ctr-(N+1));
end

%calculate FM
fM=zeros(1,(2*N+1));
for ctr=1:(2*N+1)
   for ctr2=1:M
     fM(ctr)=fM(ctr)+Ri(ctr2)*exp((j*2*pi*di_lambda(ctr2)*((ctr-(N+1))/N)));
   end
end

Rihat=zeros(1,Q);
a=Zhat'*Zhat;
if det(Zhat'*Zhat)==0
   'Warning:  Cannot invert matrix with zero determinant - script paused'
   pause;
end
Rihat=(inv(Zhat'*Zhat))*Zhat'*fM';

a=max(abs(Rihat));
for ctr=1:Q;Rihat(ctr)=abs(Rihat(ctr))/a;end %normalize Rihat

%Now order elements based on di_lambda_hat variable
[a,ix]=sort(di_lambda_hat,'ascend');
for ctr=1:Q
   Rihat2(ctr)=Rihat(ix(ctr));
end
di_lambda_hat=a;
Rihat=Rihat2;
Ri_values=Rihat;

%configure reduced array variable for simulation
a=zeros(1,Q);a(1,1)=1e10;for ctr=2:Q;a(1,ctr)=di_lambda_hat(ctr,1)-di_lambda_hat(ctr-1,1);end
%determine minimumum spacing between any two adjacent reduced array elements
halfwidth=initial_array_halfwidth*M/Q; %set reduced array elemental area equal to the initial
array elemental area (assumes all elements in initial and reduced arrays are same height)
if 2*halfwidth>a
   'Caution: Reduced Array Elements Overlap'
```

```
    pause;
end
array_struct = create_rect_planar_array(Q,neley,halfwidth,halfheight,xspacing,yspacing);
%build array structure (element locations and weights will be set in next code line)
for
ctr=1:Q;array_struct(ctr).center(1)=lambda*di_lambda_hat(ctr,1);array_struct(ctr).complex_we
ight=Rihat(1,ctr);end
N=plot_N;nelex=Q;
mpm_aux;
if replicate==0;y_reduced_plot=abs(fpp);end
if replicate==1;y_reduced_plot=abs(fpp)/max(abs(fpp));end
reduced_array_aperture=array_struct(Q).center(1)-array_struct(1).center(1); %calculate
reduced array aperture
result(:,red_elem_ctr)=y_reduced_plot;
dilh(1:length(di_lambda_hat),red_elem_ctr)=di_lambda_hat;
Rih(1:length(Rihat),red_elem_ctr)=Rihat';
square_error(:,red_elem_ctr)=(y_orig_plot-y_reduced_plot).^2;
MSE_(1,red_elem_ctr)=mean(square_error(:,red_elem_ctr));

array_struct_all(Q,1:Q)=array_struct;
red_elem_ctr
end %of red_elem_ctr

save('sim_results.mat');

%begin plotting part of script
result_index=input('Please enter the number of elements in the reduced array which will be
compared to the initial array ');

%draw initial array
figure(4)
subplot(1,2,1);
draw_array_mpm(array_struct_orig);grid;
axis equal;grid;set(gca,'LineWidth',5);
h=gca;set(h,'fontsize',font_size);
title('Initial Array');
xlabel('x (m)')
ylabel('y (m)')
zlabel('z (m)')

%draw reduced array
subplot(1,2,2);
draw_array_mpm(array_struct_all(result_index,1:result_index));grid;
axis equal;grid;set(gca,'LineWidth',5);
```

```
h=gca;set(h,'fontsize',font_size);
title('Reduced Array');
xlabel('x (m)')
ylabel('y (m)')
zlabel('z (m)')

%linear plot original vs reduced array beam patterns
figure(5)
semilogy(cos(theta_var),result(:,200),'k');set(gca,'LineWidth',5);hold;
semilogy(cos(theta_var),result(:,result_index),'--k','linewidth',2);
axis([-1 1 10 10^5]);grid;set(gca,'yminorgrid','off')
h=gca;set(h,'fontsize',font_size);
if M==20;hl=legend('Initial 20 Element Array','Reduced 12 Element Array');end
if M==32;hl=legend('Initial 32 Element Array','Reduced 18 Element Array');end
if M==64;hl=legend('Initial 64 Element Array','Reduced 34 Element Array');end
if M==128;hl=legend('Initial 128 Element Array','Reduced 66 Element Array');end
set(hl,'fontsize',legend_fontsize);
xlabel('cos($\theta$)','Interpreter','latex');
if replicate==0;ylabel('Pressure');end
if replicate==1;ylabel('Normalized Pressure');end

%linear plot original vs reduced array beam patterns (zoomed)
figure(6)
semilogy(cos(theta_var),result(:,200),'k');set(gca,'LineWidth',5);hold;
semilogy(cos(theta_var),result(:,result_index),'--k','linewidth',2);
axis([-0.4 0.4 10 10^5]);grid;set(gca,'yminorgrid','off')
h=gca;set(h,'fontsize',font_size);
if M==20;hl=legend('Initial 20 Element Array','Reduced 12 Element Array');end
if M==32;hl=legend('Initial 32 Element Array','Reduced 18 Element Array');end
if M==64;hl=legend('Initial 64 Element Array','Reduced 34 Element Array');end
if M==128;hl=legend('Initial 128 Element Array','Reduced 66 Element Array');end
set(hl,'fontsize',legend_fontsize);
xlabel('cos($\theta$)','Interpreter','latex');
if replicate==0;ylabel('Pressure');end
if replicate==1;ylabel('Normalized Pressure');end

%plot Ri (element weight) values
figure(7)
h=gca;set(h,'fontsize',font_size);
if
M==20;plot(di_lambda_hat,Rihat,'kd','markeredgecolor','k','markerfacecolor','k','markersize',30
);grid;end
```

```matlab
if
M~=20;plot(dilh(1:result_index,result_index),Rih(1:result_index,result_index),'kd','markeredgec
olor','k','markerfacecolor','k','markersize',16);grid;end
xlabel('Array Element Location (Wavelengths)');ylabel('Element Weight');
if M==20;axis([-10 10 0 1.2]);set(gca,'XTick',-10:5:10);set(gca,'XTickLabel',{'-10','-
5','0','5','10'});end
if M==32;axis([-10 10 0 1.2]);set(gca,'XTick',-10:5:10);set(gca,'XTickLabel',{'-10','-
5','0','5','10'});end
if M==64;axis([-20 20 0 1.2]);set(gca,'XTick',-20:10:20);set(gca,'XTickLabel',{'-20','-
10','0','10','20'});end
if M==128;axis([-40 40 0 1.2]);set(gca,'XTick',-40:20:40);set(gca,'XTickLabel',{'-40','-
20','0','20','40'});end

%plot reduced array spacing
figure(8)
clear reduced_array_spacing;
di_lambda_hat=dilh(:,result_index);
for ctr=2:result_index;reduced_array_spacing(ctr-1)=di_lambda_hat(ctr)-di_lambda_hat(ctr-
1);end
plot((1:(result_index-1))-
(result_index/2),reduced_array_spacing,'kd','markeredgecolor','k','markerfacecolor','k','marker
size',16);
grid;set(gca,'LineWidth',5);
if M==20;axis([-5 5 0.8 1]);end
if M==32;axis([-9 9 0.8 1]);end
if M==64;axis([-17 17 0.8 1]);end
if M==128;axis([-35 35 0.8 1]);end
h=gca;set(h,'fontsize',font_size);
set(h,'XTick',[0]);
xlabel('Array Axis');
ylabel('Element Spacing in Wavelengths');

%replace the reduced array tapered spacing with uniform spacing keeping the
%same aperture width and the same apodization
nelex_orig=nelex;nelex=result_index;
halfwidth=initial_array_halfwidth*M/result_index;
di_lambda_hat_orig=dilh(1:result_index,result_index);
di_lambda_hat=zeros(result_index,1);array_struct=array_struct(1,1:result_index);
reduced_array_aperture=(dilh(result_index,result_index)-dilh(1,result_index))*lambda;
uniform_spacing_increment=(reduced_array_aperture/lambda)/(result_index-1);
array_struct =
create_rect_planar_array(nelex,neley,halfwidth,halfheight,uniform_spacing_increment*lambd
a,yspacing);
Ri_values=Rih(1:result_index,result_index);
```

```
mpm_aux;
if  replicate==0;y_reduced_plot_forced_uniform=abs(fpp);end
if replicate ==1;y_reduced_plot_forced_uniform=abs(fpp)/max(abs(fpp));end %normalized
reduced_array_aperture_forced_uniform=array_struct(result_index).center(1)-
array_struct(1).center(1); %calculate reduced array aperture
if reduced_array_aperture_forced_uniform~=reduced_array_aperture
   'reduced array enforced uniform aperture is not same width as tapered aperture width'
end
nelex=nelex_orig;

%linear plot original vs reduced array (with enforced uniform spacing) beam patterns (zoomed)
figure(9)
semilogy(cos(theta_var),result(:,200),'k');set(gca,'LineWidth',5);hold;
semilogy(cos(theta_var),y_reduced_plot_forced_uniform,'--k','linewidth',2);
axis([-1 1 10 10^5]);grid;set(gca,'yminorgrid','off')
if replicate==1;axis([-1 1 10^-4 1]);end
h=gca;set(h,'fontsize',font_size);
if M==20;hl=legend('Initial 20 Element Array',vertcat('Reduced 12 Element Array','with Uniform
Spacing    '));end
if M==32;hl=legend('Initial 32 Element Array',vertcat('Reduced 18 Element Array','with Uniform
Spacing    '));end
if M==64;hl=legend('Initial 64 Element Array',vertcat('Reduced 34 Element Array','with Uniform
Spacing    '));end
if M==128;hl=legend('Initial 128 Element Array',vertcat('Reduced 66 Element Array','with
Uniform Spacing    '));end
set(hl,'fontsize',legend_fontsize);
xlabel('cos(${\theta}$)','Interpreter','latex');ylabel('Normalized Pressure');
if M==20;ylabel('Normalized Pressure');end
if M~=20;ylabel('Pressure');end

figure(10)
semilogy(MSE_,'--ks','markeredgecolor','k','markerfacecolor','k','markersize',7);
h=gca;set(h,'fontsize',font_size);
set(gca,'LineWidth',5);hold;
axis([0 M 10^2 10^8]);grid;set(gca,'yminorgrid','off')
xlabel('Number of Elements in the Reduced Array');ylabel('Mean Square Error');
if M==32;set(gca,'XTick',0:4:32);set(gca,'XTickLabel',{'0','4','8','12','16','20','24','28','32'});end
if M==64;set(gca,'XTick',0:8:64);set(gca,'XTickLabel',{'0','8','16','24','32','40','48','56','64'});end
if
M==128;set(gca,'XTick',0:16:128);set(gca,'XTickLabel',{'0','16','32','48','64','80','96','112','128'});
end

figure(11)
stem(max(dilh)-min(dilh),'k','linewidth',2);grid;
```

```
xlabel('Number of Elements in the Reduced Array');
ylabel('Reduced Array Aperture Width (in Wavelengths)');
```

'Finished'

```
%mpm_aux.m
% this script calculates the semi-circular focal plane and the beam pattern

max_focal_shift = ceil(0.02 / (lambda /2))* lambda/2; % 2cm max focal spacing, make sure it
aligns to the grid
xmin_source_plane = -aperture/2-max_focal_shift;
xmax_source_plane = aperture/2+max_focal_shift;
ymin_source_plane = 0;
ymax_source_plane = 0;

dx = lambda / 2;
dy = lambda / 2;
dz = lambda / 2;

zmin_source_plane = 2 * dz;
zmax_source_plane = 146 * lambda;

delta = [dx, dx, dz];

xfocus=0;
yfocus=0;
zfocus=r;

%calculate semi-circular focal region with uniformly-spaced samples in cos(theta) space
tempvar1=[-1:2/(2*N):1];
tempvar2=acos(tempvar1);
x_coord=r*cos(tempvar2);
z_coord=r*sin(tempvar2);
vector=zeros(2*N+1,3);
vector(:,1)=x_coord;
vector(:,3)=z_coord;
source_pressure_plane_coord_grid = set_coordinate_grid(delta, xmin_source_plane,
xmax_source_plane, ymin_source_plane, ymax_source_plane, zmin_source_plane,
zmin_source_plane);
vector_flag=1;
focal_plane_coord_grid = set_coordinate_grid(vector);

%set accuracy
ndiv=2;

 for ctr=1:nelex
   array_struct(ctr).complex_weight=Ri_values(ctr);
 end
```

```
 %focus simulation
if replicate==0
focal_plane_pressure = fnm_cw(array_struct, focal_plane_coord_grid, lossy, ndiv, f0, 0);
%simulate array
fpp=focal_plane_pressure;
end

%point-source calculation
if replicate==1
   [tempvar1 junk]=size(vector);
   fpp=zeros(tempvar1,1);
for focal_ctr=1:tempvar1
sumvar=0;
for array_ctr=1:nelex
r=sqrt((vector(focal_ctr,1)-array_struct(array_ctr).center(1))^2+(vector(focal_ctr,3))^2);
sumvar=sumvar+array_struct(array_ctr).complex_weight*(1/r)*exp(-j*k*r);
end
fpp(focal_ctr,1)=sumvar;
end

end
```

```matlab
function draw_array(xdc_array,ps,color)
% draw_array_mpm.m
% function to draw an array of a single transducer type
% the input argument is an array of structures
if nargin()==0
    disp('The proper usage of this function is:')
    disp('draw_array(xdc_array,color)')
    error('color is the standard matlab color notation ')
end
if nargin()==1
    color=[0.9 0.9 0.9];
    ps=[];
end
if nargin()==2 && isstruct(ps)
    color=[0.9 0.9 0.9];
end
V=[-1 1 -1 1 -1 1];
axis(V);
hold on
for i=1:length(xdc_array)
    if strcmp('circ', xdc_array(i).shape)
        draw_circ(xdc_array(i),color);
    elseif strcmp('shel', xdc_array(i).shape)
        draw_sphericalshell(xdc_array(i),color);
    elseif strcmp('shell_with_hole', xdc_array(i).shape)
        draw_sphericalshell_with_hole(xdc_array(i),color);
    elseif strcmp('circ_with_hole', xdc_array(i).shape)
        draw_circ_with_hole(xdc_array(i),color);
    elseif strcmp('rect', xdc_array(i).shape)
        draw_rect(xdc_array(i),color);
    elseif strcmp('none', xdc_array(i).shape)
        %do nothing...
    else
        warning('Invalid/Unimplemented shape XDC detected')
    end

end
if (isstruct(ps))
    x=ps.xmax;
    y=ps.ymax;
    mx=ps.xmin;
    my=ps.ymin;
    z=ps.zmin;
    maxz=ps.zmax;
```

```
    line([mx x],[y y],[z z]);
    line([mx mx],[y my],[z z]);
    line([mx x],[my my],[z z]);
    line([x x],[y my],[z z]);
    line([mx x],[y my],[z z]);
    line([mx x],[my y],[z z]);
    line([mx x],[y y],[maxz maxz]);
    line([mx mx],[y my],[maxz maxz]);
    line([mx x],[my my],[maxz maxz]);
    line([x x],[y my],[maxz maxz]);
    line([mx x],[y my],[maxz maxz]);
    line([mx x],[my y],[maxz maxz]);
    line([mx mx],[my my],[z maxz]);
    line([mx mx],[y y],[z maxz]);
    line([x x],[my my],[z maxz]);
    line([x x],[y y],[z maxz]);

end
hold off
axis tight;
V=axis;

V(1)=min([V(1) V(3)]);
V(2)=max([V(2) V(4)]);
V(3)=V(1);
V(4)=V(2);
if strcmp('shel', xdc_array(i).shape)
    V(6)=max([-V(5) V(6)]);
    V(6)=V(6)+xdc_array(i).radius;
    V(5)=-V(6);
% elseif strcmp('shell_with_hole', xdc_array(i).shape)
%    V(5)=-max([V(1)/2 -V(5)]);
%    V(6)=max([V(6) -V(5)]);
%    V(5)=-2*V(6);
else
    V(5)=V(1)/2;
%    if length(V)==6
%        V(6)=max([V(2) V(6)]);
%    else
        V(6)=V(2);
%    end
end
axis(V);
grid on;
```

REFERENCES

REFERENCES

[1]   C. Diederich, Thermal Ablation and High-Temperature Thermal therapy: Overview of Technology and Clinical Implementation, International Journal of Hyperthermia, pp. 745-753, December 2005

[2]   F. Ares-Pena, J. Rodriguez-Gonzalez, E. Villanueva-Lopez, S. Rengarajan, Genetic Algorithms in the Design and Optimization of Antenna Array Patterns, IEEE Transactions on Antennas and Propagation, vol. 47, no. 3, pp. 506-510, 1999

[3]   D. Gies, Y. Rahmat-Samii, Particle Swarm Optimization for Reconfigurable Phase-Differentiated Array Design, Microwave and Optical Technology Letters, vol. 38, issue 3, pp. 168-175, 2003

[4]   J. Rodriguez, F. Ares, E. Moreno, Linear Array Pattern Synthesis Optimizing Array Element Excitations Using the Simulated Annealing Technique, Microwave Optimization Technological Letters, vol. 23, issue 4, pp. 224-226, 1999

[5]   E. Miller, D. Goodman, A Pole-Zero Modeling Approach to Linear Array Synthesis I: the Unconstrained Solution, Radio Science, vol. 18, pp. 57-69, Jan.-Feb. 1983

[6]   Y. Hua, T. Sarkar, Matrix Pencil method for Estimating Parameter of Exponentially Damped/Undamped Sinusoids in Noise, IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 38, No. 5, May 1990, pp. 814-824

[7]   T. Sarkar and O. Pereira, Using the Matrix Pencil Method to Estimate the Parameters of a Sum of Complex Exponentials, IEEE Antennas and Propagation Magazine, vol. 37, issue 1, pp. 48-55, February 1995

[8]   Y. Liu, Z. Nie, Q.H. Liu, Reducing the Number of Elements in a Linear Antenna Array by the Matrix Pencil Method, IEEE Transactions on Antennas and Propagation, Vol. 56, No. 9, pp. 2955-2962

[9]   M. Khan, M. Tufail, Comparative Analysis of Various Matrix Pencil Methods for Direction of Arrival Estimation, International Conference on Image Analysis and Signal Processing, pp. 496-501, 2010

[10]   A. Mroue, M. Heddebaut, F. Elbahhar, A. Rivenq, Leaky Waveguide radar system for Fall on Track Object Detection and Identification

[11]   C. Gotai, A. Medouri, Using the matrix pencil Algorithm and the Singularity Expansion Method for Maneuvering Targets Detection and Identification in Maritime Environment, European Journal of Scientific Research, Vol. 71, No. 1, pp. 14-23, 2012

[12]   B. Fourestie, Z. Altman, J. Wiart, A. Azoulay, On the Use of the Matrix Pencil Method to Correlate Measurements at Different Test Sites, IEEE Transactions on Antennas and Propagation, vol. 47, No. 10, pp. 1569-1573, October 1999

[13]   J. Laroche, The Use of the Matrix Pencil Method for the Spectrum Analysis of Music Signals, Journal of the Acoustical Society of America, Vol. 94, Issue 4, pp. 1958-1965, 1993

[14]   T. Sarkar, F. Hu, Y. Hua, M. Wicks, A Real-Time Signal Processing Technique for Approximating a Function by a Sum of complex Exponentials Utilizing the Matrix-Pencil Approach, Digital Signal Processing, vol. 4, issue 2, pp. 127-140, April 1994

[15]   J. del Rio, T. Sarkar, Comparison Between the Matrix Pencil Method and the Fourier Transform Technique for High-Resolution Spectral Estimation, Digital Signal Processing 6, article No. 0011, pp. 108-125, 1996

[16]   F. Yidirim, L. Gurel, Coupled Matrix Pencil Method for Frequency Extrapolation of Electromagnetic Solutions, IEEE Antennas and Propagation Society International Symposium, Vol. 4B, pp. 152-155, 2005

[17]   Y. Hua, Estimating Two-Dimensional Frequencies by Matrix Enhancement and Matrix Pencil, IEEE Transactions on Signal Processing, Vol. 40, No. 9, pp. 2267-2280, September 1992

[18]   E. Adiba, S. Bri, M. Habibi, Application of 2-D Enhanced Matrix Pencil Method for Efficient Direction of Arrival, International Journal of Science and Advanced Technology, Vol. 1, No. 6, pp. 31-36, August 2011

[19]   B. Lu, D. Wei, B. Evans, A. Bovik, Improved Matrix Pencil Methods, Thirty-Second Asilomar Conference on Computers, Vol. 2, pp. 1433-1437, 1998

[20]   F. Harris, Multirate Signal Processing for Communication Systems, New Jersey: Prentice Hall PTR, pp. 60-64, 2004

[21]   R. J. McGough, Rapid Calculations of Time-Harmonic Nearfield Pressures Produced by Rectangular Pistons, Acoustical Society of America, vol. 115, no. 5, pp. 1934-1941, May 2004

[22]   A. Caboussat, G.K. Miers, Numerical Approximation of Electromagnetic Signals Arising in the Evaluation of Geological Formations, Computers & Mathematics with Applications, vol. 59, issue 1, January 2010

[23]   W. Xu, W. Stewart, Multiresolution-Signal Direction-of-Arrival Estimation: A Wavelets Approach, IEEE Transactions on Signal Processing Letters, vol. 7, no. 3, pp. 66-68, March 2000.