Gradient-Driven and Reduced-Rate Beamforming for Biomedical Ultrasound

by

Solmaz Khezerloo

B.Sc., Amirkabir University of Technology, Iran, 2001
M.Sc., University of Karlsruhe, Germany, 2006

A Thesis Submitted in Partial Fullfillment of the
Requirements for the Degree of

Master of Applied Science

in the Department of Electrical and Computer Engineering

Gradient-Driven and Reduced-Rate Beamforming for Biomedical Ultrasound

by

Solmaz Khezerloo

B.Sc., Amirkabir University of Technology, Iran, 2001
M.Sc., University of Karlsruhe, Germany, 2006

Supervisory Committee

Dr. Daler N. Rakhmatov, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Alexandra Branzan Albu, Department Member
(Department of Electrical and Computer Engineering)

Dr. Michael Adams, Department Member
(Department of Electrical and Computer Engineering)

Dr. Micaela Serra, Outside Member
(Department of Computer Science)

Supervisory Committee

Dr. Daler N. Rakhmatov, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Alexandra Branzan Albu, Department Member
(Department of Electrical and Computer Engineering)

Dr. Michael Adams, Department Member
(Department of Electrical and Computer Engineering)

Dr. Micaela Serra, Outside Member
(Department of Computer Science)

Abstract

Adaptive beamforming can significantly improve the image quality in biomedical ultrasound by reducing the clutter due to interfering signals arriving from undesired directions. Adaptive beamforming is computationally expensive, and the objective of this thesis is to expose and explore tradeoffs between computational complexity and quality of adaptive beamforming. We consider the conventional linearly constrained minimum variance (LCMV) adaptive beamformer, applied to B-mode ultrasound imaging, and study an alternative based on the well-known generalized sidelobe canceller (GSC) whose adaptation relies on unconstrained gradient-driven optimization. To our knowledge, this is the first time a GSC-based gradient-driven

approach has been applied and evaluated in the context of ultrasound beamforming. As another alternative to the conventional LCMV method, we also propose and evaluate a simple idea of updating the beamformer's weight vector at a reduced rate. Both approaches have lead to significant computational savings, but they also sacrifice beamforming optimality. Our simulations show that, despite suboptimal beamforming, the ultrasound image quality remains acceptable.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Analog-to-Digital Converter	ADC
Conjugate Gradient [Method]	CG
Digital-to-Analog Converter	DAC
Finite-duration Impulse Response	FIR
First In, First Out	FIFO
FLoating-point OPeration	FLOP
Fractional Delay	FD
Full Width at Half Maximum	FWHM
Generalized Sidelobe Canceller	GSC
High Voltage [Amplifier]	HV
Least-Mean-Squares [Algorithm]	LMS
Linearly Constrained Minimum Variance [Beamforming]	LCMV
Low Noise Amplifier	LNA
Minimum Variance Distortionless Response	MVDR
Multiple Sidelobe Canceller	MSC
Quasi-Newton [Method]	QN
Signal-to-Noise Ratio	SNR
Steepest Descent [Method]	SD
Time Gain Compensation	TGC

Chapter 1

Introduction

Ultrasound imaging modality is widely used in medical diagnostic applications, due to the fact that biomedical ultrasound systems are relatively inexpensive, portable, non-invasive, and practically harmless. To obtain an image, a piezoelectric transducer array transmits short sound pulses to the region of interest, and detects returning echos produced by complex tissues and organ boundaries [7]. Figure 1.1 shows a basic block diagram of a typical ultrasound system [38, 39] – its detailed description is provided in Chapter 2 (see Section 2.1). In this thesis, we focus on the *Receive Beamformer* block.

Received signals are first focused using appropriate time delays and then summed, as shown in Figure 1.2 for a transducer array with M active elements. Delay focusing (using appropriate time delays δ_1, δ_2, ..., δ_M) can be accomplished by means of first-in first-out (FIFO) queues and fractional delay (FD) filters [26], briefly described in Chapter 2 (see Section 2.1). The weighted sum of the focused inputs $u_1(t), u_2(t), ..., u_M(t)$ at the sampling instance t produces the output $y(t)$. The weights $w_1(t), w_2(t), ..., w_M(t)$ can be based on some fixed window functions (such as Hamming, von Hann, Dolph-Chebyshev, or Kaiser windowing [7, 15]), or alternatively, they can be computed adaptively based on the properties of received

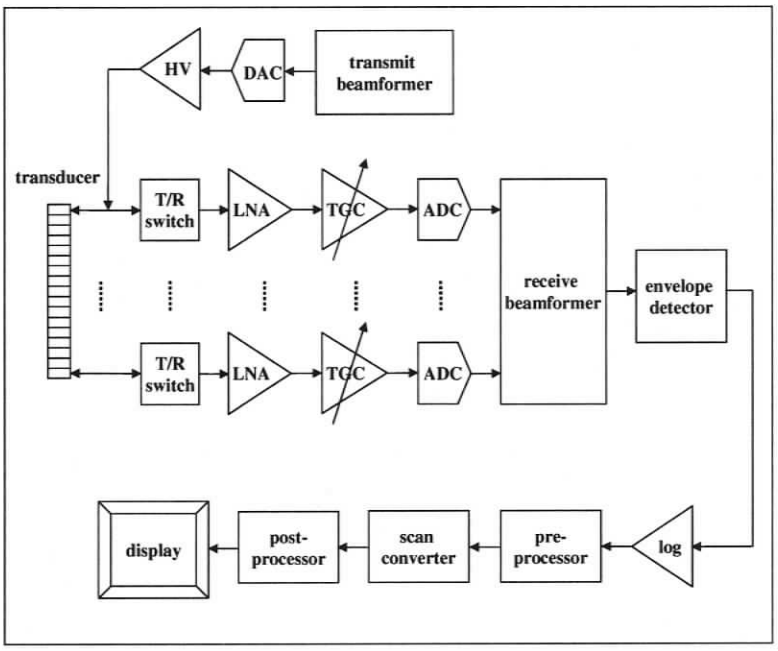


Figure 1.1: Typical ultrasound system.

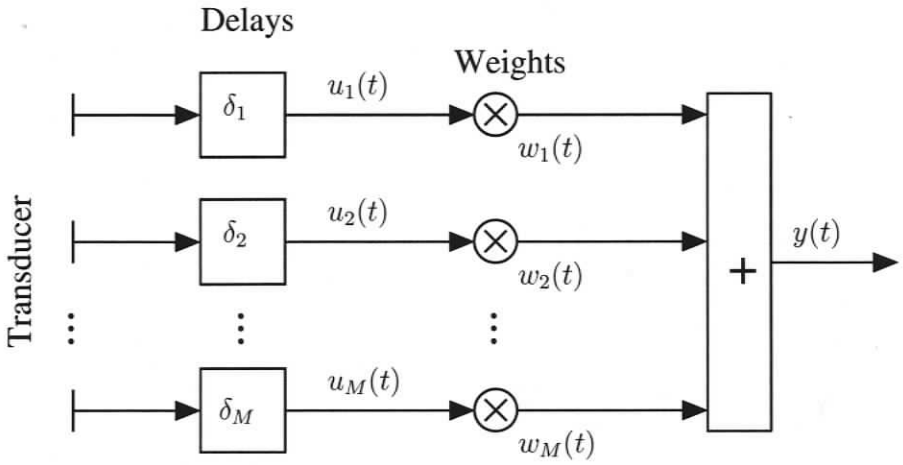


Figure 1.2: Ultrasound beamformer with delay focusing.

data. Applying the apodization weights, or beamforming, should minimize the contribution of undesired (interfering) signals. The source of these interferin g signals are the sidelobes (the undesired propagation of the acoustic wave away from the main beam). Sidelobes are also one of the sources of the artifacts, since their reflected signals are mapped along the main beam. Thus, any reduction in the sidelobes leads to improvements in the ultrasound image quality (e.g., resolution and contrast). We have investigated adaptive beamformers that produce an optimized set of apodization weights for each input snapshot (i.e., for each set of sampled and focused signals arriving from the transducer). If the desired signal were known, a typical optimization algorithm would have found the weights that minimize the mean-square error between the desired signal and the received signal [16]. In ultrasound imaging, however, the desired signal is unknown. In such cases, a common approach is to introduce constraints in the optimization problem formulation.

The subject of this thesis is linearly constrained minimum variance (LCMV) adaptive beamforming, which is one of the earliest and simplest adaptive methods

[24]. The LCMV method, presented in Chapter 2, has recently been applied to biomedical ultrasound, demonstrating significant improvements in image resolution and contrast [17,18,28,34,37,47]. Motivated by these works, we have investigated the conventional LCMV beamforming method and its suboptimal alternatives to study tradeoffs between the image quality and the computational complexity.

The first alternative is based on the use of the generalized sidelobe canceller (GSC) structure [24], which is discussed in Chapter 3. The GSC structure arises naturally, when one considers using unconstrained gradient-driven search to find a solution to a constrained optimization problem (such as LCMV beamforming). The main advantage of the GSC, coupled with iterative optimization methods, is that its output quality can be traded off for a lower computational cost depending on the number of iterations per input snapshot. In this thesis, we consider four iterative gradient-driven search methods: Newton, quasi-Newton, steepest descent, and conjugate gradient.

As another alternative to the conventional LCMV method, we also propose and evaluate a simple idea of updating the apodization weight vector at a reduced rate. In other words, we allow the LCMV beamformer to take more than one sampling period to compute and apply the weights. In the case of the 1/4-rate updating, for example, the beamformer takes as many as four sampling periods (rather than one period) to find the weights and compute the corresponding output (the same weights are applied per four buffered input snapshots). This approach reduces the computational cost per sampling period at the expense of apodization optimality. Further details are provided in Chapter 4.

In summary, this thesis investigates the gradient-driven GSC-based beamformer and the reduced-rate LCMV beamformer, compares them to the conventional LCMV beamformer, and presents numerical results that expose the tradeoffs between the image quality and the computational complexity. The main contributions of this

work have been published in [25] – they are as follows:

- We apply a GSC-based beamforming technique in the context of ultrasound beamforming.
- We evaluate the performance of Newton, quasi-Newton, steepest descent, and conjugate gradient optimization methods, as applied in the GSC-based beamformer setting.
- We propose a modification of the conventional LCMV method, whereby the apodization weights are computed at a reduced rate.

The remainder of the thesis is organized as follows. Chapter 2 provides a background on ultrasound imaging and LCMV beamforming. Chapter 3 describes the GSC-based beamforming technique and related iterative optimization methods. Chapter 4 proposes the reduced-rate LCMV beamformer. Chapter 5 presents simulated imaging results, estimated computational savings enabled by the GSC-based beamforming, and the reduced-rate LCMV beamforming schemes. Finally, the conclusions and the ideas for future work are given in Chapter 6.

Chapter 2

Background

This chapter briefly describes a typical ultrasound system, provides a basic background information on biomedical ultrasound imaging, and presents the details of the conventional LCMV beamforming method.

2.1 Typical Ultrasound System

Our descriptions here will be focused on two-dimensional brightness-mode (B-mode) ultrasound imaging. (See Figure 1.1 in Chapter 1 that shows a basic block digram of a typical ultrasound system.) High-frequency sound wave pulses, typically in the range of 1 to 15 MHz [39], are transmitted through each transducer element to propagate through the body. Instead of a single transducer, an array of transducers is used to increase the sensitivity and the spatial resolution [7]. A transducer array may consist of 8 to 512 elements [39]. A larger number of active elements of the transducer array (the aperture size) yields a better lateral spatial resolution [7]. The spacing between the elements must be less than 1/2 the resonant frequency wavelength, in order to suppress the grating lobes and increase the gray-scale resolution [7]. The grating lobes occur in symmetric locations about the mainlobe and can have amplitudes as

high as the mainlobe. As with the sidelobes, the grating lobes cause image artifacts and reduce the contrast. One-dimensional transducer arrays (used to generate two-dimensional images) come in a variety of forms and sizes. Two common types are a linear array and a phased array. In linear arrays, the beam is formed by exciting a group of adjacent elements. The desired area can be scanned by group shifting. In phased arrays, all transducer elements are used for each scan line. These arrays perform a fan-like sector scan and can cover a wider area. The line density, however, decreases with depth [7].

During transmission, the ultrasound beam is dynamically steered and focused (transmit beamforming) by applying appropriate time delays to the transmitted pulses of each transducer element. The delays can be derived mathematically by considering the path length differences between the target (the focus point) and different elements of the transducer array. Due to the path length differences, signals excited from different transducer elements arrive at the focus point with different phases. This can be avoided by delaying the middle elements with respect to the outer elements. Outputs of the transmit beamformer are converted by digital-to-analog converters (DAC) to analog signals and passed through high voltage (HV) amplifiers that drive the transducers. Ultrasound waves travel through different tissues at certain speeds, which are similar for most soft tissues – approximately 1540 m/s [7]. This makes it possible to estimate the target depths from the round trip (pulse-echo) time delays.

During reception, the reflected ultrasound waves from tissue boundaries are received at the transducer elements and transformed into electrical signals (per piezoelectric effect). The transmit/receive (T/R) switches isolate the high-voltage supply from the received echoes. Low noise amplifiers (LNA) bring the signal into a range of voltages appropriate for the signal processing. Attenuation of the ultrasound wave with time, due to absorption and scattering, is compensated by

time gain compensators (TGC). Analog-to-digital converters (ADC) digitize analog signals received, and the receive beamformer can now be used to perform the coherent summation of the echoes. Receive beamforming involves delay focusing (with identical delays as in the transmit beamforming) and apodization as shown in Figure 1.2 for a transducer array with M active elements.

The required delays for each element are, in general, non-integer numbers comprised of an integer part and a fractional part. The delay's integer part corresponds to the number of sampling periods to be skipped – this can be realized by simple FIFOs of appropriate lengths. The delay's fractional part corresponds to the timing offset needed within the sampling period – this can be realized by FIR fractional-delay (FD) filters based on Lagrange interpolation [26]. This interpolation method has several advantages: simple and explicit formulas for the filter coefficients, very good response at low frequencies, and a smooth magnitude response. In the case of dynamic receive focusing, the dynamic-length FIFOs are to be used (instead of the fixed-length FIFOs), while the FD filters are to have the Farrow structure [26]. Using this structure, filter coefficients are fixed for a specific filter order and need not be recomputed when delays change. Such FD filters have an additional input for the delay value that drives the interpolation. Dynamic receive beamforming is briefly described in the next section.

In B-mode ultrasound imaging the pixel brightness is proportional to the signal envelope that can be determined by using the Hilbert transform [7]. Next, logarithmic compression is applied to reduce the dynamic range of the signal envelope, in order to match it to the dynamic range of the display. Before displaying the data, certain pre-processing may be carried out (e.g., weak signals may be emphasized slightly). A digital scan converter remaps the data to a rectangular display. In post-processing, the amplitudes undergo a user-selectable nonlinear mapping to the final gray scale.

2.2 Ultrasound Image Quality

This section discusses the image quality measures and their determining factors.

Spatial Resolution

The image resolution along different axes is characterized by different factors. The ability to distinguish objects in the direction of the beam and perpendicular to the beam are called, respectively, the axial (or depth) resolution and the lateral resolution. The axial resolution is determined mainly by the excitation pulse width, where the reflectors with the minimum distance of 1/2 the spatial pulse length can be distinguished from each other. The shorter the pulse is, the finer the resolution will be. In other words, we can use higher frequency transducers to increase the axial resolution. A higher frequency, however, leads to a smaller penetration depth because of tissue attenuation [7].

In contrast to the axial resolution, which is primarily determined by the transmitted pulse, the lateral resolution is determined by the beam diameter [7]. Since the beam diameter is depth-dependent, the lateral resolution is not constant along the beam. The best lateral resolution is achieved at a near-field-to-far-field transition (natural focus). The near field is the region close to the transducer. For a simple piston transducer, the region farther than D^2/λ is the far field [7], where D is the radius of the transducer, and λ is the wavelength. The behavior of the beam is modeled differently in these two fields.

If the region of interest is not moving fast, multiple transmit focusing at different depths can improve the lateral resolution along the scan line. However, each focal zone requires separate echo-pulse sequences to acquire data, which reduces the frame rate and/or the number of scan lines. Another option for enhancing the lateral resolution (when using only one transmit focal zone) is dynamic receive focusing. In this case,

the delay focusing is done as a function of depth: the delays applied to each transducer element are updated at predefined intervals to focus on the point where the reflected signal was originated.

The aperture size (i.e., the number of active elements) during reception can also be dynamically increased to keep the F-number (the ratio of the focal length to the aperture size) along the beam constant. This ensures a constant beam width until the full aperture is reached. A smaller F-number results a higher lateral resolution [7]. The dynamic aperture on the transmit beam, however, compromises the frame rate.

The lateral resolution can be characterized by the full width at half maximum (FWHM). To measure the FWHM, the response of the imaging system to a point target is used. This response is the so-called point spread function (PSF), where the detected pressure is plotted versus the lateral axis.

Contrast

The difference between the image intensity of an object and the background is referred to as contrast. The contrast of an image is generally degraded by the background speckle. The object's contrast against the background is defined as the ratio [7]

$$\text{Contrast} = (S_{\text{background}} - S_{\text{object}})/S_{\text{background}}, \tag{2.1}$$

where S denotes the average log-compressed signal envelope inside the region of interest. Apodization contributes to lowering the sidelobes (and hence to lowering a signal level in the image background), which consequently increases the contrast.

Frame Rate, Scan Line Density, and Depth of Field

For real-time imaging, a minimum frame rate (usually 30 frames per second) is required to properly capture the motion within the body. In gray scale imaging,

the maximum achievable image frame rate and the scan line density are inversely related as [7]

$$F_r \cdot D_f \cdot N_s = C_0/2, \tag{2.2}$$

where F_r [1/s] is the maximum frame rate, D_f [m] the depth of field, N_s the number of scan lines and C_0 [m/s] the speed of sound. To achieve a better image detail, a larger number of scan lines for each image frame can be used. However, the field depth and/or the frame rate would be negatively affected.

Noise

There are two sources of noise in ultrasound systems: the Gaussian white noise and the speckle noise. The Gaussian white noise arises from the electronic components of the imaging system. The speckle noise is referred to the salt-and-pepper noise in the image background. Speckle limits the detectability of the desired structure against the background. The difference between the tissue dimensions and the ultrasound wavelength introduces a number of small scatterers which play a factor in the resulting image. The constructive and destructive interference of randomly distributed scatterers will result in a granular background in the image. In simulation studies when the number of scatterers is large enough (so that the simulation results are comparable to the real images), the resulting speckle is called a fully developed speckle.

2.3 Conventional LCMV Beamformer

Given the beamforming structure in Figure 1.2, at the sampling instance t, we have the input vector

$$\mathbf{u}(t) = \begin{bmatrix} u_1(t) & u_2(t) & \dots & u_M(t) \end{bmatrix}^T.$$

The beamformer output $y(t)$ and output power $P(t)$ are [24]:

$$y(t) = \mathbf{w}(t)^H \mathbf{u}(t), \tag{2.3}$$

$$P(t) = E[|y(t)|^2] = \mathbf{w}(t)^H \mathbf{R}(t) \mathbf{w}(t), \tag{2.4}$$

where

$$\mathbf{w}(t) = \begin{bmatrix} w_1(t) & w_2(t) & \dots & w_M(t) \end{bmatrix}^T$$

is the weight vector, and

$$\mathbf{R}(t) = E[\mathbf{u}(t)\mathbf{u}(t)^H]$$

is the spatial covariance matrix. Superscript H indicates conjugate transposition. In our case, all vectors and matrices of interest are real-valued, and H can be replaced with superscript T indicating ordinary transposition.

Let $\mathbf{d}$ denote the steering vector (desired direction), which is the signal from the focus point. The signal received by each transducer element is delayed by the amount needed to focus to the focus point, where the signal was originated. After applying the delay focusing, the desired direction becomes a constant vector of M ones, i.e.,

$$\mathbf{d} = \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^T.$$

The linearly constrained minimum variance (LCMV) beamformer has $\mathbf{w}(t)$ such that signals from the desired direction are passed with a unit gain, while signals from any other direction are suppressed. Equivalently, $\mathbf{w}(t)$ is a solution to the problem of minimizing the LCMV beamformer's output power $P(t)$ subject to the constraint

$$\mathbf{w}(t)^H \mathbf{d} = 1. \tag{2.5}$$

Using the method of Lagrange multipliers, the optimal solution is found to be [24]:

$$\mathbf{w}(t) = \frac{\mathbf{R}(t)^{-1}\mathbf{d}}{\mathbf{d}^H \mathbf{R}(t)^{-1}\mathbf{d}}. \tag{2.6}$$

The spatial covariance matrix $\mathbf{R}(t)$ is usually estimated based on the sample correlation matrix $\widehat{\mathbf{R}}(t)$:

$$\widehat{\mathbf{R}}(t) = \frac{1}{N} \sum_{n=t-N+1}^{t} \mathbf{u}(n)\mathbf{u}(n)^H, \tag{2.7}$$

where $\mathbf{u}(n)$ is the n-th input snapshot, and N is the number of snapshots. The value of N depends on the statistics of the signal and noise. As ultrasound signals are non-stationary, N is usually very small. Coherence between desired signals and interfering signals is another important issue in biomedical ultrasound. The LCMV beamformer performs poorly given coherent signals, unless we apply certain data preprocessing schemes to create artificial decorrelation [34]. In the next section, we will discuss a solution to this problem.

2.3.1 Spatial Smoothing Scheme

The LCMV beamformer fails to cancel the interference, if interfering signals are coherent with the desired signal, as in the case of the ultrasound data. The coherency is due to the multipath propagation, which causes the received signals to be delayed versions of each other. In order to minimize the interference power, the beam pattern needs to have deep nulls in the interfering direction. Therefore, if an interfering signal is coherent with the desired signal, the minimization is achieved if the desired signal is completely canceled out. Shan and Kailath [36] have proposed a scheme to overcome this problem, which will be used here. The proposed preprocessing scheme will restore the rank of the covariance matrix, even if the signals are completely coherent. This consequently will result in an invertible matrix, even in the case of a single or a few snapshots when estimating the spatial covariance matrix. Spatial smoothing is based on combining L-element overlapping subarrays of the original M-element array. Let

$$\mathbf{x}_k(t) = \begin{bmatrix} u_k(t) & u_{k+1}(t) & \dots & u_{k+L-1}(t) \end{bmatrix}^T$$

denote the k-th subarray within $\mathbf{u}(t)$. Then, the spatially smoothed sample correlation matrix $\widetilde{\mathbf{R}}(t)$ is defined as [36]:

$$\widetilde{\mathbf{R}}(t) = \frac{1}{(M-L+1)N} \sum_{n=t-N+1}^{t} \sum_{k=1}^{M-L+1} \mathbf{x}_k(n)\mathbf{x}_k(n)^H. \tag{2.8}$$

The size of $\widetilde{\mathbf{R}}(t)$ is $L \times L$, whereas the size of $\mathbf{R}(t)$ is $M \times M$. Consequently, using Equation (2.6) with $\widetilde{\mathbf{R}}(t)$ will produce a weight vector of size L rather than M. We denote this vector by $\widetilde{\mathbf{w}}(t)$:

$$\widetilde{\mathbf{w}}(t) = \frac{\widetilde{\mathbf{R}}(t)^{-1}\mathbf{d}}{\mathbf{d}^H\widetilde{\mathbf{R}}(t)^{-1}\mathbf{d}}. \tag{2.9}$$

Then, the beamformer output can be computed as follows [34]:

$$y(t) = \frac{1}{M-L+1} \sum_{k=1}^{M-L+1} \widetilde{\mathbf{w}}(t)^H\mathbf{x}_k(t). \tag{2.10}$$

Shan and Kailath [36] have shown that the condition

$$L \leq M/2$$

must be satisfied for this scheme to work. Increasing the subarray length L will enhance the image resolution at the cost of a less robust solution as discussed in [37]. In this work we have chosen the maximum possible

$$L = M/2$$

to get the best resolution.

While computing the output, $y(t)$, the same set of weights is applied to the subarrays, as it is seen from (2.10). To reduce the computational cost, we compute the subarray's average vector first, and then calculate the output

$$y(t) = \frac{\widetilde{\mathbf{w}}(t)^H}{M-L+1} \sum_{k=1}^{M-L+1} \mathbf{x}_k(t). \tag{2.11}$$

2.3.2 Simulation Results

For our study, we have also implemented the conventional LCMV beamformer as well as two non-adaptive beamformers and applied these methods to different phantoms. We then compared the results with the results achieved by the methods proposed in this thesis (see Section 5).

Our simulation study involves two different phantoms, a 12-point phantom and a point-scatterer-cyst phantom, and two different transducer arrays. We have used Matlab (version 7.6) and the FIELD II simulation tool (version 3.16) [21] for Linux to generate these phantoms as well as the corresponding ultrasound images (FIELD II is briefly described in Appendix A).

The 12-point phantom consists of point targets placed in the near field, at the focus point, and in the far field with a distance of 10 mm, starting at 30 mm from the transducer surface. A 4-MHz 96-element phased array transducer with 2.5-μm spacing was used. The element height was 7 mm and the width was 1/2 the wavelength. A single transmit focus was placed at 60 mm from the transducer, and receive focusing was done at 10-mm intervals. The sampling frequency was set to 100 MHz. The image contained 64 lines with the 0.47-degree separation. The purpose of this phantom is to help us evaluate the spatial resolution performance.

The point-scatterer-cyst phantom consists of a point target, a highly scattering region (with a radius of 1.5 mm), and a water-filled cyst region (with a radius of 2 mm), all centered at the transducer transmit focus and at the lateral distance of -15 mm, -5 mm and 10 mm, respectively. A large number of scatterers (10 scatterers within a resolution cell of λ^3) is used to generate a fully developed speckle [17]. Transmit and receive focusing was done as with the point phantoms. A 3-MHz 192-element linear array transducer with 64 active elements was used. The element height was 5 mm, the width was the same as the wavelength, and the spacing between

elements was 50 μm. The 40-mm image consisted of 64 image lines. The purpose of this phantom is to help us evaluate the contrast performance.

In both phantoms, image lines were saved separately for individual processing. We have configured the FIELD II tool to apply the transmit and receive focusing. Fixed apodization was applied during transmission. During reception, however, our studied receive beamformers would compute and apply the weights to the received echos. The signal envelope was compressed to a 60-dB dynamic range.

Figures 2.1 and 2.2 illustrate the advantage of the LCMV beamformer (with spatial smoothing) over two non-adaptive beamformers that use either uniform weighting (delay-and-sum) or an apodization function (here, the Kaiser window configured for 46-dB sidelobe attenuation). The Kaiser function provides a parametric window and thus we have used this window rather than fixed windows such as Hamming. Harris [15] has shown the optimality of the Kaiser window among other classical windows. By varying the shape parameter of the Kaiser window, a wide range of sidelobe attenuation can be adjusted. We have set this parameter to achieve a compromise between sidelobe attenuation and mainlobe width.

As shown in Figure 2.1, the points are imaged as lines when applying non-adaptive beamformers. Moreover, the image resolution degrades along the axial distance at points further from the transmit focus point (60 mm). Figure 2.1 shows that the LCMV beamformer (using a single snapshot: $N = 1$) gives a better resolution.

As shown in Figure 2.2, the radius of the water-filled cyst has been considerably reduced when using non-adaptive beamformers. Moreover, the cyst does not have a circular shape. The conventional LCMV beamformer results in significant reduction in the speckle. Also, the water-filled cyst has attained a more circular shape with a size closer to the real value. On the other hand, the point target is completely vanished, and the highly scattering region has been diminished (more black points

have been mapped to this region). We found that using two snapshots ($N = 2$) in this case produces a better-quality image than using only a single snapshot, as had been previously used by other groups. As the number of snapshots increases to two, a better contrast against speckled background is achieved, as illustrated in Figure 2.3. The target point is now detectable, and the highly scattering region contains fewer black points, while the water-filled cyst has preserved its circular shape. Using a larger number of snapshots ($N = 4$) worsened beamformer performance degradation (see Figure 2.3). Hence, our simulations in the sequel will always use two snapshots (to estimate the covariance matrix).

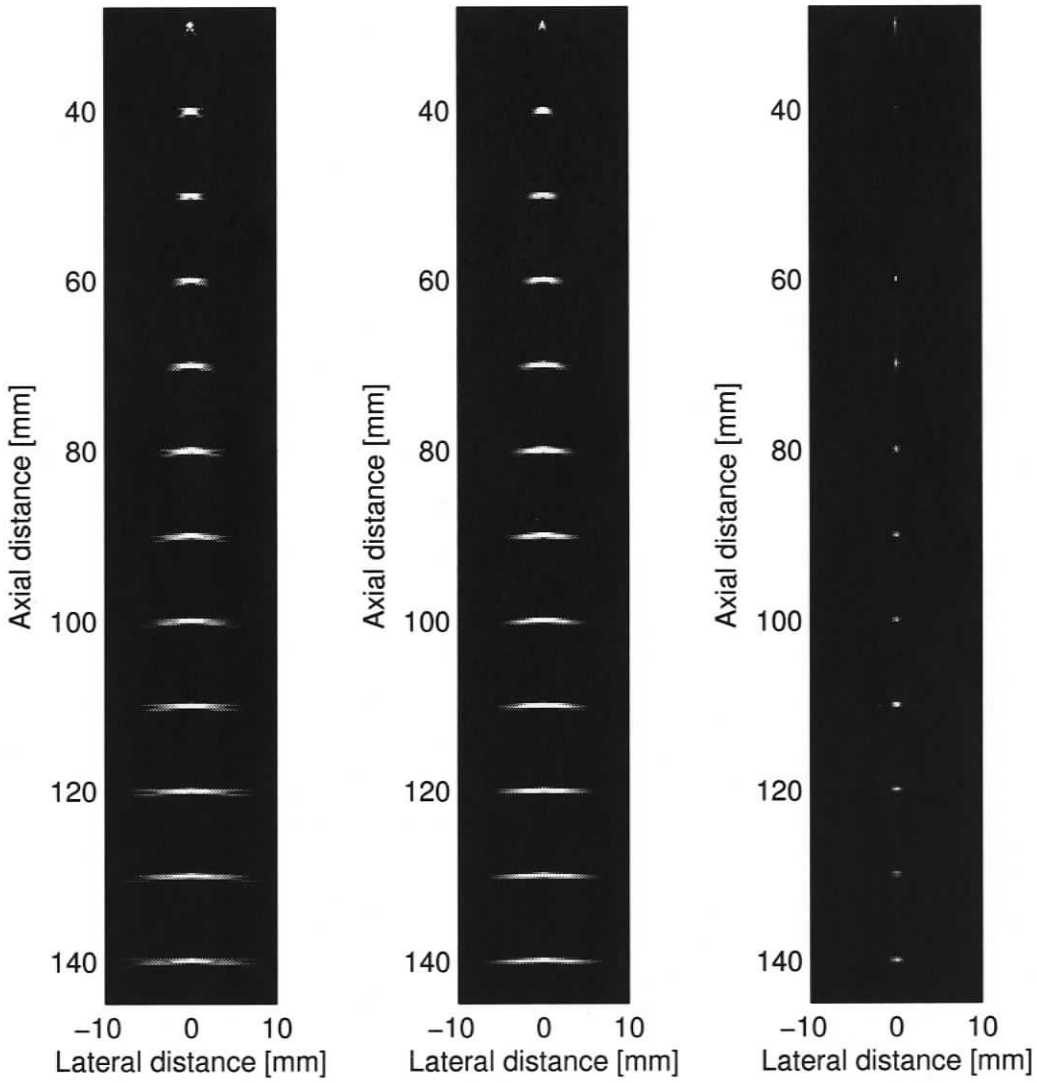


Figure 2.1: Performance of delay-and-sum (left), Kaiser window (center), and LCMV (right) beamformers: 12 single point phantom placed in 10-mm intervals, 64 image lines, 4-MHz 96-element phased array transducer (96 active elements, transmit focus at 60 mm, dynamic receive focus at 10 mm intervals), spatial smoothing with $L = 48$, using one snapshot ($N = 1$).

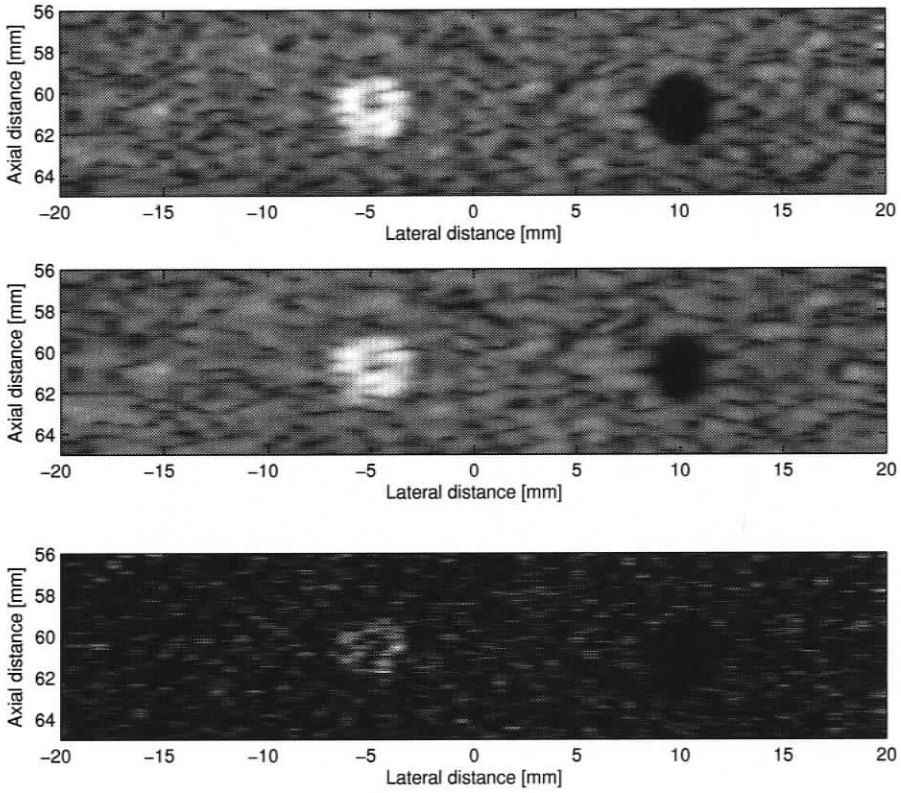


Figure 2.2: Performance of delay-and-sum (top), Kaiser window (center), and LCMV (bottom) beamformers: point-scatterer-cyst phantom placed at 60 mm (lateral placement: single point at −15 mm, 3-mm scattering region at −5 mm, 4-mm water-filled cyst at 10 mm), 64 image lines, 3-MHz 192-element linear array transducer (64 active elements, transmit focus at 60 mm, dynamic receive focus at 10 mm intervals), spatial smoothing with $L = 32$, using one snapshots ($N = 1$).

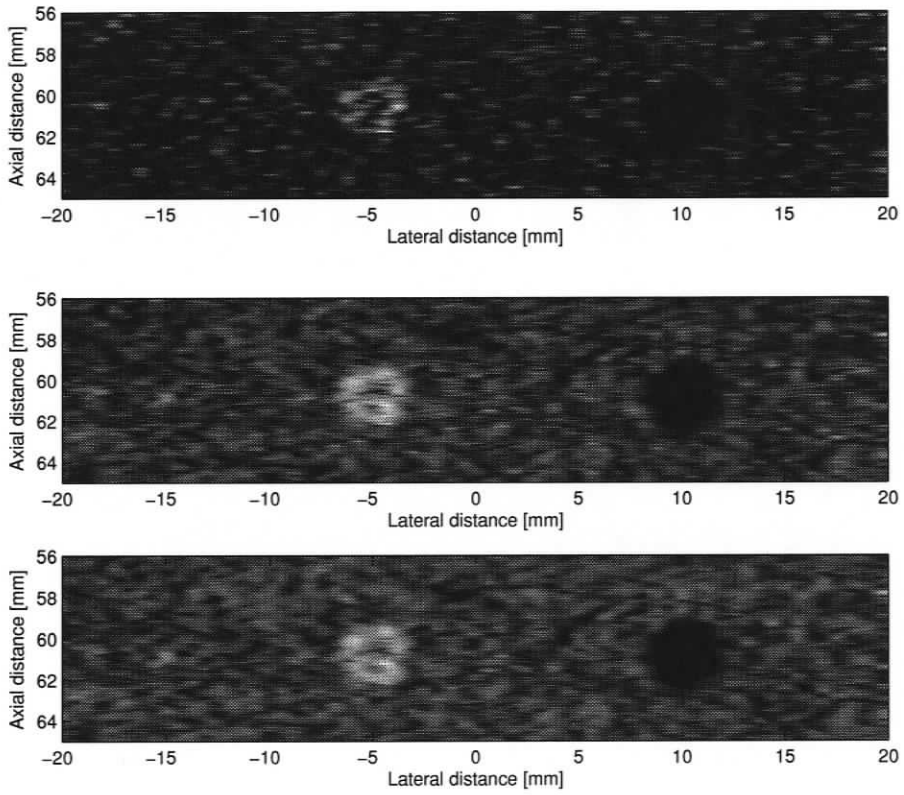


Figure 2.3: Performance of LCMV beamformer using 1 snapshot (top), 2 snapshots (center), and 4 snapshots (bottom): point-scatterer-cyst phantom (cf. Figure 2.2).

2.4 Related Work

The focus of this thesis is LCMV beamforming, which is one of the adaptive beamforming methods, applied to ultrasound imaging. We provide a brief review of literature on adaptive beamformers in Section 2.4.1 and reiterate our contributions in Section 2.4.2. Prior to that, however, we would like to mention other (non-beamforming) techniques for ultrasound image enhancement. These techniques aim at speckle reduction and are not directly related to our work; nevertheless, we mention them here for the sake of completeness.

Numerous speckle reduction methods have been proposed in the ultrasound literature. These can be categorized as follows: frequency compounding, spatial compounding, and post-processing despeckle filtering [7]. The first two are based on averaging a number of uncorrelated or partially correlated images. The third one relies on local information within an image.

In frequency compounding, the spectrum of the received signal is divided into subbands. Each of these subbands is processed individually to create an image. It has been shown that by compounding these images the background speckle can be reduced. On the other hand, however, the smaller bandwidth of each subband results a lower spatial resolution. References [7, 38, 40] provide further details on frequency compounding.

For spatial compounding, multiple images of a fixed object are produced either by slowly moving the transducer normal to the scan plane, or by steering the beams in different angular views. A loss in spatial resolution is expected in the former method. The image frame rate is reduced in both methods as a result of the averaging process. References [7, 38, 41] provide further details on spatial compounding.

Post-processing despeckle filtering approaches make use of local image statistics and homogeneity, as well as median, geometric, homomorphic, diffusion, and wavelet

filtering. Reference [27] reports a comprehensive study of 10 despeckle filtering methods applied to 440 ultrasound images of the carotid artery. Detailed assessments have involved texture analysis, comparison of various image quality metrics, and visual evaluation by medical experts. The results of this study suggest that a local-statistics moving-window filter gives the best performance. The speckle reduction filters are not without limitations [27]: they are sensitive to the size and shape of the window, which may lead to over smoothing and blurry images (large windows), or an ineffective despeckle filtering (small windows). Also, the required threshold may have to be estimated empirically. An inappropriate threshold may lead to average filtering and noisy boundaries. In most despeckle filtering methods, speckle in the neighborhood of an edge will remain after filtering.

2.4.1 Adaptive Beamforming

Beamforming has a long history (dating back to the 1960's) and finds its applications in several fields such as radar, sonar, seismology, radioastronomy, wireless communication, and biomedical imaging [8, 20, 24, 29, 30, 42]. Beamforming methods can be classified as either non-adaptive or adaptive. The former uses data-independent window functions, assuming that the desired signal characteristics are known, while the latter relies on statistical signal estimates from incoming data to minimize the interference. The LCMV beamformer belongs to the class of adaptive beamformers.

The performance of adaptive beamformers can be optimized based on various statistical criteria, e.g., minimum mean square error, maximum signal-to-noise ratio, maximum likelihood, minimum noise variance, minimum output power, maximum array gain [30]. Following Van Veen and Buckley [43] we can categorize optimum beamformers as follows.

- **Multiple sidelobe canceller:** The multiple sidelobe canceller (MSC) consists

of a main channel, and one or more auxiliary channels. The output of the auxiliary channels are subtracted from the output of the main channel. The main channel can be a data independent beamformer. All channels receive the interfering signals. The weights of the auxiliary channels are chosen to cancel the main channel interfering signals. Requiring a zero beamforming response to all interference signals, if at all possible, results in significant white noise gain. Therefore, the weights are chosen to minimize the expected value of the total output power instead, but this also cancels the desired signal. As a result, MSC is only effective in applications where the desired signal is weak relative to the interference, or when the desired signal is absent during certain time periods. In latter case, weights are computed only in the absence of the desired signal. In ultrasound imaging neither of these two conditions is satisfied, since the interference signals are in fact the scattered transmit signals that reach the transducer array with different phases due to multipath propagation. The relative strength of the signals are not known, and the desired signal is always present.

- **Use of reference signal:** Given that sufficient information is available about the desired signal, an output reference signal can be generated. Then, the apodization weights of the beamformer are chosen to minimize the mean square error between the actual beamformer output and the reference signal. It is assumed that the reference signal is uncorrelated with the interfering signals. This method, however, is not applicable in diagnostic ultrasound imaging due to the lack of knowledge about the desired signal, which depends on the patient and the tissue being imaged.

- **Maximization of signal-to-noise ratio:** The weights in this method are chosen to maximize the beamformer output signal-to-noise ratio (SNR), which

requires the knowledge of both desired signal and interfering noise covariance matrices. In some applications, e.g. active radar, a noise covariance matrix can be estimated during the time when no signal is being transmitted, and a signal covariance matrix can be obtained by knowing the transmitted pulse and the desired direction. In ultrasound imaging, however, the noise covariance matrix can not be estimated, making this method inapplicable.

- **Linearly constrained minimum variance beamforming:** As discussed above, the previous three beamforming methods are not applicable in ultrasound imaging. A more general approach that overcomes the limitations imposed by ultrasound imaging is to apply linear constraints to the weight vector. This gives rise to the LCMV (linearly constrained minimum variance) beamformer, where the weights are chosen to minimize the output power, subject to a unit-gain constraint for the signal from the desired direction. By satisfying the constraint, the desired signal is preserved, while output power minimization reduces interfering signals, as described in Section 2.3.

The above beamformers operate in time domain. It is also possible to beamform in frequency domain [42], which usually involves performing Discrete Fourier Transform (DFT) on signal samples arriving from individual elements of the transducer array, separately weighting each frequency subband, appropriately summing the subbands, and then converting the resulting output samples to time domain via Inverse Discrete Fourier Transform (IDFT). This approach is motivated by the need to handle broadband signals, and such a need often arises in ultrasound imaging. Broadband beamforming can also be done in time domain [42], by placing a tapped delay line (or FIR filter) on the signal path from each element of the transducer array.

Several research groups have applied various beamforming methods specifically to the ultrasound data. These methods include both non-adaptive beamforming

[12–14,31–33,35] and adaptive beamforming [17–19,28,34,37,45–47]. Next, we provide a brief review of the latter set of references, as they are closely related to our work [25].

Mann and Walker [28] have studied the performance of the minimum variance Frost beamformer [9], which is a broadband time-domain beamformer with multiple linear constraints. The authors used a tenth-order FIR filter with the same center frequency as the transducer, and the weight computation involved two matrix inversions. The reported results for wire targets and a cyst phantom showed improvements in point resolution and contrast. However, no attempts have been made to analyze the computational complexity or to treat it as a controllable parameter.

Holfort *et al.* [17–19] have studied the performance of the minimum variance frequency-domain broadband beamformer using 128 DFT blocks and spatial smoothing with a subarray size equal to the quarter of the 128-element aperture size. FIELD II simulation results were reported for both synthetic-aperture and plane-wave ultrasound data (point targets and a cyst pahntom). These results were compared to delay-and-sum beamforming (with Boxcar and with Hanning weights) and showed an increase in lateral resolution. Again, no attempts have been made to analyze or control the computational complexity of the beamforming process.

Viola *et al.* [46] have applied four adaptive beamformers originally proposed in [1, 4, 9, 48] during ultrasound imaging of wire targets and a cyst. It is worth mentioning here that the complexity of the three beamformers from [1, 4, 48] are comparable to or greater than that of the other (Frost) beamformer [9], which in turn is significantly more (computationally) complex than the LCMV beamformer described in Section 2.3. In [46] the results were compared against delay-and-sum beamforming with no apodization and, as expected, showed substantial improvements in resolution. Noting that the beamformer from [4] had a superior performance, the authors proposed its broadband extension in [45]. Their method uses a linear model of the image formation. The region of interest is subdivided into a collection of

hypothetical targets at arbitrary positions in a sampling grid. The spatial response of individual targets is computed through experiments, simulations or theory. Given this model and the received data, the proposed method finds the position and intensity of the real targets by using the maximum a posteriori (MAP) estimation technique. While this beamformer offers significant performance improvements over delay-and-sum beamforming, its computational complexity appears to be prohibitively high, which limits its potential use in practice.

The common feature of the beamformers described above (per [17–19,28,45,46]) is their capacity to perform broadband processing. This, however, comes at the expense of a significant increase in the computational load in comparison to narrowband beamforming. Also, none of those works have directly compared broadband beamforming to narrowband beamforming to clearly justify the added complexity. References [34,37,44,47] reviewed next, as well as this thesis, do not explicitly pursue broadband capabilities, yet the reported results still show significant improvements in resolution and contrast in comparison to conventional delay-and-sum beamforming.[1]

Sasso and Cohen-Bacrie [34] have investigated the Capon beamformer [6] operating in time domain with a single linear constraint and spatial smoothing. This beamformer has the same formulation as the LCMV beamformer described in Section 2.3.[2] The authors' simulation results for a cyst phantom showed improvements in point resolution and contrast. A similar study has been conducted by Synnevag *et al.* [37] using wire targets and a heart phantom. The authors have considered both spatial smoothing and diagonal loading for improving beamformer robustness (at the expense of reduced lateral resolution) and concluded that the former (using shorter

[1]Delay focusing described in Section 2.1 is what makes the beamforming weight computations frequency-independent.

[2]The single-constraint LCMV beamformer described in Section 2.3 is also called minimum variance distortionless response (MVDR) beamformer [42].

subarrays) is more practical than the latter. Their results again demonstrated the advantages of adaptive minimum variance beamforming over conventional delay-and-sum beamforming. Neither [34] nor [37], however, discuss computational complexity details or consider the use of a suboptimal LCMV beamformer.

Wang *et al.* [47] have directly addressed the uncertainty problem associated with the desired direction vector, in order to improve the robustness of the Capon beamformer using a synthetic aperture. Their proposed methods rely on multiple insonifications to obtain different observations of the imaged medium. Vignon and Burcher [44] use a similar approach in the sense that different observations of the medium are recorded; however, they use multiple (16) focused beams as opposed to multiple (32/64/128) diverging beams in [47]. The advantage of using focused beams instead of a synthetic aperture is a higher signal-to-noise ratio and a lower sensitivity to medium motion. In both works, the experimental results for a wire, a cyst, and a heart phantoms [47] as well as *in-vivo* for heart and abdominal images showed increased resolution and contrast in comparison to delay-and-sum beamforming. While the proposed methods improve the Capon beamformer's performance, they also increase the computational and storage burden in comparison to the single-insonification methods [34, 37]. In this thesis, we use a single focused beam to obtain the weights, as in [34, 37], but also, unlike any of the previous works, we study the performance of *suboptimal* Capon beamforming and provide a detailed analysis of the resulting computational savings.

2.4.2 Our Contribution

As mentioned above (also see the first chapter of this thesis), our objective is to address the computational complexity problem associated with the single-constraint LCMV beamformer (described in Section 2.3). This beamforming method involves an

expensive matrix inversion operation when computing a weight vector. We propose and evaluate two alternatives that sacrifice the beamformer's optimality in order to reduce the computational load.

The first alternative is based on the well-known GSC implementation of the LCMV beamformer (see Chapter 3), where the optimal weights can be computed by the single-iteration Newton method. Instead of using the Newton method, which involves a matrix inversion as well, we consider computing the weights approximately, using gradient-driven multi-iteration quasi-Newton, steepest descent, and conjugate gradient methods that do not require matrix inversions. The main advantage of this approach is that the tradeoff between the computational complexity and the image quality becomes controllable via the number of iterations performed when computing a weight vector.

The second alternative is based on a simple idea of computing the weights at a reduced rate. We allow the beamformer to take several sampling periods to compute a weight vector. In other words, a matrix inversion is performed once per several samples (rather than once per every sample), which significantly reduces the computational burden. The main advantage of this approach is due to not only the controllability of the complexity-quality tradeoff, but also the relative simplicity of its implementation.

To our knowledge, neither of the above approaches have been studied before in the context of ultrasound beamforming. Our simulation results presented in this thesis show that the proposed alternatives to the optimal Capon beamforming can yield significant computational savings with tolerable image degradation.

Chapter 3

Gradient-Driven GSC-Based Beamforming

In this chapter we present the basic GSC theory and four gradient-driven optimization algorithms for computing the beamformer's weight vector. We also provide an analysis of the computational cost associated with GSC-based beamforming.

3.1 GSC Theory

The GSC structure is shown in Figure 3.1 and produces the following output [24]:

$$y(t) = [\mathbf{w}_q - \mathbf{B}\mathbf{w}_a(t)]^H \mathbf{u}(t), \tag{3.1}$$

where $\mathbf{B}$ is the blocking matrix, $\mathbf{w}_q$ and $\mathbf{w}_a(t)$ are the non-adaptive (quiescent) and adaptive weight vectors, respectively. One can interpret $[\mathbf{w}_q - \mathbf{B}\mathbf{w}_a(t)]$ as the weight vector $\mathbf{w}(t)$ of the LCMV beamformer without spatial smoothing. Provided that delay focusing is present (i.e., $\mathbf{d}$ is a vector of M ones – see Section 2.3), $\mathbf{w}_q$ and $\mathbf{B}$ are fixed, while $\mathbf{w}_a(t)$ varies to enable adaptive response. To match the LCMV beamformer behavior, we want to minimize the output power $P(t)$, while passing the

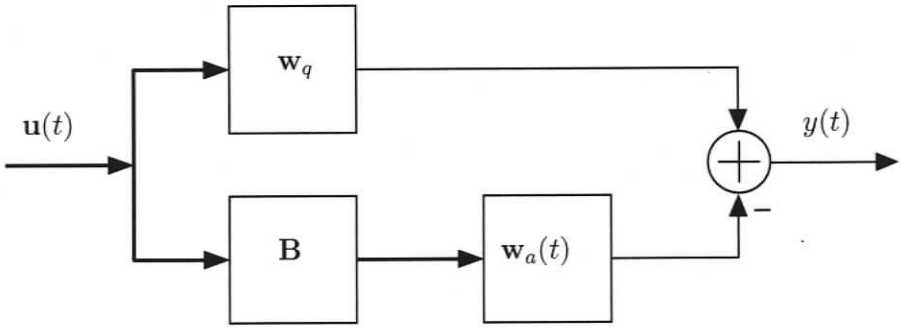


Figure 3.1: Generalized sidelobe canceller.

desired signal with a unit gain, and we also want to incorporate spatial smoothing. The unit gain constraint is satisfied by setting

$$\mathbf{w}_q = (1/M)\mathbf{d} \tag{3.2}$$

and by having $\mathbf{B}$ such that [24]

$$\mathbf{B}^H\mathbf{d} = \mathbf{0}. \tag{3.3}$$

¿From this equation we can see that $\mathbf{B}$ blocks the desired signal from flowing to the lower path of the GSC structure. Blocking is accomplished if the columns of $\mathbf{B}$ are linearly independent and sum up to zero (as a result, the number of columns of matrix $\mathbf{B}$ is equal or smaller than $M-1$). Griffiths and Jim [11] have discussed various blocking matrices. For the case of $M = 4$, two possibilities are

$$\mathbf{B}_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & -1 \\ -1 & -1 & 1 \\ -1 & 1 & -1 \end{bmatrix} \tag{3.4}$$

and

$$\mathbf{B}_2 = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{bmatrix}. \tag{3.5}$$

The columns of matrix (3.4) are computed from Walsh functions. In this work, we have chosen sparse matrix (3.5), as it involves less computation.

The output power

$$P(t) = [\mathbf{w}_q - \mathbf{B}\mathbf{w}_a(t)]^H \widehat{\mathbf{R}}(t)[\mathbf{w}_q - \mathbf{B}\mathbf{w}_a(t)] \tag{3.6}$$

is minimized by updating $\mathbf{w}_a$ based on the gradient $\mathbf{g} = \nabla P$ and the Hessian $\mathbf{H} = \nabla^2 P$ (both with respect to $\mathbf{w}_a$). Without spatial smoothing, we would have

$$\mathbf{g}(t) = -2\mathbf{B}^H \widehat{\mathbf{R}}(t)[\mathbf{w}_q - \mathbf{B}\mathbf{w}_a(t)], \tag{3.7}$$

$$\mathbf{H}(t) = 2\mathbf{B}^H \widehat{\mathbf{R}}(t)\mathbf{B}. \tag{3.8}$$

With spatial smoothing, however, we want to use $\widetilde{\mathbf{R}}$ whose size $(L \times L)$ is smaller than that of $\widehat{\mathbf{R}}$. Consequently, the sizes of the other vectors and matrices must be adjusted accordingly. Using symbol $\widetilde{\ }$ to indicate a size reduction due to the presence of spatial smoothing, we now redefine P, $\mathbf{g}$ and $\mathbf{H}$ as

$$P(t) = [\widetilde{\mathbf{w}}_q - \widetilde{\mathbf{B}}\widetilde{\mathbf{w}}_a(t)]^H \widetilde{\mathbf{R}}(t)[\widetilde{\mathbf{w}}_q - \widetilde{\mathbf{B}}\widetilde{\mathbf{w}}_a(t)], \tag{3.9}$$

$$\mathbf{g}(t) = -2\widetilde{\mathbf{B}}^H \widetilde{\mathbf{R}}(t)[\widetilde{\mathbf{w}}_q - \widetilde{\mathbf{B}}\widetilde{\mathbf{w}}_a(t)], \tag{3.10}$$

$$\mathbf{H}(t) = 2\widetilde{\mathbf{B}}^H \widetilde{\mathbf{R}}(t)\widetilde{\mathbf{B}}. \tag{3.11}$$

The sizes of $\widetilde{\mathbf{B}}$ and $\mathbf{H}$ are $L \times (L-1)$ and $(L-1) \times (L-1)$, respectively. The size of $\widetilde{\mathbf{w}}_a$ and $\mathbf{g}$ is $(L-1) \times 1$, and the size of $\widetilde{\mathbf{w}}_q$ is $L \times 1$. In the next section, we consider four options for computing $\widetilde{\mathbf{w}}_a(t)$, based on four gradient-driven optimization methods.

Given $\widetilde{\mathbf{B}}$, $\widetilde{\mathbf{w}}_q$, and final $\widetilde{\mathbf{w}}_a(t)$, we can compute the output of the GSC combined with spatial smoothing as

$$y(t) = \frac{[\widetilde{\mathbf{w}}_q - \widetilde{\mathbf{B}}\widetilde{\mathbf{w}}_a(t)]^H}{M-L+1} \sum_{k=1}^{M-L+1} \mathbf{x}_k(t). \tag{3.12}$$

One can interpret $[\widetilde{\mathbf{w}}_q - \widetilde{\mathbf{B}}\widetilde{\mathbf{w}}_a(t)]$ as the weight vector $\widetilde{\mathbf{w}}(t)$ of the LCMV beamformer with spatial smoothing.

3.2 Gradient-Driven Optimization Methods

Gradient-driven optimization methods are based on the gradient information used in the first-order and second-order Taylor series [3]. Our objective function P, given by Equation (3.9), is quadratic. As its Hessian $\mathbf{H}$ turns out to be positive definite, P is also convex [3]. To claim that $\mathbf{H}$ is indeed positive definite, we need to prove that the following necessary and sufficient condition is satisfied for all vectors $\mathbf{v} \neq \mathbf{0}$ [3]:

$$\mathbf{v}^T\mathbf{H}(t)\mathbf{v} > 0.$$

Using Equation (3.11), we rewrite it as

$$(\mathbf{v}\widetilde{\mathbf{B}})^H\widetilde{\mathbf{R}}(t)(\widetilde{\mathbf{B}}\mathbf{v}) > 0.$$

The correlation matrix $\widetilde{\mathbf{R}}$ is positive definite, based on its definition. The condition is satisfied if the vector $\widetilde{\mathbf{B}}\mathbf{v}$ is non-zero for all $\mathbf{v} \neq \mathbf{0}$. As a concrete example, consider $\widetilde{\mathbf{B}}$ given by (3.5) and

$$\mathbf{v} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^T,$$

which yields

$$\widetilde{\mathbf{B}}\mathbf{v} = \begin{bmatrix} v_1 & (v_2 - v_1) & (v_3 - v_2) & -v_3 \end{bmatrix}^T.$$

For the vector $\widetilde{\mathbf{B}}\mathbf{v}$ to be zero, all elements of the vector $\mathbf{v}$ must be zero. This, however, contradicts the assumption that $\mathbf{v} \neq \mathbf{0}$; therefore, the product $\widetilde{\mathbf{B}}\mathbf{v}$ must be non-zero.

Same reasoning applies to the larger-sized $\widetilde{\mathbf{B}}$ and $\mathbf{v}$ as well. Let $\widetilde{B}_{ij}$ denote the element of $\widetilde{\mathbf{B}}$ in row i and column j. Our choice of the matrix $\widetilde{\mathbf{B}}$, whose size is $L \times (L-1)$ in general, requires that:

- $\widetilde{B}_{kk} = 1$, for $k = 1, 2, ..., L-1$;
- $\widetilde{B}_{k(k-1)} = -1$, for $k = 2, 3, ..., L$;
- All other elements are zero.

Consequently, the product $\widetilde{\mathbf{B}}\mathbf{v}$, where $\mathbf{v}$ is of size $(L-1) \times 1$, will be in the form

$$\widetilde{\mathbf{B}}\mathbf{v} = \begin{bmatrix} v_1 & (v_2 - v_1) & (v_3 - v_2) & \dots & (v_{L-2} - v_{L-3}) & (v_{L-1} - v_{L-2}) & -v_{L-1} \end{bmatrix}^T.$$

For $\widetilde{\mathbf{B}}\mathbf{v}$ to be a vector of all zeros, we must have $v_1 = v_2 = ... = v_{L-1} = 0$, which contradicts the assumption that $\mathbf{v} \neq \mathbf{0}$, i.e., $\widetilde{\mathbf{B}}\mathbf{v}$ cannot be equal to $\mathbf{0}$. Hence, $\mathbf{H}$ is indeed positive definite.

Thus, the optimization problem at hand is *convex quadratic*. Consequently, the line search (usually required at each iteration) is omitted, and the step size can be computed by a closed formula [3], which drastically reduces the complexity of the iterative solution algorithm. At each iteration, the weights are updated as

$$\widetilde{\mathbf{w}}_{a[i+1]}(t) = \widetilde{\mathbf{w}}_{a[i]}(t) + \mu_{[i]}(t)\mathbf{d}_{s[i]}(t), \tag{3.13}$$

where i is the iteration number, $\mu_{[i]}(t)$ is the step size, and $\mathbf{d}_{s[i]}(t)$ is the search direction.

It is important to note that we permit multiple iterations per snapshot (i.e., per fixed t). As ultrasound signals are non-stationary, $\widetilde{\mathbf{R}}(t)$ is usually computed using a very few snapshots, and it may change rapidly with time. Consequently, $\mathbf{H}(t)$ may change rapidly with t as well, per Equation (3.11). The rationale for permitting multiple iterations per snapshot (i.e., per fixed t) is to facilitate searching with fixed

$\mathbf{H}(t)$. For the current snapshot at time t, we initialize $\widetilde{\mathbf{w}}_a$ to the the vector produced by the last iteration for the previous snapshot:

$$\widetilde{\mathbf{w}}_{a[0]}(t) = \widetilde{\mathbf{w}}_a(t-1).$$

In the special case of a single iteration per snapshot, we obtain the usual form of an adaptive weight update [16]: $\widetilde{\mathbf{w}}_a(t) = \widetilde{\mathbf{w}}_a(t-1) - ....$ Next, we present four gradient-driven methods used in this work.

3.2.1 Newton Method

Using the Newton method, weights are updated as

$$\widetilde{\mathbf{w}}_{a[i+1]}(t) = \widetilde{\mathbf{w}}_{a[i]}(t) - \mathbf{H}(t)^{-1}\mathbf{g}_{[i]}(t). \tag{3.14}$$

In relation to Equation (3.13), this method uses the search direction

$$\mathbf{d}_{s[i]}(t) = -\mathbf{H}(t)^{-1}\mathbf{g}_{[i]}(t),$$

and the step size

$$\mu_{[i]}(t) = 1.$$

Replacing $\mathbf{g}_{[i]}(t)$ by $[2\widetilde{\mathbf{B}}^H\widetilde{\mathbf{R}}(t)\widetilde{\mathbf{w}}_q + \mathbf{H}(t)\widetilde{\mathbf{w}}_{a[i]}(t)]$ in Equation (3.14) yields:

$$\widetilde{\mathbf{w}}_a(t) = 2\mathbf{H}(t)^{-1}\widetilde{\mathbf{B}}^H\widetilde{\mathbf{R}}(t)\widetilde{\mathbf{w}}_q, \tag{3.15}$$

that is the Newton method finds optimal $\widetilde{\mathbf{w}}_a(t)$ after a single iteration, which is to be expected when minimizing quadratic convex functions using this method. The conventional LCMV method also finds the optimal weight vector at once, according to Equation (2.9). The GSC-based beamformer employing the Newton method and the conventional LCMV beamformer produce the same-quality (optimal) output.

3.2.2 Steepest Descent Method

Despite their single-iteration convergence, the conventional LCMV and the Newton methods are computationally expensive because they require a matrix inversion. The steepest descent (SD) method provides a much simpler means to update the weights [3]:

$$\widetilde{\mathbf{w}}_{a[i+1]}(t) = \widetilde{\mathbf{w}}_{a[i]}(t) - \frac{\mathbf{g}_{[i]}(t)^T \mathbf{g}_{[i]}(t)}{\mathbf{g}_{[i]}(t)^T \mathbf{H}(t) \mathbf{g}_{[i]}(t)} \mathbf{g}_{[i]}(t), \tag{3.16}$$

where the search direction

$$\mathbf{d}_{s[i]}(t) = -\mathbf{g}_{[i]}(t),$$

and the step size

$$\mu_{[i]}(t) = \frac{\mathbf{g}_{[i]}(t)^T \mathbf{g}_{[i]}(t)}{\mathbf{g}_{[i]}(t)^T \mathbf{H}(t) \mathbf{g}_{[i]}(t)}.$$

The SD method does not involve the inverse Hessian, which results in significant computational savings; however, it suffers from slow convergence [3].

Remark: The steepest descent method with one iteration per snapshot is closely related to the classical least-mean-squares (LMS) algorithm. Combining spatial smoothing with the GSC based on the LMS algorithm was studied in [2]. Without spatial smoothing, the LMS algorithm would update $\mathbf{w}_a$ as follows [16]:

$$\mathbf{w}_a(t+1) = \mathbf{w}_a(t) + \mu \mathbf{B}^H \mathbf{u}(t) \mathbf{u}(t)^H [\mathbf{w}_q - \mathbf{B}\mathbf{w}_a(t)], \tag{3.17}$$

where μ is the constant step-size parameter. With spatial smoothing, one can use the following update method from [2]:

$$\begin{aligned} \widetilde{\mathbf{w}}_a(t+1) &= \widetilde{\mathbf{w}}_a(t) + \mu \widetilde{\mathbf{B}}^H \widetilde{\mathbf{R}}(t) [\widetilde{\mathbf{w}}_q - \widetilde{\mathbf{B}} \widetilde{\mathbf{w}}_a(t)] \\ &= \widetilde{\mathbf{w}}_a(t) - \mu \mathbf{g}(t)/2. \end{aligned} \tag{3.18}$$

The main difference between Equations (3.18) and (3.16) is the step-size parameter: in our case,

$$\mu(t) = \frac{\mathbf{g}(t)^T \mathbf{g}(t)}{\mathbf{g}(t)^T \mathbf{H}(t) \mathbf{g}(t)},$$

i.e., it is exactly prescribed, optimal (fastest convergence), and time-dependent.

3.2.3 Quasi-Newton Method

The steepest descent method can be very slow to converge to the optimal solution (especially at high Hessian eigenvalue spread as in ultrasound applications). The quasi-Newton (QN) method offers a compromise between the convergence rate and computational complexity and ranks among the most efficient algorithms [3]. The search direction in the QN methods does not rely on $\mathbf{H}(t)^{-1}$; instead, it is computed based on an $(L-1)\times(L-1)$ matrix $\mathbf{S}_{[i]}(t)$ that progressively converges to $\mathbf{H}(t)^{-1}$ as i grows, i.e., as the number of iterations increases. In our case (featuring a convex quadratic objective function), $\mathbf{S}_{[i]}(t)$ becomes identical to $\mathbf{H}(t)^{-1}$ in L iterations.

Using the QN method, weights are updated as

$$\widetilde{\mathbf{w}}_{a[i+1]}(t) = \widetilde{\mathbf{w}}_{a[i]}(t) - \frac{-\mathbf{g}_{[i]}(t)^T\mathbf{d}_{s[i]}(t)}{\mathbf{d}_{s[i]}(t)^T\mathbf{H}(t)\mathbf{d}_{s[i]}(t)}\mathbf{d}_{s[i]}(t), \tag{3.19}$$

where the search direction

$$\mathbf{d}_{s[i]}(t) = -\mathbf{S}_{[i]}(t)\mathbf{g}_{[i]}(t). \tag{3.20}$$

Note that in relation to Equation (3.13), the step size is

$$\mu_{[i]}(t) = \frac{-\mathbf{g}_{[i]}(t)^T\mathbf{d}_{s[i]}(t)}{\mathbf{d}_{s[i]}(t)^T\mathbf{H}(t)\mathbf{d}_{s[i]}(t)}.$$

Several methods have been developed to iteratively update matrix $\mathbf{S}_{[i]}$. Davidon, Fletcher, and Powell (DFP) and the Broyden, Fletcher, Goldfarb, and Shanno (BFGS) formulas are among the most efficient ones. Although there are no clear theoretical advantages, BFGS formula tends to yield more efficient algorithms [3]. Therefore, we

have applied the BFGS update formula:

$$\mathbf{S}_{[i+1]}(t) = \mathbf{S}_{[i]}(t) + \left(1 + \frac{\boldsymbol{\gamma}_{[i]}(t)^T \mathbf{S}_{[i]}(t) \boldsymbol{\gamma}_{[i]}(t)}{\boldsymbol{\gamma}_{[i]}^T(t) \boldsymbol{\delta}_{a[i]}(t)}\right) \frac{\boldsymbol{\delta}_{a[i]}(t) \boldsymbol{\delta}_{a[i]}^T(t)}{\boldsymbol{\gamma}_{[i]}^T(t) \boldsymbol{\delta}_{a[i]}(t)} - \frac{\boldsymbol{\delta}_{a[i]}(t) \boldsymbol{\gamma}_{[i]}^T(t) \mathbf{S}_{[i]}(t) + \mathbf{S}_{[i]}(t) \boldsymbol{\gamma}_{[i]}(t) \boldsymbol{\delta}_{a[i]}^T(t)}{\boldsymbol{\gamma}_{[i]}^T(t) \boldsymbol{\delta}_{a[i]}(t)} \tag{3.21}$$

where

$$\boldsymbol{\gamma}_{[i]}(t) = \mathbf{g}_{[i+1]}(t) - \mathbf{g}_{[i]}(t) \tag{3.22}$$

and

$$\boldsymbol{\delta}_{a[i]}(t) = \widetilde{\mathbf{w}}_{a[i+1]}(t) - \widetilde{\mathbf{w}}_{a[i]}(t). \tag{3.23}$$

Matrix $\mathbf{S}_{[i+1]}(t)$ is positive definite, if $\mathbf{S}_{[i]}(t)$ is positive definite and the condition

$$\boldsymbol{\delta}_{a[i]}^T(t) \boldsymbol{\gamma}_{[i]}(t) > 0 \tag{3.24}$$

is satisfied [3]. Having initialized the matrix $\mathbf{S}$ with the identity matrix, matrix $\mathbf{S}$ stays positive definite if this condition is satisfied at all time. Therefore, the above condition is checked in every iteration: if it is not satisfied, $\mathbf{S}_{[i+1]}(t)$ is assigned to the identity matrix. This ensures that the next change is in the steepest descent direction.

3.2.4 Conjugate Gradient Method

One of the most effective methods for solving convex quadratic problems is the conjugate gradient (CG) method [3]. It offers both low computational cost (comparable to the SD method) and fast convergence (comparable to the QN method). The CG method does not require the inverse Hessian and converges in $L - 1$ iterations in our case. At the same time, its computational complexity is only slightly higher than that of the SD method, which can be seen by examining the equations below.

The weights are computed as

$$\widetilde{\mathbf{w}}_{a[i+1]}(t) = \widetilde{\mathbf{w}}_{a[i]}(t) + \frac{\mathbf{g}_{[i]}(t)^T \mathbf{g}_{[i]}(t)}{\mathbf{d}_{s[i]}(t)^T \mathbf{H}(t) \mathbf{d}_{s[i]}(t)} \mathbf{d}_{s[i]}(t), \tag{3.25}$$

where the search direction is updated as

$$\mathbf{d}_{s[i+1]}(t) = -\mathbf{g}_{[i+1]}(t) + \beta_{[i]}(t) \mathbf{d}_{s[i]}(t) \tag{3.26}$$

using

$$\beta_{[i]}(t) = \frac{\mathbf{g}_{[i+1]}(t)^T \mathbf{g}_{[i+1]}(t)}{\mathbf{g}_{[i]}(t)^T \mathbf{g}_{[i]}(t)}. \tag{3.27}$$

We initialize $\mathbf{d}_s$ as follows [3]:

$$\mathbf{d}_{s[0]}(t) = -\mathbf{g}_{[0]}(t),$$

or equivalently,

$$\mathbf{d}_{s[0]}(t) = 2\widetilde{\mathbf{B}}^H \widetilde{\mathbf{R}}(t) \widetilde{\mathbf{w}}_q - \mathbf{H}(t) \widetilde{\mathbf{w}}_{a[0]}(t).$$

Note that the first iteration of the CG method produces the same results as the first iteration of the SD method, as Equations (3.16) and (3.25) become identical for $i = 0$.

3.3 Computational Complexity Analysis

Table 3.3 summarizes the computational costs of the conventional LCMV method and the four described GSC-based gradient-driven methods. To obtain this table, we have first divided each method into smaller computational steps, and then estimated the complexity of each step as shown in Tables 3.1 and 3.2. In our analysis, we take advantage of the symmetry of the correlation matrix $\widetilde{\mathbf{R}}(t)$.

For all methods, we need to compute $\widetilde{\mathbf{R}}(t)$ as shown in (2.8). The complexity of this step depends on the number of snapshots, N. For $N = 1$ (as in the case of our point phantoms), computing $\widetilde{\mathbf{R}}(t)$ involves $L(M-1)$ additions and $L(M+1)$ multiplications. For $N = 2$ (as in the case of our point-scatterer-cyst phantom), the

current correlation matrix must be added to the matrix computed at the previous time instance, and hence the number of addition is increased by $L(L+1)/2$.

The gradient-based methods require the gradient and Hessian, computed based on (3.10) and (3.11), respectively. Note that the blocking matrix $\widetilde{\mathbf{B}}$ is a sparse matrix with elements equal to -1, 0 or 1. Therefore, all products involving this matrix translate into addition operations only. Thus, zero multiplications are listed in Table 3.1 when computing intermediate matrix $\mathbf{D} = 2\widetilde{\mathbf{B}}^H\widetilde{\mathbf{R}}(t)$, intermediate vector $\mathbf{e} = \widetilde{\mathbf{w}}_q - \widetilde{\mathbf{B}}\widetilde{\mathbf{w}}_a(t)$, and Hessian $\mathbf{H}(t) = \mathbf{D}\widetilde{\mathbf{B}}$.

Both the conventional LCMV and Newton methods must compute a product in the form $\mathbf{A}^{-1}\mathbf{b}$, where $\mathbf{A}$ and $\mathbf{b}$ represent either $\widetilde{\mathbf{R}}(t)$ and $\mathbf{d}$ (for conventional LCMV), or $\mathbf{H}(t)$ and $2\widetilde{\mathbf{B}}^H\widetilde{\mathbf{R}}(t)\widetilde{\mathbf{w}}_q$ (for Newton). This product can be computed by solving $(\mathbf{C}\mathbf{C}^H)(\mathbf{A}^{-1}\mathbf{b}) = \mathbf{b}$, where $\mathbf{C}$ is the result of the Cholesky decomposition of $\mathbf{A}$, using gaxpy [10]. Following the standard practice, we first solve $\mathbf{C}\mathbf{a} = \mathbf{b}$ for $\mathbf{a}$, then solve $\mathbf{C}^H(\mathbf{A}^{-1}\mathbf{b}) = \mathbf{a}$ for $(\mathbf{A}^{-1}\mathbf{b})$, using backward and forward substitutions [10].

Finally, computing output $y(t)$ is of the same computational complexity for both the conventional LCMV method (using $\widetilde{\mathbf{w}}$ in Equation (2.11)), and the GSC-based methods (using $\mathbf{e}$ in Equation (3.12)). We need to perform $ML - L^2 + L - 1$ additions and $L+1$ multiplications.

Table 3.3 shows the total computational costs of the conventional LCMV method and the GSC-based gradient-driven methods. In this table, symbol I denotes the number of iterations per snapshot. The number of additions shown corresponds to the case of $N = 1$. If $N = 2$, we need to include $L(L+1)/2$ extra additions, as explained previously.

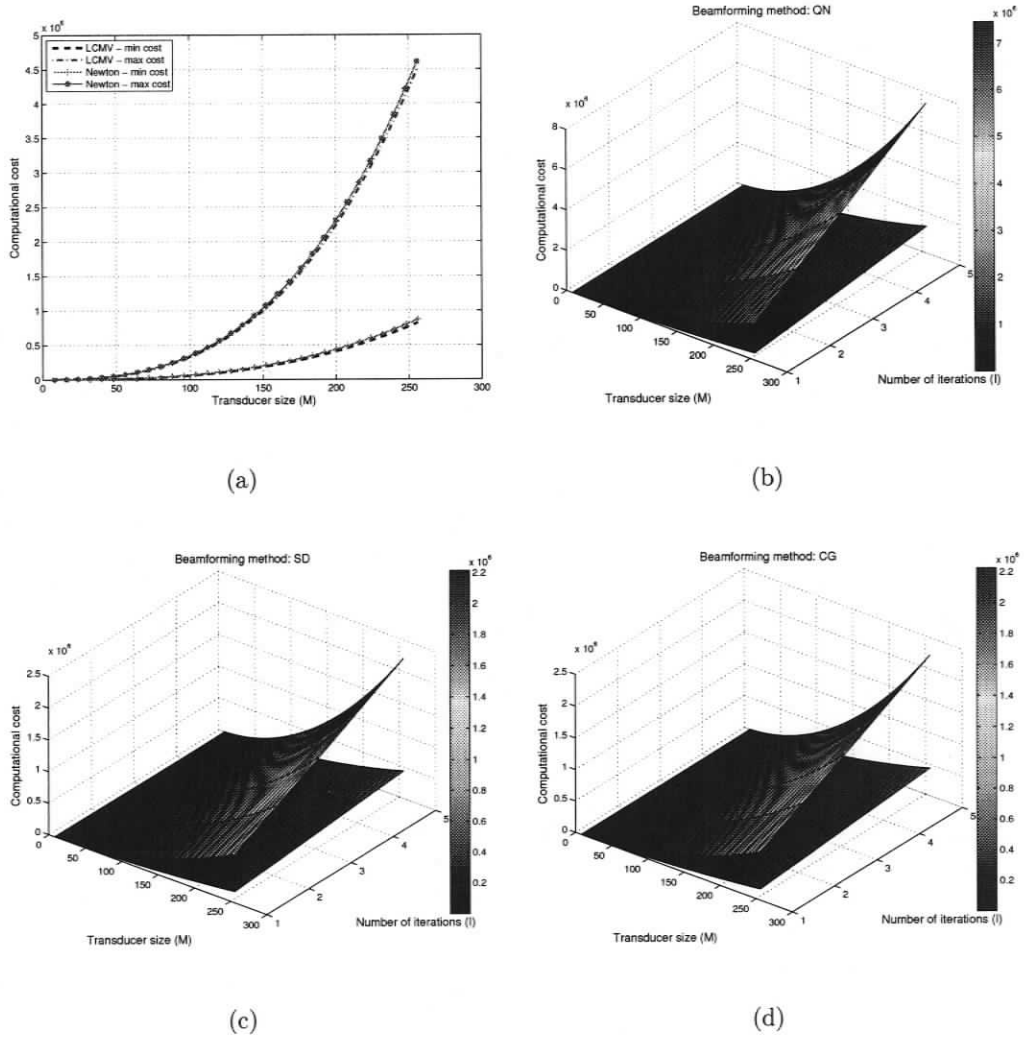
Figure 3.2 graphically illustrates the cost information provided in Table 3.3. We let I vary from 1 to 5 and M vary from 1 to 256, while setting $L = M/2$. We compute two values of the total cost for each method and each setting of variables I, M, and L. The first cost value, MIN, is computed assuming the same (unit) cost

Table 3.1: Computational cost of each beamforming step.

	Addition	Multiplication	Square Root
$\widetilde{\mathbf{R}}(t)$: $N=1$	$L(M-1)$	$L(M+1)$	0
$\widetilde{\mathbf{R}}(t)$: $N=2$	$L(M-1)+L(L+1)/2$	$L(M+1)$	0
$\mathbf{D}=2\widetilde{\mathbf{B}}^H\widetilde{\mathbf{R}}(t)$	$L(L-1)$	0	0
$\mathbf{e}=\widetilde{\mathbf{w}}_q-\widetilde{\mathbf{B}}\widetilde{\mathbf{w}}_a(t)$	$2(L-1)$	0	0
$\mathbf{g}(t)=-\mathbf{De}$	$(L-1)^2$	$L(L-1)$	0
$\mathbf{H}(t)=\mathbf{D}\widetilde{\mathbf{B}}$	$(L-1)^2$	0	0
$\mathbf{CC}^H$: LCMV	$(L^3-L+12)/6$	$(L^3+3L^2+2L)/6$	L
$\mathbf{CC}^H$: Newton	$L^3/6-L^2/2+L/3+2$	$(L^3-L)/6$	$L-1$
$(\mathbf{CC}^H)(\mathbf{A}^{-1}\mathbf{b})=\mathbf{b}$: LCMV	L^2-1	L^2+2L	0
$(\mathbf{CC}^H)(\mathbf{A}^{-1}\mathbf{b})=\mathbf{b}$: Newton	L^2-3L+2	L^2-L	0
$y(t)$	$ML-L^2+L-1$	$L+1$	0

Table 3.2: Computational complexity of weights updating.

Method	Addition	Multiplication	Square Root
LCMV	$L^3/6+L^2-L/6+1$	$L^3/6+3L^2/2+7L/3$	L
Newton	$L^3/6+L^2/2-5L/3+3$	$L^3/6+L^2-7L/6$	$L-1$
QN	$6L^2-9L$	$7L^2-9L+5$	0
SD	L^2-3	L^2+L-1	0
CG	L^2+2L-5	L^2+3L-2	0



(a) (b)

(c) (d)

Figure 3.2: Total computational costs of beamforming methods under consideration (see Table 3.3, $L = M/2$).

Table 3.3: Total computational cost of beamforming methods under consideration.

Method	Additions ($N = 1$)	Multiplications	Square Root
LCMV	$L^3/6 + 2ML - L/6$	$L^3/6 + 3L^2/2 + ML + 13L/3 + 1$	L
Newton	$L^3/6 + 5L^2/2 + 2ML - 14L/3 + 2$	$L^3/6 + 2L^2 + ML - L/6 + 1$	$L - 1$
QN	$2ML + L^2 - 3L + I(7L^2 - 9L - 1)$	$ML + 2L + 1 + I(8L^2 - 10L + 5)$	0
SD	$2ML + L^2 - 3L + I(2L^2 - 4)$	$ML + 2L + 1 + I(2L^2 - 1)$	0
CG	$2ML + L^2 - 5L + 2 + I(2L^2 + 2L - 6)$	$ML + 2 + I(2L^2 + 2L - 2)$	0

for each addition, multiplication, and square root operation. The second cost value, MAX, is computed assuming that each multiplication is 10 times more expensive than an addition, and each square root operation is 32 times more expensive than an addition. Thus, the range [MIN, MAX] captures, to some extent, a variety of possible implementations of adders, multipliers, and square-root circuits. As M increases, the difference between MIN and MAX increases as well, i.e., the computational cost becomes more sensitive to implementation details. Also note that the QN method quickly becomes more expensive than the others. In Chapter 5 (see Section 5.1), Table 5.1 gives specific numerical values of MIN and MAX for our two test examples: the single point phantom and the point-scatterer-cyst phantom.

Chapter 4

Reduced-Rate LCMV Beamforming

In the previous chapter we have shown that iterative gradient-driven beamforming, based on the GSC structure, can offer lower computational complexity than that of the conventional LCMV method, but at the expense of the reduced output quality (i.e., the weights are no longer optimal). A similar tradeoff between the computational cost and the output quality can be achieved with a different approach: we are to use the conventional LCMV beamformer, but also to reduce the rate at which its weights are computed.

4.1 Reduced-Rate Weight Updating

For each sampled input $\mathbf{u}(t)$, the conventional LCMV beamformer finds and applies optimal weights $\widetilde{\mathbf{w}}(t)$ to compute output $y(t)$, according to Equations (2.9) and (2.11). Let COST_w and COST_y denote the respective cost of computing the weights and the output for given $\mathbf{u}(t)$. The total cost per sampling period is $(\text{COST}_w + \text{COST}_y)$ for the conventional full-rate LCMV beamformer. If we update the weights once every N_w sampling periods (rather than every period), then the cost per sampling

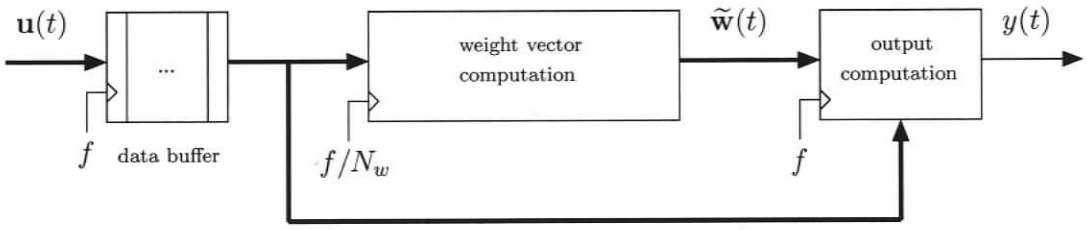


Figure 4.1: Block diagram of educed-rate LCMV beamformer.

period becomes (COST_w/N_w + COST_y). As the total cost is dominated by the cost of computing the weights, the potential savings can be substantial.

Effectively, we want to let the LCMV beamformer take as many as $(N_w - 1)$ extra sampling periods to compute a weight vector. For a given sampling instance t, the weights are computed based on $\widetilde{\mathbf{R}}(t)$, but they will not be ready for the next $(N_w - 1)$ sampling periods. While waiting for the new weight vector, the beamformer buffers new inputs $\mathbf{u}(t), \mathbf{u}(t+1), ..., \mathbf{u}(t+N_w-1)$ and produces the outputs by applying the previous weight vector, computed based on $\widetilde{\mathbf{R}}(t-N_w)$, to the previously buffered inputs $\mathbf{u}(t-N_w), \mathbf{u}(t-N_w+1), ..., \mathbf{u}(t-1)$. Figure 4.1 shows a basic block diagram of the reduced-rate LCMV beamformer. The weight updating rate is f/N_w, where f denote the beamformer's input data rate. In the sequel we normalize the weight updating rate to f for the sake of simplicity: e.g., $N_w = 2$ will correspond to the 1/2-rate updating, where 1/2 represents the fraction of f.

Note that in Figure 4.1 the input data buffer and the output computation block are clocked at the f rate. The weight computation block, however, is clocked at the f/N_w rate, i.e., its clock period is N_w-times longer. Let $1/f$ represent one time unit. The weight computation block should be able to complete its task in one time unit, thus being able to handle the case of $N_w = 1$ (full-rate LCMV beamforming), When $N_w > 1$, it has more time units available within its clock period, but it only needs

one time unit to complete its task. In other words, at each clock period the weight computation block is idle during $(N_w - 1)$ time units. This presents an opportunity for scaling down the block's supply voltage. Voltage scaling is a powerful technique for reducing the energy consumption – it is discussed next.

Digital CMOS circuits exhibit an approximately linear relationship between the circuit delay and the supply voltage [5]. Let V_{DD}, α, and C_L denote, respectively, the supply voltage, switching activity, and switched capacitance of the weight computation block. Then, the energy consumption can be expressed as [5]:

$$E = \alpha C_L V_{DD}^2. \tag{4.1}$$

It is important to note that both α and C_L, and therefore E, are specific to the choice of a reference time frame. In our case, we let the time frame of interest be of length N_w time units. Intuitively, C_L characterizes the block's physical structure, and α characterizes the block's workload (e.g., the amount of switching needed to complete the block's task). The value of C_L does not change, as the block's structure is assumed to fixed. The value of α can be reduced by letting the block do less work (within the reference time frame), which yields linear savings in E. When we reduce the weight updating rate by the factor of N_w, the weight computation block does N_w-times less work, i.e., α (and therefore E) is reduced by the factor of N_w. The most impact, however, is due to V_{DD}, as E is proportional to V_{DD}^2. In the case of $1/N_w$-rate updating, the delay of the weight computation block can be increased N_w-fold (while still fitting within the block's clock period), which means that the block's supply voltage can be scaled by the factor of $1/N_w$, yielding additional quadratic savings in energy. The total amount of energy saved by the weight computation block, due to the reduce-rate weight updating, can be estimated as follows:

$$\frac{\alpha C_L V_{DD}^2 - (\alpha/N_w) C_L (V_{DD}/N_w)^2}{\alpha C_L V_{DD}^2} \times 100\% = (1 - 1/N_w^3) \times 100\%. \tag{4.2}$$

Figure 4.2 shows a simple example, given $N_w = 2$.

- Figure 4.2(a): At the full rate, the reference time frame, from T_1 to T_2, contains two clock periods. The weight computation block computes two weight vectors between T_1 and T_2. We let E_1 denote the amount of energy consumed in this case (see the shaded area in the figure).

- Figure 4.2(b): Given $N_w = 2$, the time interval from T_1 to T_2 contains one clock period. The weight computation block is clocked at the rate of $f/N_w = f/2$ and computes one weight vector between T_1 and T_2. No voltage scaling is applied, and the energy consumption in this case is equal to $E_1/2$.

- Figure 4.2(c): The delay is increased two-fold, after we scale the block's supply voltage by the factor of $1/N_w = 1/2$. We denote the energy consumption in this case by E_2 – it is only $1/4$ of $E_1/2$ ($N_w^2 = 4$). Thus, $E_2 = E_1/8$.

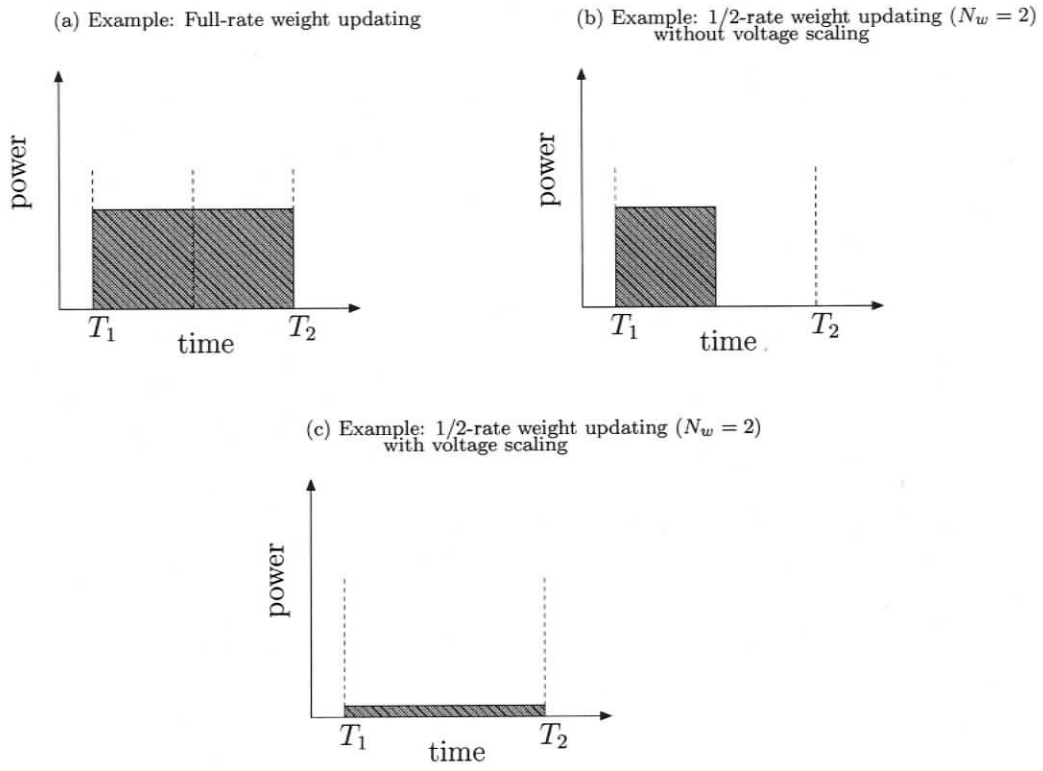


Figure 4.2: Weight vector computation: Energy savings given $N_w = 2$.

4.2 Computational Complexity Analysis

The computational complexity of the reduced-rate LCMV beamformer is summarized in Table 4.1. We let COST_w^* denote the computational cost of the reduced-rate weight computation block shown in Figure 4.1. Table 3.2 in Chapter 3 shows COST_w, the cost of the full-rate weight updating, when $N = 1$ (i.e., one snapshot is used). Note that when $N = 1$, COST_w^* is N_w-times less than COST_w, while COST_y (the cost of the output computation block in Figure 4.1) is the same as in the case of the full-rate LCMV beamformer. Recall that when $N = 2$ (i.e., two snapshots are used), COST_w must also include the cost of $L(L+1)/2$ extra additions. The extra operations in the reduced-rate beamforming case (included in COST_w^* when $N = 2$) are summarized in Table 4.2. This extra cost is due to averaging the input correlation matrix over two snapshots. In the case of the full-rate weight updating ($N_w = 1$), the correlation matrix

$$\frac{1}{M-L+1}\sum_{k=1}^{M-L+1}\mathbf{x}_k(t)\mathbf{x}_k(t)^T \tag{4.3}$$

is computed at time t and also saved for use at time $(t+1)$, when computing the average according to Equation (2.8). When $N_w > 1$, however, the weight computation block receives every N_w-th input snapshot, i.e., each received input snapshot will be separated from the next one by N_w cycles of the f clock. The correlation matrix computed and saved at t can no longer be paired with the correlation matrix computed at $(t+N_w)$ when computing the average. Instead, we average the correlation matrices of the input snapshots at times $(t+N_w)$ and $(t+N_w+1)$. Note that we use the next snapshot at time $(t+N_w+1)$, rather than the previous snapshot at time $(t+N_w-1)$. We have made this choice because the former is more relevant than the latter when computing the weight vector for the current group of N_w snapshots (the latter belongs to the previous group of N_w snapshots). Equation (4.3) is computed twice

Table 4.1: Reduced-rate LCMV beamforming: Computational costs per sampling period $1/f$.

COST	Addition	Multiplication	Square Root
COST_w^* $(N=1)$	$(L^3/6+ML+L^2-7L/6+1)/N_w$	$(L^3/6+ML+3L^2/2+10L/3)/N_w$	L/N_w
COST_w^* $(N=2)$	$(L^3/6+2ML+3L^2/2-5L/3+1)/N_w$	$(L^3/6+2ML+L^2+23L/6)/N_w$	L/N_w
COST_y	$ML-L^2+L-1$	$L+1$	0

Table 4.2: Reduced-rate LCMV beamforming: Extra computational cost per sampling period $1/f$ when $N=2$ (two snapshots).

Addition	Multiplication	Square Root
$(ML-L)/N_w$	$(ML-L^2/2+L/2)/N_w$	0

for every N_w-th input snapshot (i.e., for every input snapshot received by the weight computation block), causing the extra computational cost shown in Table 4.2.

Next, we derive an equation for estimating the energy savings of the reduced-rate LCMV beamformer in comparison to the conventional (full-rate) one. In accordance with the previous section, we let our reference time frame be N_w time units, where each time unit equals $1/f$. We use E_w and E_y to denote, respectively, the amount energy consumed by the weight computation block and the output computation block of the full-rate LCMV beamformer.

Full-rate LCMV beamforming: Over the interval of N_w time units (i.e., over N_w sampling periods of length $1/f$), the full-rate LCMV beamformer would compute both the weight vector and the output N_w times. The total computational cost is $(\text{COST}_w N_w + \text{COST}_y N_w)$, and the total amount energy consumed is $(E_w + E_y)$.

Reduced-rate LCMV beamforming with $N = 1$: Over the interval of N_w time units (i.e., over N_w sampling periods of length $1/f$), the reduced-rate LCMV beamformer computes the weight vector once and the output N_w times. The total computational cost is $(\text{COST}_w + \text{COST}_y N_w)$, and the total energy consumption is $(E_w/N_w^3 + E_y)$. Note that COST_w is the same as $\text{COST}_w^* N_w$. The energy savings in this case amount to

$$\frac{(E_w + E_y) - (E_w/N_w^3 + E_y)}{E_w + E_y} \times 100\% = \frac{1 - 1/N_w^3}{1 + E_y/E_w} \times 100\%, \tag{4.4}$$

where larger N_w and/or smaller E_y/E_w yield greater energy savings. The ratio E_y/E_w of the full-rate LCMV beamformer depends on the differences between the workload (i.e., switching activity α), the physical structure (i.e., switched capacitance C_L), and the supply voltage V_{DD} of the weight computation block and those of the output computation block. Let subscripts w and y refer to the weight computation block and the output computation block, respectively. For the sake of simplicity, we assume that both blocks have the same physical structure (e.g., a multi-core digital signal processor), so that $C_{Ly}/C_{Lw} = 1$. We also assume that the ratio α_y/α_w is equal by the ratio $\text{COST}_y/\text{COST}_w$. If we allow these two blocks to have different supply voltages, then V_{DDy} can be made smaller than V_{DDw}. Indeed, both blocks operate within the same reference time frame, yet the output computation block does only the α_y/α_w fraction of the weight computation block's work – this presents an opportunity for having $V_{DDy} = (\alpha_y/\alpha_w)V_{DDw}$. Consequently, $E_y/E_w = (\text{COST}_y/\text{COST}_w)^3$, and the energy savings can be expressed as follows:

$$\text{Energy Savings} = \frac{1 - 1/N_w^3}{1 + (\text{COST}_y/\text{COST}_w)^3} \times 100\%. \tag{4.5}$$

Reduced-rate LCMV beamforming with $N = 2$: When two snapshots are used, the ratio $\text{COST}_w^*/\text{COST}_w$ is not exactly $1/N_w$, as the reduced-rate and the full-rate beamformers require different amounts of extra computation to find the

correlation matrix. In this chapter so far, we have used $1/N_w$ as the scaling factor for the clock and the voltage of the weight computation block. Recall that the ratio of the reduced-rate and full-rate α's is what determines the voltage scaling factor. We have also let the ratio of α's be equal to the computational cost ratio. Consequently, we should now use $\text{COST}_w^*/\text{COST}_w$ as our scaling factor, which is slightly worse than $1/N_w$. Then,

$$\text{Energy Savings} = \frac{1-(\text{COST}_w^*/\text{COST}_w)^3}{1+(\text{COST}_y/\text{COST}_w)^3} \times 100\%. \tag{4.6}$$

Note that Equation (4.5) follows from Equation (4.6) when $\text{COST}_w^*/\text{COST}_w = 1/N_w$ in the single snapshot case.

Remark: In our derivations we have assumed that the supply voltage can be scaled as desired. Realistic scaling factors, however, may be limited by technological and operational constraints – it may not always be possible to match the ratio of α's. In such cases, our energy savings equation becomes

$$\text{Energy Savings} = \frac{1-(\text{COST}_w^*/\text{COST}_w)s_w^2}{1+(\text{COST}_y/\text{COST}_w)s_y^2} \times 100\% \tag{4.7}$$

where s_w and s_y represent, respectively, the scaling factors for the voltage of the reduced-rate weight computation block and the voltage of the output computation block, both in relation to the voltage of the full-rate weight computation block. Ideally, $s_w = \text{COST}_w^*/\text{COST}_w$ and $s_y = \text{COST}_y/\text{COST}_w$, i.e., both scaling factors are at their best, which would yield Equations (4.5) and (4.6).

Chapter 5

Evaluation Results

In this chapter we provide simulated images generated with the FIELD II tool [21], using gradient-driven GSC-based beamforming (described in Chapter 3) and reduced-rate LCMV beamforming (described in Chapter 4). We evaluate these imaging results against those produced by the conventional LCMV beamformer (described in Chapter 2).

5.1 Gradient-Driven GSC-Based Beamforming

Figures 5.1 - 5.9 illustrate the performance of the GSC-based beamformer based on the Newton, QN, SD, and CG methods. These images have been obtained with the FIELD II simulation tool [21] for our two test examples: the single point phantom ($M = 96$, $L = 48$, $N = 1$) and the point-scatterer-cyst phantom ($M = 64$, $L = 32$, $N = 2$). Table 5.1 shows the corresponding computational savings due to the use of GSC-based beamforming. These savings are with respect to the baseline cost of the conventional LCMV beamformer.

Each entry in Table 5.1 is a pair of numbers [MIN, MAX]. As discussed in Chapter 3 (see Section 3.3), MIN is computed assuming the same (unit) cost for each

addition, multiplication, and square root operation, while MAX is computed assuming that each multiplication is 10 times more expensive than an addition, and each square root operation is 32 times more expensive than an addition. Under such assumptions, which capture a variety of possible implementations of adders, multipliers, and square-root circuits, the baseline cost of the conventional LCMV beamformer ranges from 54393 to 296226 for the point phantom, and from 19297 to 102954 for the point-scatterer-cyst phantom. Note that greater M and L yield greater savings. Next, we take a closer look at the Newton, QN, SD, and CG methods individually.

- **Newton Method:** The Newton method finds optimal $\widetilde{\mathbf{w}}_a(t)$ after a single iteration. The conventional LCMV method also finds the optimal weight vector at once, according to Equation (2.9). Both beamformers produce the same optimal output, as illustrated by Figures 5.1 and 5.7. According to Table 5.1, however, the Newton method does not have a practical advantage over the conventional LCMV beamformer.

- **QN Method:** Table 5.1 shows that, given $M = 96$ and $L = 48$ (the point phantom), the QN method with one iteration per snapshot saves approximately 10% of computations. We cannot afford more than one iteration per snapshot, as it would make the QN method more expensive than the conventional LCMV method. The 1-iteration QN method produces poor $\widetilde{\mathbf{w}}_a(t)$, as illustrated by the image quality in Figure 5.3. The situation is worsened by the fact that $\mathbf{H}(t)$ is time-dependent, which may affect the continuing convergence from the previously computed $\mathbf{S}(t-1)$ to the current $\mathbf{H}^{-1}(t)$. The 1-iteration QN method may offer some practical value when $M = 128$ and $L = 64$ (over 25% savings), but its use still remains weakly motivated due to poor-quality images. As for the point-scatterer-cyst phantom ($M = 64$ and $L = 32$), the 1-iteration QN method

produces acceptable images; however, according to Table 5.1, the conventional LCMV method (producing optimal weights) is less expensive.

- **SD Method:** The SD method can afford as many as four iterations per snapshot for the point phantom, and three iterations per snapshot for the point-scatterer-cyst phantom, without exceeding the computational cost of the conventional LCMV beamformer. The savings range approximately from 0% to 60%, but we also face a degradation in image quality, especially in the case of the single point phantom (see Figures 5.2). From the practical point of view, the CG method is preferable over the SD method, as the former exhibits faster convergence yet has a similar computational cost in comparison to the latter (see Table 5.1).

- **CG Method:** The first iteration of the CG method produces the same results as the first iteration of the SD method (see Chapter 3). The subsequent iterations of the CG method are expected to produce better results than those of the SD method, which is confirmed by Figures 5.2 and 5.8. According to Table 5.1, the computational savings of both methods are comparable to each other, which makes the CG method a preferred choice.

A more detailed assessment of the image quality in the case of the single point phantom can be done based on Figures 5.4 and 5.5, showing the point spread function at the transmit focus. The full width at half maximum (FWHM) is used to evaluate the lateral resolution, and the sidelobe energy is used to evaluate the contrast – a smaller value of either evaluation metric indicates a better image. The sidelobe energy is measured for attenuation larger than 25 dB to include all sidelobe ripples. (This threshold is more stringent than a typical value of 30 dB required to avoid ultrasound imaging artifacts [7].) Table 5.2 shows the corresponding numerical values of interest, including the ratio of the sidelobe energy to the mainlobe energy. This ratio may also

be used as a contrast evaluation metric. (However, for the cases where the mainlobe width varies considerably, the ratio of the sidelobe energy to the mainlobe energy is not very informative.)

As expected, the conventional LCMV and Newton methods produce identical FWHM (0.351 mm), sidelobe energy E_{SL} (-36.9 dB), and mainlobe energy E_{ML} (1.3 dB): they are the best among all the methods considered here. The performance of the 1-iteration QN method (E_{SL}/E_{ML} = -19.8 dB) is worse than that of the 3-iteration SD method (E_{SL}/E_{ML} = -25.2 dB), and the latter is outperformed by the 3-iteration CG method (E_{SL}/E_{ML} = -30.6 dB). The practical utility of the 1-iteration SD/CG method is limited due to the poor image quality. On the other hand, using 5 iterations for the SD or CG methods exceeds the computational cost of the conventional LCMV beamformer, without producing optimal images (see Figure 5.3). Consequently, the 3-iteration CG method is recommended, offering approximately 20% computational savings (see Table 5.1) while matching the 0.351-mm FWHM of the LCMV method. As illustrated by Figure 5.5 and Table 5.2, the quality improvement per iteration achieved by the CG method is considerably greater in comparison to the SD method. Indeed, increasing the number of iterations from 1 to 3 improves E_{SL}/E_{ML} from -23.1 dB to -25.2 dB in the case of the SD method, while in the case of the CG method, the imporvment is from -23.1 dB to -30.6 dB.

Table 5.3 shows the contrast values, computed according to Equation (2.1), in the case of the point-scatterer-cyst phantom (see Figures 5.7 - 5.9). Figure 5.6 (obtained with the delay-and-sum beamformer) shows the sampled rectangle areas used when computing the average log-compressed signal envelope (to have a better estimation of the background signal the average is computed over six rectangles). As expected, the conventional LCMV and Newton methods produce the best contrast values for both the scattering region (2.214) and the water-filled cyst (0.943). The 1-iteration QN method is more expensive than the conventional LCMV beamformer;

Table 5.1: Gradient-driven GSC-based beamforming: Computational savings in our test examples (see Table 3.3 and Figures 2.1-5.9).

Beamforming Method	Savings (%): $M = 96$, $L = 48$, $N = 1$	Savings (%): $M = 64$, $L = 32$, $N = 2$
Newton	[-11.9, -5.0]	[-14.4, -5.9]
1-iteration QN	[8.6, 14.4]	[-16.2, -9.1]
1-iteration SD	[53.5, 63.2]	[39.1, 52.2]
2-iteration SD	[36.6, 46.1]	[17.9, 30.3]
3-iteration SD	[19.6, 29.0]	[-3.3, 8.5]
4-iteration SD	[2.7, 11.9]	[-24.5, -13.4]
5-iteration SD	[-14.2, -5.3]	[-45.7, -35.3]
1-iteration CG	[53.5, 63.2]	[39.1, 52.2]
2-iteration CG	[35.9, 45.4]	[16.6, 29.0]
3-iteration CG	[18.6, 27.9]	[-5.3, 6.5]
4-iteration CG	[1.3, 10.4]	[-27.1, -16.1]
5-iteration CG	[-16.0, -7.0]	[-49.0, -38.6]

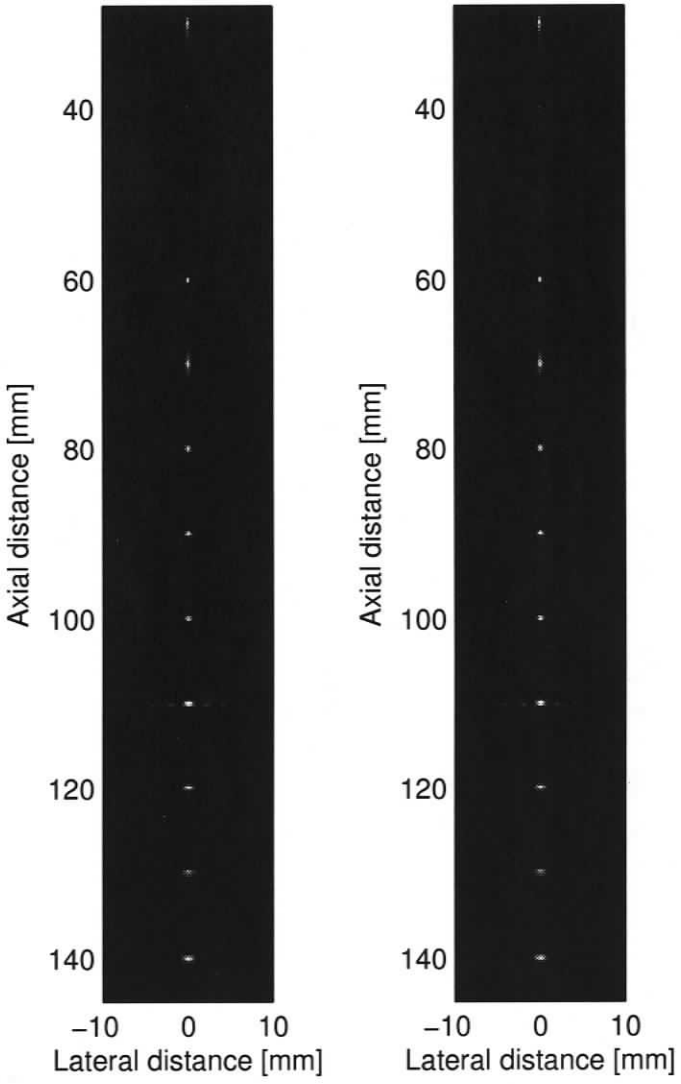


Figure 5.1: Performance of conventional LCMV (left) and Newton (right) methods: 12 single point phantom.

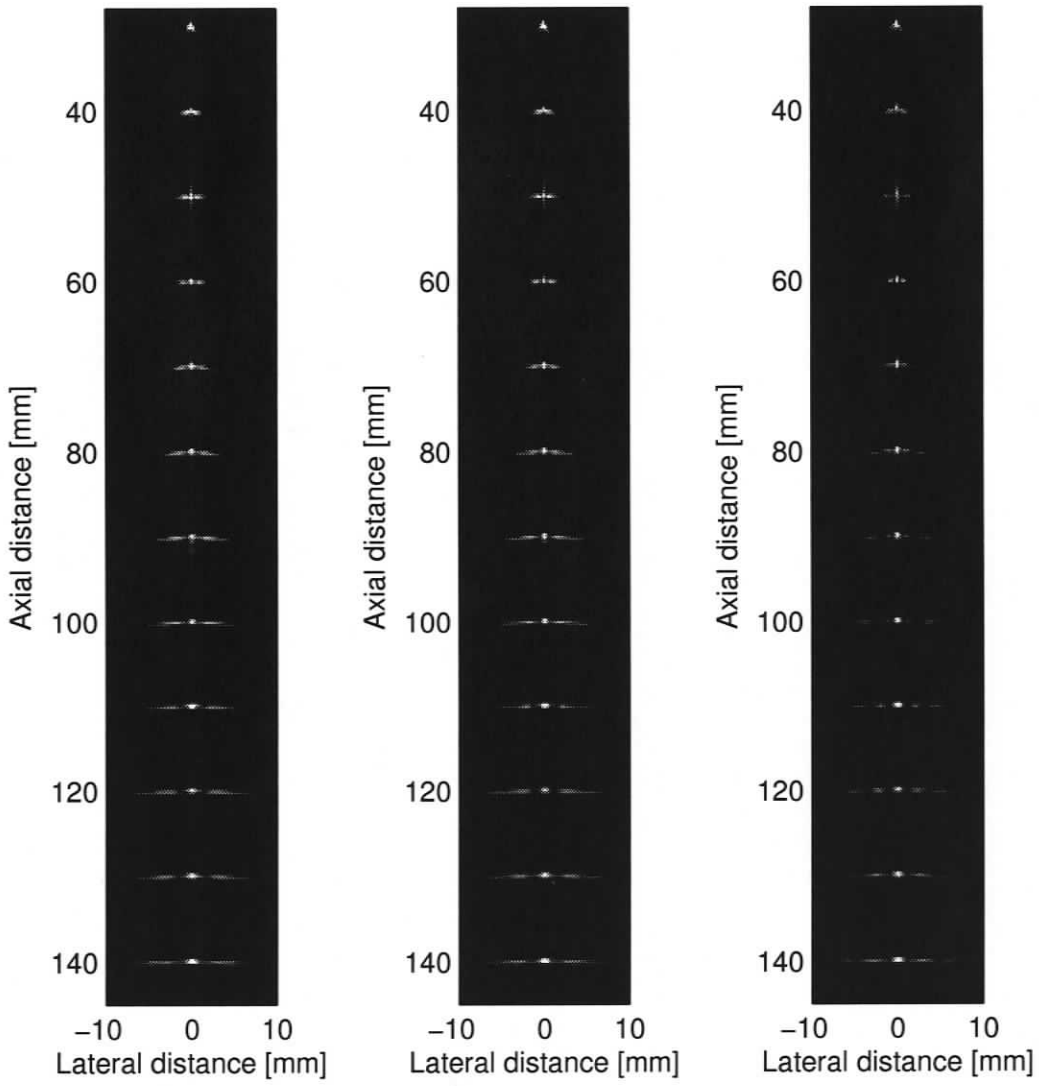


Figure 5.2: Performance of 1-iteration SD/CG (left), 3-iteration SD (center), and 3-iteration CG (right) methods: 12 single point phantom (cf. Figures 5.1 and 5.3).

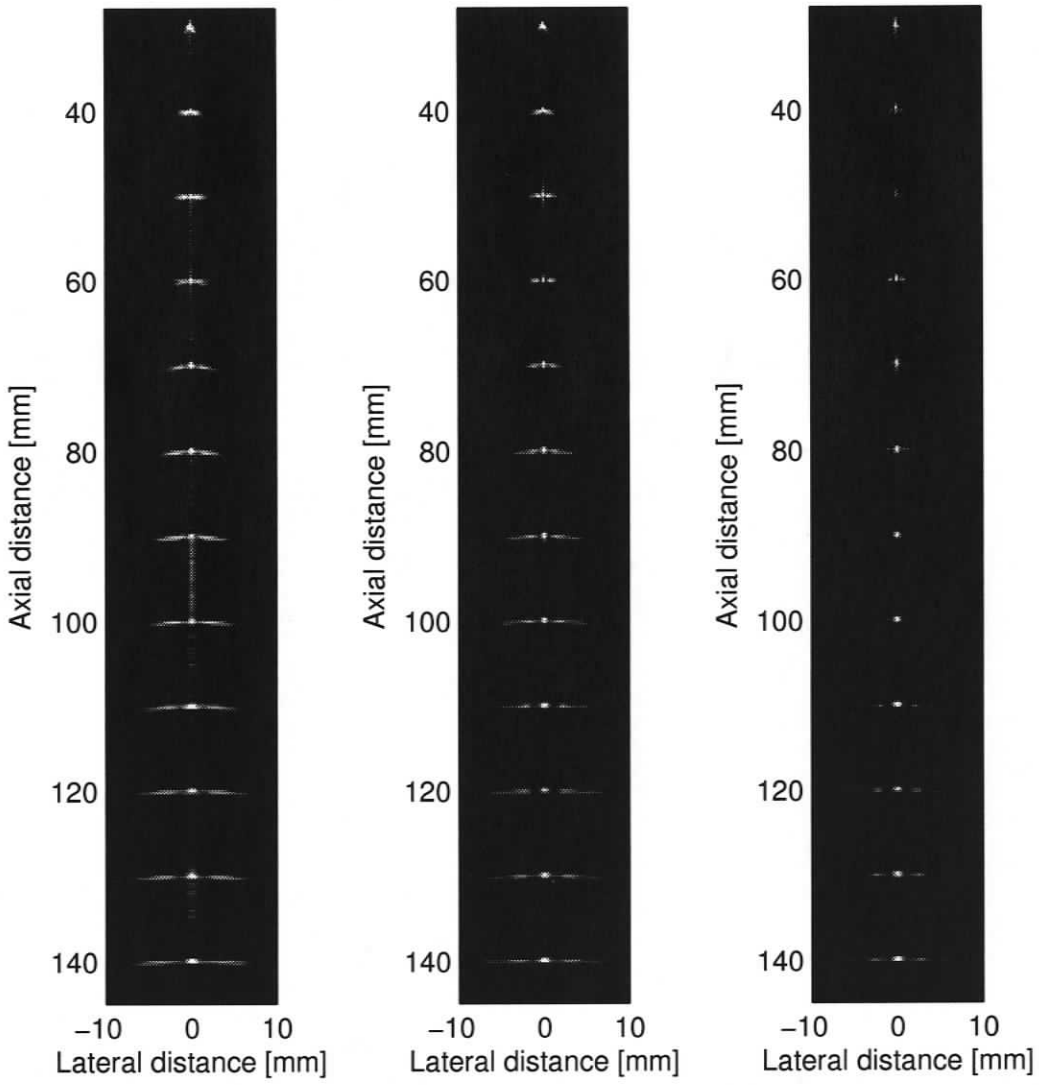


Figure 5.3: Performance of 1-iteration QN (left), 5-iteration SD (center), and 5-iteration CG (right) methods: 12 single point phantom (cf. Figures 5.1 and 5.2).

Table 5.2: Gradient-driven GSC-based beamforming: FWHM, sidelobe energy E_{SL}, and mainlobe energy E_{ML} at transmit focus (12 single point phantom).

Beamforming Method	FWHM (mm)	E_{SL} (dB)	E_{ML} (dB)	E_{SL}/E_{ML} (dB)
Delay-and-Sum	0.885	-23.777	2.964	-26.741
Kaiser Window	1.186	-24.770	4.134	-28.903
LCMV/Newton	0.351	-36.919	-1.291	-35.628
1-iteration QN	0.358	-20.962	-1.184	-19.779
3-iteration QN	0.361	-24.144	-1.152	-22.992
1-iteration SD/CG	0.360	-24.259	-1.172	-23.088
3-iteration SD	0.354	-26.410	-1.248	-25.161
5-iteration SD	0.352	-27.138	-1.282	-25.857
3-iteration CG	0.351	-31.910	-1.290	-30.620
5-iteration CG	0.351	-34.030	-1.290	-32.740

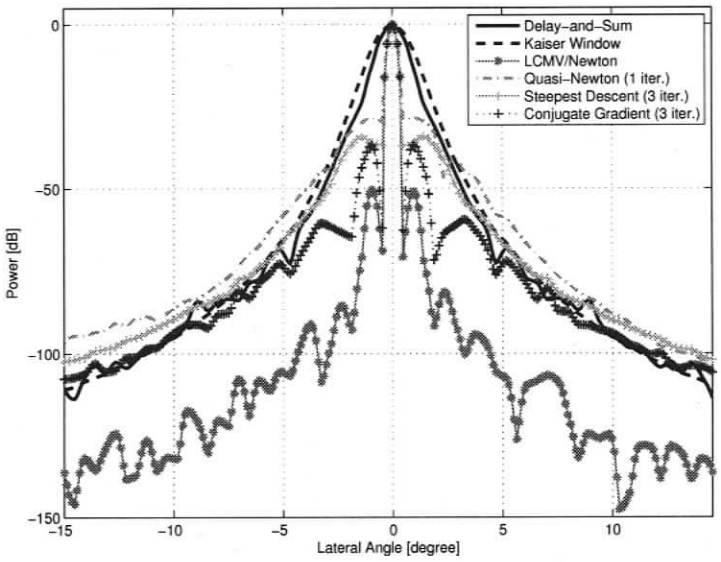


Figure 5.4: Gradient-driven GSC-based beamforming: Point spread function at transmit focus (12 single point phantom).

in fact, according to Table 5.3, its performance is worse than that of the simplest non-adaptive delay-and-sum beamformer. The 1-iteration SD method performs relatively well (contrast values of 1.494 for scattering region and 0.931 for water-filled cyst), considering that it offers approximately 40% computational savings (see Table 5.1). Using the 3-iteration SD or CG method yields approximately the same computational cost as the conventional LCMV method, but without producing optimal images (see Figure 5.8). Hence, the 1-iteration SD method is recommended. Note that, in the case of the point-scatterer-cyst phantom, the 5-iteration SD and CG methods are not practical at all from the computational point of view (see Table 5.1), despite some contrast improvement in comparison to the 1-iteration SD method (see Table 5.3 and Figure 5.9).

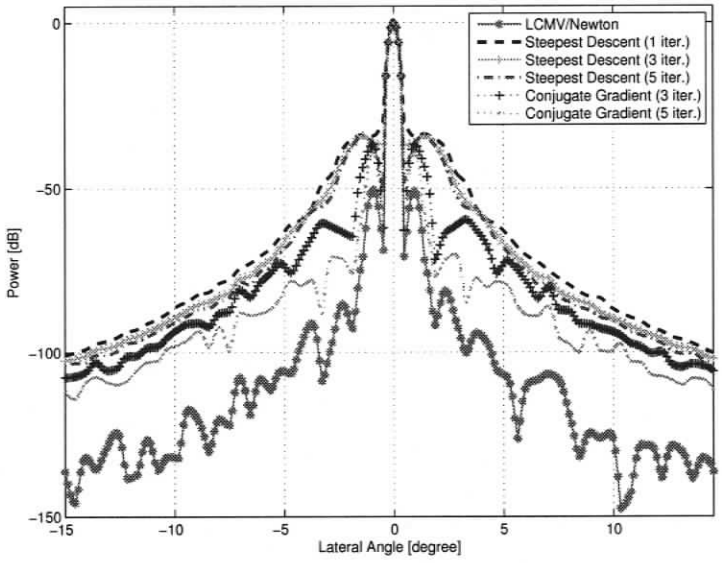


Figure 5.5: Gradient-driven GSC-based beamforming: Point spread function at transmit focus using SD and CG methods with different iteration numbers (12 single point phantom).

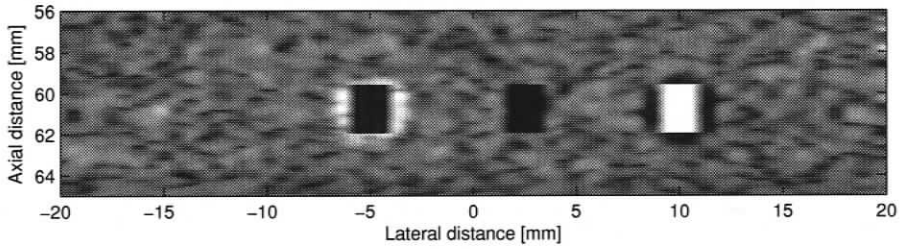


Figure 5.6: Sampled areas used for contrast computations: point-scatterer-cyst phantom.

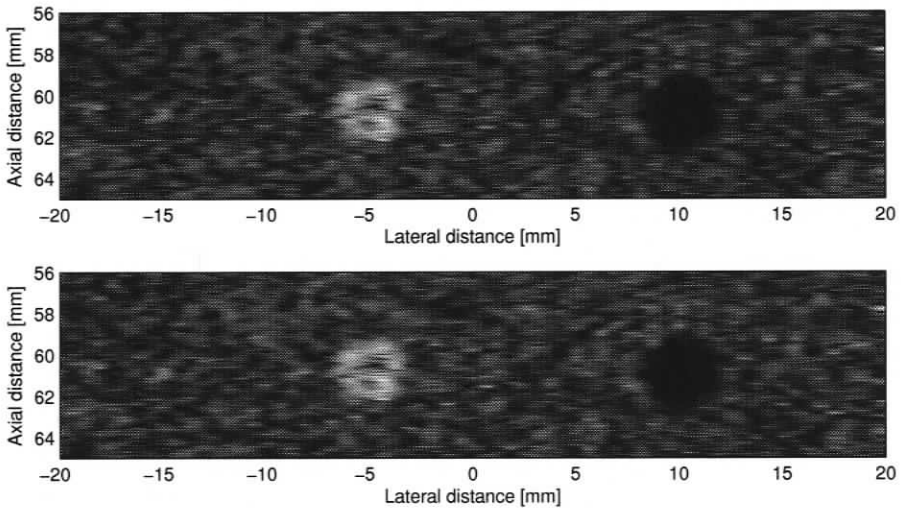


Figure 5.7: Performance of conventional LCMV (top) and Newton (bottom) methods: point-scatterer-cyst phantom.

Table 5.3: Gradient-driven GSC-based beamforming: Contrast with respect to speckled background (point-scatterer-cyst phantom).

Beamforming Method	Scattering Region	Water-Filled Cyst
Delay-and-Sum	1.290	0.931
Kaiser Window	1.210	0.888
LCMV/Newton	2.214	0.943
1-iteration QN	1.247	0.927
1-iteration SD/CG	1.494	0.931
3-iteration SD	1.696	0.936
5-iteration SD	1.701	0.934
3-iteration CG	1.820	0.931
5-iteration CG	2.021	0.939

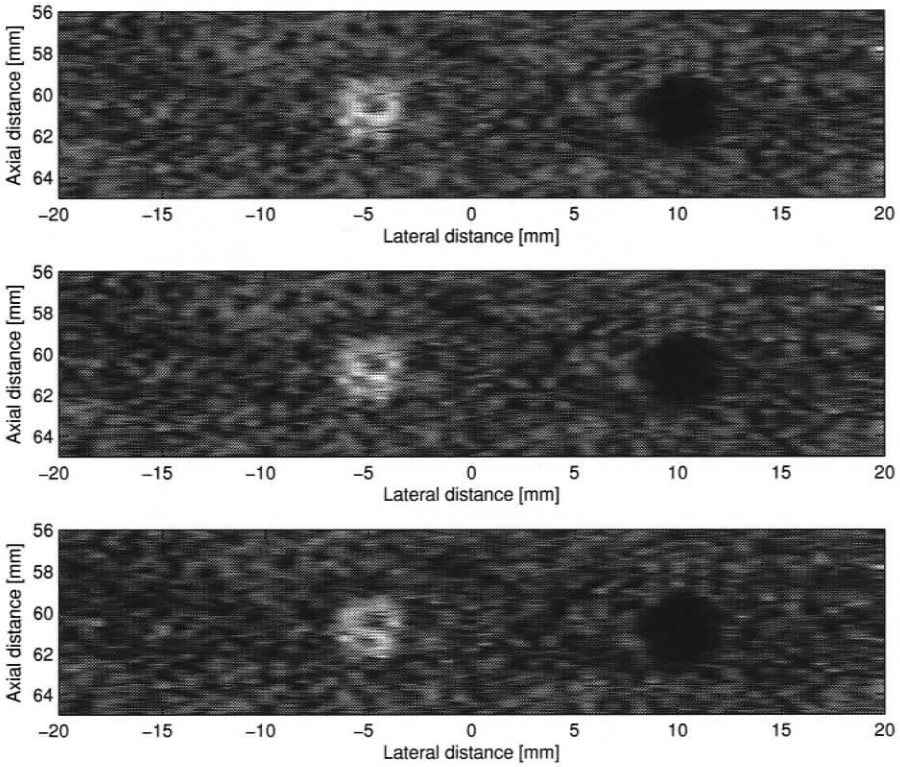


Figure 5.8: Performance of 1-iteration SD/CG (top), 3-iteration SD (center), and 3-iteration CG (bottom) methods: point-scatterer-cyst phantom (cf. Figures 5.7 and 5.9).

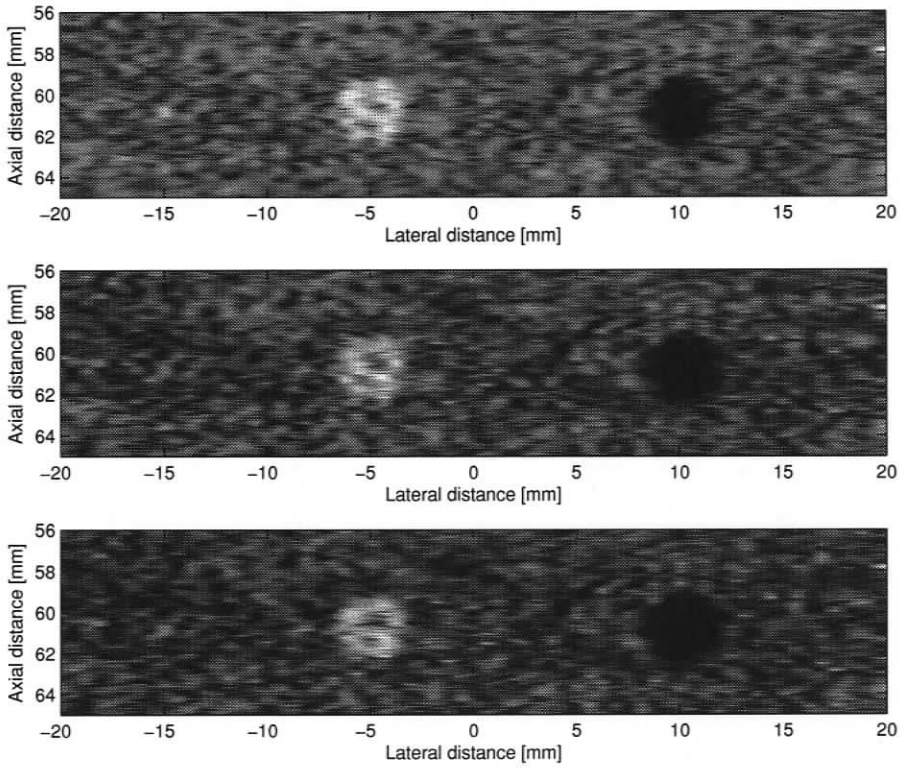


Figure 5.9: Performance of 1-iteration QN (top), 5-iteration SD (center), and 5-iteration CG (bottom) methods: point-scatterer-cyst phantom (cf. Figures 5.7 and 5.8).

5.2 Reduced-Rate LCMV Beamforming

Figures 5.10 and 5.12 show the simulation results for the single point phantom and the point-scatterer-cyst phantom, given $N_w = 2, 4, 8$, and 16. Tables 5.4 and Table 5.5 show the corresponding computational savings (ranging approximately from 40% to 90%) and energy savings (exceeding 75%), respectively. The former is computed using Table 4.1, while the latter is due to Equation (4.7) assuming ideal voltage scaling. We assume that the same relationship among the costs of addition, multiplication, and square root operations as in the previous section.

The point spread function at the transmit focus in the case of single point phantom is shown Figure 5.11. Table 5.6 shows the corresponding numerical values of interest. The 1/2-rate updating ($N_w = 2$) performs very well, with negligible degradation in FWHM and E_{SL}. Switching to $N_w = 8$ worsens FWHM and E_{SL} by approximately 3% and 35%, respectively, but the computational savings are doubled in comparison to the 1/2-rate updating. Note that the 1/4-rate updating does not yield significantly better images than those corresponding to the 1/8-rate updating, while the 1/16-rate updating yields clearly unacceptable images. Hence, $N_w = 8$ appears to be a good recommendation, striking a favorable tradeoff between the image quality and the computational and energy savings (over 80% and 90%, respectively) in comparison to the cases with $N_w = 2$, $N_w = 4$, and $N_w = 16$.

For the point-scatterer-cyst phantom (see Figure 5.12), Table 5.7 shows the relevant contrast values, identifying $N_w = 4$ as a recommended choice. Indeed, the 1/4-rate updating not only offers over 60% and 90% computational and energy savings, respectively, but also yields (surprisingly) the best contrast values (2.711 for scattering region and 0.968 for water-filled cyst).

Table 5.4: Reduced-rate LCMV beamforming: Computational savings in our test examples (see Figures 5.10 and 5.12).

Update Rate	Savings (%): $M = 96$, $L = 48$, $N = 1$	Savings (%): $M = 64$, $L = 32$, $N = 2$
2	[47.8, 49.5]	[37.9, 40.8]
4	[71.7, 74.3]	[66.1, 69.7]
8	[83.6, 86.7]	[80.3, 84.2]
16	[89.6, 92.9]	[87.3, 91.4]

Table 5.5: Reduced-rate LCMV beamforming: Energy savings in our test examples (see Figures 5.10 and 5.12).

Test Case	Savings (%)			
	$N_w = 2$	$N_w = 4$	$N_w = 8$	$N_w = 16$
12 single point phantom	[87.49, 87.50]	[98.43, 98.44]	[99.80, 99.80]	[99.97, 99.98]
Point-scatterer-cyst phantom	[78.60, 79.84]	[97.31, 97.48]	[99.64, 99.68]	[99.94, 99.96]

Table 5.6: Reduced-rate LCMV beamforming: FWHM, E_{SL}, E_{ML} at transmit focus (12 single point phantom).

Update Rate	FWHM (mm)	E_{SL} (dB)	E_{ML} (dB)	E_{SL}/E_{ML} (dB)
1	0.351	-36.919	-1.291	-35.628
1/2	0.352	-36.929	-1.284	-35.545
1/4	0.360	-26.958	-1.109	-25.894
1/8	0.361	-24.053	-1.120	-22.933
1/16	0.471	-24.560	-1.009	-25.604

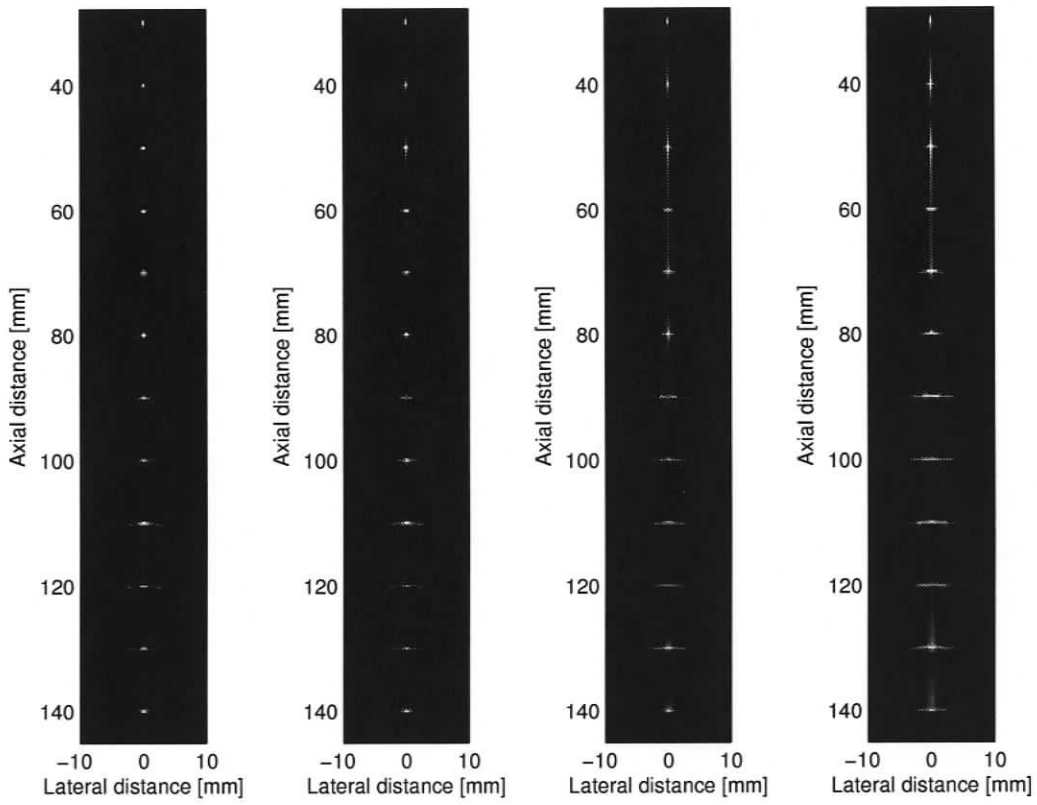


Figure 5.10: From left to right: Performance of LCMV beamformer with 1/2-rate, 1/4-rate, 1/8-rate, and 1/16-rate weight updating (12 single point phantom, cf. Figure 2.1).

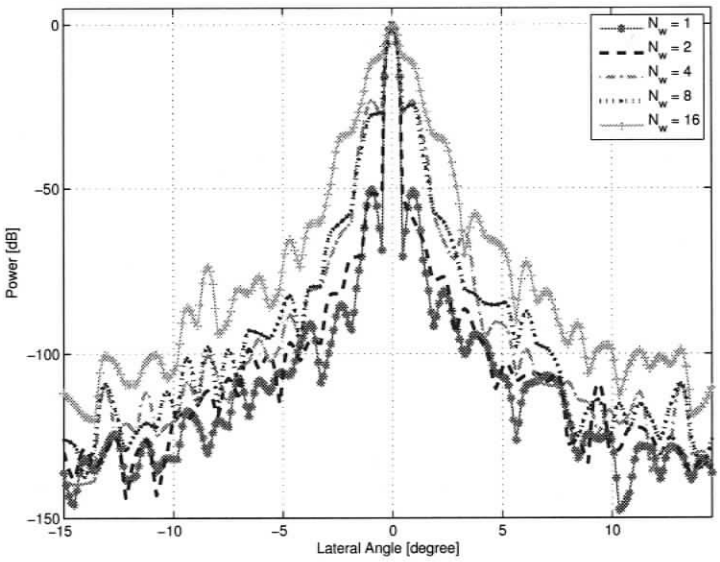


Figure 5.11: Reduced-rate LCMV beamforming: Point spread function at transmit focus (12 single point phantom).

Table 5.7: Reduced-rate LCMV beamforming: Contrast with respect to speckled background (point-scatterer-cyst phantom).

Update Rate	Scattering Region	Water-Filled Cyst
1	2.214	0.943
1/2	2.341	0.941
1/4	2.711	0.968
1/8	1.756	0.918
1/16	1.592	0.886

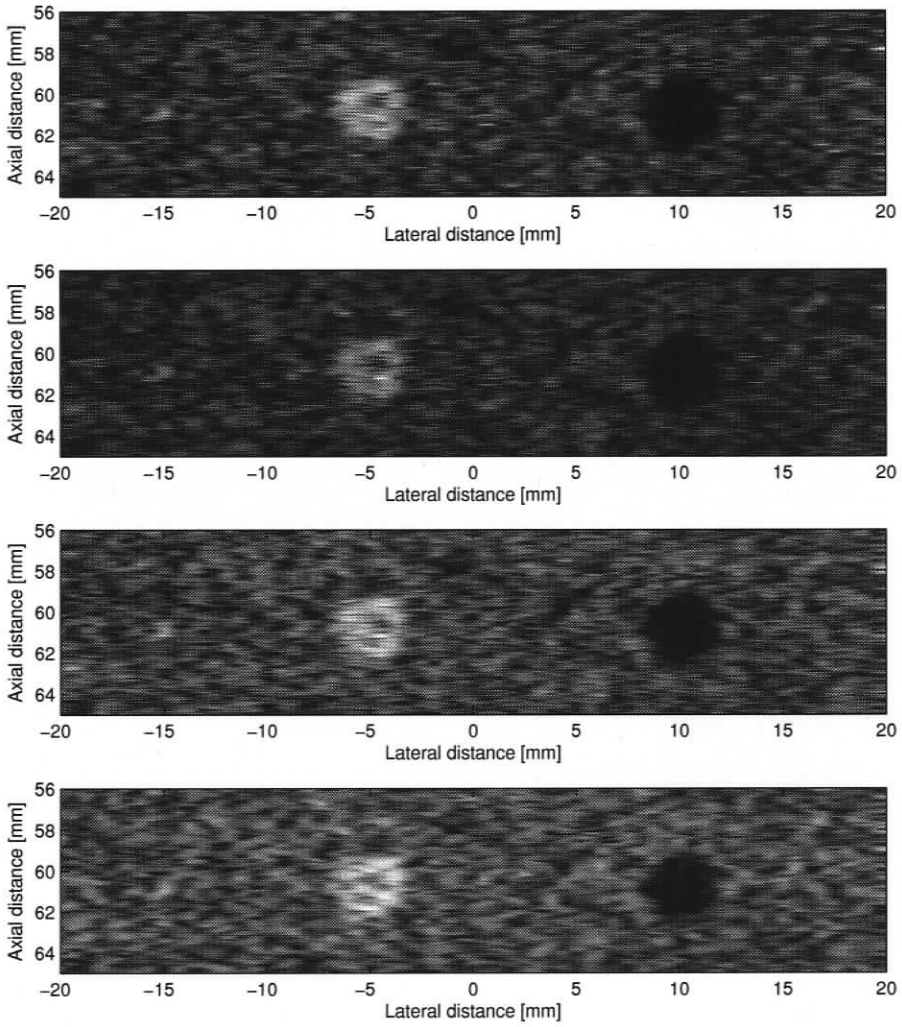


Figure 5.12: From top to bottom: Performance of LCMV beamformer with 1/2-rate, 1/4-rate, 1/8-rate, and 1/16-rate weight updating (point-scatterer-cyst phantom, cf. Figure 2.3).

Chapter 6

Conclusions and Future Work

We have described the application of the GSC structure to adaptive ultrasound beamforming as an alternative to the conventional LCMV method. We have considered adjusting the GSC's adaptive weight vector using four gradient-driven methods: Newton, quasi-Newton, steepest descent, and conjugate gradient. The investigated gradient-driven GSC-based beamformer allows for a flexible tradeoff between the image quality and the computational complexity (determined by the number of algorithm iterations). Furthermore, the GSC can easily be configured as the (cheapest) delay-and-sum beamformer by deactivating the GSC's adaptive part. Based on our evaluation results for the single points phantom, we recommend using the 3-iteration conjugate gradient method, whose computational cost is approximately 20% less than that of the conventional LCMV beamformer. In the case of the point-scatterer-cyst phantom, we recommend the 1-iteration steepest descent method offering approximately 40% in com putational savings. These savings, however, lead to (tolerable) degradation of the image quality.

As an alternative approach to lower computational costs, we have also proposed to use the LCMV beamformer whose weights are updated at a reduced rate. We have considered the update rates of 1/2, 1/4, 1/8, and 1/16 (the denominator denotes

the number of sample periods alloted for weight vector computations). Based on our evaluation results for the single points phantom, we recommend using the 1/8-rate updating that yields over 80% in computational savings (in comparison to the conventional full-rate LCMV beamformer) without excessively degrading the image quality. In the case of the point-scatterer-cyst phantom, we recommend the update rate of 1/4, which produces images of (unexpectedly) better contrast than the full-rate beamformer, while at the same time offering over 60% in computational savings. Also, the reduced-rate updating leads to underutilization of the weight computation hardware (when working at full speed), which allows us to reduce the hardware speed by reducing the supply voltage. Lower supply voltages result in significant energy savings. In our evaluation cases, the estimated energy savings were over 75%.

The evaluation results presented in this thesis indicate that the studied beamforming techniques can be valuable in ultrasound practice. This work, however, can certainly benefit from further extensions in many possible directions. Some of the suggested future research efforts are outlined next.

Reduced-Rate GSC-Based Beamforming: The idea of the reduced-rate weight updating can be applied to the gradient-driven GSC-based beamformer as well. As more time becomes available for weight computations, we can reduce the supply voltage accordingly, which may lead to significant energy savings.

Gradient-Driven LCMV Beamforming: Recall that the conventional LCMV beamformer needs to compute a product in the form $\mathbf{A}^{-1}b$, where $\mathbf{A}$ corresponds to the sample correlation matrix, and $\mathbf{b}$ corresponds the steering vector. In other words, we need to solve the linear system $\mathbf{Ac} = \mathbf{b}$ for unknown $\mathbf{c}$. Many iterative methods have been developed for solving such linear systems. One of them is the conjugate gradient method, which is immediately applicable here because $\mathbf{A}$ is positive-definite and symmetric.

Dynamic-Rate Weight Updating: Rather than committing to the fixed-rate

weight updating (whether it is full-rate or specific reduced-rate), it may be feasible for the beamforming hardware to decide on the update rate dynamically, based on the received data. For example, there may be relatively long delays between returning echos from the target – during those time intervals, high-performance beamforming is not important, and low-rate updating can be used. On the other hand, when the target echos arrive, the sampled data received by the beamformer is critical, and the high-rate weight updating becomes desirable. There are technical challenges, however. Dynamic voltage/clock scaling introduces a certain overhead (in terms of latency and energy) associated with switching from one voltage/clock setting to another. More importantly, we would need a proper estimator that makes a decision when and to which voltage/clock setting the hardware should switch. Such an esti mator would have only the past history information, while having to make a prediction about the importance of the data to be received next. Mispredictions are possible, and if a lower than needed voltage/clock setting is incorrectly decided upon, the image quality may suffer.

Beamforming Hardware Design and Implementation: Our evaluation results have been obtained from Matlab-based simulations, where the simulated sampling rate is 100 MHz, and computations are performed using 64-bit floating-point numbers. In practical hardware realizations, however, we are likely to use a lower sampling rate and a shorter fixed-point number representation. These two issues (i.e., the sampled data's phase resolution and amplitude quantization) may have a notable impact on the beamformer performance. Also, the actual implementation cost of various beamforming hardware designs (subject to area, latency, power, and other possible constraints) may expose practical advantages of some beamforming methods over the others (e.g., easier pipelining, greater parallelism, easier time-multiplexing, greater regularity, etc).

System-Level Design and Evaluation (with Real Medical Images): The

receive beamformer is only one component within a complex ultrasound system. Other blocks (e.g., transmit beamforming, envelope detection, log-compression, pre-processing, etc.) can also impact the quality of ultrasound images. They may also offer additional opportunities for reducing the receive beamforming cost. For example, the beamformer's output data is substantially modified by the envelope detection and log-compression blocks – it may be possible to relax beamforming performance requirements (thus reducing the beamformer's computational cost), and compensate for the losses later (if needed) during envelope detection and log-compression. Ideally, comprehensive system evaluations would be performed to draw definite engineering conclusions, using real medical images and real hardware implementations, as opposed to using synthetic phantoms and computer simulations.

Bibliography

[1] M. Ali and P. Schreib, "Adaptive single snapshot beamforming: A new concept for the rejection of nonstationary and coherent interferers," *IEEE Trans. Signal Processing*, vol. 40, no. 12, pp. 3055–3058, Dec. 1992.

[2] P. Anderson and M. Ingram, "The performance of the least mean squares algorithm combined with spatial smoothing," *IEEE Trans. Signal Processing*, vol. 45, no. 4, pp. 1005–1012, Apr. 1997.

[3] A. Antoniou and W. Lu, *Practical Optimization.* NY: Springer, 2007.

[4] R. Bethel, B. Shapo, and H. Van Trees, "Single snapshot spatial processing: Optimized and constrained," in *Proc. IEEE Sensor Array and Multichannel Signal Processing Workshop*, July 2002, pp. 508–512.

[5] T. Burd and R. Brodersen, *Energy Efficient Microprocessor Design.* Kluwer Academic Publishers, 2002.

[6] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proc. IEEE*, vol. 57, no. 8, pp. 1408–1418, Aug 1969.

[7] R. Cobbold, *Foundations of Biomedical Ultrasound.* NY: Oxford, 2007.

[8] R. Compton, *Adaptive Antennas.* NJ: Prentice-Hall, 1988.

[9] O. Frost, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, no. 8, pp. 926–935, Aug 1972.

[10] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed. MD: Johns Hopkins, 1996.

[11] L. Griffiths and C. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas and Propagation*, vol. 30, no. 1, pp. 27–34, Jan. 1982.

[12] D. Guenther and W. Walker, "Optimal apodization design for medical ultrasound using constrained least squares. part i: Theory," *IEEE Trans. Ultrasonics, Ferroelectrics and Frequency Control*, vol. 54, no. 2, pp. 332–342, feb 2007.

[13] D. Guenther and W. Walker, "Optimal apodization design for medical ultrasound using constrained least squares. part ii: Simulation results," *IEEE Trans. Ultrasonics, Ferroelectrics and Frequency Control*, vol. 54, no. 2, pp. 343–358, feb 2007.

[14] D. Guenther and W. Walker, "Optimal contrast resolution beamforming," in *Proc. IEEE Ultrasonics Symp.*, Oct. 2007, pp. 37–41.

[15] F. Harris, "On the use of windows for harmonic analysis with the discrete fourier transform," in *Proc. IEEE, Vol. 66*, Jan. 1978, pp. 51–83.

[16] S. Haykin, *Adaptive Filter Theory*, 4th ed. NJ: Prentice-Hall, 2001.

[17] I. Holfort, F. Gran, and J. Jensen, "Minimum variance beamforming for high frame-rate ultrasound imaging," in *Proc. IEEE Ultrasonics Symp.*, Oct. 2007, pp. 1541–1544.

[18] I. Holfort, F. Gran, and J. Jensen, "Plane wave medical ultrasound imaging using adaptive beamforming," in *Proc. IEEE Sensor Array and Multichannel Signal Processing Workshop*, July 2008, pp. 288–292.

[19] I. Holfort, F. Gran, and J. Jensen, "Broadband minimum variance beamforming for ultrasound imaging," *IEEE Trans. Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 56, no. 2, pp. 314–325, Feb. 2009.

[20] J. Hudson, *Adaptive Array Principles.* UK: Peregrinus, 1981.

[21] J. Jensen, "Field II Simulation Program," 2008, http://server.elektro.dtu.dk/personal/jaj/field.

[22] J. Jensen, "Field: A program for simulating ultrasound systems," *10th Nordic-Baltic Conference on Biomedical Imaging Published in Medical and Biological Engineering and Computing*, vol. 34, pp. 351–353, 1996.

[23] J. Jensen and N. Svendsen, "Calculation of pressure fields from arbitrarily shaped, apodized, and excited ultrasound transducers," *IEEE Trans. Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 32, no. 2, pp. 262–267, Mar 1992.

[24] D. Johnson and D. Dudgeon, *Array Signal Processing.* NJ: Prentice-Hall, 1993.

[25] S. Khezerloo and D. Rakhmatov, "Gradient-driven beamforming for biomedical ultrasound," in *Proc. International Conf. IEEE Engineering in Medicine and Biology Society, to appear*, Sept. 2009, pp. 1–7.

[26] T. Laakso, V. Valimaki, M. Karjalainen, and U. Laine, "Splitting the unit delay [fir/all pass filters design]," *IEEE Signal Processing Magazine*, vol. 13, no. 1, pp. 30–60, Jan 1996.

[27] C. Loizou, C. Pattichis, C. Christodoulou, R. Istepanian, M. Pantziaris, and A. Nicolaides, "Comparative evaluation of despeckle filtering in ultrasound imaging of the carotid artery," *IEEE Trans. Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 52, no. 10, pp. 1653–1669, oct 2005.

[28] J. Mann and W. Walker, "A constrained adaptive beamformer for medical ultrasound: Initial results," in *Proc. IEEE Ultrasonics Symp.*, Oct. 2002, pp. 1807–1810.

[29] R. Monzingo and T. Miller, *Introduction to Adaptive Arrays.* NY: Wiley, 1980.

[30] E. Nicolau and D. Zaharia, *Adaptive Arrays.* NY: Elsevier, 1989.

[31] K. Ranganathan, M. Santy, T. Blalock, J. Hossack, and W. Walker, "Direct sampled I/Q beamforming for compact and very low-cost ultrasound imaging," *IEEE Trans. Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 51, no. 9, pp. 1082–1094, sep 2004.

[32] K. Ranganathan and W. Walker, "A novel beamformer design method for medical ultrasound. part i: Theory," *IEEE Trans. Ultrasonics, Ferroelectrics and Frequency Control*, vol. 50, no. 1, pp. 15–24, jan 2003.

[33] K. Ranganathan and W. Walker, "A novel beamformer design method for medical ultrasound. part ii: Simulation results," *IEEE Trans. Ultrasonics, Ferroelectrics and Frequency Control*, vol. 50, no. 1, pp. 25–39, jan 2003.

[34] M. Sasso and C. Cohen-Bacrie, "Medical ultrasound imaging using the fully adaptive beamformer," in *Proc. IEEE International Conf. Acoustics, Speech, and Signal Processing, Vol. II*, Mar. 2005, pp. 489–492.

[35] C. Seo and J. Yen, "Sidelobe suppression in ultrasound imaging using dual apodization with cross-correlation," *IEEE Trans. Ultrasonics, Ferroelectrics and Frequency Control*, vol. 55, no. 10, pp. 2198–2210, oct 2008.

[36] T. Shah and T. Kailath, "Adaptive beamforming for coherent signals and interference," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 33, no. 3, pp. 527–536, June 1985.

[37] J. Synnevag, A. Austeng, and S. Holm, "Adaptive beamforming applied to medical ultrasound imaging," *IEEE Trans. Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 54, no. 8, pp. 1606–1613, Aug. 2007.

[38] T. Szabo, *Diagnostic Ultrasound Imaging: Inside Out.* MA: Elsevier, 2004.

[39] Texas Instruments, "Ultrasound System," 2009, http://focus.ti.com/docs/solution/folders/print/346.html.

[40] G. Trahey, J. Allison, S. Smith, and O. von Ramm, "Speckle pattern changes with varying acoustic frequency: Experimental measurement and implications for frequency compounding," in *Proc. IEEE Ultrasonics Symp.*, 1986, pp. 815–818.

[41] G. Trahey, S. Smith, and O. von Ramm, "Speckle pattern correlation with lateral aperture translation: Experimental results and implications for spatial compounding," *IEEE Trans. Ultrasonics, Ferroelectrics and Frequency Control*, vol. 33, no. 3, pp. 257–264, May 1986.

[42] H. Van Trees, *Optimum Array Processing.* NY: Wiley, 2002.

[43] B. V. Veen and K. Buckley, "Beamforming: a versatile approach to spatial filtering," *ASSP Magazine, IEEE*, vol. 5, no. 2, pp. 4–24, apr 1988.

[44] F. Vignon and M. Burcher, "Capon beamforming in medical ultrasound imaging with focused beams," *IEEE Trans. Ultrasonics, Ferroelectrics and Frequency Control*, vol. 55, no. 3, pp. 619–628, mar 2008.

[45] F. Viola, M. Ellis, and W. Walker, "Time-domain optimized near-field estimator for ultrasound imaging: Initial development and results," *IEEE Trans. Medical Imaging*, vol. 27, no. 1, pp. 99–110, jan 2008.

[46] F. Viola and W. Walker, "Adaptive signal processing in medical ultrasound beamforming," in *Proc. IEEE Ultrasonics Symp.*, Oct. 2002, pp. 1807–1810.

[47] Z. Wang, J. Li, and R. Wu, "Time-delay- and time-reversal-based robust Capon beamformers for ultrasound imaging," *IEEE Trans. Medical Imaging*, vol. 24, no. 10, pp. 1308–1322, Oct. 2005.

[48] B. Widrow, K. Duvall, R. Gooch, and W. Newman, "Signal cancellation phenomena in adaptive antennas: Causes and cures," *IEEE Trans. Antennas Propagation*, vol. 30, no. 3, pp. 469–478, Mar. 1982.

Appendix A

FIELD II Simulation Tool

FIELD II [21–23] is a software program for simulating ultrasound transducer fields and ultrasound imaging using linear acoustics. Both emitted and pulse-echo fields can be calculated by this program for a variety of different transducers. Also any kind of linear imaging can be simulated, including realistic images of human tissue. The program runs in the Matlab environment and is available in several versions for different operating systems. It is currently free of charge and can be downloaded from the program's website at the Technical University of Denmark [21]. The program consists of a pre-compiled C code and a number of Matlab m-functions. All calculations are performed by the C code, which is called by the user through the provided Matlab functions.

After initialization, the scatterers' positions and amplitudes are generated and stored in a file first. This file can be then used to compute the image lines. Each image line is saved in a separate file, which makes it possible to restart the simulation from the last simulated image line (if something goes wrong) without having to perform the whole simulation again. The delay focusing and apodization during both transmission and reception can be controlled dynamically. Thus, many different kinds of ultrasound imaging systems can be simulated and studied.

The examples included in the FIELD II tool distribution [21] provide a good starting point for generating desired phantoms. For our 12-point phantom the positions of the points are set first, and then the amplitude of each scatterer is fixed to 1. For the point-scatterer-cyst phantom, a collection of randomly placed point scatterers are required to generate the speckled background. The scattering strength is determined with a Gaussian distribution. The scatterers in the phantom are positioned randomly within a $50 \times 10 \times 20$ mm^3 volume, starting at 50 mm from the transducer surface. If a scatterer resides within a cyst region, the amplitude is set to zero. Within the highly scattering region the amplitude is multiplied by 10. The point targets have a fixed amplitude of 20.