In Support of High Quality 3-D Ultrasound Imaging for Hand-held Devices

by

Ming Yang

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

Approved December 2014 by the
Graduate Supervisory Committee:

Chaitali Chakrabarti, Chair
Antonia Papandreou-Suppappola
Lina Karam
David Frakes
Umit Ogras

ARIZONA STATE UNIVERSITY

May 2015

ABSTRACT

Three dimensional (3-D) ultrasound is safe, inexpensive, and has been shown to drastically improve system ease-of-use, diagnostic efficiency, and patient throughput. However, its high computational complexity and resulting high power consumption has precluded its use in hand-held applications.

In this dissertation, algorithm-architecture co-design techniques that aim to make hand-held 3-D ultrasound a reality are presented. First, image enhancement methods to improve signal-to-noise ratio (SNR) are proposed. These include virtual source firing techniques and a low overhead digital front-end architecture using orthogonal chirps and orthogonal Golay codes.

Second, algorithm-architecture co-design techniques to reduce the power consumption of 3-D SAU imaging systems is presented. These include (i) a subaperture multiplexing strategy and the corresponding apodization method to alleviate the signal bandwidth bottleneck, and (ii) a highly efficient iterative delay calculation method to eliminate complex operations such as multiplications, divisions and square-root in delay calculation during beamforming. These techniques were used to define Sonic Millip3De, a 3-D die stacked architecture for digital beamforming in SAU systems. Sonic Millip3De produces 3-D high resolution images at 2 frames per second with system power consumption of 15W in 45nm technology.

Third, a new beamforming method based on separable delay decomposition is proposed to reduce the computational complexity of the beamforming unit in an SAU system. The method is based on minimizing the root-mean-square error (RMSE) due to delay decomposition. It reduces the beamforming complexity of a SAU system by 19× while providing high image fidelity that is comparable to non-separable beamforming. The resulting modified Sonic Millip3De architecture supports a frame

rate of 32 volumes per second while maintaining power consumption of 15W in 45nm technology.

Next a 3-D plane-wave imaging system that utilizes both separable beamforming and coherent compounding is presented. The resulting system has computational complexity comparable to that of a non-separable non-compounding baseline system while significantly improving contrast-to-noise ratio and SNR. The modified Sonic Millip3De architecture is now capable of generating high resolution images at 1000 volumes per second with 9-fire-angle compounding.

DEDICATION

To my parents

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Motivation

Three-dimensional (3-D) ultrasound is a particularly attractive modality for hand-held imaging of the future. Ultrasound transducers use little power (limited by FDA regulations to a few hundred milliwatts [1]) and pose no known dangers or side-effects, in contrast to X-ray and MRI [2,3]. While most current systems are still 2-D, it is now established that 3-D ultrasound provides numerous benefits over its 2-D counterpart. 3-D imaging not only simplifies tasks such as volumetric measurements of cysts and tumors, but also provides capabilities that are unavailable with 2-D, such as accurate measurement of volumetric flow or 3-D shear wave tissue sonoelastography. In addition, the portability of these imaging devices is not simply a matter of convenience; clinical studies have demonstrated that patient outcomes improve [4, 5], especially for patients in critical condition. Moreover, improved portability holds the potential to bring advanced medical imaging to traditionally underserved populations in the rural and developing world.

Industry has already recognized these advantages and marketed several portable ultrasound devices marketed today. Examples include LOGIQ Book XP by GE [6], SonoSite 180 by SonoSite [7], Primedic by HandyScan [7], and Sonic Window by Analogic [8]. However, the current devices are targeted for emergency diagnosis and produce low-resolution two-dimensional views. Furthermore, the features (e.g., color and pulse Doppler) and image quality of portable systems vary, in part due to hardware limitations and part due to market considerations. For example, some portable

1

systems limit the number of processing channels and array elements used to produce images to reduce power. In this thesis we describe our work on development of a hand-held ultrasound imaging device that produces high resolution 3-D images similar to the ones produced by large tethered systems.

1.2 Challenges of 3-D Ultrasound Imaging

3-D ultrasound comes with many challenges that are compounded when implementing a system in a hand-held form factor. To construct a 3-D volumetric image, a conventional linear transducer array (e.g., 120 elements) for 2-D imaging must be replaced with a rectangular array (120×88), increasing the incoming data rate by $88\times$. Furthermore, rather than reconstruct a typical 2D image resolution of 50×4096 focal points, the 3D image comprises $50 \times 50 \times 4096$ focal points, another factor of 50 increase. The computational requirements increase by the product of these factors (at least $4000\times$). At the same time, the data rate (as high as 5 Tb/s) of the received echo signals is so high that the data cannot easily be transferred off-chip for image formation; current 3-D systems typically transfer data for only a fraction of receive channels, sacrificing image quality or aperture size. In addition to the extreme computational requirements, power is of the utmost importance, not only to ensure adequate battery life, but more importantly because the device is in direct contact with the patient's skin, placing tight constraints on safe operating temperature. As a result, the challenge of 3-D hand-held ultrasound lies in performing these computations within a typical 5-W budget for hand-held devices.

Implementing a hand-held 3-D system with commercially available digital signal processor (DSP) or graphics accelerator chips using conventional beamforming algorithms designed for software is simply infeasible. Our analysis indicates that it would take 700 ultrasound DSP chips with a total power budget of 7.1 kW to meet typical

3-D imaging computational demands at just 1 frame per second (fps). To enable such demanding computation on such a low power budget, a complete rethink of both the algorithm and architecture is required.

1.3   Problems Addressed

1. Image Enhancement Techniques for SAU Systems

Synthetic aperture ultrasound (SAU) imaging is a promising technique for emerging 3-D applications, because it can achieve a higher frame rate and higher resolution than traditional phased array imaging. However, classic SAU systems suffer from low signal-to-noise ratio (SNR), because they are based on single-element firing schemes. Virtual source firing technique improves SNR by firing multiple transducers to emulate a virtual source in each transmit, but the SNR gain is limited because of signal tail artifacts. Orthogonal coded excitation can further improve the SNR for SAU systems but with large increase in computational complexity. To reduce the high complexity of coded excitation based systems, two approaches are presented. First, by combining the decoding and beamforming process, the overall computations are reduced by $31\times$ for chirp based systems and by $5\times$ for Golay coded based systems. Second, by optimizing the code length and the number of transmit elements (which is the same as the number of orthogonal codes), the total number of computations of the system is reduced by $20\times$ for the same SNR gain. Unfortunately, the use of orthogonal coded excitation exacerbates the motion artifacts on SAU imaging systems. To address this issue, we propose a cost-effective motion compensation method that dynamically compensates for the body motion by adjusting the beamforming delay. Field II simulation results show that the method is able to significantly improve the SNR and reduce sidelobe levels in presence of motion. The studies on orthogonal coded excitation and motion compensation were presented in our papers [9, 10].

3

2. Algorithm and Architecture Innovation for a Low Power 3-D SAU System

Building a 3-D ultrasound imaging system in hardware within the tight power budget of a handheld device is significantly challenging. Use of a 2-D transducer array and the large number of scanlines required for a 3-D volume make the computational complexity of a 3-D ultrasound imaging system at least two to three orders of magnitude higher than a 2-D system. So algorithm and architecture innovations are clearly required for a handheld 3-D SAU system.

First, we present a subaperture firing and receive scheme for a 3-D SAU system, which reduces the number of firings per frame by 2×. To optimize a subaperture based 3-D SAU system, we propose an apodization design algorithm that maximizes the SNR and reduces the sidelobe level. For delay calculation, instead of storing the delays in large look-up tables, we iteratively calculate the delays of focal points along a scanline. The iterative delay calculation only requires three additions per focal point, thereby reducing the complexity of the delay calculation unit significantly.

At the architecture level, we propose the Sonic Millip3De hardware accelerator in collaboration with researchers at the University of Michigan. The Sonic Millip3De employs state-of-art 3-D die stacking structure, allocating transducer and analog components, SRAM storage and 1,024-beamforming processing units into three silicon layers for a compact design with short wires. The accelerator array is organized according to a streaming design paradigm that supports massive parallel processing. Based on RTL-level design and floorplanning for an industrial 45nm process, it is shown that Sonic Millip3De can support a 2Hz frame rate for a 3-D ultrasound system with a 15W full-system power budget and will meet a 5W safe power target by the 16nm node. This work appeared in [11–13].

4

3. Separable Beamforming For 3-D SAU Systems

Although, Sonic Millip3De has very low power consumption, it is able to achieve a frame rate of only 2Hz. In order to improve the frame rate without affecting the power consumption, we focus on reducing the complexity of beamforming which accounts for 99% of the computational complexity of the digital front end. We propose to use separable beamforming, wherein 2-D array beamforming is decomposed into a series of 1-D beamforming problems. The separable beamforming method is based on decomposing the delay in a way that minimizes the root-mean-square (RMS) phase error introduced the decomposition. We show how this method can be used to achieve separable beamforming in subaperture based 3-D synthetic aperture ultrasound (SAU) systems with $19\times$ reduction in computation complexity compared to the non-separable method. The separable delays obtained by this method could be stored in large look-up tables and directly used for delay calculations. However, to reduce look-up table storage requirements, we propose a method to iteratively compute these delays at runtime using far fewer pre-computed constants. We demonstrate, using Field II simulations of cyst images, that our proposed separable beamforming method achieves almost identical image quality compared to the non-separable baseline method.

Finally, we implement the separable beamforming algorithm into a modified version of Sonic Millip3De. The $19\times$ reduction in computation can be leveraged either to reduce hardware requirements (by scaling down the number of parallel pipelines in the design), increase frame rate (by operating the existing design at higher throughput), or save power (by scaling down frequency to maintain current frame rates). The synthesis results based on an implementation using a 45nm industrial library show that the proposed method boosts the frame rate by $16\times$ (from 2Hz to 32Hz) while

keeping power consumption at 15W. The work on separable beamforming for SAU system appeared in [14, 15].

4. Separable Beamforming and Coherent Compounding For 3-D Plane-wave Imaging

3D plane-wave imaging systems can support the high volume acquisition rates that are essential for 3D vector flow imaging and sonoelastography but suffer from low resolution and low SNR. Coherent compounding is a technique to improve the image quality of plane-wave systems at the expense of significant increase in beamforming computational complexity.

We propose a new separable beamforming method for 3-D plane-wave imaging with coherent compounding. The separable beamforming method is based on a delay decomposition that minimizes the RMS phase error. Such a system has computational complexity comparable to that of a non-separable non-compounding baseline system and yet has superior imaging quality performance compared to the baseline system. The new method with 9-fire-angle compounding helps improve average CNR from 1.6 to 2.2 and achieve a SNR increase of 9.0 dB compared to the baseline system. Overall, the system is capable of generating high resolution images at 1000 volumes per second. The work on separable beamforming for plane-wave systems appeared in [16, 17].

1.4 Thesis Organization

This report is organized as follows.

In Chapter 2, the ultrasound system enhancement techniques including coded excitation, motion compensation and virtual source are presented. The performance of these techniques are verified for a 2-D system using Field II simulations.

In Chapter 3, algorithm techniques to facilitate the design of low power beam-former accelerator are presented. The algorithm level innovations including subaperture firing, subaperture apodization optimization and a new low-cost iterative delay calculation. These innovations helped define a 3-D die-stacked beamforming accelerator, Sonic Millip3De, which had power consumption of 15W in 45nm technology

In Chapter 4, a new separable beamforming based on minimum RMSE delay decomposition is presented. An analysis on factors affecting approximation error is presented. Hardware architecture modifications on Sonic Millip3De architecture for the separable beamforming method are presented along with image quality evaluation using Field II.

In Chapter 5, separable beamforming is used to reduce the complexity of plane-wave systems with coherent compounding. The modifications to our beamforming accelerator, Sonic Millip3De, are presented. The proposed system achieves volume acquisition rates of over 1000 volumes per second for compounding with 9 firing angles.

Chapter 6 concludes the report along with description of future work.

Chapter 2

ULTRASOUND IMAGING SYSTEM ENHANCEMENT TECHNIQUES

2.1 Ultrasound Imaging System Overview
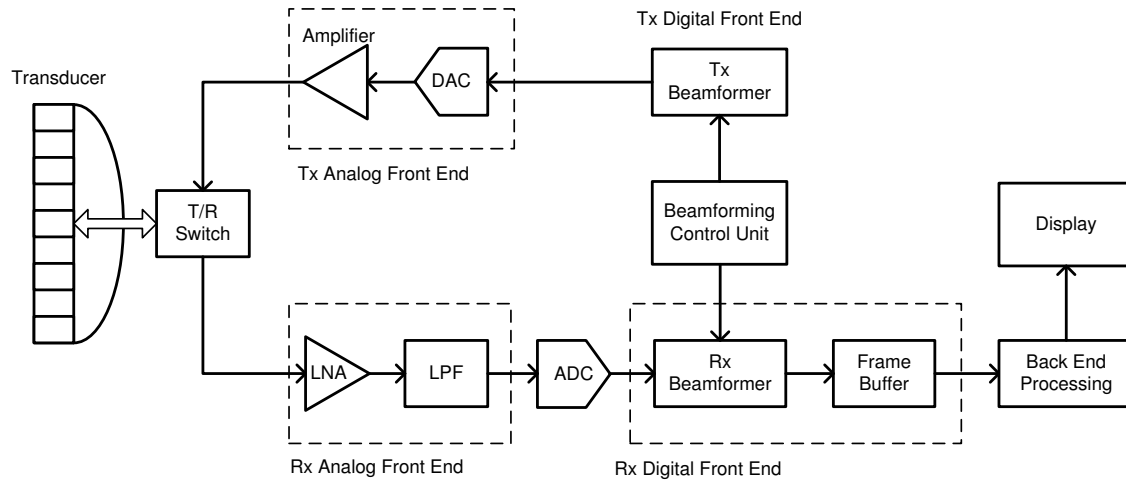
Figure 2.1: Ultrasound Imaging System Block Diagram

The block diagram of a typical ultrasound imaging system is shown in Figure 2.1. In the front is the ultrasound transducer array, which consists of a number of transducer elements. Each transducer element is made of piezoelectric material and converts electrical energy into sound energy or vice versa. The imaging system can be divided into the transmit part (on the top of Figure 2.1) and receive part (on the bottom of Figure 2.1). The transmit front end generates excitation waveforms to drive transducers. It also controls the transmit delay for each transducer so that the ultrasound wave can be focused at a certain depth. As the ultrasound wave propagates through soft tissue, echoes that are generated propagates back to the transducer array. To collect echo signals, the system is switched to receive mode by the transmit/receive (T/R) switch. The transducer array converts ultrasound echoes

into electronic signals, which are sent to an analog front end. The analog front end consists of amplifiers that compensate for tissue attenuation, analog filters and A/D converters that sample and quantize the signals. The beamformer then adjusts delays of each channel so that the echo signals in a certain direction are collected and combined. This process repeats for all scanlines, and an image is constructed. The image is processed further in back end processing module and sent to the display unit.

In this work, we focus on two different types of imaging systems: synthetic aperture ultrasound (SAU) imaging systems and plane-wave imaging systems. In a classic SAU system, in each transmit/receive, only one transducer fires, and all transducers receive. Then a low resolution image is obtained after beamforming. A high resolution image is constructed by combining multiple low resolution images obtained in multiple firings. SAU systems can produce multiple scanlines in each firing and hence support higher frame rates. In this chapter, several enhancement techniques for SAU systems are discussed. This is followed by description of beamforming accelerator for SAU systems in Chapter 3 and techniques to reduce the complexity of beamforming in Chapter 4.

The other imaging system discussed in this work is a plane-wave system. In a plane-wave system, all transducers fire at the same time to emulate an unfocused plane wave that insonifies the imaging region. All scanlines are parallel to each other and perpendicular to the plane wave. Compared to SAU systems, plane-wave systems offer very high frame rates at the cost of smaller field of view, and lower resolution. Such high frame rates are essential for applications such as flow imaging, sonoelastography, etc. 3-D plane-wave imaging technique is described in Chapter 5.

In this chapter, several imaging enhancement techniques for SAU based systems are presented. First, a virtual source firing technique that helps improve SNR is studied. Second, we present an orthogonal coded excitation technique that supports

higher SNR gain compared to virtual source at the cost of higher computational complexity. Third, we propose a simple motion compensation method that improves the performance of coded excitation based systems in presence of body motion.

We presented orthogonal coded excitation and motion compensation in our papers [9, 10].

2.2 Virtual Source Firing Technique

An SAU system can support high frame rates compared to phased array imaging, since its frame rate does not depend on the number of scanlines. It also has better resolution since its beamforming is equivalent to performing dynamic focusing on both transmit and receive ends. Unfortunately, traditional SAU systems suffer from low signal-to-noise (SNR) and low contrast. Virtual source technique is a simple method that helps improve SNR [18, 19].
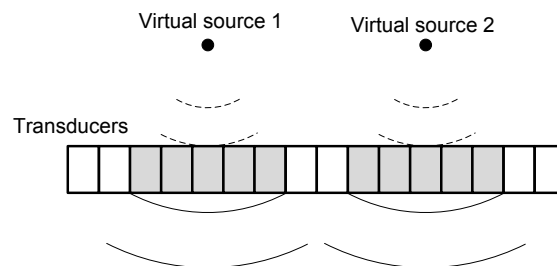
Figure 2.2: Principle of Virtual Source

In a classic SAU system, only one transducer element fires at a time, which limits the total transmit power, and results in low SNR. Virtual source technique employs multiple transducers in each firing to emulate a spherical wavefront as if it is generated by a virtual source. This process is illustrated in Figure 2.2, where each of the two virtual sources is emulated by five transducers in a 1-D transducer array. Each transducer is delayed by a certain amount of time according to the relative distance between the virtual source and the transducer. Assuming a virtual source is located

in $(x_v, 0, z_v)$ and a transducer is located in $(x, 0, z)$ and $z > z_v$, the transmit delay value is given by

$$\tau_{tx} = \frac{(x - x_v)^2}{2c(z - z_v)}$$ (2.1)

where c is the speed of sound [18]. The virtual source firing creates an approximate spherical wave. As the wave propagates, it insonifies the imaging area. The receive and beamforming process in such a system is the same as that in a classic SAU system.

Assuming the virtual source is emulated by K_t transducer elements, then the signal-to-noise ratio (SNR) gain is about $10 \log(K_t)$ dB compared to single element firing scheme [18]. The SNR in this work is defined as the ratio between signal power and thermal and electronic noise power, and is represented by Eq. (2.2).

$$\mathrm{SNR} = \frac{P_{\mathrm{signal}}}{P_{\mathrm{noise}}}$$ (2.2)

The SNR gain increases with the number of transducers K_t. However, there is a limitation: using a large number of transducers for one virtual source leads to imaging artifacts. The artifacts are verified by Field II simulations, where an 1-D array with 128 transducers is used. Figure 2.3 shows the ultrasound fields at 3cm depth varying with time. The three cases correspond to no virtual source, 11-element virtual source and 31-element virtual source. The use of virtual source incurs undesired tails; as the number of elements increases, the tail becomes more significant. In the 31-element virtual source case, there are significant gaps between the tails and the original pulse waves. The artifacts on point target images are shown in Figure 2.4. The 31-element virtual source incurs two peaks in axial direction and has significant sidelobes. The 11-element virtual source case is better, but the width of the PSF in axial direction increases compared to the case where virtual source is not used. In a 1-D array with half wavelength spacing, 11-element virtual source configuration is a good trade-off

(a) No Virtual Source

(b) 11-Element Virtual Source

(c) 31-Element Virtual Source
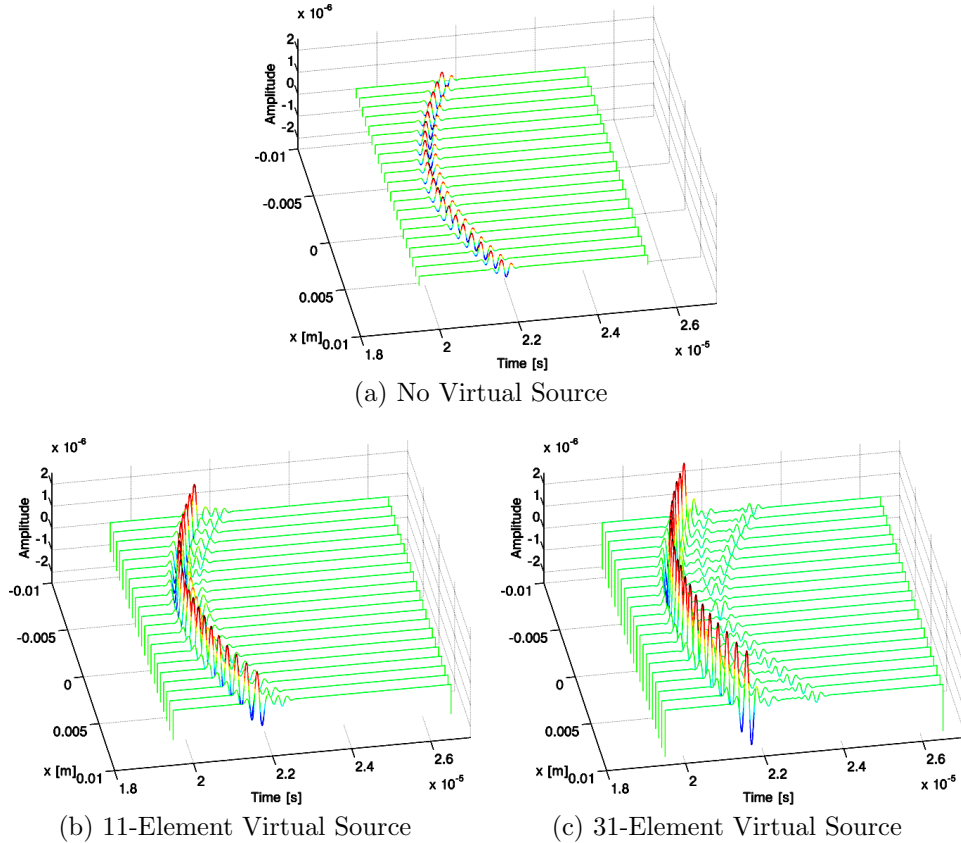
Figure 2.3: Virtual Source Field II Simulation

between SNR gain and tail artifacts [18]. Thus this method increases SNR gain but suffers from tail artifacts.

2.3 Orthogonal Coded Excitation For Synthetic Aperture Ultrasound

The tail artifact precludes virtual source firing technique from supporting SNR gain higher than 10 dB for a 2-D imaging system and 15 dB for a 3-D imaging system. On the other hand, temporal coded excitation using chirps and Golay codes have large time-bandwidth product and can improve the SNR of SAU systems by about 15dB [20]. Orthogonal coding in spatial domain can further improve the SNR by typically 10-20 dB as shown in [21–23].

12

(a) No Virtual Source

(b) 11-Element Virtual Source

(c) 31-Element Virtual Source
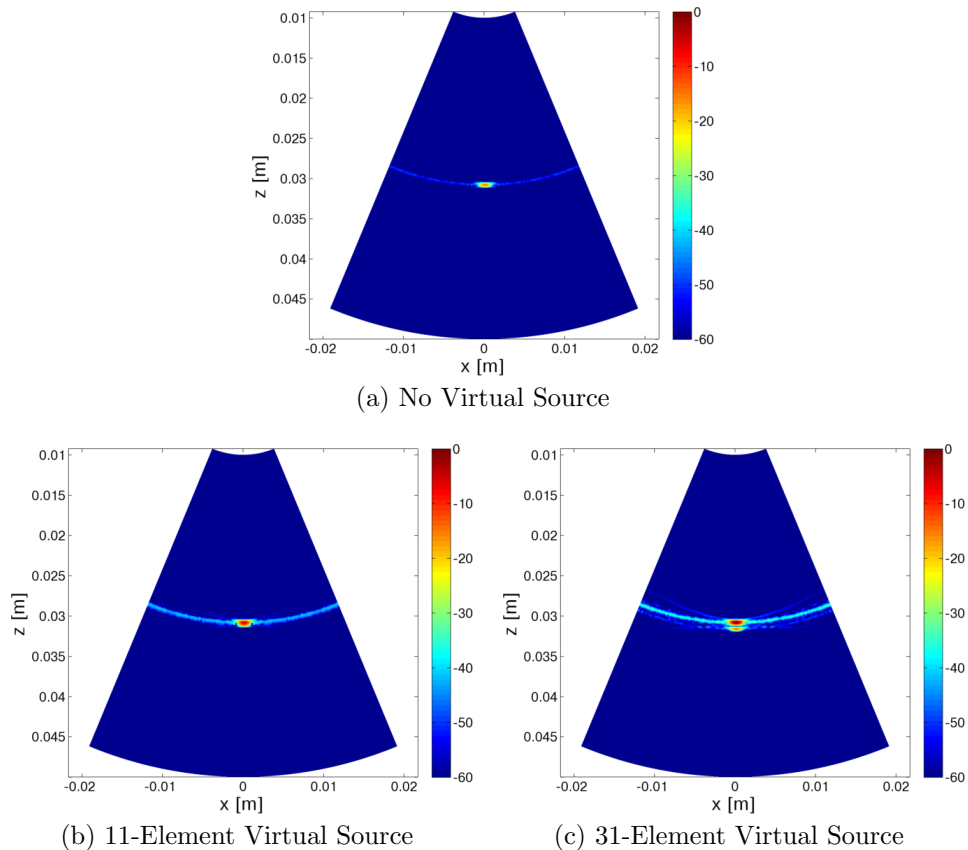
Figure 2.4: Virtual Source Artifacts on Point Target Images

There exist several coded excitation systems based on orthogonal Golay codes and chirps [21–23]. In these systems, the RF-data from the A/D converter is first decoded using a Hadamard transform. This data is then processed by a compression filter in case of a chirp-based system or two code correlators in case of a Golay code-based system. The decoded data is then sent to a beamformer. We define such a system as decoding-first since the decoding is done before beamforming. While such a system has the advantage of simple beamforming, it results in high computation and storage complexity. Moreover SAU systems with orthogonal codes are sensitive to body motion and in the decoding-first architecture, the motion artifacts can not be easily compensated.

13

In this chapter, we first present an efficient architecture for orthogonal chirp and orthogonal Golay code-based systems that integrates decoding with beamforming. The proposed architecture significantly reduces computation complexity and storage space compared to existing decoding-first systems. Then we compare the performance of the two proposed systems in presence of motion and propose a simple motion compensation scheme which can significantly improve the system performance in terms of SNR and range sidelobe level (RSLL). We compare the implementation complexity of both systems and show that while orthogonal Golay code-based system has lower computation complexity compared to chirp-based system, when motion compensation is included, both systems have comparable complexity. Then we discuss several techniques to reduce the complexity of orthogonal code based SAU systems. Since the complexity of such a system is a function of N, the number receive elements, M, the number of transmit elements and L the code length, we first describe a framework to choose the value of these parameters such that the overall complexity is minimized for a given SNR gain. We consider both the number of multiplications and the number of additions in the formulation. We show that this procedure reduces the complexity of the system by about $20\times$.

2.3.1 Orthogonal Golay Code

Biphase code is a type of coded excitation that has low implementation complexity. Examples include Barker codes, pseudo noise series, Golay complementary codes, etc. Correlators of biphase code can be implemented with only additions and subtractions. In ultrasound imaging, the requirement for the dynamic range has to be quite large (40 dB - 80 dB), and biphase codes with zero sidelobes are desired. Golay code is a type of biphase code with perfect sidelobe cancellation [23–25].

Single transmit biphase codes inevitably have range sidelobes; Golay complimentary pairs that need two transmissions can perfectly cancel range sidelobes. The autocorrelation function of Golay complementary pairs meet the following condition.

$$X_g[k] = \sum_{i=1}^{2} \sum_{l=1}^{L-k} g_i[l]g_i[l+k] = 2L\delta[k] \tag{2.3}$$

A simple example of Golay complementary pair is $g_1[n] = \{1,1\}$ and $g_2[n] = \{1,-1\}$. The autocorrelation of $g_1[n]$ is $\{1,2,1\}$, and the autocorrelation of $g_2[n]$ is $\{-1,2,-1\}$ and the sum of the two autocorrelations generate the result $\{0,4,0\}$.

The Golay complementary pairs can be generated in the following way. According to [26], given a complementary pair $g_1[n]$ and $g_2[n]$ with length L, the Golay codes with length $2L$ can be obtained by

$$g_1'[n] = \begin{cases} g_1[n], & 0 \leqslant n < L \\ g_2[n-L], & L \leqslant n < 2L \end{cases} \tag{2.4}$$

$$g_2'[n] = \begin{cases} g_1[n], & 0 \leqslant n < L \\ -g_2[n-L], & L \leqslant n < 2L \end{cases} \tag{2.5}$$

Orthogonal Golay pairs are constructed as follows. First, construct a complementary Golay pair $\mathbf{g}_{1,1}$ and $\mathbf{g}_{2,1}$ with desired length L [27, 28]. $\mathbf{g}_{1,1}$ is transmitted by the first transmit element in the first transmission, and $\mathbf{g}_{2,1}$ in its second transmission. The orthogonal pair, $\mathbf{g}_{1,2}$ and $\mathbf{g}_{2,2}$, are constructed from $\mathbf{g}_{1,1}$ and $\mathbf{g}_{2,1}$; $\mathbf{g}_{1,2}[l] = -\mathbf{g}_{2,1}[L-1-l]$ and $\mathbf{g}_{2,2}[l] = \mathbf{g}_{1,1}[L-1-l]$, where $l \in \{0,1,\ldots,L-1\}$. $\mathbf{g}_{1,2}$ is transmitted by the second transmit element in the first transmission, and $\mathbf{g}_{2,2}$ is transmitted in the second transmission. The two orthogonal complementary pairs are used to recursively construct codes transmitted by M elements in M transmissions. According to [23], the code set can be constructed as follows

15

$$
\mathbf{G} = \begin{bmatrix} \mathbf{g}_{1,1} & \cdots & \mathbf{g}_{1,M} \\ \vdots & \ddots & \vdots \\ \mathbf{g}_{M,1} & \cdots & \mathbf{g}_{M,M} \end{bmatrix} = \mathbf{H}_{\frac{M}{2} \times \frac{M}{2}} \otimes \begin{bmatrix} \mathbf{g}_{1,1} & \mathbf{g}_{1,2} \\ \mathbf{g}_{2,1} & \mathbf{g}_{2,2} \end{bmatrix} \tag{2.6}
$$

where $\mathbf{H}$ is a $\frac{M}{2}$-by-$\frac{M}{2}$ Hadamard matrix which is composed of -1s and $+1$s, and $\otimes$ is Kronecker product. $\mathbf{g}_{m,i}$ is transmitted by the ith Tx element in the mth transmission.

2.3.2 Orthogonal Chirp

Chirp is a family of signals that are widely used in sonar and radar applications. In this report, chirp refers to linear frequency modulation (LFM) signal. The LFM chirp signal with transmit window can be expressed as

$$
e(t) = W(t, T) \cos(\omega_0 t + \frac{\gamma t^2}{2}) \tag{2.7}
$$

where $W(t, T)$ is a window function, which limits the signal in region $[-T/2, T/2]$. Note that the derivative of the phase with respect to time is the frequency $f(t) = \omega_0 + \gamma t$, and since the frequency is linearly increasing with time, it is called Linear Frequency Modulation (LFM) signal.

The receiving matched filter can be defined by

$$
s(t) = W'(t, T) \cos(\omega_0 t - \frac{\gamma t^2}{2}) \tag{2.8}
$$

where $W'(t, T)$ is also a window function. Assume $W(t, T) = W'(t, T) = \Pi(t/T)$, where $\Pi(t)$ is a rectangular window defined by

$$
\Pi(t) = \begin{cases} 1, & \text{for } -\frac{1}{2} \leqslant t \leqslant \frac{1}{2} \\ 0, & \text{for else} \end{cases} \tag{2.9}
$$

In such a case, the output signal can be approximated by

$$e_0(t) = \Pi\left(\frac{t}{2T}\right)\sqrt{\frac{2\gamma}{\pi}}\cos(\omega_0 t)\frac{\sin[\gamma t(T - |t|)/2]}{\gamma t} \tag{2.10}$$

However, the sidelobe of this signal is big, which can degrade the imaging quality. Different methods are used to overcome this deficiency. One of the solutions is to use a different time-domain window such as the Dolph-Chebyshev window, whose expression is given by

$$G(t/T) = \frac{\cos(\pi\sqrt{(t/T)^2 - A^2})}{\cosh(\pi A)} \tag{2.11}$$

By properly choosing parameter A, the sidelobe level can be greatly reduced [29–31].

The orthogonal chirps can be constructed using the following expression.

$$\mathbf{S}(t) = \begin{bmatrix} s_{1,1}(t) & \cdots & s_{1,M}(t) \\ \vdots & \ddots & \vdots \\ s_{M,1}(t) & \cdots & s_{M,M}(t) \end{bmatrix} = \mathbf{H}_{M \times M} \otimes s(t) \tag{2.12}$$

where $s(t)$ is a chirp signal, and signal $s_{m,i}$ is transmitted from the ith transmitter element in the mth transmission, and $\mathbf{H}$ is a M-by-M Hadamard matrix.

2.3.3   System Design Optimization For SAU System Using Coded Excitation

Beamforming and Decoding Architecture

The proposed digital front-end architectures for orthogonal chirp and orthogonal Golay based systems corresponding to one receive chain are shown in Figure2.5a and Figure2.5b, respectively. The first unit is apodization, which is a fixed-coefficient spatial window that can reduce lateral sidelobe levels. Next, interpolation filtering is done to increase the sampling rate from 40MHz at the A/D to 120MHz. For the

17

(a) Orthogonal Chirp Beamforming Architecture Part I

(b) Orthogonal Golay Beamforming Architecture Part I

(c) Beamforming part II
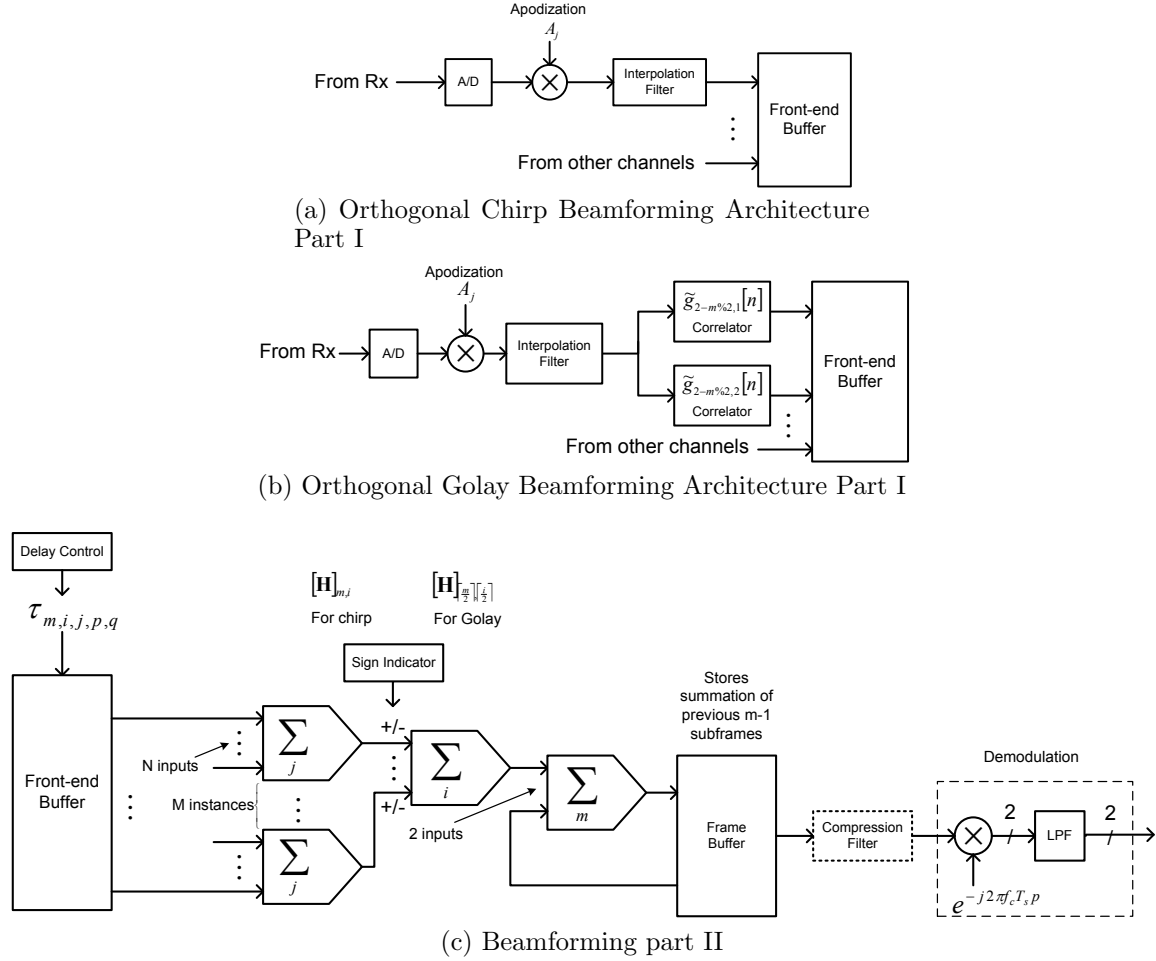
Figure 2.5: Beamforming Architectures for Orthogonal Golay and Chirp with Demodulation after Beamforming

Golay code-based system, two correlators are used to generate two versions of correlated signals for decoding. The output of the interpolation filter in case of chirp-based system or the output of the two correlators in case of Golay code-based system, is stored in the front-end buffer for further processing by the beamforming unit.

The beamforming architecture, shown in Figure2.5c, is almost the same for Golay code and chirp-based systems. The delay control unit chooses the signal samples from the front-end buffer according to delay $\tau_{m,i,j,p,q}$, where m is the transmission index, i is the transmit (Tx) element index, j is the receive (Rx) element index, p is the focal

point index and q is the scanline index. Clearly, τ is a function of i, j, p and q, and in presence of motion, it is also a function of m.

There are three stages of summation. The first stage, which consists of M summers, sums signal samples from N receiving elements. In the second stage, the signal samples corresponding to M transmit elements are summed up, thereby generating the value corresponding to a focal point. The Hadamard matrices $\mathbf{H}_{M \times M}$ for chirp-based system and $\mathbf{H}_{\frac{M}{2} \times \frac{M}{2}}$ for Golay code-based system are used to decide whether to add or subtract the streams corresponding to different transmit elements. For example, if $[\mathbf{H}]_{\lceil \frac{m}{2} \rceil, \lceil \frac{i}{2} \rceil} = -1$ in a Golay code-based system, the stream corresponding to the mth transmission from the ith element is subtracted from the streams from the mth transmission of all the other transmit elements. Actually this stage merges the spatial decoding and beamforming together. For the mth transmission, this process is repeated PQ times and an image containing PQ focal points for the mth transmission is generated. In the third stage, the image frame stored in the frame buffer is updated with the one generated in the mth transmission.

The data in the frame buffer is then demodulated. In case of a chirp-based system, there is a compression filter before the demodulation stage. Both compression filtering and demodulation are done scanline by scanline.

The complexity of the Golay code-based and chirp-based systems are compared using the parameters defined in Table 2.1. While the Golay code-based system requires two correlation units, they involve only additions and subtractions. The chirp-based system, on the other hand, requires a compression filter which increase the complexity significantly – about $PQT_{code}f_s$ additional multiplications per image. For the setting in Table 2.1, the chirp-based system needs about 1.1×10^{10} multiplications per image, while the Golay code-based system needs only 2.9×10^8 multiplications.

Table 2.1: Parameter Definitions and Values of SAU Systems

Symbol	Description	Value
N	Number of receiving elements	128
M	Number of transmitting elements and number of transmissions	32
f_c	Transducer central frequency	4 MHz
B	6 dB bandwidth of transducer	4 MHz
f_s	A/D sampling frequency	40 MHz
f_s'	Sampling frequency after interpolation	120 MHz
T_{code}	Duration of the coded excitation	32 μs
K_{LPF}	Number of taps of demodulation lowpass filter	36
K_{INT}	Number of taps of interpolation lowpass filter	5
D	Maximum detection depth	20 cm
c	Speed of sound in body tissue	1540 m/s
P	Number of focal points in one scanline	1.04×10^4
Q	Number of scanline in one image	200

A comparison of the complexity of the proposed chirp-based system and the chirp-based system implemented with decoding-first scheme shows that the proposed system requires significantly fewer multiplications, $PQT_{code}f_s$ compared to $PMNT_{code}f_s$. Thus for the setting in Table 2.1, the proposed chirp-based system needs only ~5% of the multiplications required by the corresponding decoding-first system. In terms of space complexity, the decoding-first architecture has to buffer all M subframes received in M transmissions. In contrast, the proposed architecture only needs to buffer the receive data in one subframe though this data is stored at a higher sampling rate

because of interpolation. Thus, the space complexity is reduced to $\frac{f'_s}{f_s M}$ or 9.4% of decoding-first architecture for the setting in Table 2.1.

For orthogonal Golay code-based systems proposed in [23, 32], $2M$ correlators perform N times for one image, while in the proposed system, $2N$ correlators perform M times. Hence the complexity of the two systems in terms of code correlation is the same. However in terms of space complexity, the proposed architecture only needs to buffer two correlated versions of receive data in one subframe. In contrast, the decoding-first system stores M subframes. Hence the proposed architecture only needs $\frac{2f'_s}{f_s M}$ of the storage needed by the decoding-first architecture.

Design Parameter Optimization

Since the complexity of such a system is a function of N, the number receive elements, M, the number of transmit elements and L the code length, we first describe a framework to choose the value of these parameters such that the overall complexity is minimized for a given SNR gain. We consider both the number of multiplications and the number of additions in the formulation. We show that this procedure reduces the complexity of the system by about $20\times$. Next we reduce the complexity of the motion compensation method by first operating in the polar domain and then exploiting the property of uniformity of velocity field in a small region. This method helps reduce the complexity of motion compensation by $3000\times$. Finally we show that as a result of these reductions, the beamsum is now the bottleneck of orthogonal coded excitation based SAU systems.

In this section we describe how the parameters N, M, L can be chosen to satisfy the performance requirements while minimizing the complexity. We define the reference system as one with only one transmit and one receive element, and with excitation of unit time-bandwidth product (TBP). We specify the performance re-

quirement in terms of SNR gain which is defined as the ratio of the SNR of a system and the SNR of the reference system. It has been proven that longer code length of wide band signal results in higher TBP, and higher TBP contributes to higher SNR gain [30]. The number of transmissions and the number of simultaneous transmit elements also contributes to SNR gain [33]. Thus the theoretical SNR gain is approximately $10 \log_{10}(NM^2L)$ dB for both chirp-based and Golay-based systems [22].

Since $M \leqslant N$, N determines the aperture size, and consequently the lateral resolution. Thus N should be chosen according to the desired lateral resolution. Axial resolution is hard to quantify. It is a function of system bandwidth, type of coded excitation and motion speed, and is not considered in the formulation below. Now for a given SNR gain constraint, there are multiple choices of M and L. In the following, we formulate this as an optimization problem which minimizes computation complexity given SNR gain constraint $NM^2L \geqslant A$, where $L \in [L_{min}, L_{max}]$ and $M \in [M_{min}, M_{max}]$. In our system, $L_{min} = 1$, $L_{max} = 128$, $M_{min} = 1$ and $M_{max} = 64$.

Optimization of Golay code-based systems. For the Golay code based system, the computation complexity is given by

$$K_{INT}PNM + \alpha[K_{INT}PNM + PQNM^2 + 2PNML] \tag{2.13}$$

Here $K_{INT}PNM$ is the number of multiplications and additions required by the interpolation filter, $PQNM^2$ is the number of additions needed by the beamsum unit, and $2PNML$ is the number of additions for the Golay correlation units. We combine the effect of multiplications and additions by scaling the number of additions by a constant α, where α is the ratio of the complexity of an adder to the complexity of a multiplier. For a 16-bit-fixed-point system, that range of α varies from 1/14 to 1/8, depending on the specifics of the adder and multiplier implementations. In this work, we use $\alpha = 1/12$ to evaluate the complexity.

Optimization of chirp-based systems. For the chirp-based system, the complexity is given by

$$K_{INT}PNM + PQL_0L + \alpha[K_{INT}PNM + PQNM^2 + PQL_0L] \qquad (2.14)$$

where $K_{INT}PNM$ is the number of multiplications and additions in the interpolation filter, $PQNM^2$ is the number of additions in the beamsum unit and PQL_0L the number of multiplications and additions in the compression filter with L_0L taps. Here L_0 is the number of samples in one period of a sinusoid whose frequency is the same as the transducer's central frequency.

FFT-based compression filtering. Since complexity of the chirp compression filter is very high and the number of taps can be as high as 1280, we propose to use FFT for filter implementation. We use the overlap-save method, where the signal is divided into s chunks of equal length, and length of FFT U is chosen according to $U = 2^{\lceil \log_2(L_0L + (L_0L+P)/s-1) \rceil}$. For this implementation the number of multiplications per scanline is $s(2U \log_2 U + 2U)$, and the number of additions per scanline is $s(3U \log_2 U + U)$. The computation complexity of chirp based system using FFT-based compression filter is given by

$$K_{INT}PNM + Qs(2U \log_2 U + 2U) + \alpha[K_{INT}PNM + PQNM^2 + Qs(3U \log_2 U + U)]$$
$$(2.15)$$

Complexity Results

Figure 2.6 plots the number of effective multiplications for different values of SNR gain for the two systems. We see that the complexity of the system is proportional to the SNR gain as expected. The step shape is caused by the constraint that M is
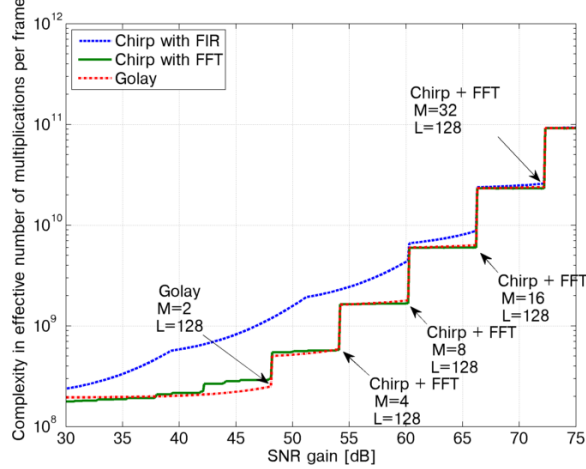
Figure 2.6: Choice of M and L to Achieve Lowest Complexity for Different Values of SNR Gain

an integer power of 2 — a constraint that is set by the Hadamard matrix required to generate orthogonal codes. We repeated this experiment for $\alpha = 1/8$ and $\alpha = 1/16$. We find that the optimal values of M, L do not change with α. This is because the constraint that M has to be an integer power of 2 is a strong constraint and changing the value of α has little effect on the choice of the parameters M and L.

To evaluate the optimization efficiency, we compare the complexity of the optimal solution with the average of all feasible solutions for a specific SNR gain. Figure 2.7a and 2.7b show the comparison result of the average versus the optimal configuration for five different SNR constraints. The complexity of the optimal solution is only 5.8% of the average for Golay code based system and 5.5% of the average for chirp based system. Of all the units, the beamsum is the most complex with a complexity of 99% for chirp based system and 98% for Golay code based systems. The reduction in the number of computations for beamsum is quite significant. For 60 dB SNR gain, the number of effective multiplications for optimized beamsum is only 4.7% of those needed by the average. The complexity reduction of each unit is summarized in Table 2.2.

24
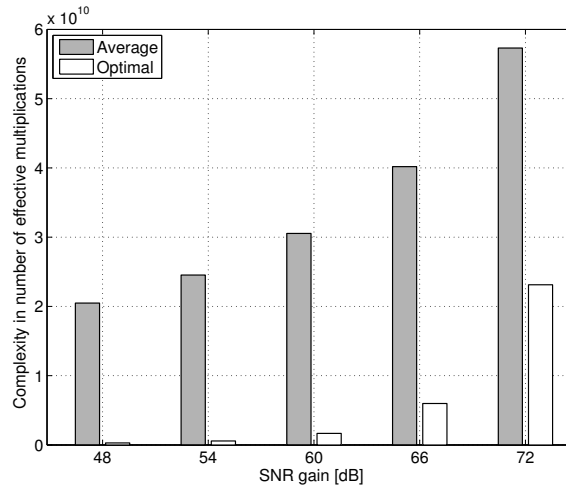
(a) Chirp-based System
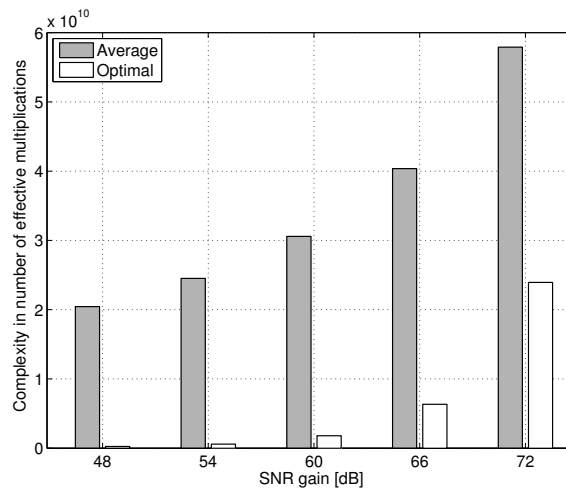
(b) Golay Code-based System

Figure 2.7: Complexity Comparison between Average and Optimal Configuration of Chirp and Golay Code Based Systems

2.4 Motion Compensation

Orthogonal coded excitation boost SNR gain up to 70 dB, however in presence of body motion, both chirp-based system and Golay code based systems suffers from significant SNR loss, as shown in Figure 2.8. Golay code based systems are more sensitive to body motion because orthogonal code based SAU systems require perfect

Table 2.2: Complexity Reduction for 60 dB SNR Gain Case

Component	Golay Avg.	Golay Opt.	Percentage
Interpolation	2.1×10^8	5.8×10^7	27.6%
Beamsum	3.0×10^{10}	1.4×10^9	4.7%
Correlation	1.1×10^8	2.3×10^8	209%
Demodulation	9.2×10^7	9.2×10^7	100%
Total	3.1×10^{10}	1.8×10^9	5.8%
Component	**Chirp Avg.**	**Chirp Opt.**	**Percentage**
Interpolation	2.1×10^8	5.8×10^7	27.6%
Beamsum	3.0×10^{10}	1.4×10^9	4.7%
Correlation	6.9×10^7	1.0×10^8	145%
Demodulation	9.2×10^7	9.2×10^7	100%
Total	3.1×10^{10}	1.7×10^9	5.5%

timing alignment to insure the signal coherency and sidelobe suppression. Thus to design a high performance system, motion compensation has to be included.

2.4.1 Baseline Motion Compensation Method

In this section we describe a simple motion compensation scheme for the proposed architecture. The idea is that if the motion velocity can be estimated (as in [34]), then the beamforming algorithm can dynamically adjust the delay and phase correction term and the focal point can follow the moving target points [35]. Here we propose
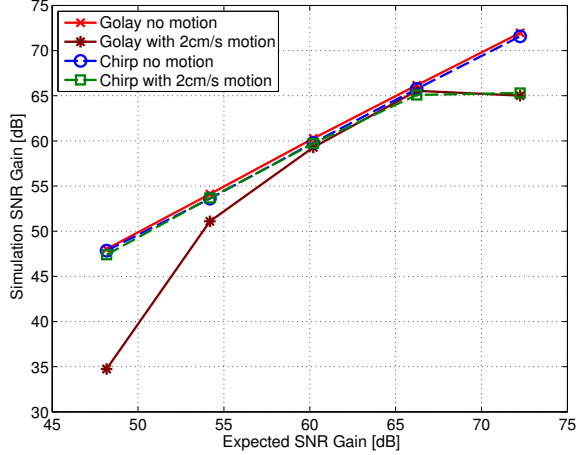
Figure 2.8: SNR Gain in Presence of Motion

a simplified version of the method in [35] using Taylor expansion to update $\tau_{m,i,j,p,q}$, which requires only 2 multiplications for each delay value.

In the absence of motion, the delay $\tau_{m,i,j,p,q}$ is constant for all transmissions. This can be calculated by $\tau_{m,i,j,p,q} = (A_{j,p,q} + B_{i,p,q})/c$, where $A_{j,p,q} = \sqrt{(x_j - x_{p,q})^2 + z_{p,q}^2}$ and $B_{i,p,q} = \sqrt{(x_i - x_{p,q})^2 + z_{p,q}^2}$ are the distance between the focal point and the jth Rx element and the ith Tx element, respectively. This part can be pre-calculated and stored in a look up table.

If motion artifacts are significant, then it is necessary to add a compensation value $\Delta\tau_{m,i,j,p,q}$ to the delay value of the next transmission.

$$\Delta\tau_{m,i,j,p,q} = \frac{1}{c}\left(\frac{x_{p,q} - x_j}{A_{j,p,q}} + \frac{x_{p,q} - x_i}{B_{i,p,q}}\right)v_x\Delta t + \frac{1}{c}\left(\frac{z_{p,q}}{A_{j,p,q}} + \frac{z_{p,q}}{B_{i,p,q}}\right)v_z\Delta t \qquad (2.16)$$

where $z_{p,q}$ is the z coordinate of the focal point p in scanline q, Δt is the time interval between two transmissions. For efficient computation of $\Delta\tau_{m,i,j,p,q}$, we need two look-up tables, one to store $\frac{(x_{p,q} - x_j)\Delta t}{cA_{j,p,q}}$ and the other to store $\frac{z_{p,q}\Delta t}{cA_{j,p,q}}$. Each table has NPQ elements, which is fairly large. But in a real implementation, the symmetry of the scanlines and transducer elements and the fact that the delay will be eventually mapped to an integer memory address, is used to reduce the size of the tables.

27

In baseline method, we used Taylor expansion to simplify the calculation in motion compensation. Unfortunately, the complexity of that method is still very high. In this subsection, we propose a scheme that reduces the complexity by first doing the computations on data represented in the polar system and second by assuming uniformity of velocity field in a small region.

Mapping Computations into Polar Domain
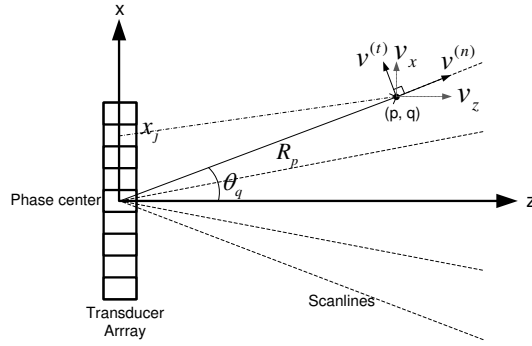
Figure 2.9: Motion Decomposition Using Polar Coordinates

We propose to represent the motion velocity in polar coordinates instead of rectangular coordinates with v_x and v_y as shown in Figure 2.9. Such a representation is better in terms of both computation and storage complexity.

Suppose the target at point (p, q) is moving, so the distance $R_{p,q}$ and the angle $\theta_{p,q}$ vary with time. The two new velocity components are defined as $v_{p,q}^{(n)} = \frac{dR_{p,q}}{dt}$ and $v_{p,q}^{(t)} = R_{p,q}\frac{d\theta_{p,q}}{dt}$, where $v_{p,q}^{(n)}$ is parallel to the scanline and $v_{p,q}^{(t)}$ is vertical to the scanline.

We find that when $R_{p,q}$ is large, the derivative of single trip delay from point (p, q) to receive element j can be approximated by equation (2.17), where x_j is the coordinate of the receive element j.

28

$$\frac{d\tau_{rx}}{dt} \approx \frac{1}{c}v_{p,q}^{(n)} - \frac{x_j \cos \theta_{p,q}}{cR_{p,q}}v_{p,q}^{(t)} \tag{2.17}$$

For the round trip delay from transmit element i to point (p, q) to receive element j, the approximation of the total delay adjustment term for mth transmission is represented by

$$\Delta\tau_{m,i,j,p,q} \approx \frac{2}{c}v_{p,q}^{(n)}m\Delta t - \frac{(x_j + x_i) \cos \theta_{p,q}}{cR_{p,q}}v_{p,q}^{(t)}m\Delta t \tag{2.18}$$

where x_i is the x coordinate of the ith transmit element, Δt is the transmit interval between two consecutive transmissions, m is the index of transmission which varies from 0 to $M - 1$

To ensure good accuracy at small depths, only a few transmit/receive elements near the center are used. Since here the signal strength is good, this operation incurs only a mild performance loss. As the distance $R_{p,q}$ increases, more streams are used for beamforming. Simulation results show that $\frac{R_{p,q}}{\max(|x_i|,|x_j|)} > 3$ reaches a balance between approximation accuracy and performance loss due to reduction in the aperture size.

Neighborhood Approximation

Body motion is likely to have significant regional correlation. In abdominal ultrasound imaging, the most likely motion is caused by patient's breathing or moving of transducer head. As a result, it is reasonable to assume that the velocity field generated by body motion is continuous and locally correlated. Based on this assumption, delay adjustment term due to motion does not have to be calculated for every focal point, and can be calculated once in every local neighborhood.

Here we use a simple deformation velocity field to simulate a mild body motion [36]. The velocity field shown in Figure 2.10a corresponds to the case where the tissues

29

(a) Simulation Velocity Field   (b) Neighborhood Approximation Error
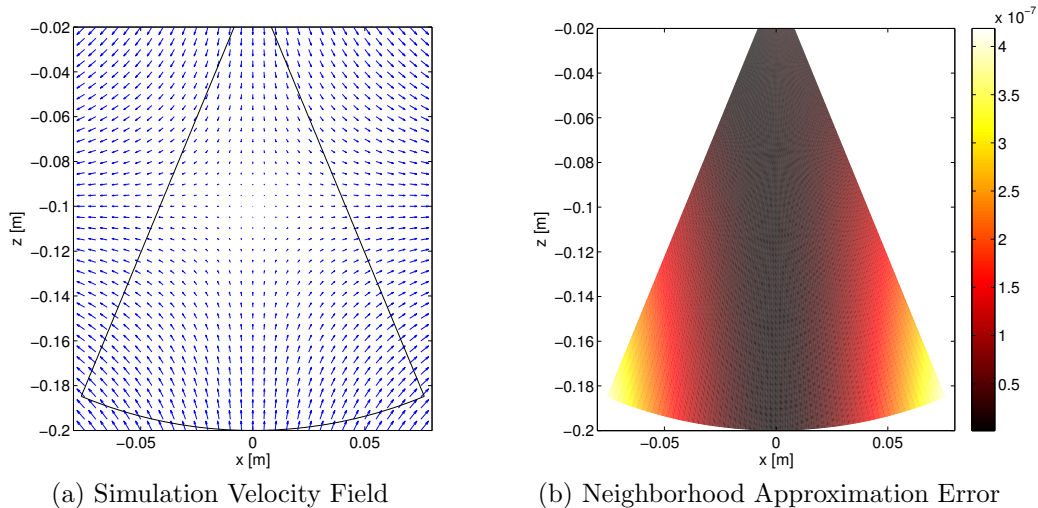
Figure 2.10: Velocity Field and Neighborhood Approximation Error when the Neighborhood is of Size 20×3

are being compressed in the vertical direction and being expanded in the horizontal direction. In motion fields where the velocity vectors have similar orientations, the neighborhood approximation has very good performance even for grid sizes as large as 200×5 (200 samples along the scanline and 5 scanlines wide). However in the motion field investigated in this paper, the motion vectors are in all possible directions in the 2D plane. In this case, even some small neighborhood size such as 50×1 or 30×5 results in large Range Sidelobe Level (RSLL). In the worst case where motion speed reaches 20mm/s, the RSLL is larger than -41 dB. As a result, smaller grid size is necessary for such a motion field. To choose a good grid size, we further tested grid sizes of 10×6, 20×3 and 30×2. All three configurations provide acceptable performance and have the same reduction in complexity. However, when motion speed is 20mm/s, 20×3 neighborhood approximation has about 3 dB better performance than the other two cases and approaches the performance of motion compensation without approximation. The mean square error of the delay values in

30

each neighborhood is shown in Figure 2.10b. In this configuration the highest mean square error is proportional to 4.2×10^{-7}, and occurs in the bottom two corners.

Complexity Analysis

We first analyze the number of multiplications and additions that are required to compute Equation (2.18). The first term in the equation does not depends on i or j, which means it can be shared by all streams. As a result, this part only needs 1 multiplications per focal point, so if we recalculate the delay for every subframe, PQM multiplications are required for one image. The second term is a linear function of $x_i + x_j$. When elements are equally spaced, one can calculate the second term with only add/subtract according to the adjustment term used in $i - 1$ or $j - 1$ streams. As a result, this part needs PQM multiplications and PQM^2N addition per frame. Compared to the motion compensation in [9], the number of multiplications needed is only 0.12% of that method.

Now, for the neighborhood approximation method, there are $P'Q' = PQ/(20 \times 3)$ small neighborhoods, and the samples in the same neighborhood can share the same adjustment term. Taking this into account, the new motion compensation needs $2P'Q'M$ multiplications and $2P'Q'M^2N$ additions per frame. Compared to the method in [9] using the same configuration, the total number of effective multiplications needed by motion compensation is reduced from 6.8×10^{11} per frame to 2.3×10^8 per frame, which corresponds to about $3000\times$ reduction.

2.4.3   Simulation Results

We ran simulations for the gradient field described in Figure 2.10a for maximum motion speed varying from 0 mm/s to 20 mm/s. We put 30 target points in the

observation area; the space between each point is 30mm along range direction and 10° along the azimuth direction.

Figure 2.11 and 2.12 show the average SNR performance and average RSLL performance of 30 target points for different motion speeds. As shown in Figure2.11, the new reduced complexity motion compensation method can significantly improve the SNR for both Golay code and chirp based systems. The SNR performance of Golay code based systems is slightly better than a chirp based system.
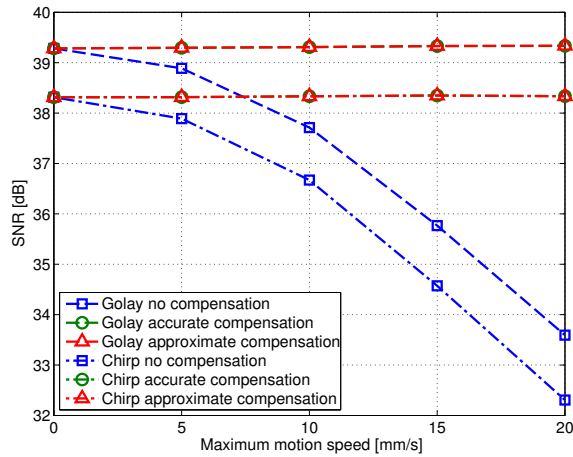
Figure 2.11: SNR Performance for Different Motion Speeds

From Figure 2.12 we see that for both Golay code based and chirp based systems, the RSLL performance of the simplified motion compensation method is very close to the one without the approximations.
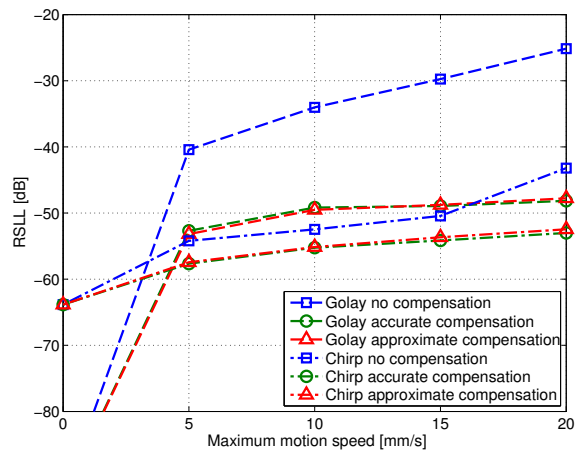
Figure 2.12: RSLL Performance for Different Motion Speeds

Chapter 3

SONIC MILLIP3DE: ALGORITHM INNOVATIONS AND HARDWARE DESIGN

3.1 Motivation

In Chapter 1 we have made the case for 3-D ultrasound imaging. Not only are 3-D images easier to interpret, reducing effort (and errors) for technicians to locate relevant anatomy, they also provide accurate volumetric measurements of cysts and tumors that 2-D cannot match.

However, the benefits of 3-D also come with numerous hardware challenges that are only exacerbated when trying make the system hand-held. The construction of a 3-D volumetric image requires at least $4000\times$ computations compared to a 2-D system. On the other hand, because it is in close contact with human skin, an ultrasound scan head must operate within a tight power budget (about 5W) to maintain safe temperatures.

In this chapter, we present algorithm level optimizations for *Sonic Millip3De*, a low-power hardware accelerator architecture for 3-D ultrasound imaging. This work was done in collaboration with researchers in University of Michigan. The algorithm level optimizations include subaperture processing, apodization optimization (Section 3.2.2) and iterative delay calculation method (Section 3.2.3). Subaperture processing reduces the number of concurrent channels and the number of firings for each frame by a factor of 2 thereby reducing computational complexity by a factor of 2. Subaperture apodization optimization not only maximizes SNR but also reduces sidelobe levels. Iterative delay calculation eliminates complex calculations such as square roots, divisions and multiplications. It reduces the storage by $400\times$ compared

to direct lookup table implementation. For each focal point, it only requires three additions, thereby greatly reduces the power consumption on address calculations.

Next, the Sonic Millip3De hardware architecture is briefly described in Section 3.3. It makes use of a massively parallel design and state-of-the-art 3-D die stacking [37–40], splitting analog components, analog-to-digital (ADC) converters and SRAM storage, and a 1024-unit beamsum accelerator array across three silicon layers for a compact design with short (and hence low-power) wires. Based on RTL-level design and floorplanning for an industrial 45nm process, we estimate a full-system power requirement of 15W for Sonic Millip3De and project that it will meet the 5W target power budget by the 11nm node.

The study was presented in [11–13].

3.2   Algorithm Design

Conventional beamforming algorithms (intended for implementation in software) are too storage- and compute-intensive for hand-held 3-D ultrasound. In this section, algorithmic innovations that enable a low-power hardware implementation are presented.

3.2.1   Subaperture Processing

To reduce the number of firings and to improve image quality, in this section overlapped subaperture processing is presented. This method helps reduce the computation requirements for one frame by a factor of 2 so that the power consumption can be reduced. Alternatively, it can increase the frame rate by a factor of 2 without increasing the computational requirements.

In a naive implementation of an SAU system with a 2-D array of $128\times96{=}12{,}228$ transducers and non-overlapping subapertures of size 32×32, there are a total of

12,228/1,024=12 subapertures. If there are 4×4=16 virtual sources, each virtual source has to fire 12 times, and there are 12×16 = 192 firings for generating a 3-D frame.

In this section, a firing method based on overlapping subapertures that helps increase the frame rate is presented. This method is a generalization of the 1-D scheme in [41]. As shown in Figure 3.1, each subaperture is of size 32×32 elements and neighboring subapertures overlap with each other. There are 96 virtual sources and each virtual source is located in the center of the corresponding subaperture. Each virtual source fires only once and is emulated by 76 transducers. The use of virtual source increases SNR by about 19 dB compared to the single transducer firing scheme [18]. After each firing, both virtual source and receive subaperture shift by 8 elements to an adjacent location. Consequently, the number of firings required by this scheme is 96, which is half the number required in [11]. As a result, computations for processing one frame are reduced by a factor 2. Although this method only uses 120×88 transducer elements, the equivalent aperture size is increased, because more virtual sources fire from wider angles. To achieve the desired image quality for such a system, the apodization coefficients of each subaperture have to be optimized.

3.2.2 Apodization Scheme for Subaperture Processing

1-D Subaperture Apodization

In this section we used the equivalent aperture concept introduced in [19, 41] to formulate the subaperture apodization problem for 2-D subapertures. The equivalent aperture of a transducer array is defined as the receive aperture that can exactly produce the same radiation pattern as the original transducer array but with only
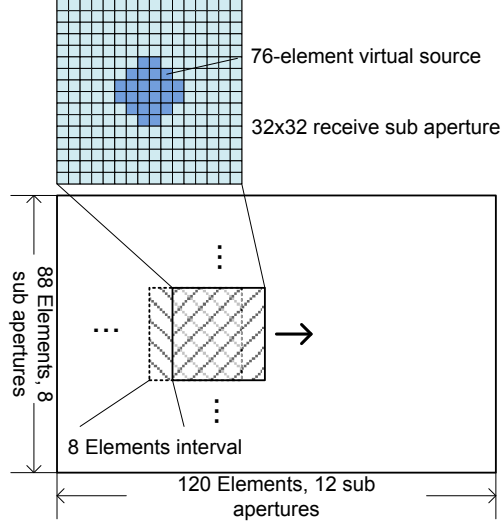
Figure 3.1: Overlapped Subaperture Processing for SAU Systems

one single transmit element [19]. In the far-field scenario, the equivalent aperture is simply the convolution of transmit and receive apertures.

Assuming that the power for each firing is unity, apodization of the equivalent aperture $y[n]$ can be represented by $y[n] = \sum_{i=0}^{L-1} a_i[n - m_i]$, where $a_i[n]$ is the 1-D receive subaperture apodization for the ith transmission, and m_i is the location of ith virtual source. The convolution process can be represented in matrix-vector form as $\mathbf{y} = \mathbf{Ga}$ where $\mathbf{a} = [\mathbf{a}_1^T, \ldots, \mathbf{a}_L^T]^T$ consists of all nonzero subaperture apodization windows, $\mathbf{a}_i$ a vector comprising of nonzero apodization coefficients of ith subaperture. $\mathbf{G}$ is a $N \times M$ matrix with only zeros and ones, which represents the summations and shifts required by the convolution of transmit and receive apertures. Here N is the size of the equivalent aperture, and $M = KL$, where K is the size of a subaperture, and L is the number of subapertures.

Since subapertures overlap with each other, matrix $\mathbf{G}$ does not have full rank. Thus there are multiple solutions to this linear equation, and the best solution is the one that maximizes SNR and reduces sidelobes. Given that the power for each firing

is unity, the lower bound of the SNR after beamforming is proportional to $1/(\mathbf{a}^T\mathbf{a})$, maximizing SNR is equivalent to minimizing $\mathbf{a}^T\mathbf{a}$.

In order to reduce sidelobes, the apodization window should be continuous and smooth. This is because any discontinuities or spikes in the apodization function introduces higher spatial frequency components, which translate to sidelobe energy. We find that the smoothness is inversely proportional to $\mathbf{a}^T\mathbf{V}^T\mathbf{V}\mathbf{a}$ where

$$
\mathbf{V} = \begin{bmatrix} \mathbf{D} & & & \\ & \mathbf{D} & & \\ & & \ddots & \\ & & & \mathbf{D} \end{bmatrix} \quad \text{and} \quad \mathbf{D} = \begin{bmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{bmatrix}
$$

Here $\mathbf{D}$ is of size $K \times K$, and $\mathbf{V}$ is of size $M \times M$. We choose $\mathbf{D}$ to have the above structure so it works like a high pass filter that can detect spikes or discontinuities in a subaperture window. Thus the smoothness should be inversely proportional to the energy of the filter outputs, which is calculated as follows.

$$
\sum_{i=1}^{L} \|\mathbf{D}\mathbf{a}_i\|_2^2 = \mathbf{a}^T\mathbf{V}^T\mathbf{V}\mathbf{a} \tag{3.1}
$$

Thus to maximize smoothness, we minimize $\mathbf{a}^T\mathbf{V}^T\mathbf{V}\mathbf{a}$. Hence the objective function can formulated by the weighted sum of the two terms as follows.

$$
\min_{\mathbf{a}} \quad \gamma\mathbf{a}^T\mathbf{a} + \mathbf{a}^T\mathbf{V}^T\mathbf{V}\mathbf{a} = \mathbf{a}^T\boldsymbol{\Phi}\mathbf{a}
$$

$$
\text{Subject to} \quad \mathbf{y} = \mathbf{G}\mathbf{a} \tag{3.2}
$$

where $\boldsymbol{\Phi} = \gamma\mathbf{I} + \mathbf{V}^T\mathbf{V}$ and γ is a coefficient to control the relative importance between the SNR performance and aperture smoothness. When $\gamma \geqslant 0$, $\boldsymbol{\Phi}$ is positive definite, and this quadratic optimization problem has an analytical solution given by

$$
\mathbf{a} = \boldsymbol{\Phi}^{-1}\mathbf{G}^T(\mathbf{G}\boldsymbol{\Phi}^{-1}\mathbf{G}^T)^{-1}\mathbf{y} \tag{3.3}
$$
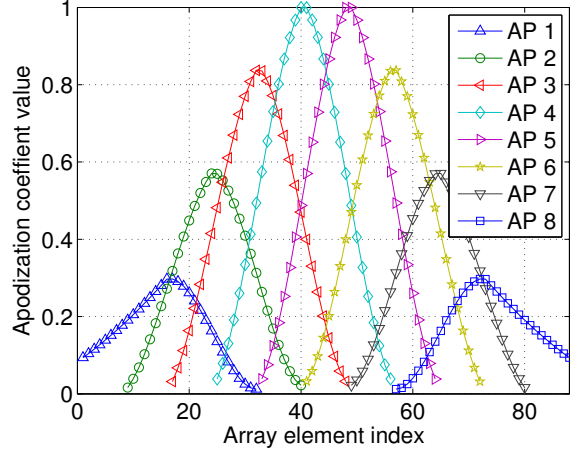
Figure 3.2: Subaperture Apodization using Blackman Window with $M = 88$, $K = 32$

Note that this method can be applied to any apodization window. Figure 3.2 shows the receive subaperture apodization coefficients generated by this method for Blackman window when the 1-D transducer array is of size 88 and subaperture is of size $K = 32$. There are 8 receive subapertures AP1 through AP8 with very different apodization coefficients.

2-D Separable Subaperture Apodization

For the 2-D transducer array, we decompose the 2-D subaperture apodization design problem into two 1-D problems. The method outlined in Section 3.2.2 is now used to derive the coefficients in each of the dimensions. The 2-D subaperture apodization window is the product of two 1-D apodization windows as represented by $A_{i,j}(n_x, n_y) = a_i(n_x) \cdot a_j(n_y)$. The 2-D apodization generated by the proposed method is shown in Figure 3.3. Note that, although the figure presents apodization windows separately, actually the adjacent windows are overlapping with each other as illustrated in Figure 3.1.
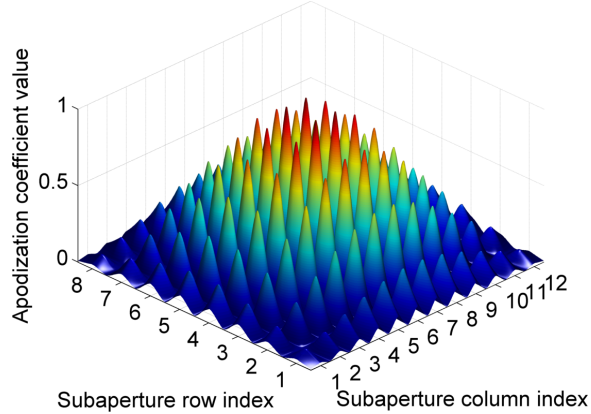
Figure 3.3: Subaperture Apodization using Kaiser Window with $M = 88$, $K = 32$

3.2.3   Iterative Index Calculation

Delay calculation for beamforming is enormously compute-intensive, requiring either numerous processors or a large LUT of pre-calculated values. Neither of these approaches is feasible in a small hand-held device. Prior work has reduced delay calculation computational complexity through iterative methods [42, 43], but these methods still require billions of expensive square root operations. Instead, we re-design the algorithm to require tractable storage and eliminate the trigonometric and square root operations required in a straight-forward implementation.

The key insight of the proposed algorithm is to replace prior iterative index calculations [42] with a piece-wise quadratic approximation that can be computed using only add operations. Because focal points are evenly spaced, the delta function between adjacent focal point delays form a smooth curve and indices can be approximated accurately (with error similar to that introduced by interpolation) over short intervals with quadratic approximations. We replace these exact delta curves with a per-transducer pre-computed piece-wise quadratic approximation constrained to allow an index error of at most 3 (corresponding to at most 30μm error between the estimated and exact focal point) thus resulting in negligible blur. Figure 3.4a compares

40
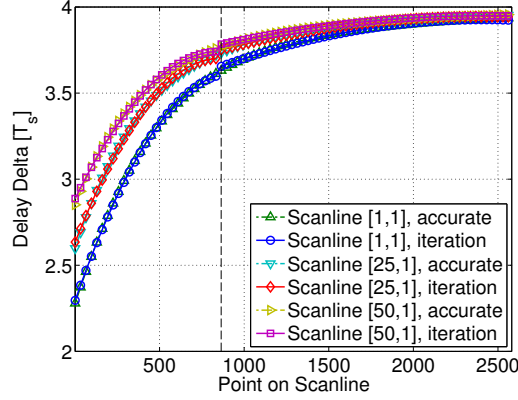
(a) The Exact Delta vs. Estimates from Our Iterative Algorithm
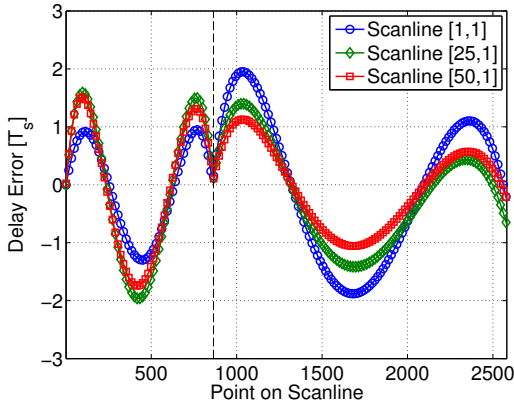
(b) The Error between Our Approximation and The Exact Delta
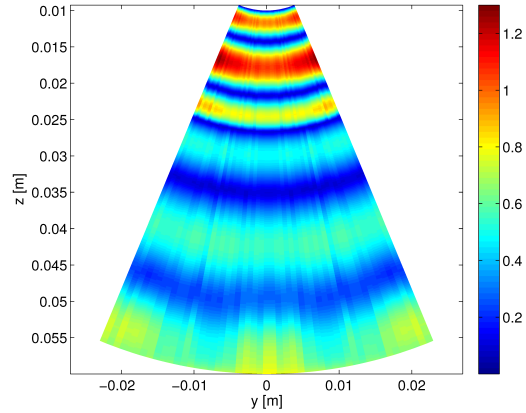
(c) Root Mean Square Error For an Entire y-z Image Slice

Figure 3.4: Delay Curve Fitting to Facilitate Iterative Index Calculation

our approximation to the exact difference between adjacent delays for three representative scanlines. The dotted line indicates the boundary of the 2-section piecewise approximation. Figure 3.4b shows the corresponding round-trip delay error. Figure 3.4c shows the root mean square (RMS) error for the full y-z slice through the middle of the image. Our approach drastically reduces storage requirements relative to pre-computing all delays because only four constants (three for the delta function; one for the section boundary) are pre-computed and stored per section. Because of its simplicity, this approximation requires only table lookups (to retrieve constants) and adds (to iteratively calculate the delay).

Through quality studies, we have determined that our piece-wise quadratic estimation method requires only two sections to meet our accuracy target for the typical abdominal imaging aperture used in our evaluation. Hence, we must store at most nine (two sections of four constants and a start value) constants per scanline. In contrast, complete delay pre-computation requires a 4096-entry LUT per scanline, over $400\times$ more storage than our method. By exploiting symmetry in the imaging geometry, we can further reduce the required constant storage by a factor of four, for an aggregate storage requirement of 77MB. Nevertheless, 77MB exceeds the storage capacity we can provision on chip. However, our system only uses 250kB of constants at a time, allowing us to load the constants as needed from DRAM.

3.3   Hardware Architecture

We next describe the Sonic Millip3De system architecture and its key features, including the beamforming accelerator that implements our iterative delay calculation algorithm in a massively parallel array.

3.3.1   System Architecture

The Sonic Millip3De system (Figure 3.5) is comprised of three stacked silicon dies (transducers and analog electronics, ADC and storage, and computation) connected vertically using through-silicon vias (TSVs) and off-stack LPDDR2 memory. These components are integrated in the ultrasound scanhead, the wand-like device a radiologist manipulates to obtain ultrasound images.

Using 3-D die-stacked design provides several architectural benefits. First, it is possible to stack dies manufactured in different technologies. Hence, the transducer layer can be manufactured in a cost-effective process for the analog circuitry, higher voltages, and large geometry of ultrasonic transducers, while the beamforming acceler-

42
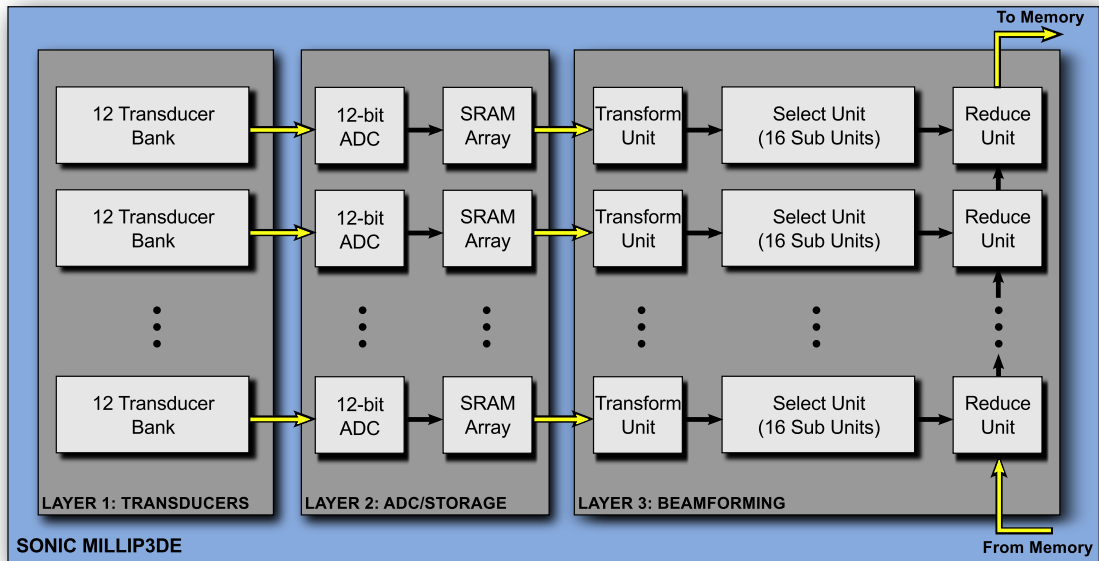
Figure 3.5: Sonic Millip3De Hardware Overview

ator can exploit the latest digital logic process technology. Second, ultrasound systems typically feature interchangeable scan heads for different imaging tasks (e.g., different imaging depths and resolutions). By separating the transducer array, ADC/storage, and computation engine into separate dies, a standard interface (i.e., TSV layout) between each enables dies to be reused with varying transducer array layers, reducing design costs. Finally, as in recent 3-D-stacked processor architectures where caches and cores are connected vertically [37], the face-to-face connections between SRAM arrays and corresponding computation units avoid the need for long wires.

The transducer die comprises an array of 120×88 transducers with $\lambda/2$ spacing. The area between transducers contains the analog electronics and routing to the TSV interface to the ADC/storage die. Transducers within a bank are multiplexed onto a single signal per bank that is passed over a TSV to the ADC/storage layer for digitization.

The ADC/storage layer comprises 1024 12-bit ADCs, each connected to an incoming analog signal from the transducer layer. The ADCs sample at a frequency of 40MHz. This sampling frequency balances energy efficiency and flexibility for ultrasound applications requiring varying transmit frequencies. After digitization, the received signals are stored in 1024 independent SRAM arrays, each storing 4096 12-bit samples. The SRAMs are clocked at 1GHz. Each SRAM array is connected vertically to a corresponding functional unit on the computation layer.

The computation layer includes the beamforming accelerator units, a unidirectional pipelined interconnect, a control processor (e.g., an M-class ARM core), and an LPDDR2 memory controller. The die area is dominated by the beamforming accelerator array and interconnect, which are described in the following subsections. The control processor manages memory transfers from the LPDDR2 interface to the accelerator array, controls the transducer array, and performs other general purpose functions. The off-stack LPDDR2 memory stores index delay constants and a frame buffer for the final volumetric image.

The Sonic Millip3De memory system comprises a 192-bit wide memory channel striped across 6 2Gb x16 LPDDR2-800 parts. This unusual arrangement matches the width of our on-chip interconnect, provides sufficient capacity (1.5 GB) and sufficient memory bandwidth (38.4 GB/sec) to load beamforming constants (requiring 6.2 GB/sec) and read/write image data (requiring 5.5 GB/sec) for the frame rate of 2 frames per second while still requiring little power [44, 45].

Accelerator Overview

The beamforming accelerator is the central element of Sonic Millip3De, and is the key
to achieving our performance and power objectives. The accelerator relies on massive
parallelism (1024 beamforming units operate in concert) and achieves energy efficiency
through carefully optimized 12-bit data paths that perform only add, compare, and
table lookup operations.

Recall (from Section 3.2.1) that a single ultrasound frame is obtained by summing
the received data from 96 receive subapertures. For each of these 96 receive opera-
tions, the entire imaging volume is read from memory (15MB), the (single) correct
sample from each transducer in the sub-aperture is added to each focal point, and
the volume is stored back to DRAM. Below, we describe a single of these 96 receive
operations: the data flow during each receive is identical, only the apodization and
delay constants differ across receives.

The accelerator follows a streaming *transform-select-reduce* data flow paradigm.
We first describe the principle of this approach and how the algorithm described in
Section 3.2 maps to this framework.

Principle of Operation

As shown in Figure 3.5, the accelerator streams data in parallel from all 1024 SRAM
arrays on the ADC/storage layer (corresponding to the 1024 transducer banks) to
1024 corresponding beamforming units. The data streams pass through three con-
ceptual stages: *transform*, *select*, and *reduce*. Each stage is implemented in a separate
pipelined functional unit and a unidirectional pipeline interconnect (starting and ter-
minating at the LPDDR2 interface) links the reduce units together.
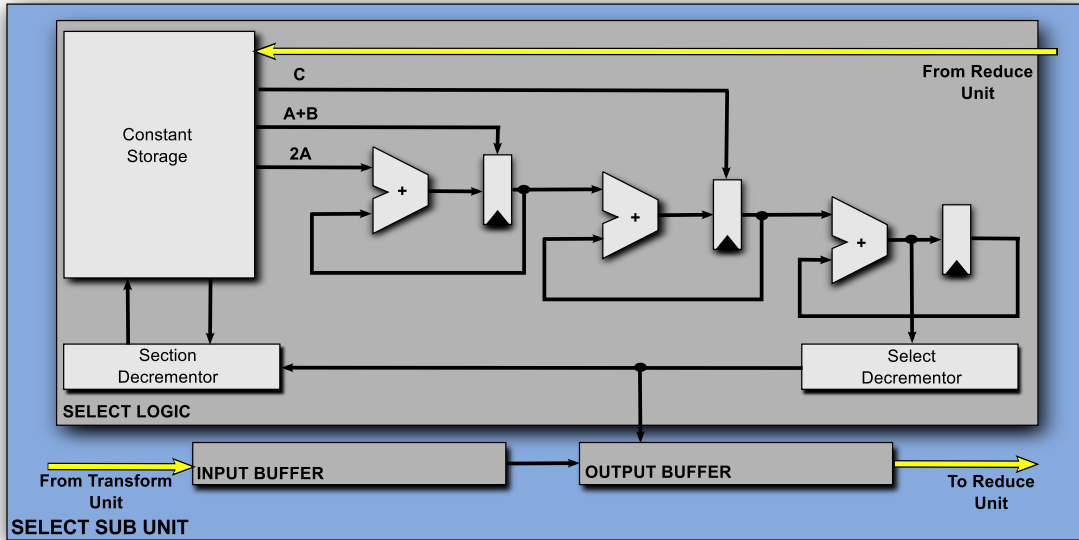
Figure 3.6: Select Unit Microarchitecture

In the conceptual model, the *transform* stage performs pre-processing that must be applied to all data. For beamforming, this stage performs interpolation (which upsamples the 40MHz signal to 160MHz).

The *select* stage transforms data from the receive time domain into the image space domain as shown in Figure 3.6. Select units map incoming samples from the receive time domain to image focal points. Sample data arrives from the interpolation unit at the input buffer, and each sample is either discarded or copied to the output buffer to accumulate a particular focal point. The unit selects the correct sample for each focal point using the indexing algorithm in Section 3.2.3. The key innovation of the select stage is that we have eliminated the index-load-add-store sequence that software implementations use to map from the time domain to the image space, thus significantly reduces the number of times data must be read from the SRAM.

Finally, the *reduce* stage performs a reduction operation across the *transform-select* pipelines. For beamsum, the reduction operation is simply an add. The main feature of the *reduce* stage is the interconnect that links the pipelines. In cases where

46

Table 3.1: 3-D Ultrasound System Parameters

Parameter	Value
Total Transmits per Frame	96
Total Transducers	10,560
Receive Transducers per Sub-aperture	1024
Storage per Receive Transducer	4096 x 12-bits
Focal Points per Scanline	4096
Image Depth	10cm
Image Total Angular Width	$\pi/6$
Sampling Frequency	40MHz
Interpolation Factor	4x
Interpolated Sampling Frequency	160MHz
Speed of Sound (tissue)	1540m/s
Target Frame Rate	2fps

the reduce operation is commutative, this could be an arbitrary interconnect. In our design, we use a unidirectional pipeline with wide, short links forming a 1024-stage pipeline.

3.4 Simulation Results

3.4.1 Subaperture Processing and Apodization Simulation Results

Figure 3.7 plots the radiation pattern due to a point object located at 3cm depth using Field II [46,47]. The 2-D subaperture configuration is as described in Figure 3.1. Compared to the method using 2-D Blackman window on each subaperture, the
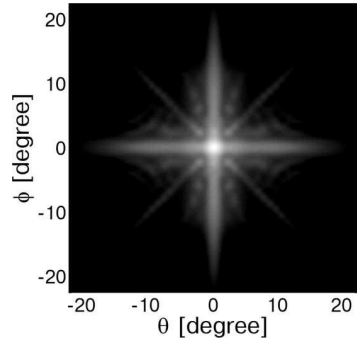
(a) Using Blackman Window on Each Subaperture
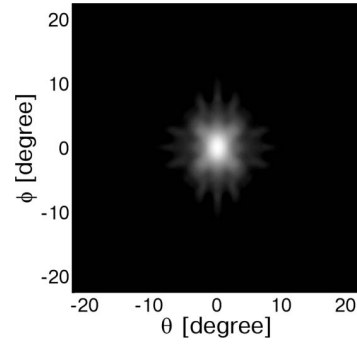
(b) Using Proposed Subaperture Apodization Scheme Results in Lower Sidelobes

Figure 3.7: Radiation Pattern due to a Point Object, 60 dB Display Dynamic Range
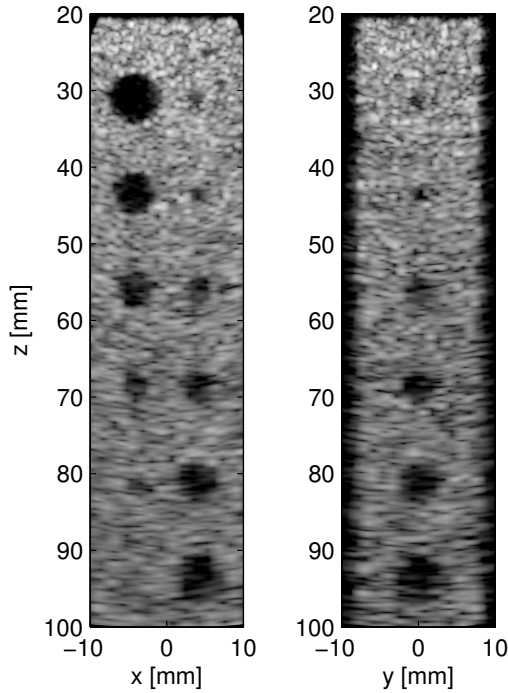
(a) With Plain Apodization Window
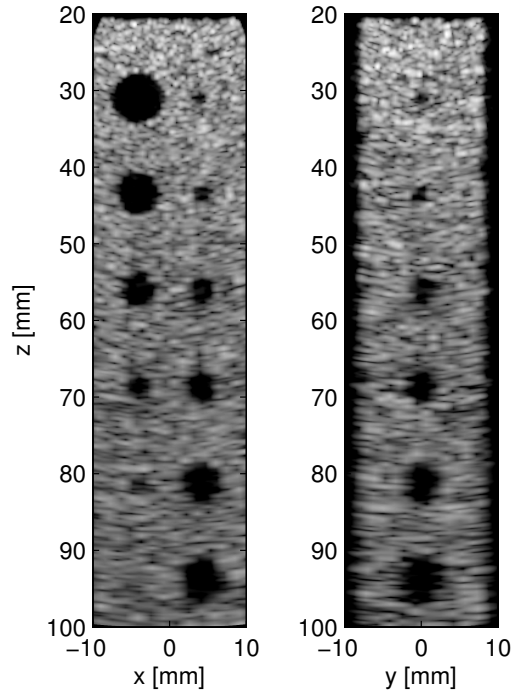
(b) With Optimized Apodization Window

Figure 3.8: 2-D Slice of 3-D Volume for Plain and Optimized Apodization, 40 dB Display Dynamic Range

radiation pattern of the proposed decomposable 2-D subaperture apodization is much smaller, and the sidelobes are under -60 dB.

48

We use Field II [46, 47] and MATLAB to simulate a 3-D imaging system with parameters listed in Table 3.1. The simulation case consists of 10 spherical anechoic cysts in a 20mm×15mm×80mm volume under the transducer array. The diameter of the cysts range from 2mm to 7mm. To verify the proposed apodization coefficient design scheme, we compare 2-D slices of 3-D volume generated by the subaperture based SAU imaging system with plain apodization and optimized apodization. In the plain apodization case, all subapertures employ an identical Kaiser window. For the proposed method, we generate subaperture apodization coefficients according to the target equivalent aperture apodization that is a large 2-D separable Kaiser window. The 2-D x-z plane slices and y-z plane slices of the two cases are compared in Figure 10. Images produced by plain apodization window have more artifacts inside the cysts which not only blur the edge of the cyst but also reduce the average Contrast-to-Noise Ratio (CNR) to 1.6. On the other hand, images generated with the proposed apodization optimization method is clearer and improved the average CNR on all ten cysts from 1.6 to 2.0. The CNR is calculated according to

$$\mathrm{CNR} = \frac{|\mu_{\mathrm{cyst}} - \mu_{\mathrm{bgnd}}|}{\sqrt{\sigma_{\mathrm{cyst}}^2 + \sigma_{\mathrm{bgnd}}^2}} \tag{3.4}$$

where μ_{cyst} and mean μ_{bgnd} correspond to brightness of cyst and background, while σ_{cyst} and σ_{bgnd} correspond to the standard deviation of cyst and background. The pixels inside the sphere whose radius is 80% the radius of the cyst are used to calculate μ_{cyst} and σ_{cyst}, while pixels outside the sphere whose radius is 120% the radius of the cyst are used to calculate μ_{bgnd} and σ_{bgnd}. Note that the noise here is speckle noise which is different from the thermal noise used in the definition of SNR.

Table 3.2: CNR Values for Both Ideal System and Sonic Millip3De v1.0

Left		Right	
Ideal	SM3D	Ideal	SM3D
3.59	3.58	1.93	1.85
3.18	3.21	1.51	1.41
2.68	2.67	1.94	1.85
1.61	1.62	2.10	2.01
1.10	1.18	2.39	2.30
0.33	0.39	2.43	2.34

3.4.2 Fixed-point Simulation Results

We contrast CNR of simulated cysts in tissue for images generated using an ideal system (precise index calculation and double-precision floating-point) against our new Sonic Millip3De design, which uses 14-bit fixed-point beamsum, iterative delay calculation, and dynamic focus. An x-z slice through the middle of the cysts is shown in Figure 3.9 for both the ideal case and our design. Table 3.2 shows a CNR breakdown for all cysts for both configurations. Neither design is effective in resolving the smallest (2mm) cyst at depth, but Sonic Millip3De's image quality is nearly indistinguishable from the ideal case, providing high image quality at all depths for the larger cysts.

3.4.3 Power Analysis and Scaling

To evaluate full system power computation, we use a combination of RTL-level synthesis for the beamformer, SRAM, and interconnect and published estimates [44, 48, 49] for other system components. The beamformer along with the interconnect consumes 8.4W and accounts for 58% of the total system power. The next largest

(a)   Double-Precision Floating Point and Exact Delay Index Calculation

(b) Fixed-Point Precision, and Dynamic Focus and Iterative Delay Calculation
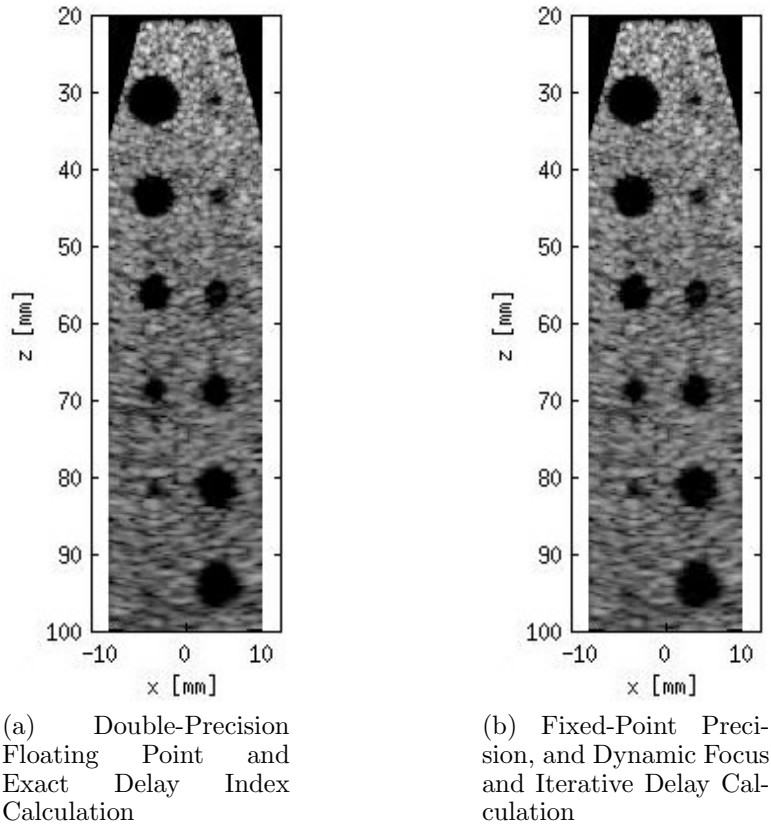
Figure 3.9: Image Quality Comparison, Floating-Point vs. Fixed-Point

power consuming unit is DRAM, which consumes 4.4W and accounts for 30% of the total power. The front-end that includes ADCs and transducers consumes 1.5W and accounts for 10% of the total system power. We determine that our design requires a full system power of 14.6W in 45nm technology (Figure 3.10). Using published scaling trends for ADCs [50] and CMOS logic [51], we project that this design will achieve a 5W power budget (our target for safe contact with human skin) by the 16nm node.
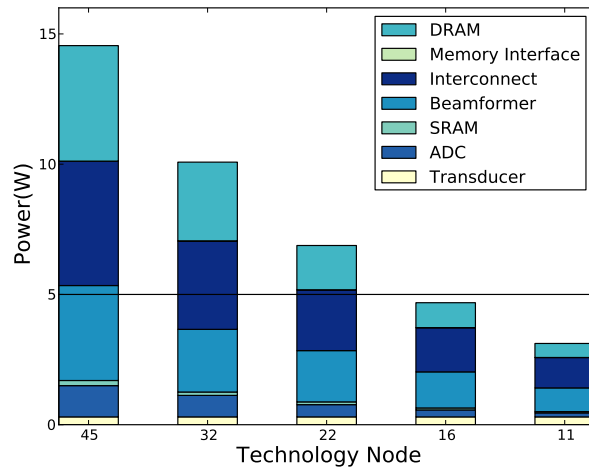
Figure 3.10: Power Breakdown of Sonic Millip3De v1.0 Across Technology Nodes

Chapter 4

SEPARABLE BEAMFORMING FOR 3-D SYNTHETIC APERTURE ULTRASOUND SYSTEMS

4.1 Introduction

Although the optimized subaperture apodization and firing scheme presented in Chapter 3 reduces the number of firings by a factor of 2 and thereby reduces computation complexity by a factor of 2, the existing design still falls short of desirable power targets (e.g., 5W for safe contact with human skin), calling for further reductions in front-end processing requirements.

Consider a 3-D subaperture-based SAU system shown in Table 4.1. To generate a 3-D image of size $10\text{cm} \times 45° \times 45°$, 1.1×10^{12} delay-sum operations must be performed per frame. Such high computational complexity results in correspondingly high power consumption and can limit hand-held devices for 3-D ultrasound imaging.

To support 3-D ultrasound imaging with reduced computation complexity, 2-D sparse array designs have been studied in [52–54]. Although sparse array based systems employ fewer transducer elements than traditional 2-D arrays, such systems usually have high sidelobe levels and suffer from low SNR. Alternatively, the computation complexity of digital front end can be greatly reduced without compromising the image quality by a promising method called separable beamforming. Separable beamforming reduces computational complexity by decomposing 2-D array beamforming into a series of 1-D array operations.

Separable beamforming was first introduced in [55,56] for frequency domain beamforming in sonar applications and later to ultrasound imaging [57]. Although amenable

Table 4.1: System Parameters of 3-D SAU System

Property	Value
Pitch, μm	192.5
Array size, element	120×88
Subaperture size, element	32×32
Number of scanlines	48×48
View angle, square degree	$45° \times 45°$
Max depth, cm	10
Center frequency, MHz	4
6 dB transducer bandwidth, MHz	2
A/D sampling rate, MHz	40

to parallelization [57], prior delay decomposition methods incur large error for small f-number, resulting in poor resolution. A later separable beamforming proposal [58] improved delay decomposition error at the cost of only using vertical scanlines, limiting the approach to plane-wave systems with rectangular views that are too narrow to image large anatomies.

In this chapter, a new separable beamforming method that is not restricted to any specific scan system is presented. It is based on a delay decomposition method that minimizes the root-mean-square (RMS) phase error introduced by the separable delay approximation. This method can be used to achieve separable beamforming in subaperture based 3-D synthetic aperture ultrasound (SAU) systems with 19-fold reduction in computation complexity compared to the non-separable method. The separable delays obtained by this method can be stored in large look-up tables and directly used for delay calculations. However, to reduce look-up table storage requirements, we propose a method to iteratively compute these delays at runtime using far

fewer pre-computed constants. We demonstrate, using Field II simulations of cyst images, that our proposed separable beamforming method achieves almost identical image quality compared to the non-separable baseline method.

We also briefly describe the architectural modifications that are required to support 2-stage separable beamforming on Sonic Millip3De. These modifications include additional SRAM storage and reorganization of the 1,024 pipelines into 32 clusters with 32 nodes per cluster to facilitate the 2-stage beamforming. The modified Sonic Millip3De architecture implements separable beamforming with $19\times$ reduction in computational complexity. This reduction can be leveraged either to reduce hardware requirements (by scaling down the number of parallel pipelines in the design), increase frame rate (by operating the existing design at higher throughput), or save power (by scaling down frequency to maintain current frame rates).

This work on separable beamforming method for SAU systems appeared in [14,15].

4.2   Algorithm Design For Separable Beamforming

4.2.1   Separable Beamforming Process
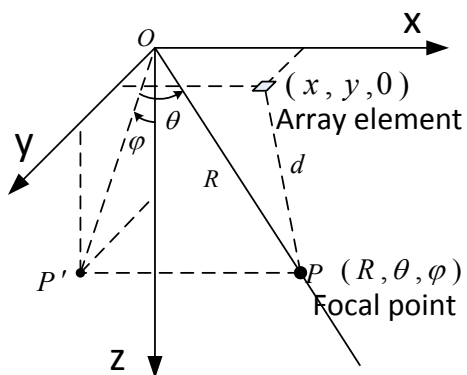
Figure 4.1: Array and Beamforming Coordinate System for SAU Imaging

55

Without loss of generality, we assume the 3-D coordinate system shown in Figure 4.1. Let (R, θ, ϕ) be the coordinates of a focal point P. Here R is the radial distance from the origin O to point P. Point P' is the orthogonal projection of P in the yz plane. ϕ is the elevation angle between line OP' and the z axis. θ is the azimuth angle between OP and its orthogonal projection OP' in the yz plane. For a transducer array element at $(x, y, 0)$, the distance between the transducer element and the focal point P is given by

$$d_{\text{rx}} = \sqrt{R^2 + x^2 - 2Rx\sin(\theta) + y^2 - 2Ry\cos(\theta)\sin(\phi)} \qquad (4.1)$$

Similarly the distance between the firing virtual source located at (x_v, y_v, z_v) and the focal point P is given by

$$d_{\text{tx}} = \sqrt{R^2 + x_v^2 + y_v^2 + z_v^2 - 2x_v R\sin\theta - 2Ry_v\cos\theta\sin\phi - 2Rz_v\cos\theta\cos\phi} \qquad (4.2)$$

Assuming that the ultrasound speed is c, and the round-trip delay between the origin and the focal point is $2R/c$, the round-trip delay at the transducer relative to that at the origin is given by

$$\tau(x, y, R, \theta, \phi) = (2R - d_{\text{tx}} - d_{\text{rx}})/c \qquad (4.3)$$

Let $\tau(n_x, n_y, m_R, m_\theta, m_\phi)$ be the discrete form of $\tau(x, y, R, \theta, \phi)$, where n_x and n_y are variables associated with the coordinates of receive elements, and m_R, m_θ and m_ϕ are variables associated with the coordinates of focal points. Then the non-separable beamforming corresponding to subaperture l of size $N_x \times N_y$ whose left corner indices are i_l and j_l, is described as

$$F_l(m_R, m_\theta, m_\phi; t) = \sum_{n_x=i_l}^{i_l+N_x-1} \sum_{n_y=j_l}^{j_l+N_y-1} A_l(n_x, n_y) \cdot S_l(n_x, n_y, t - \tau(n_x, n_y, m_R, m_\theta, m_\phi)) \qquad (4.4)$$

where $S_l(n_x, n_y, t)$ is the signal received by transducer element (n_x, n_y) at lth firing and $A_l(n_x, n_y)$ is the corresponding apodization coefficient. $F_l(m_R, m_\theta, m_\phi; t)$ is the low resolution 3-D image generated by subaperture l. F_l should be sampled at $t = 2R/c$ for dynamic focusing. For a synthetic aperture ultrasound system, the final high resolution image is obtained by summing all the low resolution images from all subapertures.

Now, if $\tau(n_x, n_y, m_R, m_\theta, m_\phi)$ can be decomposed as

$$\tau(n_x, n_y, m_R, m_\theta, m_\phi) = \tau_1(n_x, n_y, m_R, m_\theta) + \tau_2(n_y, m_R, m_\theta, m_\phi) \qquad (4.5)$$

then equation (4.4) can be represented by a two-stage separable beamforming process:

$$F_l^{(1)}(n_y, m_R, m_\theta; t) = \sum_{n_x=i_l}^{i_l+N_x-1} A_l(n_x, n_y) S_l(n_x, n_y, t - \tau_1(n_x, n_y, m_R, m_\theta)) \qquad (4.6)$$

$$F_l^{(2)}(m_R, m_\theta, m_\phi; t) = \sum_{n_y=j_l}^{j_l+N_y-1} F_l^{(1)}(n_y, m_R, m_\theta; t - \tau_2(n_y, m_R, m_\theta, m_\phi)) \qquad (4.7)$$

In the first stage, the beamforming is along the x axis, which functions as a spatial filter that steers the receive plane to azimuth angle θ. The process repeats for all combinations of m_R, n_y and m_θ and results in a partially beamformed intermediate signal $F_l^{(1)}$. In the second stage, 1-D beamforming is performed along the y axis, and corresponds to steering receive plane to elevation angle ϕ. The second stage beamforming is repeated for all combinations of m_R, m_θ and m_ϕ. The principle of the proposed separable beamforming method is shown in Figure 4.2.

The number of delay-sum operations of separable beamforming for one subaperture is $N_x N_y M_R M_\theta + N_y M_R M_\theta M_\phi$ in contrast to $N_x N_y M_R M_\theta M_\phi$ in conventional, non-separable beamforming. Thus, the computational complexity reduction is $N_x M_\phi/(N_x + M_\phi)$. For the configuration shown in Table 4.1 with a 32×32 subaperture size and 48×48 scanlines, our approach achieves about $19\times$ complexity reduction.
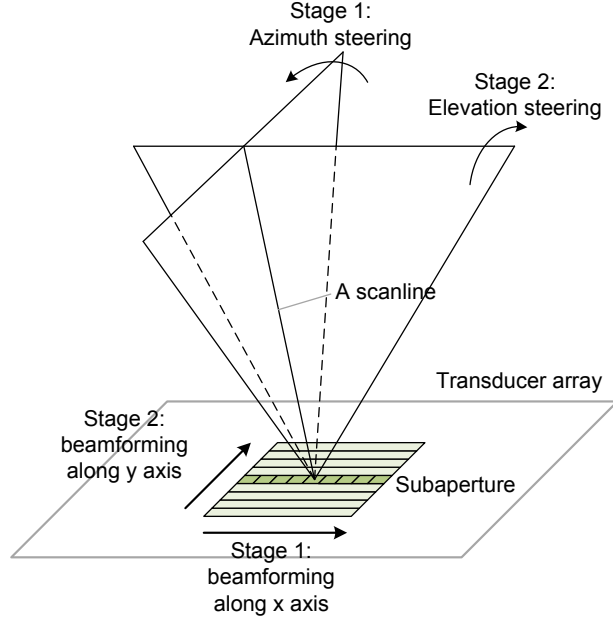
Figure 4.2: The Principle of Separable Beamforming

The separable beamforming method is based on the assumption that the wave propagation delay τ can be decomposed into τ_1 and τ_2. However the decomposition is not exact and its effectiveness depends on the accuracy of the τ_1 and τ_2 approximations. Next, we describe the proposed decomposition method, which designs τ_1 and τ_2 to minimize RMS phase error.

4.2.2   Delay Decomposition Method

The beamsum delay τ, which is a function of five variables, cannot be strictly decomposed into a sum of two functions with fewer variables because the distance calculation involves a square root operation (as shown in equation (4.1) and (4.2)). The Taylor series of the square root includes functions involving multiple variables, also referred to as cross terms, which can not be easily decomposed. To make the delay equation separable, some of these cross terms must be dropped. Although the effect of the cross terms diminish with large R, for small depths, the cross terms in

the delay calculation can be significant. There are several factors that affect the error incurred by the delay decomposition: the number of variables in each decomposed function, the combination of variables, and the sequence of computations in the two-stage beamforming.

To build foundation for our proposed decomposition, we first consider a simpler strawman wherein τ_1 and τ_2 are each functions of three variables and the delay decomposition is given by

$$\tau(x, y, R, \theta, \phi) \approx \tau_1(x, R, \theta) + \tau_2(y, R, \phi) \tag{4.8}$$

We justify this first decomposition as follows. For dynamic focusing, both τ_1 and τ_2 depend on R, thus R should be included in the variable list of both τ_1 and τ_2. τ_1 is also a function of θ and x because 1-D beamforming along the x direction allows the array system to filter out signals from all azimuth directions except those with azimuth angle θ. Hence θ and x should not be separated. Similarly, 1-D beamforming along the y direction allows the array system to filter out signals from all elevation directions except those whose elevation angle is ϕ, so ϕ and y should not be separated. In this decomposition, since (x, θ) and (y, ϕ) are interchangeable, either τ_1 or τ_2 can be used in the first stage of beamforming. Unfortunately, this simple decomposition approach leads to large errors primarily because θ and ϕ are separated and the cross terms involving θ and ϕ are lost.

To improve the approximation, τ_1 and τ_2 must capture additional important dependencies. To simplify our presentation, let us assume that we perform first-stage beamforming along the x axis while the second stage is along the y axis. For τ_1, consider adding ϕ or y to the variable list already consisting of R, x and θ. There is no benefit in adding ϕ because 1-D beamforming in the first stage is along the x direction, and does not have enough resolution along ϕ. However there is an advantage

of adding y to the variable list of τ_1, because τ_1 already includes x and thus adding y helps to retain cross terms between x and y. For τ_2, there are two candidate variables that can be added to the variable list, namely x and ϕ. There is no benefit of adding x to τ_2's variable list because neither the input signal of second-stage beamforming $F^{(1)}$ nor output of second stage beamforming $F^{(2)}$ relates to x. However, adding θ to τ_2 has the advantage of preserving cross terms involving θ and ϕ. Thus, both τ_1 and τ_2 are best represented as functions of four variables and our final decomposition is of the form

$$\tau(x, y, R, \theta, \phi) \approx \tau_1(x, y, R, \theta) + \tau_2(y, R, \theta, \phi) \tag{4.9}$$

Alternatively, if the first beamforming stage is performed along the y axis, the candidate decomposition is instead

$$\tau(x, y, R, \theta, \phi) \approx \tau_1(x, y, R, \phi) + \tau_2(x, R, \theta, \phi) \tag{4.10}$$

Compared to the strawman (Eq. (4.8)), τ_1 includes ϕ because beamforming along the y axis allows the system to distinguish signals coming from different elevation angles ϕ.
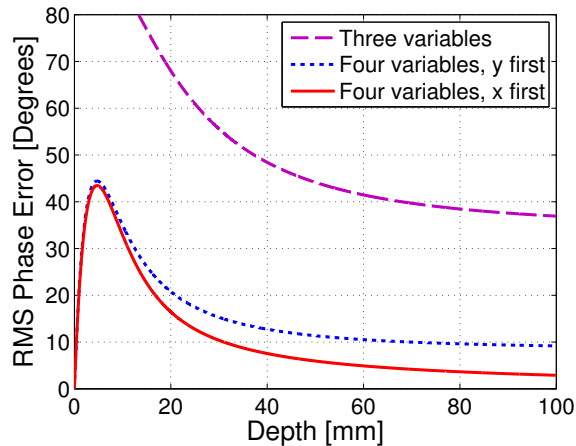

Figure 4.3: RMS Phase Errors for Three Types of Decomposition

Figure 4.3 compares the RMS errors for the three alternative decompositions defined in (4.8), (4.9) and (4.10). Each decomposition is obtained by setting up an RMSE minimization problem and is solved using the Euler-Lagrange equation. By increasing the number of variables of τ_1 and τ_2 from three to four, the approximation error is reduced by one decade for relatively large depth. As depth decreases towards zero, the error of three-variable decomposition increases rapidly, while the error of four-variable decomposition approaches zero. This behavior occurs because, as R approaches 0, the cross terms involving x and y become significant, and both the four-variable decomposition forms retain these cross terms in τ_1.

Also note that beamforming along x first results in lower RMSE. We find this ordering is better because the RMSE of approximation (4.9) is primarily due to dropping cross terms involving x and ϕ, which is less than the RMSE caused by dropping cross terms involving y and θ in approximation (4.10). The cross term involving y and θ is an artifact of the coordinate system; the distance calculation (Eq. (4.1)) contains a natural cross term involving y and θ, namely $2Ry\cos(\theta)\sin(\phi)$. The proposed decomposition (Eq. (4.9)) results in delay error under $9°$ when depth is larger than 3cm, and reduces rapidly as depth increases.

4.2.3   Generating Delay Functions τ_1 and τ_2

Given this decomposition in Eq. 4.9, we must next generate $\tau_1(x, y, R, \theta)$ and $\tau_2(y, R, \theta, \phi)$ such that the error due to approximation is minimized. Minimizing RMSE is equivalent to minimizing

$$E = \int_{\phi_1}^{\phi_2} \int_{x_1}^{x_2} [\tau(x, y, R, \theta, \phi) - (\tau_1(x, y, R, \theta) + \tau_2(y, R, \theta, \phi))]^2 dx d\phi \qquad (4.11)$$

where x_1 and x_2 are the lateral boundary of the corresponding subaperture, and ϕ_1 and ϕ_2 are the range of the elevation angle—a classic calculus of variation problem. By solving the corresponding Euler-Lagrange equations [59] and discretizing τ_1 and τ_2, we get the following solution that minimizes RMSE:

$$\tau_1(n_x, n_y, m_R, m_\theta) = \frac{1}{M_\phi} \sum_{m_\phi=1}^{M_\phi} \tau(n_x, n_y, m_R, m_\theta, m_\phi) - \rho(n_y, m_R, m_\theta) \qquad (4.12)$$

$$\tau_2(n_y, m_R, m_\theta, m_\phi) = \frac{1}{N_x} \sum_{n_x=i_l}^{i_l+N_x-1} \tau(n_x, n_y, m_R, m_\theta, m_\phi) - \rho(n_y, m_R, m_\theta) \qquad (4.13)$$

$$\rho(n_y, m_R, m_\theta) = \frac{1}{2N_x M_\phi} \sum_{n_x=i_l}^{i_l+N_x-1} \sum_{m_\phi=1}^{M_\phi} \tau(n_x, n_y, m_R, m_\theta, m_\phi) \qquad (4.14)$$

Note that equations (4.12) and (4.13) are not the only form that minimizes RMSE. Assuming we add an arbitrary term $\xi(n_y, m_R, m_\theta)$ on the right-hand side of (4.12) while subtracting it from the right-hand side of (4.13), the summation of τ_1 and τ_2 and the RMSE both remain the same. In this work, we choose $\xi(n_y, m_R, m_\theta) = 0$ so that the mean values of τ_1 and τ_2 are the same. From an architectural perspective, this formulation makes the delay line length or buffer depth roughly equal in the two beamforming stages.

4.2.4 Online Iterative Separable Delay Calculation

Next we focus on efficient calculation of τ_1 and τ_2. Equations (4.12) and (4.13) are straight forward, and can be used to generate look-up tables for τ_1 and τ_2. However, storing τ_1 and τ_2 as look-up tables is not practical due to the large size of look-up table. For our system configuration, look-up tables of τ_1 and τ_2 for 96 subapertures include at least (considering symmetry) 5.7 billion and 8.9 billion constants, respectively.

Fortunately, the delay values of consecutive samples on a scanline do not change much. Hence, it is possible to iteratively calculate the delay value for the ith focal
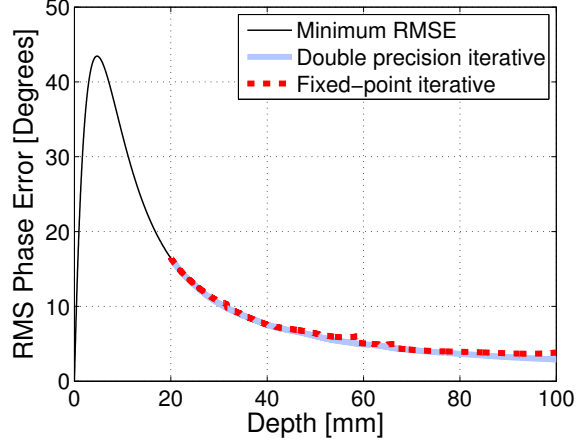
Figure 4.4: RMS Phase Errors using Iterative Delay Calculation

point on a scanline from the delay value of the $(i-1)$th focal point. We use piece-wise quadratic curves to approximate the delay difference between consecutive samples along a scanline. For example, let $\tilde{\tau}(m_R)$ be the delay corresponding to the m_Rth focal point for fixed n_x and n_y. Let $\eta(m_R) = \tilde{\tau}(m_R + 1) - \tilde{\tau}(m_R)$, then $\eta(m_R)$ can be approximated by $am_R^2 + bm_R + c$. Instead of storing the delay look-up table directly, the coefficients a, b and c and the initial delay are stored, and the delays are iteratively calculated using these coefficients. The iterative calculation method does not require multiplications, it can be implemented in a simple circuit using only three additions. We employ a similar iterative delay calculation for non-separable beamforming in [11].

To get an accurate approximation, each scanline is divided in to 2-4 sections and the delay in each section is approximated by a quadratic curve. For our system configuration, where the depth ranges from 2cm to 10cm, we cannot use a 2 section configuration since it results in significantly large approximation error. We choose a 3 section configuration over a 4 section configuration since it requires 23% lower storage with comparable approximation error.

The storage requirements of this method are as follows. Each section is character-ized by three constants and an initial point, and each scanline requires an additional

start index. Thus, each scanline requires 13 constants. A total of 38M constants must be stored; 15M constants are required for τ_1 and the remaining 23M for τ_2. The 15M constants for τ_1 correspond to 13 constants/scanline $\times$ 48 scanlines $\times$ 1,024 transducers/subaperture $\times$ 96 subapertures, divided by 4 due to symmetry (the delay term is symmetric in both x dimension and y dimension and so it is sufficient to store only 1/4 of the constants). The number of constants for τ_2 is calculated in a similar way. Each constant requires 12 bits on average [12], resulting an overall storage requirement of 55MB.

Figure 4.4 shows simulation results of this iterative delay calculation method with double-precision floating-point and 12-bit fixed-point precision. The iterative method with double precision floating-point coincides with minimum RMS curve, and fixed-point approximation only slightly increases RMSE.

4.3 Simulation Results

We evaluate image quality through simulated beamforming of cyst phantoms using Field II [46,47] and MATLAB. The simulation parameters are listed in Table 4.1. The system employs a 2-D transducer array comprising 120×88 transducer elements with a central frequency of 4MHz and 50% fractional bandwidth. The scan view is 45° in both elevation and azimuth angles. The maximum depth of view is 10cm.

4.3.1 Analysis of Delay Decomposition Error

The delay decomposition error arises due to dropping cross terms in our delay decomposition approximation. We simulate various configurations using MATLAB, and analyze the RMS phase errors for different subaperture sizes and wide view angles.
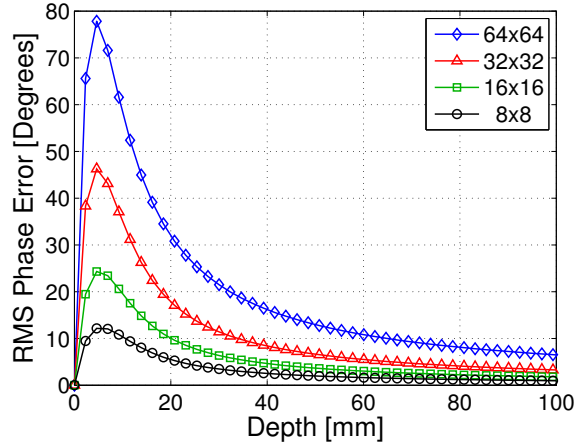
Figure 4.5: RMS Phase Errors for Different Subaperture Sizes
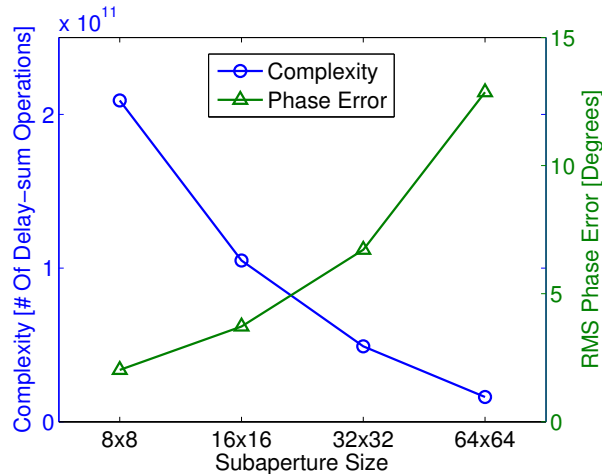
Figure 4.6: Complexity and RMS phase error as a function of subaperture size

As shown in Figure 4.5, RMS phase error is approximately proportional to subaperture size, but the error decreases rapidly as depth increases. For depths larger than 2cm, even for a subaperture size of 64×64, the error drops below 28°.

Although smaller subapertures result in less delay error, a smaller subaperture implies more firings to traverse the same transducer array, and correspondingly more computations, when compared to a larger subaperture. For instance a (16×16) configuration requires $4 \times$ more firings, resulting in a doubling of computational complexity and cutting the peak frame rate (due to transmit limits) by $4 \times$. On the other hand, the

65
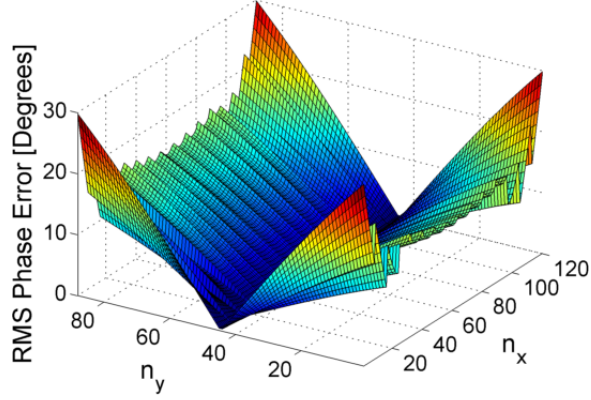
(a) RMS Phase Error Map for Different Transducers
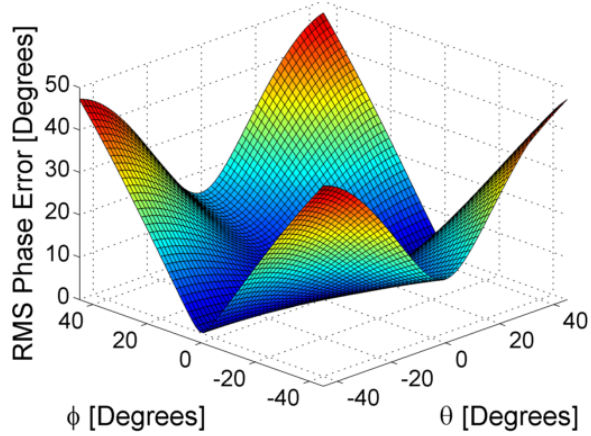
(b) RMS Phase Error Map for Different Angles

Figure 4.7: RMS Phase Error Maps

larger (64×64) configuration quadruples on-chip storage requirements and increases the number of parallel pipelines in the accelerator, increasing area requirements. The trade-off between delay error and beamforming complexity is shown in Figure 4.6. The beamforming complexity in this figure is reported as the number of delay-sum operations required to produce a complete frame with multiple firings. The RMSE is averaged over depths from 2cm to 10cm, for a $45° \times 45°$ angle view. We see that the (32×32) configuration has both low RMSE ($< 6°$) and low complexity, so we choose this configuration for our system.

Next, we present the RMS phase error map across the 120×88 transducer array for subaperture size of 32 × 32. The RMS errors are averaged across different depths (2-10cm), different angles (45° × 45° field of view) and 96 subapertures. The result is shown in Figure 4.7a. The RMS phase errors are mostly below 20°; the worst case is at the corners where RMS phase error reaches 30°.

We also present the RMS phase errors for different angles. Although our proposed system is for 45° × 45° field of view, in order to study the performance of the separable beamforming method for larger angles, we extend the RMS phase error map to 90° × 90°. The RMS phase errors are averaged across different depths (2-10cm), different transducers and different subapertures, and presented in Figure 4.7b. The highest error arises in the four corners where both θ and ϕ are large.

Ideally, the phase error of our configuration (4MHz central frequency and 160MHz sampling rate after 4× interpolation), should be ±9 degrees. In our system this can be achieved when the depth is larger than 3cm or the elevation angle is within ±10 degrees. Our simulation results for different depths and different angles are presented in Section 4.3.2. They show that the overall image quality of the separable system is good and comparable to that of the non-separable system.

4.3.2   Separable Beamforming

We evaluate image quality use Field II [46, 47] and MATLAB to simulate a 3-D imaging system with the parameters listed in Table 4.1. We consider two simulation cases, as illustrated in Figure 4.8. Both cases have twelve anechoic cysts located in a 20mm×15mm×80mm volume of random scatterers. The diameters of the cysts range from 2mm to 7mm. In Case A (Figure 4.8a), the volume containing cysts and scatterers is vertical, corresponding to $\theta = \phi = 0°$. In Case B (Figure 4.8b), the volume containing cysts and scatterers, with cysts located at $\theta = \phi = 30°$.

67

Consequently, the field of scan view is increased from $45° \times 45°$ to $90° \times 90°$, and the number of scanlines is increased from 48×48 to 96×96.

We quantify image quality via CNR and Contrast Ratio (CR). The CNR is defined in Eq. (3.4) and CR is defined as follows.

$$CR = \frac{\mu_{\text{bgnd}} - \mu_{\text{cyst}}}{\mu_{\text{bgnd}} + \mu_{\text{cyst}}} \tag{4.15}$$

where μ_{cyst} and μ_{bgnd} correspond to mean brightness of cyst and background.

The image quality of the 2-D slice images obtained in Case A by the baseline non-separable beamforming (shown in Fig. 4.9a) and our proposed separable beamforming method (shown in Fig. 4.9b) are nearly indistinguishable; both achieve an average CNR of 2.0 and an average CR of 0.55.

(a) Case A: Scatterers and Cysts in Upright Position

(b) Case B: Scatterers and Cysts Swung To $\theta = \phi = 30°$
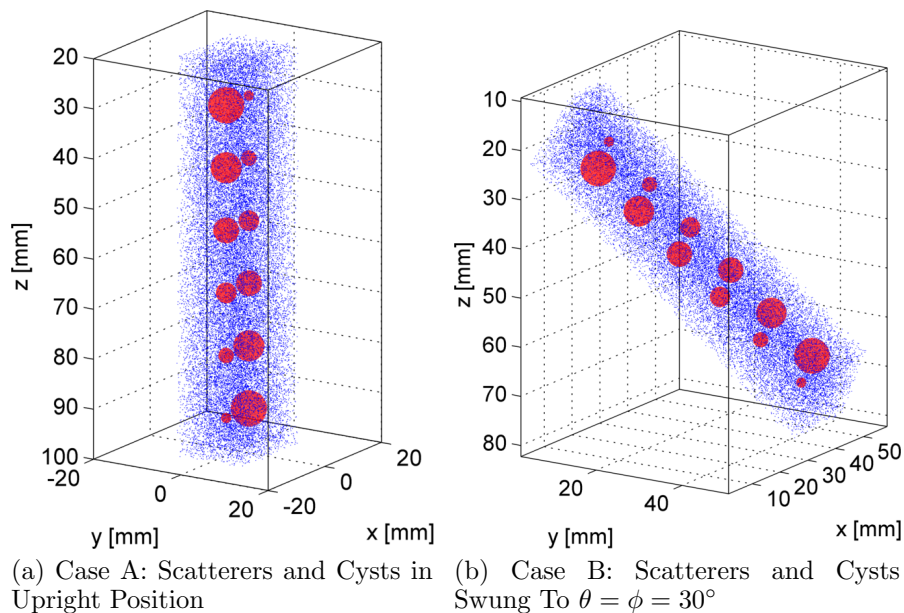
Figure 4.8: The Scatterer and Cyst Distribution for Two Field II Simulation Cases: 12 Anechoic Cysts with Diameters Ranging from 2mm to 7mm.

The 2-D slices of 3-D images obtained in Case B by non-separable beamforming and separable beamforming method are shown in Fig. 4.10a and Fig. 4.10b, respectively. We perform coordinate transformation and scan conversion in order to display

(a) Non-Separable Beamforming. Average   (b) Separable Beamforming. Average CNR of
CNR of 2.0, Average CR of 0.55   2.0, Average CR of 0.55

Figure 4.9: 2-D Slices of 3-D Simulation Images for 40 dB Dynamic Range. Case A: $\theta = \phi = 0°$.

the 2-D slices vertically. In these images, the vertical axis indicates depth R rather than the z axis coordinate. The images produced by non-separable method achieve an average CNR of 1.55 an average CR 0.55, while the images produced by separable method achieve an average CNR of 1.45 and an average CR of 0.55.

Finally, we confirm that the fixed-point performance of the proposed methods matches the quality of full double-precision floating point. We compare results of 12-bit and 14-bit separable beamforming in Fig. 4.11a and Fig. 4.11b. Both the 14-bit and 12-bit implementation achieve the same average CNR of 2.0, as in the double-precision-floating-point separable beamforming, although the 12-bit implementation has a slightly lower average CR compared to the 14-bit implementation (0.54 vs 0.55). Compared to the 14-bit nonseparable beamforming suggested in [12], 12-bit

(a) Non-Separable Beamforming. Average CNR of 1.55, Average CR of 0.55

(b) Separable Beamforming. Average CNR of 1.45, Average CR of 0.55
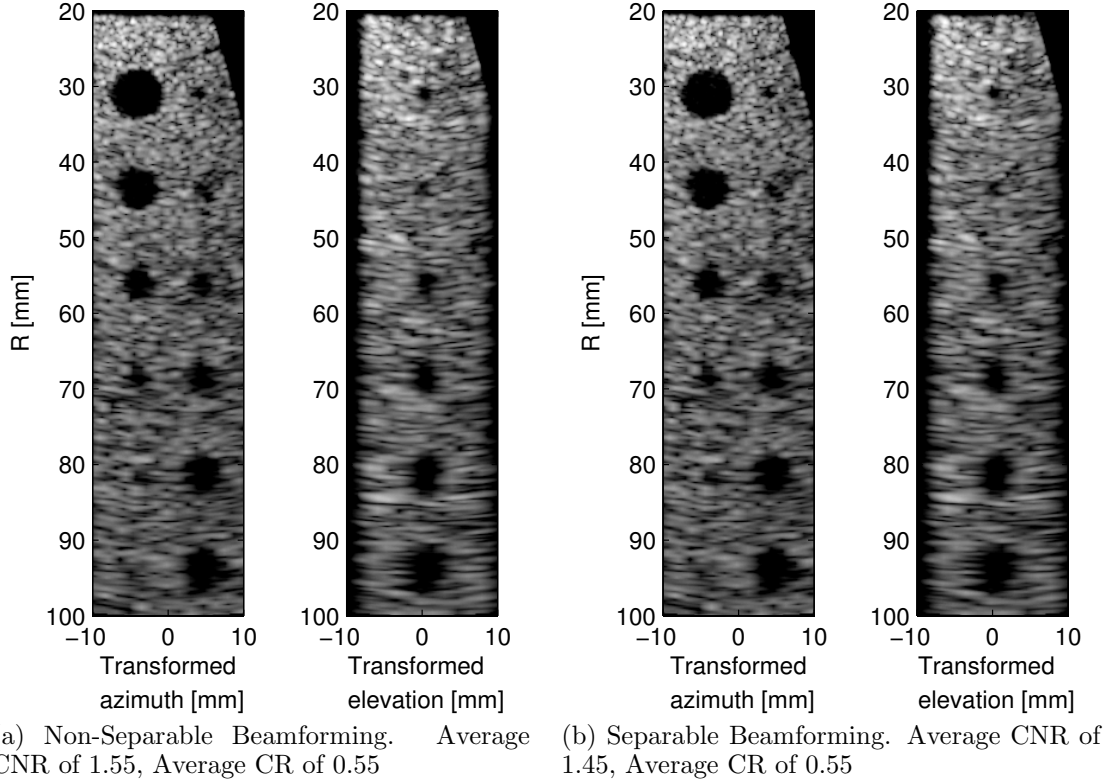
Figure 4.10: 2-D Slices of 3-D Simulation Images for 40 dB Dynamic Range. Case B: $\theta = \phi = 30°$.

is sufficient for separable beamforming, because truncations can be done on partial beamforming data $F_l^{(1)}(n_y, m_R, m_\theta; t)$ to prevent overflow without affecting image quality. Hence we propose a 12-bit data path in our hardware implementation for separable beamforming.

4.4 Modified Sonic Millip3De Architecture

4.4.1 System Architecture Overview

We extend the existing Sonic Millip3De beamforming accelerator described in Section 3.3 to handle separable beamforming. Sonic Millip3De v2.0 architecture also consists three distinct die layers that are stacked vertically and are connected TSVs as shown in Figure 4.12.
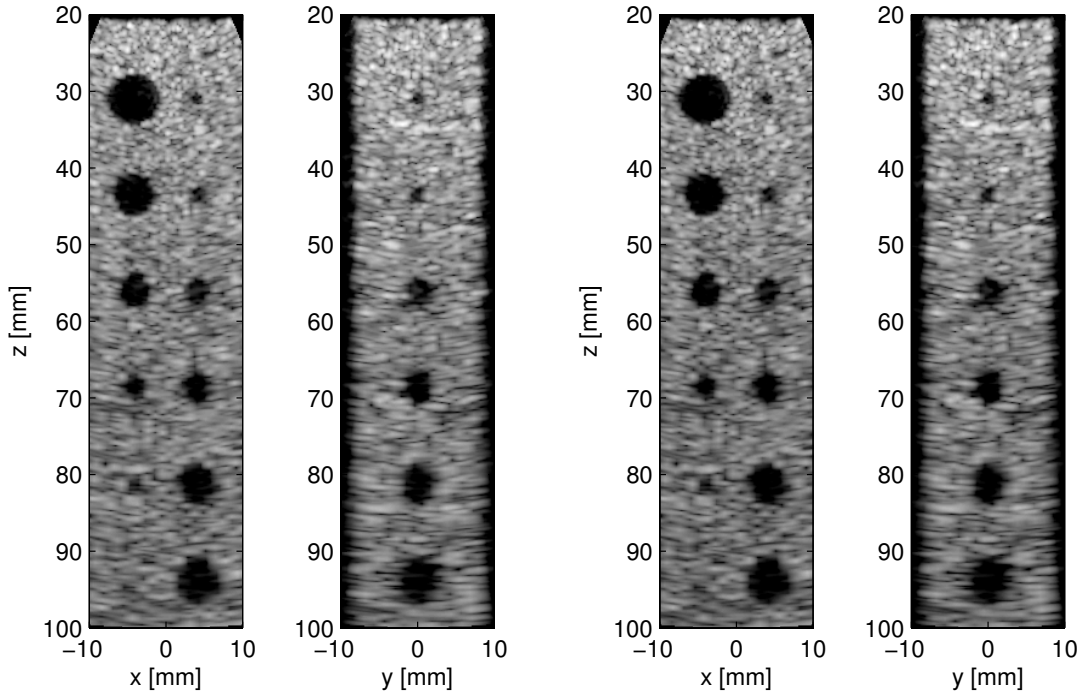
(a) Separable Beamforming, for 12-bit Data Path, Average CNR of 2.0, Average CR of 0.54

(b) Separable Beamforming, for 14-bit Data Path, Average CNR of 2.0, Average CR of 0.55

Figure 4.11: 2-D Slices of 3-D Simulation Images Generated by Fixed-Point Algorithm for 40 dB Dynamic Range. Case A: $\theta = \phi = 0°$.
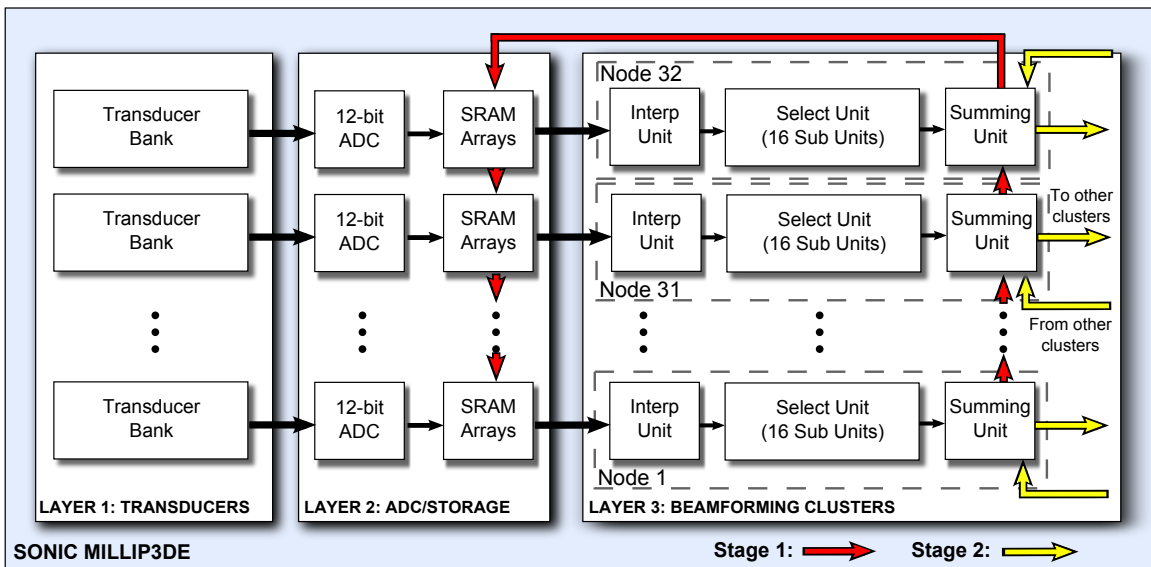

Figure 4.12: Sonic Millip3De v2.0 Hardware Overview

The first of these layers contains a 120×88 grid of capacitive-micromachined ultrasonic transducers (CMUTs) and associated analog support circuitry [60]. As before, these transducers are multiplexed into 1,024 output channels, which are then fed into the second layer. The second layer is made up of ADCs and SRAM storage. For each of the 1,024 processing channels, there is a 12-bit ADC as well as a 6kB SRAM array to store the digital signal during the first stage of beamforming. Additionally, this layer has a secondary set of 1,536 6kB SRAM arrays which are used to store partially beamformed data for the second stage of separable-beamforming with data being fed back to this storage from the accelerator for the second pass. This second set of arrays is necessary to prevent the original echo data from being overwritten during beamforming as it is reused over a series of scanlines.

The final layer is the beamforming accelerator, which reads echo data from the SRAM arrays and generates beamformed output. To perform separable beamforming, data must pass through this layer twice, once for each partial beamforming operation. In the following section, we provide a more complete description of this layer and its operation.

4.4.2 Beamforming Accelerator

The beamforming accelerator is the central component of Sonic Millip3De, combining massive parallelism with a hardware-efficient implementation of the piecewise quadratic approach to delay estimation. The accelerator comprises 1,024 parallel processing channels, which each read data from separate input channels and process 16 scanlines at a time. Each of these channels is further broken into a three-unit pipeline, which translates raw echo data stored in the SRAM layer into the beamformed data for the image (Figure 4.12). During the first stage of separable beamforming, partial beamforming is performed within 32-channel clusters that perform a summation

within the cluster and write partially beamformed data back to secondary SRAM storage in the second (memory) layer. The partially beamformed data is then fed through the accelerator a second time, where it is again delay-aligned and summed across all 1,024 channels to generate the final image. The image is then written to external memory via a ARM Cortex M-3 control processor.

As noted, each beamforming channel comprises three units. The first unit (interpolation unit) reads echo data from the SRAM storage and applies a pre-loaded channel-specific constant apodization to the signal. The apodization weights the channel's impact on the final image based on the corresponding transducer's position in the sub-aperture. After apodization, this unit then performs a 4× linear interpolation to up-sample the signal from 40MHz to 160MHz, a common optimization in existing commercial designs to reduce the ADC sampling frequency.

Next, the expanded data is streamed into the next unit for the beamformation process to begin. The interpolated signal is transferred from the interpolation unit to the select unit. The select unit iteratively calculates the delays between consecutive focal points along a scanline and identifies the interpolated sample that most closely corresponds to the focal point (i.e., it selects the sample from its channel nearest to each focal point). The select unit operates in parallel on 16 scanlines. 16 sub-units iterate over the interpolated data in a block-synchronized fashion each aligning the input signal to its assigned scanline. As described previously, the iterative delay calculation algorithm determines how many samples to advance an input channel to arrive at the sample nearest a focal point using our piecewise quadratic delay estimation formula. The hardware is easily able to estimate the delta between selected samples using three adders and the pre-computed quadratic constants, thereby iteratively solving the quadratic equation and producing each estimated delta as needed. Using these estimates, the sub-units know how far along the data stream to iterate

73
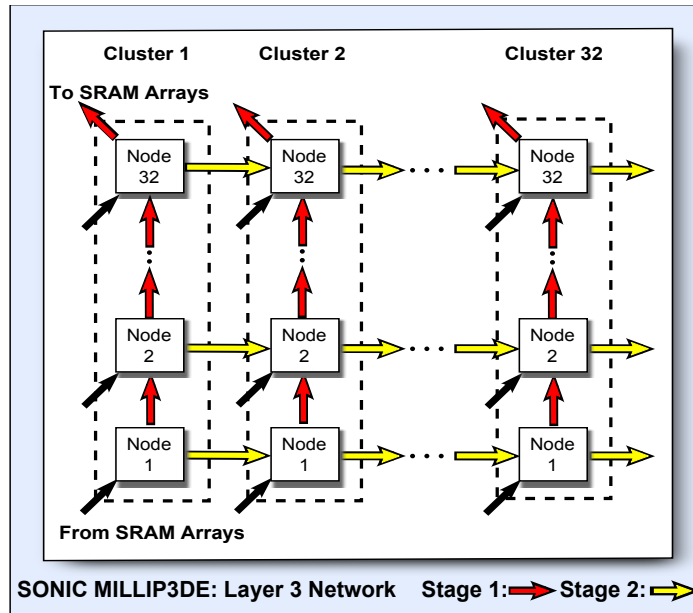
Figure 4.13: Network Data Flow in Stages 1 and 2 of Sonic Millip3De v2.0

before selecting their next output value. The delay-adjusted scanline data for the 16 neighboring scanlines is then fed forward to the summation network.

The final unit of each channel sums partially beamformed data across the channels though the use of adders connected via a reconfigurable mesh network. The network is reconfigured between beamforming operations to connect adders into a pipeline appropriate to the necessary summation operation: within clusters of 32 channels in the first beamforming stage, and across clusters in the second beamforming stage. The reconfigurability of the summation network is one of the key changes required over the baseline Sonic Millip3De design to enable separable beamforming.

The output of the summation network is written either to secondary SRAM arrays on the memory layer (for the first beamforming stage), or are passed to an ARM Cortext M-3 control processor to write final image data to external memory. (Figure 4.13) illustrates the reconfigurable network and the data flow in the two beamforming stages.

4.4.3   Power Analysis

To analyze the power performance of Sonic Millip3De architecture, we estimate the power for each component of the design. We use RTL-level Verilog synthesis results of the accelerator hardware using an industrial 45nm standard cell library. SRAM values are generated using an industrial SRAM compiler, and our network power is obtained using SPICE models of our wires in 45nm. Published state-of-the-art power numbers are used for ADC [49], DRAM [44], and memory interconnect (ARM Cortex M-3) [48]. Figure 4.14a shows the complete power breakdown of the architecture. The beamformer along with the interconnect consumes 9.8W and accounts for 64% of the total system power. The front-end that includes ADCs and transducers consumes 1.5W and accounts for 10% of the total system power. The total system power at 45nm is just about 15W for a frame rate of 32Hz.

In addition to our power analysis at 45nm, we also project power requirements to 11nm technology using published trends. ADC scaling uses values from [50], technology scaling is taken from [51], and we assume network wire power does not scale other than the shortening of the wires due to transistor area scaling. Based on these scaling trends, Sonic Millip3De v2.0 is just within our 5W target at the 16nm node and falls well below the target power by the 11nm node.

If the frame rate is reduced to 16 frames/second, the front-end power remains at 1.5W, but the beamformer power reduces to 5.9W at 45nm. The total power consumption at 45nm is now 11W, and the 5W power target can be met at 22nm technology as shown in Figure 4.14b.
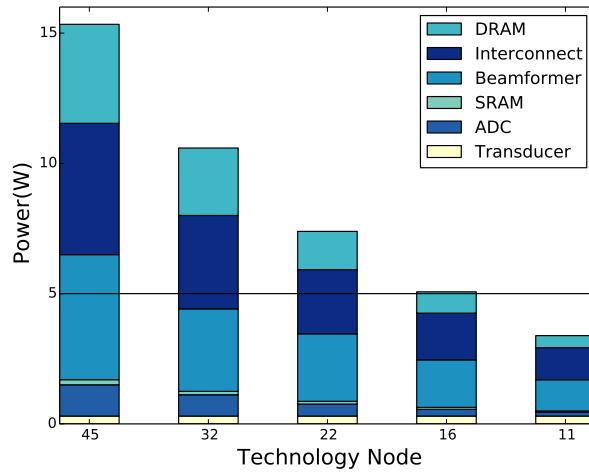
(a) Power Consumption for Frame Rates of 32 frames/second
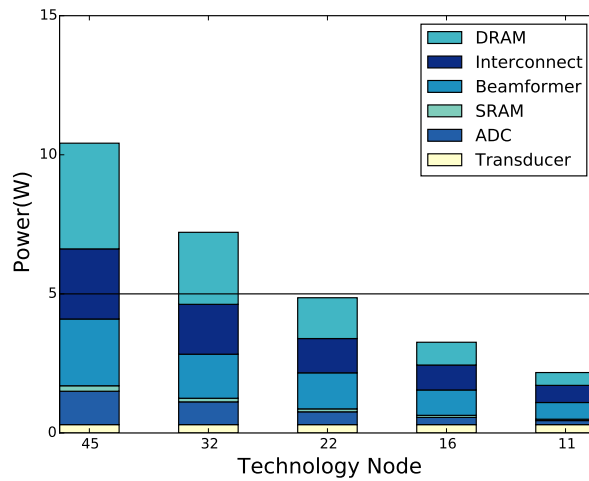
(b) Power Consumption for Frame Rates of 16 frames/second

Figure 4.14: Power Breakdown of Sonic Millip3De v2.0 and Projection across Technology Nodes

Chapter 5

SEPARABLE BEAMFORMING FOR PLANE-WAVE 3-D IMAGING

3-D vector flow imaging and 3-D elastography require high volume acquisition rates of over 1000 volumes per second. 3-D plane-wave imaging has the potential to achieve such high volume acquisition rates because it utilizes a defocused plane-wave excitation and produces multiple scanlines in each firing. Unfortunately, plane-wave imaging systems suffer from low resolution and low SNR due to lack of transmit focus. A coherent image compounding scheme was proposed in [61] to compensate for the lower quality of plane-wave based 2-D imaging systems at the expense of significant increase in computational complexity.

In this chapter, a low complexity 3-D plane-wave imaging system that applies coherent image compounding to improve lateral resolution and SNR is presented. The increased computation requirement of compounding can be offset by applying separable beamforming. Use of the separable beamforming method helps achieve $11\times$ reduction in computation, creating headroom to compound multiple images. The Field II simulations show that the proposed system with 9-firing-angle compounding has better image quality in terms of sidelobe levels, SNR values, and contrast-to-noise ratios (CNR), while having lower computational complexity compared to the baseline (non-separable, non-compounded) plane-wave system. Hardware modifications to the Sonic Millip3De accelerator is also proposed. These include additional low power embedded DRAM and a configurable interconnect to support on-chip compounding and separable beamforming. These architecture level upgrades allow us to achieve 1000 volumes per second.
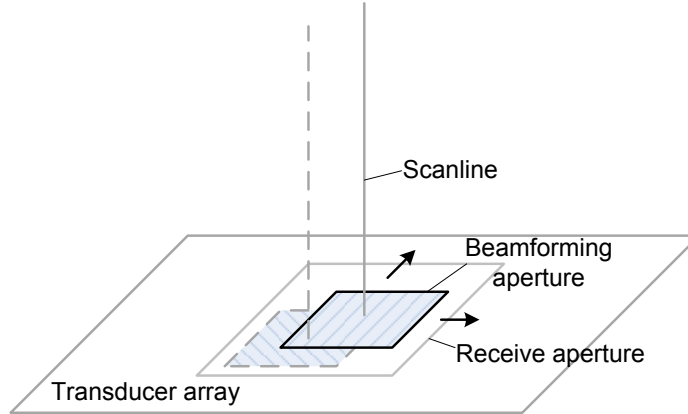
Figure 5.1: 2-D Plane-Wave Transmit and Receive Scheme

This chapter is organized as follows. First, the plane-wave imaging system overview is presented. Then the separable beamforming and delay decomposition for plan-wave systems using coherent compounding are presented. After that, we present the modifications of Sonic Millip3De architecture in order to support the coherent compounding and very high volume acquisition rates. Finally, we present Field II simulation results and image quality analysis of the proposed system

5.1 Plane-Wave Imaging System Overview

The transmit and receive scheme of a plane-wave system is shown in Figure 5.1. We assume that only a subset of transducers in the physical aperture is used for plane-wave imaging. In each firing, the transducers in the selected aperture are used to generate a plane-wave that propagates through the region of interest. All the elements in the selected aperture are used to receive echo signals. Within the receive aperture, subsets of elements form a sequence of beamforming apertures (one such aperture is shown by a bold box in Figure 5.1), traversing all possible positions within the receive aperture. In each position, the beamforming aperture generates a single vertical scanline located at its center. When part of the beamforming aperture is outside the selected aperture, the corresponding apodization coefficients are set to
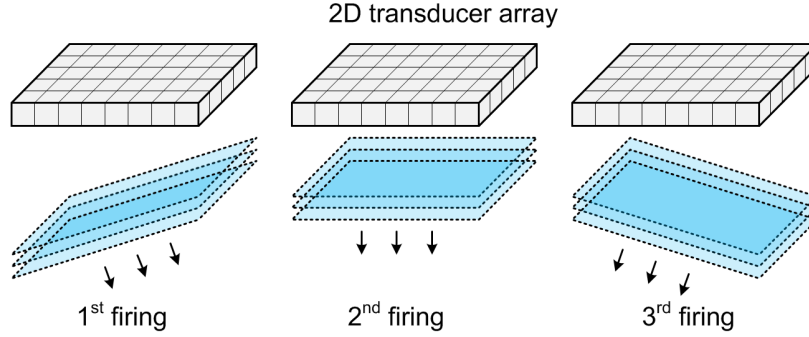
Figure 5.2: Firing Scheme of 3-D Plane-Wave System with Compounding

zero. A 3-D plane-wave system with coherent image compounding fires the plane-wave at multiple firing angles as shown in Figure 5.2. The volumes obtained by these firings are coherently combined resulting in improved SNR and lateral resolution.

5.2 Separable Beamforming For Plane-Wave Systems with Coherent Compounding

The scanline geometry of a 3-D plane-wave system is shown in Figure 5.3. In this system, the scanlines are all parallel to each other and perpendicular to the transducer plane. $(x, y, 0)$ is the coordinate of a transducer element, and $(\hat{x}, \hat{y}, \hat{z})$ is the coordinate of focal point P.
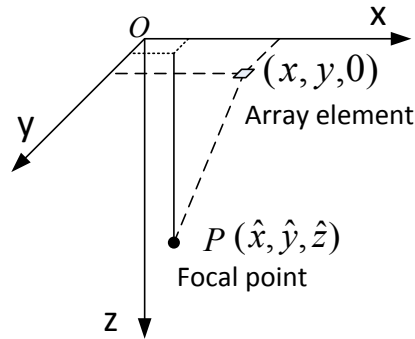
Figure 5.3: Scan Geometry of Plane-Wave System

The plane-wave firing angles can be defined by (α, β) as shown in Figure 5.4, where α is the angle between the z axis and the normal vector, and β is the angle between the x axis and the projection of the normal vector on the xy plane.
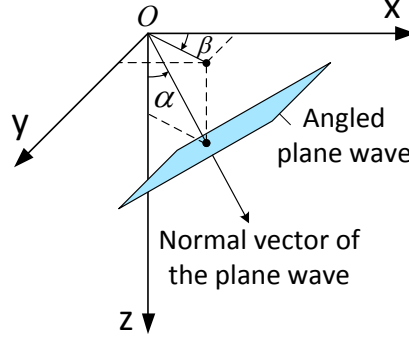
Figure 5.4: Angle Definition of 3-D Plane-Wave System with Coherent Compounding

The beamforming delay is given by $\tau = (2|\hat{z}| - d_{tx} - d_{rx})/c$, where c is the speed of sound, d_{tx} is the distance between the wavefront plane and the focal point P at $t = 0$, and d_{rx} is the distance between the focal point P and the receive transducer at $(x, y, 0)$. d_{tx} and d_{rx} are calculated as follows.

$$d_{tx} = (\hat{x} - x_0)\sin\alpha\cos\beta + (\hat{y} - y_0)\sin\alpha\sin\beta + (\hat{z} - z_0)\cos\alpha \qquad (5.1)$$

$$d_{rx} = \sqrt{(x - \hat{x})^2 + (y - \hat{y})^2 + \hat{z}^2} \qquad (5.2)$$

where (x_0, y_0, z_0) is the coordinate of an arbitrary point on the wavefront at $t = 0$. Thus, the beamforming delay τ is a function of five variables, namely $\tau(x, y, \hat{x}, \hat{y}, \hat{z})$.

Assuming the receive signal at transducer (x, y) is $S(x, y, t)$, the non-separable beamforming process for a plane-wave system is represented by

$$F(\hat{x}, \hat{y}, \hat{z}; t) = \int_{\hat{y}-\frac{D_y}{2}}^{\hat{y}+\frac{D_y}{2}} \int_{\hat{x}-\frac{D_x}{2}}^{\hat{x}+\frac{D_x}{2}} A(x - \hat{x}, y - \hat{y}) \cdot S(x, y, t - \tau(x, y, \hat{x}, \hat{y}, \hat{z}))dxdy \qquad (5.3)$$

where D_x and D_y are the width of the beamforming aperture in x dimension and the height of the beamforming aperture in y dimension, respectively.

Now if the delay term can be decomposed as

$$\tau(x, y, \hat{x}, \hat{y}, \hat{z}) = \tau_1(x, y, \hat{x}, \hat{z}) + \tau_2(y, \hat{x}, \hat{y}, \hat{z}) \tag{5.4}$$

and apodization coefficients can be represented by $A(x-\hat{x}, y-\hat{y}) = A_x(x-\hat{x}) \cdot A_y(y-\hat{y})$, the beamforming process can be decomposed as

$$F^{(1)}(y, \hat{x}, \hat{z}; t) = \int_{\hat{x}-\frac{D_x}{2}}^{\hat{x}+\frac{D_x}{2}} A_x(x-\hat{x}) S(x, y, t - \tau_1(x, y, \hat{x}, \hat{z})) dx \tag{5.5}$$

$$F^{(2)}(\hat{x}, \hat{y}, \hat{z}; t) = \int_{\hat{y}-\frac{D_y}{2}}^{\hat{y}+\frac{D_y}{2}} A_y(y-\hat{y}) F^{(1)}(y, \hat{x}, \hat{z}; t - \tau_2(y, \hat{x}, \hat{y}, \hat{z})) dy \tag{5.6}$$

Figure 5.5 demonstrates this process. In the first stage, beamforming is performed along the x dimension. The 1-D beamforming aperture traverses the entire selected aperture. For each combination of $(\hat{x}, y)$, a 1-D beamformer steers the azimuth angle to be normal to the xy plane, and records partially beamformed data in $F^{(1)}(y, \hat{x}, \hat{z}; t)$. In the second-stage, the 1-D beamforming aperture moves along the y dimension. For each combination of $(\hat{x}, \hat{y})$, the beamformer steers the elevation angle to be normal to the xy plane and a scanline is generated.

For a 3-D plane-wave system with coherent image compounding, the delay τ is a function of five variables, to capture the effect of angled plane-wave firing, but $\tau_1(x, y, \hat{x}, \hat{z})$ and $\tau_2(y, \hat{x}, \hat{y}, \hat{z})$ are functions of four variables. To reduce the approximation error due to this decomposition, the significant cross terms in the Taylor series expansion of the beamforming delay function should be kept. The functions τ, τ_1 and τ_2 all contain $\hat{z}$. τ_1 contains x and $\hat{x}$, because after first-stage beamforming along x, the beamformer is able to focus in the vertical azimuth angle at position $\hat{x}$ (recall Figure 5.5a). Adding y to the list helps to keep the cross terms due to the square root operation. Similarly, τ_2 contains y and $\hat{y}$, and adding $\hat{x}$ to variable list of τ_2 helps keep cross terms such as $\hat{x}y$ or $\hat{x}\hat{y}$.

(a) Plane-Wave Separable Beamforming, Stage 1

(b) Plane-Wave Separable Beamforming, Stage 2
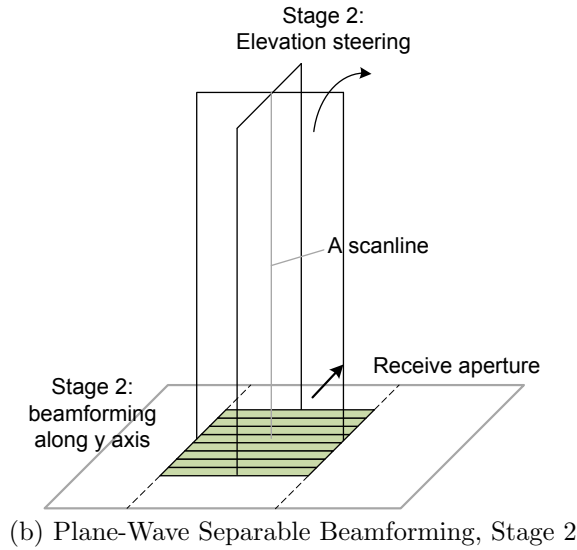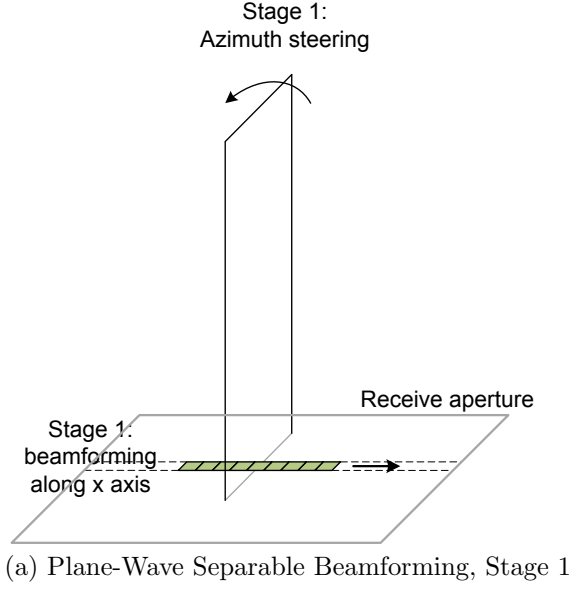
Figure 5.5: Plane-Wave Separable Beamforming Principle

To get the optimal solution for τ_1 and τ_2, we minimize the RMSE, which is equivalent to minimizing the error function below.

$$E(y, \hat{x}, \hat{z}) = \int_{y-\frac{D_y}{2}}^{y+\frac{D_y}{2}} \int_{\hat{x}-\frac{D_x}{2}}^{\hat{x}+\frac{D_x}{2}} [\tau(x, y, \hat{x}, \hat{y}, \hat{z}) - (\tau_1(x, y, \hat{x}, \hat{z}) + \tau_2(y, \hat{x}, \hat{y}, \hat{z}))]^2 dx d\hat{y} \quad (5.7)$$

The discrete version of a solution to this problem is given by:

$$\tau_1(n_x, n_y, m_{\hat{x}}, m_{\hat{z}}) = \frac{1}{N_y} \sum_{m_{\hat{y}}=n_y-\frac{N_y}{2}}^{n_y+\frac{N_y}{2}-1} \tau(n_x, n_y, m_{\hat{x}}, m_{\hat{y}}, m_{\hat{z}}) - \rho(n_y, m_{\hat{x}}, m_{\hat{z}}) \qquad (5.8)$$

$$\tau_2(n_y, m_{\hat{x}}, m_{\hat{y}}, m_{\hat{z}}) = \frac{1}{N_x} \sum_{n_x=m_{\hat{x}}-\frac{N_x}{2}}^{m_{\hat{x}}+\frac{N_x}{2}-1} \tau(n_x, n_y, m_{\hat{x}}, m_{\hat{y}}, m_{\hat{z}}) - \rho(n_y, m_{\hat{x}}, m_{\hat{z}}) \qquad (5.9)$$

$$\rho(n_y, m_{\hat{x}}, m_{\hat{z}}) = \frac{1}{2N_x N_y} \sum_{m_{\hat{y}}=n_y-\frac{N_y}{2}}^{n_y+\frac{N_y}{2}-1} \sum_{n_x=m_{\hat{x}}-\frac{N_x}{2}}^{m_{\hat{x}}+\frac{N_x}{2}-1} \tau(n_x, n_y, m_{\hat{x}}, m_{\hat{y}}, m_{\hat{z}}) \qquad (5.10)$$

where n_x and n_y are transducer column index and row index, respectively; $m_{\hat{x}}$, $m_{\hat{y}}$ and $m_{\hat{z}}$ are scanline column index, scanline row index and focal point index, respectively; N_x and N_y are the number of columns and the number of rows of the transducer array, respectively; $M_{\hat{x}}$ and $M_{\hat{y}}$ are the number of scanlines in x dimension and y dimension, respectively; $M_{\hat{z}}$ is the number of focal point per scanline.
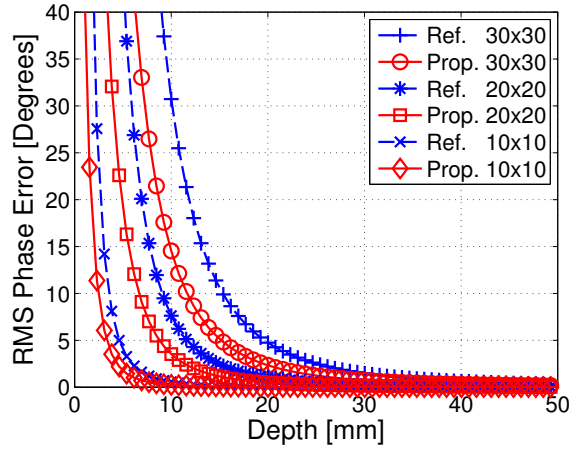
Figure 5.6: RMS Phase Error of Proposed Delay Decomposition Compared with Prior Work for Different Beamforming Aperture Sizes

The computation complexity of non-separable beamforming in terms of delay-sum operations per volume is $N_x N_y M_{\hat{x}} M_{\hat{y}} M_{\hat{z}}$. In comparison, separable beamforming requires $N_x N_y M_{\hat{x}} M_{\hat{z}} + N_y M_{\hat{x}} M_{\hat{y}} M_{\hat{z}}$ delay-sum operations per volume, hence the complexity reduction with respect to non-separable beamforming is $N_x M_{\hat{y}}/(N_x + M_{\hat{y}})$.

83
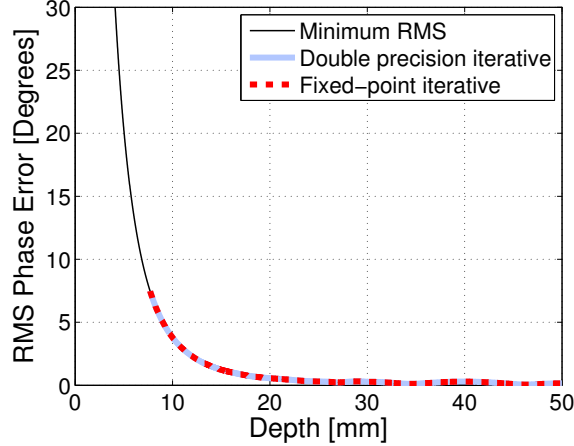
Figure 5.7: RMS Phase Error of Plane-Wave System for Aperture Size of 20×20

Next, we compare our delay decomposition method with the approximation proposed by Owen et al. [58]. Figure 5.6 presents the comparison of the RMS error of our proposed delay decomposition method with that in [58]. The results indicate that our proposed method has lower phase error for beamforming aperture ranging from 10×10 to 30×30. For both methods, as depth increases the error rapidly reduces. Also, as subaperture size increases, the error increases for both methods. We present Field II simulation results for both methods in Section 5.5.

5.3 Online Iterative Separable Delay Calculation

The iterative delay calculation for separable plane-wave systems is very similar to that for separable SAU system described in Section 4.2.4. Here too we use piece-wise quadratic curves to approximate the delay difference between consecutive samples along a scanline. For example, let $\tilde{\tau}(m_R)$ be the delay corresponding to the m_Rth focal point for fixed n_x and n_y. Let $\eta(m_R) = \tilde{\tau}(m_R + 1) - \tilde{\tau}(m_R)$, then $\eta(m_R)$ can approximated by $am_R^2 + bm_R + c$. Instead of storing the delay look-up table directly, the coefficients a, b and c and the initial delay are stored, and the delays are iteratively
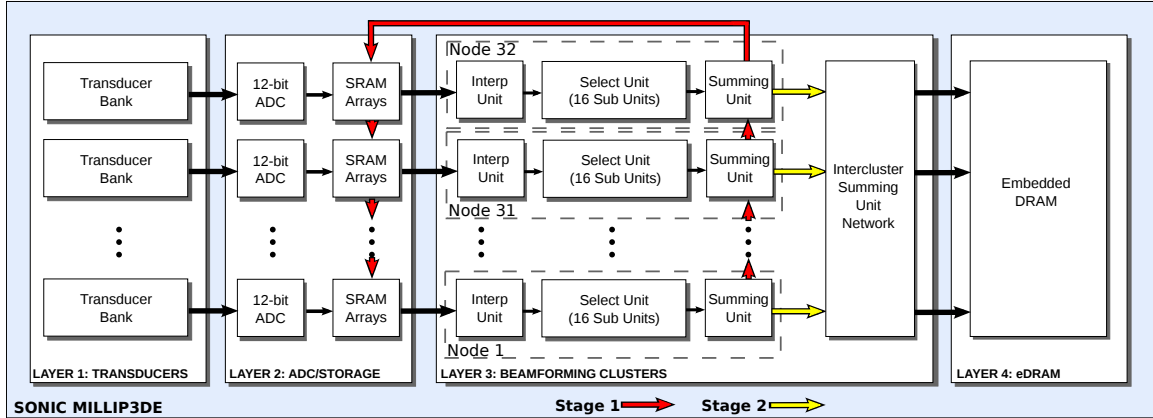
Figure 5.8: Sonic Millip3De v3.0 for Plane-Wave Systems that includes an Additional Embedded DRAM Layer

calculated using these coefficients. The iterative calculation method does not require multiplications; it can be implemented in a simple circuit using only three additions.

By using iterative delay calculation method, the proposed method requires storage of 8.5×10^5 coefficients for each firing angle. Assuming each coefficient has 12 bits on average, this is about 1.2MB storage for each firing angle, and for a system with 9 firing angles, this is about 10.8MB of storage. The storage requirement can be reduced by about $2\times$ if the firing angles are chosen to be symmetric.

Figure 5.7 shows simulation results of the iterative delay calculation method with double-precision floating-point and 12-bit fixed-point precision on the SAU and the plane-wave systems, respectively. The iterative method with double precision floating-point coincides with the minimum RMSE curve, and fixed-point approximation's performance is very close to the minimum RMSE curve. Thus the iterative calculation method helps reduce the storage requirement significantly with almost no increase in RMSE.
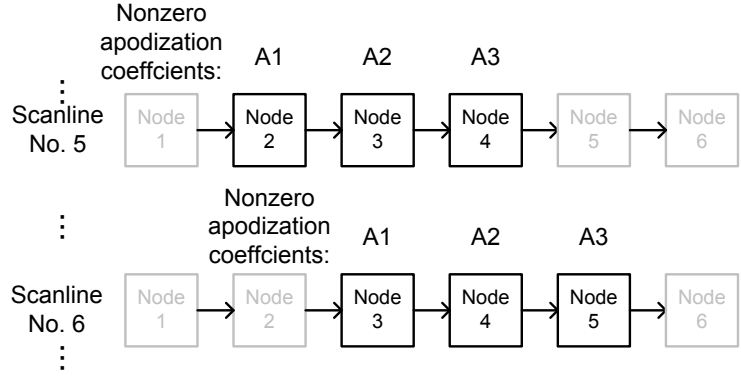
Figure 5.9: Dataflow Diagram of Plane-Wave Separable Beamforming

5.4 Extension of Sonic Millip3De for Separable Plane-Wave Imaging

In the SAU system design, sub-volume data from each firing was temporarily stored in off-chip DRAM before being combined to produce the final volume, however, due to the extremely high rate that these sub-volumes are produced for our planar technique (over 9,000 sub-volumes per second), bandwidth to off-chip DRAM is insufficient for temporary storage. To remove this bottleneck, we have modified our design to include an additional 4th die layer of embedded DRAM (eDRAM) to handle the temporary storage of the 21MB sub-volumes locally, as shown in Figure 5.8. Previous work [62] has demonstrated eDRAM as an efficient, high-bandwidth alternative to traditional DRAM in 3-D-stacked designs. Additionally eDRAM is more dense and consumes less power than SRAM storage, minimizing the additional power of our modified design. Furthermore, we can avoid the refresh power conventionally required for DRAM since sub-volumes are overwritten so rapidly that there is no need to refresh them.

The modified Sonic Millip3De v3.0 architecture is able to support separable plane-wave beamforming with only minor change in the accelerator core layer. As described previously, in a separable plane-wave system, each scanline is generated using echo

86

data from only a small neighboring sub-set of channels, in contrast to the SAU system, where all channels contribute to every scanline. However, in Sonic Millip3De v3.0, each channel processes 16 scanlines independently and data that is not needed can easily be zeroed out using the apodization coefficients. Figure 5.9 shows a simple example where the 5th scanline uses data from channels 2, 3, and 4 (represented by their respective network nodes) and scanline 6 uses data from channels 3, 4 and 5. In the example, the image data processed in sub-unit of Node 5 for scanline 6 will have A3 as the apodization; however, the data generated by the sub-unit for scanline 5 will use 0 for apodization, and hence will not contribute to the beamforming. Overall the only additional hardware required to perform this operation is storage for 16 apodization constants (one per scanline processed) in each channel instead of the single constant required by SAU. The rest of the processing pipeline remains the same as in Section 4.4.2.

5.5   Simulation Results

Table 5.1: System Parameters of 3-D Plane-Wave System

Property	Value
Pitch, μm	385
Receive aperture size, transducers	32×32
f-number	2.0
Number of scanlines	32×32
Max depth, cm	5
Center frequency, MHz	4
6 dB transducer bandwidth, MHz	2
A/D sampling rate, MHz	40

We use Field II [46, 47] and MATLAB simulation to verify the algorithm and the proposed system according to the configuration shown in Table 5.1. For the plane-wave system, the transducer array is the same as the SAU system, but the transducer spacing is a full wavelength, which is twice that of the SAU system. The receive aperture consists of $32{\times}32$ transducers, but the beamforming aperture size increases as depth increases in order to maintain consistent lateral resolution at different depths. The depth of our plane-wave system ranges from 8mm to 50mm.

The baseline system employs non-separable beamforming method. The plane-wave fire angles used are $(\alpha, \beta) \in \{(0°, 0°), (3°, 0°), (3°, 90°), (3°, 180°), (3°, 270°), (6°, 0°), (6°, 90°), (6°, 180°), (6°, 270°), (9°, 0°), (9°, 90°), (9°, 180°), (9°, 270°)\}$. For 5-fire-angle compounding, only the first five firing angles are used, for 9-fire-angle compounding the first nine angles are used, and for 13-angle compounding all the above firing angles are used.

In the first simulation case, three point targets are set at depths of 13mm, 23mm, and 33mm. Compared to the baseline system, the separable beamforming with 5-angle compounding, 9-angle compounding and 13-angle compounding provide 6.4 dB, 9.0 dB and 10.4 dB SNR improvement, respectively. The xy plane projections of the point spread function at depth of 23mm are shown in Figure 5.10. We see that the separable beamforming with coherent compounding method helps to reduce the mainlobe width and the artifacts near the boundary.

Next, three 6 mm anechoic cysts located in phantom scatterers at depths of 12mm, 23mm and 33mm are simulated. The average CNR value provided by the non-separable beamforming without compounding is 1.6 and the corresponding CR value is 0.32. The proposed method with 5-angle, 9-angle and 13-angle compounding improves the average CNR values to 1.9, 2.2, and 2.4, respectively, and the average CR values is 0.46, 0.52 and 0.56, respectively.

(a) Non-separable beamform-
ing, non-compounding

(b) Separable beamforming,
non-compounding

(c) Separable Beamforming,
5-Fire-Angle Compounding

(d) Separable Beamforming,
9-Fire-Angle Compounding

(e) Separable Beamforming,
13-Fire-Angle Compounding
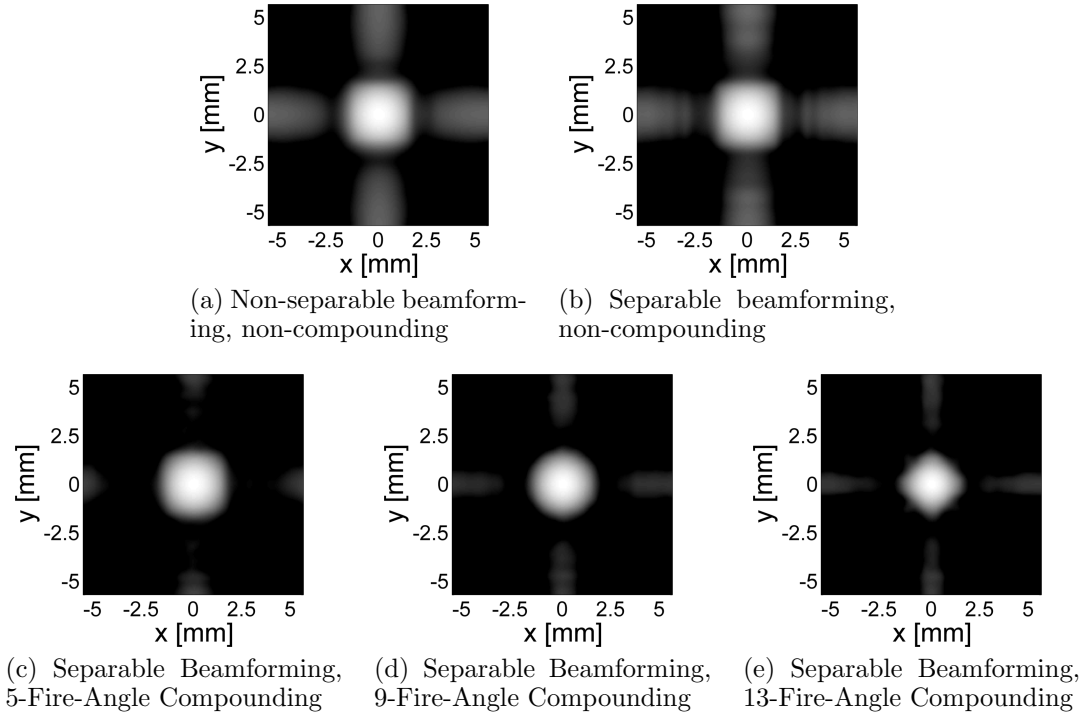
Figure 5.10: Point Spread Functions of Non-Separable and Separable Beamforming Systems with and without Coherent Image Compounding, Displayed in 40 dB Dynamic Range

The xz slices of the 3-D volume are shown in Figure 5.11. The figure demonstrates that the image quality of the plane-wave 3-D imaging system is significantly improved by the combination of separable beamforming with coherent compounding method.

5.5.1 Image Quality and Computational Complexity Trade-offs

The complexity reduction due to separable beamforming is $N_x M_{\hat{y}}/(N_x + M_{\hat{y}})$ as shown in Section 5.2. Since the beamforming aperture size depends on f-number and depth, the computational complexity is also a function of f-number and depth. Based on the configuration in Table 5.1, for f-number = 2.0 at 10mm depth, $N_x = 13$, and the separable beamforming method reduces computational complexity by about 9×. At 25mm depth, $N_x = 32$, and the complexity reduction is increased to 16×. The total delay-sum operations per volume required by the non-separable system is

(a) Non-Separable Beamform-
ing, Non-Compounding

(b) Separable Beamforming,
5-Fire-Angle Compounding

(c) Separable Beamforming,
9-Fire-Angle Compounding

(d) Separable Beamforming,
13-Fire-Angle Compounding
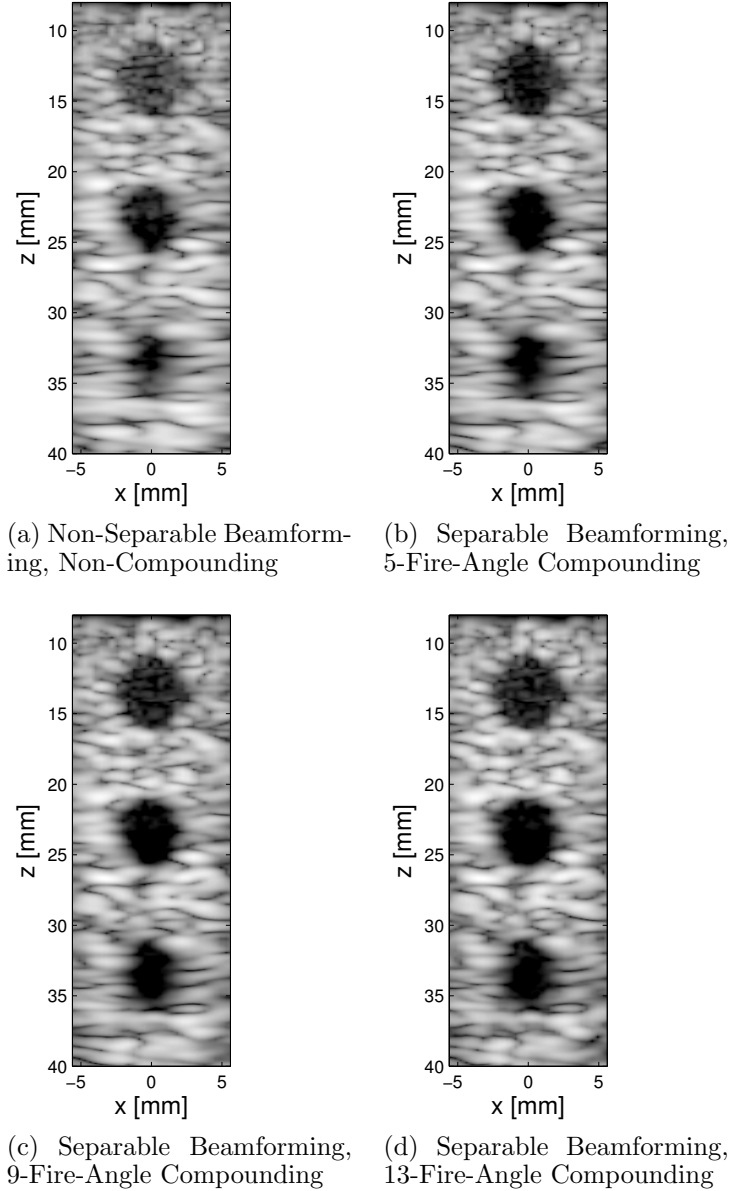
Figure 5.11: 2-D Slices of 3-D Cyst Phantom Simulation Images

about 9.9×10^8, while the delay-sum operations required by separable beamforming
is 9.1×10^7, equivalent to $11\times$ reduction in complexity per image volume.

Finally we provide the trade-offs between quality performance and complexity as
functions of the number of firing angles. The benefits of the proposed method with
different number of firing angles are summarized in Figure 5.12a. We see that as the

(a) Gains

(b) Costs
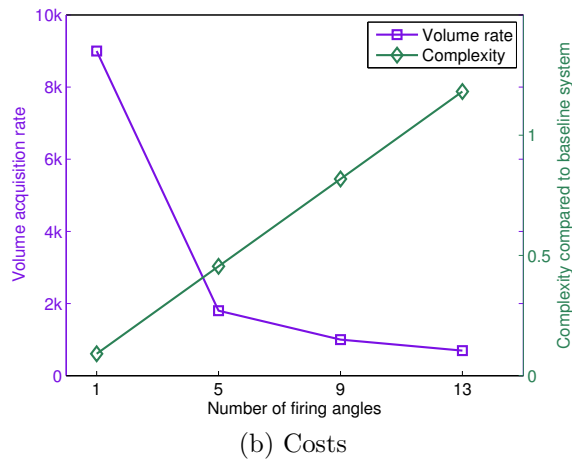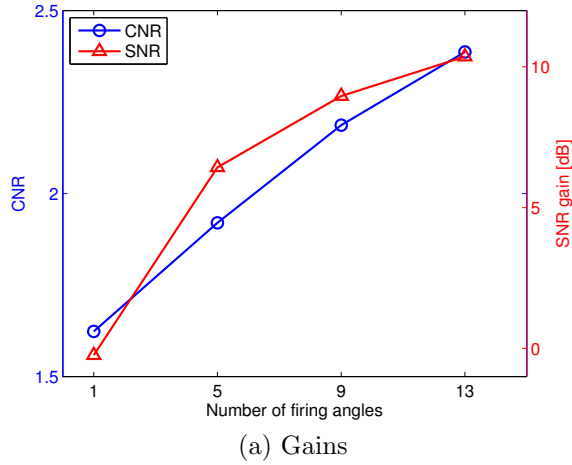
Figure 5.12: Choice of the Number of Firing Angles

number of firing angles increase, both the SNR and the CNR improves. However for large firing angles, the increase is moderate. On the other hand, the computational complexity is proportional to the number of firing angles, and volume acquisition rate is inversely proportional to the number of firing angles, as shown in Figure 5.12b. Hence we choose 9-fire-angle compounding configuration, as it is able to support significantly improved SNR, image quality and high volume rate at lower complexity compared to the baseline system.

Chapter 6

CONCLUSIONS AND FUTURE WORK

6.1   Main Contributions

In this work we developed efficient algorithms to support low-power hand-held
3-D ultrasound systems. Our contributions are as follows.

1. Image enhancement techniques for synthetic aperture ultrasound systems.

- Proposed SNR improvement techniques including virtual source firing and or-
  thogonal coded excitation. We found that the complexity of orthogonal coded
  excitation based systems can be significantly reduced by optimizing the decod-
  ing algorithm and choosing design parameters such as code length and number
  of orthogonal codes judiciously.

- Proposed a low overhead motion compensation method that improves the per-
  formance of coded excitation based systems in presence of motion. Specifically,
  this method improves SNR and reduces RSLL, and has been verified by Field
  II simulation.

2. Algorithm optimizations to support low power design of the 3D die-stacking hard-
ware accelerator, Sonic Millip3De:

- Proposed a subaperture transmit/receive scheme and apodization optimization
  method that halves the number of firings required by an SAU system.

- Proposed an iterative method that significantly simplifies the delay calculation
  for beamforming. Instead of storing the delays in huge look-up tables, this

method can iteratively generate the delays of focal points along a scanline, using only three additions per focal point.

- Designed a 3D die-stacking beamforming accelerator, Sonic Millip3De, that produces 2 volumes per second and consumes 15W when synthesized in 45nm technology.

3. Separable beamforming method for 3-D SAU imaging systems:

- Proposed a separable beamforming method that decomposes beamforming for 2-D array into a series of 1-D beamforming processes. This method is based on decomposing the delay such that the RMS phase error is minimized. This method results in about 19× complexity reduction for our system.

- Studied key factors including the number of variables, combination of variables and sequence of beamforming (along x first vs. along y first), which affect decomposition errors. RMS phase error maps for different depths, different angles, and transducer locations have been studied. The performance of the separable method has been validated by Field II simulations. We found that the quality of images produced by the separable beamforming method is identical to the quality of images produced by the non-separable beamforming method.

- Proposed modifications to the original Sonic Millip3De architecture to support the separable beamforming method. The synthesis results show that Sonic Millip3De v2.0 is able to produce images 16× faster (from 2Hz to 32Hz in frame rate) while keeping total power consumption at 15W for 45nm technology node.

4. Separable beamforming and coherent compounding method for 3-D plane-wave imaging systems:

- Proposed an RMS delay decomposition based method for 3-D plane-wave imaging systems using plane-wave compounding. The proposed system has improved SNR and CNR with lower computational complexity compared to the non-separable non-compounding baseline system.

- Validated the image quality improvement by Field II simulations. The proposed method with 9-fire-angle compounding helps improve average CNR from 1.6 to 2.2 and SNR by 9.0 dB.

- Proposed Sonic Millip3De v3.0 that achieves volume acquisition rates over 1000 volumes per second for the proposed system configuration.

6.2   Future Work

We plan to study algorithm-architecture codesign for sonoelastography applications. The elasticity of soft tissues provides useful diagnostic information to detect thyroid, breast, and liver abnormalities, because changes in tissue stiffness often relate to pathological changes. For instance, [63,64] show that a map of tissue stiffness can significantly improve the accuracy of diagnosis of thyroid and breast cancers. To support shear wave based 3-D sonoelastography, we propose to scale our 3-D speckle-tracking accelerator to compute displacement of tissue scatterers due to shear wave propagation. Shear waves propagate at 2-6 m/s, hence, the required frame rate for robust tracking is on the order of 5000 frames/second. We will rely on plane wave transmission (Chapter 5) to enable the required image acquisition rate in the region of interest. We will then develop algorithms to reconstruct the 3-D elasticity distribution [65] that is robust to speckle tracking error and is amenable to hardware implementation.

REFERENCES

[1] T.R. Nelson, J.B. Fowlkes, J.S. Abramowicz, and C.C. Church, "Ultrasound biosafety considerations for the practicing sonographer and sonologist," *J. of Ultrasound in Medicine*, vol. 28, no. 2, pp. 139, 2010.

[2] Stergios Stergiopoulos, Ed., *Advanced Signal Processing: Theory and Implementation for Sonar, Radar, and Non-Invasive Medical Diagnostic Systems*, The Electrical Engineering and Applied Signal Processing Series. CRC Press, 2009.

[3] Jerry L. Prince and Jonathan M. Links, *Medical Imaging: Signals and Systems*, Pearson Prentice Hall, 2006.

[4] T. Gunnarsson, A. Theodorsson, P. Karlsson, S. Fridriksson, S Boström, J. Persliden, I. Johansson, and J. Hillman, "Mobile computerized tomography scanning in the neurosurgery intensive care unit: increase in patient safety and reduction of staff workload," *J. of Neurosurgery*, vol. 93, no. 3, pp. 432–436, Sept. 2000.

[5] D. Weinreb and J. Stahl, "The introduction of a portable head/neck ct scanner may be associated with an 86% increase in the predicted percent of acute stroke patients treatable with thrombolytic therapy," *Radiological Soc. of N. America*, 2008.

[6] "LOGIQ Book XP," http://www.gehealthcare.com/usen/ultrasound/products/lbook_index.html.

[7] Bret P. Nelson, Edward R. Melnick, and James Li, "Portable ultrasound for remote environments, part i: Feasibility of field deployment," *The Journal of Emergency Medicine*, vol. 40, no. 2, pp. 190 – 197, 2011.

[8] M.I. Fuller, K. Owen, T.N. Blalock, J.A. Hossack, and W.F. Walker, "Real time imaging with the Sonic Window: A pocket-sized, C-scan, medical ultrasound device," in *IEEE International Ultrasonics Symposium (IUS)*, Sept. 2009, pp. 196 –199.

[9] M. Yang and C. Chakrabarti, "Design of orthogonal coded excitation for synthetic aperture imaging in ultrasound systems," in *IEEE International Symposium on Circuits and Systems*, May 2012.

[10] M. Yang, S. Wei, and C. Chakrabarti, "Reducing the complexity of orthogonal code based synthetic aperture ultrasound system," in *Proceedings of IEEE Workshop on Signal Processing Systems (SiPS)*, Oct. 2012, pp. 270–275.

[11] R. Sampson, M. Yang, S. Wei, C. Chakrabarti, and T. F. Wenisch, "Sonic Millip3De: Massively parallel 3D stacked accelerator for 3D ultrasound," in *19th IEEE International Symposium on High Performance Computer Architecture*, Feb. 2013, pp. 318–329.

[12] R. Sampson, M. Yang, S. Wei, C. Chakrabarti, and T. F. Wenisch, "Sonic Millip3De with dynamic receive focusing and apodization optimization," in *Proceedings of IEEE International Ultrasonics Symposium*, July 2013, pp. 557–560.

[13] R. Sampson, M. Yang, S. Wei, C. Chakrabarti, and T. F. Wenisch, "Sonic Millip3De: An architecture for hand-held 3d ultrasound," *to appear in Special Issue of IEEE Micro: Micro's Top Picks from the Computer Architecture Conferences*, 2014.

[14] M. Yang, R. Sampson, T. F. Wenisch, and C. Chakrabarti, "Separable beamforming for 3-D synthetic aperture ultrasound imaging," in *Proceedings of IEEE Workshop on Signal Processing Systems*, Taipei, Taiwan, Oct. 2013, pp. 207–212.

[15] M. Yang, R. Sampson, S. Wei, T. F. Wenisch, and C. Chakrabarti, "Separable beamforming for 3-D medical ultrasound imaging," *IEEE Transactions on Signal Processing*, vol. 63, no. 2, pp. 279–290, Jan. 2015.

[16] M. Yang, R. Sampson, S. Wei, T. F. Wenisch, and C. Chakrabarti, "High frame rate 3-D ultrasound imaging using separable beamforming," *Journal of Signal Processing Systems*, vol. 78, no. 1, pp. 73–84, 2015.

[17] M. Yang, R. Sampson, S. Wei, T. F. Wenisch, B. Fowlkes, O. Kripfgans, and C. Chakrabarti, "High volume rate, high resolution 3D plane wave imaging," in *IEEE International Ultrasonics Symposium (IUS)*, Sep. 2014, pp. 1253–1256.

[18] M. Karaman, Pai-Chi Li, and M. O'Donnell, "Synthetic aperture imaging for small scale systems," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 42, no. 3, pp. 429 –442, May 1995.

[19] G. R. Lockwood, J. R. Talman, and S. S. Brunke, "Real-time 3-D ultrasound imaging using sparse synthetic aperture beamforming," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control,*, vol. 45, no. 4, pp. 980 –988, July 1998.

[20] M. O'Donnell and Yao Wang, "Coded excitation for synthetic aperture ultrasound imaging," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 52, no. 2, pp. 171 –176, Feb. 2005.

[21] T.X. Misaridis and J.A. Jensen, "Space–time encoding for high frame rate ultrasound imaging," *Ultrasonics*, vol. 40, no. 1, pp. 593–597, 2002.

[22] T. Misaridis and J.A. Jensen, "Use of modulated excitation signals in medical ultrasound. part III: high frame rate imaging," *IEEE Transactions on Ultrasonics Ferroelectrics and Frequency Control*, vol. 52, no. 2, pp. 208 –219, Feb. 2005.

[23] R.Y. Chiao and L.J. Thomas, "Synthetic transmit aperture imaging using orthogonal Golay coded excitation," in *Proceedings of IEEE Ultrasonics Symposium*, Oct. 2000, vol. 2, pp. 1677 –1680.

[24] C. Leavens, R. Williams, F.S. Foster, P.N. Burns, and M.D. Sherar, "Golay pulse encoding for microbubble contrast imaging in ultrasound," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 54, no. 10, pp. 2082 –2090, Oct. 2007.

[25] R. Y. Chiao and Xiaohui Hao, "Coded excitation for diagnostic ultrasound: a system developer's perspective," in *Proceedings of IEEE Symposium on Ultrasonics*, Oct. 2003, vol. 1, pp. 437 – 448.

[26] M. Golay, "Complementary series," *IRE Transactions on Information Theory*, vol. 7, no. 2, pp. 82 –87, April 1961.

[27] Chin Chong Tseng and C. Liu, "Complementary sets of sequences," *IEEE Transactions on Information Theory*, vol. 18, no. 5, pp. 644 – 652, Sep. 1972.

[28] Chin Chong Tseng, "Signal multiplexing in surface-wave delay lines using orthogonal pairs of Golay's complementary sequences," *IEEE Transactions on Sonics and Ultrasonics*, vol. 18, no. 2, pp. 103 – 107, Apr. 1971.

[29] R.S.C. Cobbold, *Foundations of Biomedical Ultrasound*, Oxford University Press, USA, 2007.

[30] T. Misaridis and J.A. Jensen, "Use of modulated excitation signals in medical ultrasound. part I: basic concepts and expected benefits," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 52, no. 2, pp. 177 –191, Feb. 2005.

[31] T. Misaridis and J.A. Jensen, "Use of modulated excitation signals in medical ultrasound. part II: design and performance for medical imaging applications," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 52, no. 2, pp. 192 –207, Feb. 2005.

[32] R.Y. Chiao and L.J. Thomas III, "Method and apparatus for ultrasonic synthetic transmit aperture imaging using orthogonal complementary codes," Apr. 11 2000, US Patent 6,048,315.

[33] R.Y. Chiao, L. J. Thomas, and S.D. Silverstein, "Sparse array imaging with spatially-encoded transmits," in *Proceedings of IEEE Ultrasonics Symposium*, Oct. 1997, vol. 2, pp. 1679–1682.

[34] F.W. Mauldin, F. Viola, and W.F. Walker, "Complex principal components for robust motion estimation," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 57, no. 11, pp. 2437 –2449, Nov. 2010.

[35] S. Nikolov, K. Gammelmark, and J. Jensen, "Velocity estimation using recursive ultrasound imaging and spatially encoded signals," in *IEEE Ultrasonics Symposium*, 2000, vol. 2, pp. 1473 –1477.

[36] J. Meunier and M. Bertrand, "Ultrasonic texture motion analysis: theory and simulation," *IEEE Transactions on Medical Imaging*, vol. 14, no. 2, pp. 293 –300, June 1995.

[37] D. Fick, R. Dreslinski, B. Giridhar, G. Kim, S. Seo, M. Fojtik, S. Satpathy, Y. Lee, D. Kim, N. Liu, et al., "Centip3De: a 3930 DMIPS/W configurable near-threshold 3D stacked system with 64 arm cortex-m3 cores," in *Proc. of International Solid-State Circuits Conference*, Feb. 2012.

[38] Taeho Kgil, Shaun D'Souza, Ali Saidi, Nathan Binkert, Ronald Dreslinski, Trevor Mudge, Steven Reinhardt, and Krisztian Flautner, "Picoserver: using 3D stacking technology to enable a compact energy efficient chip multiprocessor," in *Proc. of the 12th Conf. on Arch. Support for Programming Languages and Operating Systems*, 2006.

[39] Gabriel H. Loh, "3D-stacked memory architectures for multi-core processors," in *Proc. of the 35th International Symp. on Computer Architecture*, 2008.

[40] Bryan Black, Murali Annavaram, Ned Brekelbaum, John DeVale, Lei Jiang, Gabriel H. Loh, Don McCaule, Pat Morrow, Donald W. Nelson, et al., "Die stacking (3D) microarchitecture," in *Proc. of the 39th International Symp. on Microarchitecture*, 2006.

[41] M. Karaman, H. S. Bilge, and M. O'Donnell, "Adaptive multi-element synthetic aperture imaging with motion and phase aberration correction," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 45, no. 4, pp. 1077 –1087, July 1998.

[42] David Patrick Magee, "Iterative time delay values for ultrasound beamforming," 2010.

[43] Murtaza Ali, Dave Magee, and Udayan Dasgupta, "Signal Processing Overview of Ultrasound Systems for Medical Imaging," Tech. Rep., Texas Instraments, Nov 2008.

[44] K.T. Malladi, F.A. Nothaft, K. Periyathambi, B.C. Lee, C. Kozyrakis, and M. Horowitz, "Towards energy-proportional datacenter memory with mobile DRAM," *Proc. of 39th International Symp. on Computer Architecture (ISCA '12)*, pp. 37–48, June 2012.

[45] R. Palmer, J. Poulton, B. Leibowitz, Y. Frans, S. Li, A. Fuller, J. Eyles, J. Wilson, M. Aleksic, T. Greer, et al., "A 4.3 gb/s mobile memory interface with power-efficient bandwidth scaling," in *Symp. on VLSI Circuits*, 2009.

[46] J. A. Jensen, "FIELD: A program for simulating ultrasound systems," *10th Nordicbaltic Conference on Biomedical Imaging, Vol. 4, Supplement 1, Part 1:351–353*, pp. 351–353, 1996.

[47] J. A. Jensen and N. B. Svendsen, "Calculation of pressure fields from arbitrarily shaped, apodized, and excited ultrasound transducers," *IEEE Transactions on Ultrasonics Ferroelectrics and Frequency Control*, vol. 39, no. 2, pp. 262 –267, Mar. 1992.

[48] ARM, "Cortex-m3 40g specifications," http://www.arm.com/products /processors/cortex-m/cortex-m3.php.

[49] B. Verbruggen, M. Iriguchi, and J. Craninckx, "A 1.7mw 11b 250ms/s 2x interleaved fully dynamic pipelined sar adc in 40nm digital CMOS," in *Proc. of 2012 IEEE International Solid-State Circuits Conf. (ISSCC '12)*, Feb. 2012.

[50] B. Murmann, "ADC performance survey 1997-2013," http://www.stanford.edu/ murmann/adcsurvey.html.

[51] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, "Dark silicon and the end of multicore scaling," *Proc. of the 38th International Symp. on Computer Architecture (ISCA '11)*, pp. 365–376, June 2011.

[52] M. Cetin, E. Bossy, R. Cleveland, and W. C. Karl, "Sparsity-driven sparse-aperture ultrasound imaging," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006, vol. 2, pp. 89–92.

[53] M. Karaman, I. O. Wygant, O. Oralkan, and B. T. Khuri-Yakub, "Minimally redundant 2-D array designs for 3-D medical ultrasound imaging," *IEEE Transactions on Medical Imaging*, vol. 28, no. 7, pp. 1051 –1061, July 2009.

[54] B. Diarra, H. Liebgott, M. Robini, P. Tortoli, and C. Cachard, "Optimized 2D array design for ultrasound imaging," in *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, 2012, pp. 2718–2722.

[55] A. Tawfik, S. Stergiopoulos, and A. C.Dhanantwari, "A generic beamforming structure allowing implementation of adaptive processing schemes for 2-D and 3-D arrays of sensors," in *Proceedings of the MTS/IEEE OCEANS Conference*, 1997, vol. 1, pp. 369–373.

[56] S. Stergiopoulos, *Advanced signal processing handbook: theory and implementation for radar, sonar, and medical imaging real time systems*, CRC press, 2010.

[57] A. C. Dhanantwari, S. Stergiopoulos, L. Song, C. Parodi, F. Bertor, P. Pellegretti, and A. Questa, "An efficient 3D beamformer implementation for real-time 4D ultrasound systems deploying planar array probes," in *Proceedings of IEEE Ultrasonics Symposium*, Aug. 2004, vol. 2, pp. 1421–1424.

[58] K. Owen, M. I. Fuller, and J. A. Hossack, "Application of X-Y separable 2-D array beamforming for increased frame rate and energy efficiency in handheld devices," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 59, no. 7, pp. 1332–1343, July 2012.

[59] B Dacorogna, *Introduction to the Calculus of Variations*, World Scientific, 2004.

[60] O. Oralkan, A. S. Ergun, J. A. Johnson, M. Karaman, U. Demirci, K. Kaviani, T. H. Lee, and B. T. Khuri-Yakub, "Capacitive micromachined ultrasonic transducers: next-generation arrays for acoustic imaging?," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 49, no. 11, pp. 1596–1610, Nov. 2002.

[61] G. Montaldo, M. Tanter, J. Bercoff, N. Benech, and M. Fink, "Coherent plane-wave compounding for very high frame rate ultrasonography and transient elastography," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 56, no. 3, pp. 489–506, March 2009.

[62] D. Fick, R.G. Dreslinski, B. Giridhar, Gyouho Kim, Sangwon Seo, M. Fojtik, S. Satpathy, Yoonmyung Lee, Daeyeon Kim, N. Liu, M. Wieckowski, G. Chen, T. Mudge, D Sylvester, and D. Blaauw, "Centip3De: A 3930DMIPS/W configurable near-threshold 3d stacked system with 64 ARM Cortex-M3 cores," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, Feb 2012, pp. 190–192.

[63] Yurong Hong, Xueming Liu, Zhiyu Li, Xiufang Zhang, Meifeng Chen, and Zhiyan Luo, "Real-time ultrasound elastography in the differential diagnosis of benign and malignant thyroid nodules," *Journal of Ultrasound in Medicine*, vol. 28, no. 7, pp. 861–867, 2009.

[64] Hui Zhi, Bing Ou, Bao-Ming Luo, Xia Feng, Yan-Ling Wen, and Hai-Yun Yang, "Comparison of ultrasound elastography, mammography, and sonography in the diagnosis of solid breast lesions," *Journal of Ultrasound in Medicine*, vol. 26, no. 6, pp. 807–815, 2007.

[65] J. Bercoff, M. Tanter, and M. Fink, "Supersonic shear imaging: a new technique for soft tissue elasticity mapping," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 51, no. 4, pp. 396–409, April 2004.