2017

Modeling and Simulation of Acoustic Pressure Field for Ultrasonic Tactile Displays

Hyo Won Park
Lehigh University

Modeling and Simulation of Acoustic Pressure Field for Ultrasonic Tactile Displays

by

Hyo Won Park

A Thesis

Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy of the Degree of

Master of Science

in

Electrical Engineering

Lehigh University

August, 2017

This thesis is accepted and approved in partial fulfillment of the requirements for the Master of Science.

_____
Date

_____Miltiadis K. Hatalis
Thesis Advisor

_____Svetlana Tatic-Lucic
Chairperson of Department

TABLE OF CONTENTS

ABSTRACT

As the virtual and augmented reality industry continues to grow, it is important to develop a tactile display technology that can seamlessly integrate into a multimodal VR experience. Ultrasonic haptic display technology uses a phased array of ultrasound transducers to create a mid-air pressure focal point, and a modulation of this radiation field at a frequency around 100-300 Hz can stimulate the mechanoreceptors in the skin to produce a tactile sensation. Optimizing this technology to create a strong pressure intensity and focality at low cost and in small space can help open up a new commercial market for tactile displays.

This study explores the creation of a simple and modularized pressure field simulator for ultrasonic haptic displays using a simplified model of transducer radiation pattern. The radiation behavior is broken down to a combination of an on-axis radiation behavior and a directivity behavior, each modeled by an exponential and a Gaussian function, respectively. Then, some physical characteristics of phased array are examined to evaluate their influence on peak intensity of focal peak, focal radius, and number of significant secondary focal peaks. The results of the simulator are then compared against the real pressure field of a haptic display prototype.

CHAPTER 1: INTRODUCTION AND BACKGROUND

It may come to surprise to some, especially those who may consider words "augmented reality" (AR) or "virtual reality" (VR) as buzzwords that have started to dominate the tech news only recently, that the first VR head-mounted display (HMD) system was developed in 1968. Created by Ivan Sutherland, The Sword of Damocles is the predecessor to the large commercial VR headsets market, and its age stands as a testament to our society's long-established fascination with using technology to emulate full human sensory experience. Being able to artificially stimulate the human sensory system, which includes sight, hearing, touch, smell, and taste, signifies being able to surpass space and time constraints in sensory experience, as well as being able to strategically control the stimulations to transmit information in a way that expands the human capacity. But despite momentous technological advances in enhancing the visual and auditory experiences, there have yet to be significant improvements in a commercially available VR products that most closely mimic our natural tactile experience.

This is in no way to say that there has not been a substantial history of tactile displays. This is to no surprise, since the skin is the largest organ of human body, and is capable of transmitting a large amount of information through its 16 different classes of receptors. Three of the skin's mechanoreceptors, specifically, the Meissner corpuscle, Merkel disk receptor, and Pacinian corpuscle, are most often stimulated in modern approaches to tactile display [1] due to their closeness to the skin's surface and their

temporal adaptive characteristics. As each of these mechanoreceptors react distinctly to different sense modalities such as stroking, pressure, and vibration, and at different frequency ranges, various approaches to emulate these modalities have been implemented in tactile displays. Some of these approaches include pressure based devices that use a vertically shifting array of pins, vibration based devices using a small array of vibratory pins [2] or a set of vibrating devices on a wearable device (such as *ActiveBelt*) [3], magneto-rheological (MR) devices that use changes in consistency of MR fluids under magnetic field to create shapes, and electro-cutaneous devices that use direct electrical activation of the skin's nerve fibers (such as *SmartTouch*) [4].

Table 1
Human mechanoreceptors and their corresponding sensory modalities

Receptor	Class	Sense modality	Frequency range (most sensitive at)	Receptors (cm^2) at fingertip
Meissner corpuscle	RAI	Stroking, fluttering	10–200 Hz (200–300 Hz)	140
Merkel disk receptor	SAI	Pressure, texture	0.4–100 Hz (7 Hz)	70
Pacinian corpuscle	RAII	Vibration	40–800 Hz (200–300 Hz)	21
Ruffini ending	SAII	Skin stretch	7 Hz	9
Hair follicle	RA	Stroking, fluttering	?	–
Hair	–	Light stroking	?	–
Field	–	Skin stretch	?	–

Figure 1: Mechanoreceptors of skin [1]

Despite these efforts in exploring the modalities of tactile stimulations and methods to fabricate commercial tactile display devices, they have struggled to find a way to integrate with the rising visual VR devices. The main point disjunction emerges from the fact that while visual VR devices such as HMDs and 3D hologram projectors create illusion of mid-air objects, most of the previously mentioned tactile displays still rely on direct contact between skin and solid display surface. This limitation can easily break the user's immersion into the VR world. As many studies show that the human capacity to quickly react to the changes in the environment is significantly improved

with the use of multimodal displays [1], ability to seamlessly integrate with other sensory displays is pivotal in the success of tactile displays.

In order to mend this point of disjunction, various studies have suggested using ultrasound radiation pressure to create focal points mid-air and create a haptic image that can easily be detected on palms or fingertips. In 2010, Shinoda Lab at University of Tokyo has conducted a promising research on tactile display that creates a focal point of concentrated acoustic pressure using an array of 40kHz ultrasonic transducers. [5] This focal point creates a mid-air haptic sensation that can be easily controlled with a processing circuit. In 2013, the same lab also presented a visuo-tactile projector that incorporated both downward projected moving images and corresponding tactile sensation using this haptic technology, showing that this technology can also be used for information transfer and a dynamic alarm system. [6] The same group then further expanded their project to study and demonstrate the potential of using large scale haptic display to create pinchable 3D images and a tactile mirroring system called HaptoClone. [7][8][9]

In 2013, a research group in University of Bristol also showcased their version of the ultrasonic haptic display called *Ultrahaptics,* which showcased the capability to create multiple focal points and vary them in modulation frequency to create smaller granularity in tactile image. Their display system also has interactive component that uses small IR sensors to detect the position and movement of the hand, adding an important level of robustness and versatility to the technology. [10] The same group

formed a startup company called *Ultrahaptics™ in* order to develop and manufacture consumer products based on these researches. [11]

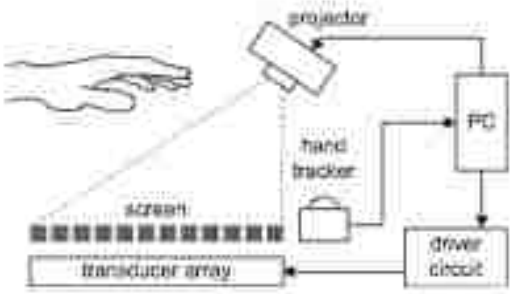


Figure 2: Diagram of UltraHaptics setup [7]

As significant as Shinoda Lab and University of Bristol's researches are to achieving a full tactile integration to the current VR technology, it is still far from being readily incorporated to our day-to-day lives. Their tactile projector requires a complex installation and is very large in size, as well as very costly to build. Lehigh University's Display Research Lab (DRL), for the past years, has been focusing on developing a haptic display system that is relatively low in cost of production, as well as improving the pressure level and focality of haptic points to minimize the size of the display system without compromising its efficiency. DRL has also been exploring methods to improve and diversify the driving methods of tactile display system, while successfully experimenting with novel methods to use ultrasonic radiation pressure field as both an input and output device. [12] [13] [14]

The goal of this MS thesis is to provide an overview of an ultrasonic haptic display interface, including the core theory behind the technology and a hardware

prototype designed and assembled to test its capacity. This thesis will especially focus on a simple and modular pressure field simulator that allows further insight into how physical structure of ultrasonic array affects the focality and the intensity of the field, and suggests optimal configurations for the array. Further improvements to the ultrasonic haptic display would greatly strengthen the possibility of tactile integration to the commercial VR industry, which has potential to enhance our lives through advances in entertainment, education, industrial and medical fields as well as enhancing the lives of visually impaired people.

CHAPTER 2: THEORY

Principal theories that compose an ultrasonic haptic display is relatively well-known and simple. Decades-long studies have shown that it is possible to produce a local stimulation of human skin surface by applying pulses of focused ultrasonic pressure, which in turn puts a small shear force on the skin tissue. Numerous studies have explored the main factor responsible for the neural structure stimulation, and have found that it is the radiation force of the ultrasound wave that determines the strength of the shear force, while the pulse length and variation is responsible for the simulation of various tactile sensations. [15]. A controlled region of ultrasonic pressure can be generated through a well-parameterized array of piezoelectric ultrasonic transducers, and a technique of phased array structure that is commonly used in ultrasound imaging and non-destructive testing (NDT) fields is used to generate focused regions of acoustic radiation. In summary, ultrasonic haptic display technology uses application of modulating signal to an array of ultrasonic transducers to create a controlled acoustic pressure field that can stimulate the mechanoreceptors of the skin.

To better understand how ultrasound radiation field can be generated and controlled, it is important to understand some basic theories of ultrasound transducer radiation. For a planar piston ultrasonic transducer that radiate an acoustic beam directly into a medium, with an approximation of an infinite planar baffle, the Rayleigh-Sommerfeld integral is used to model the pressure at point x in space above the planar surface of the transducer.

$$p(x,\omega) = \frac{-i\omega\rho v_0}{2\pi} \int_S \frac{\exp(ikr)}{r} dS$$

Rayleigh-Sommerfeld integral for planar piston transducer:
p=complex acoustic pressure amplitude at point x,
ω =angular frequency of ultrasonic wave,
ρ =distance between point source and center of transducer plane,
v_0 =constant velocity of sound wave at z direction in space S,
k=wave number, r=distance between point source and point x

This model uses the superposition of spherical radiation on point x caused by all the elementary point sources on active piston surface of the transducer, and is very costly to model numerically due to its infinite summation nature. However, if we only consider the far-field region of the radiation, the complexity of the radiation can be reduced significantly. Far field of radiation is area where z, the axis-distance from the point transducer plane to the point of interest, is significantly greater than the transducer radius, and since for the ultrasonic haptic display application, the mid-air display plane is 10-30cm above the transducer plane while the transducers have 4.5-8mm radius, we focus on characterizing the radiation behavior in the far-field. In this far-field region, complex pressure field of the transducer can be broken down into behavior of on-axis pressure (along the z-axis directly above the center of transducer surface) and off-axis pressure to create a simplified model. Simplification of the on-axis pressure uses the circular symmetry in the transducer piston, and directly integrates the above Rayleigh-Sommerfeld integral to the equation below.

$$p(z,\omega) = \rho c v_0 \left[\exp(ikz) - \exp\left(ik\sqrt{z^2 - a^2}\right)\right]$$

$$= \rho c v_0 \left[\delta\left(t - \frac{z}{c}\right) - \delta\left(t - \frac{\sqrt{z^2 + z^2}}{c}\right)\right]$$

Equation for on-axis pressure of circular piston transducer:
z=distance between on-axis point x and the center of transducer surface
a=radius of transducer, c=speed of sound

This mathematically shows that along the perpendicular axis of the planar transducer surface, the pressure can be simply interpreted as the combination of a direct pulse (due to impulse-like motion of the piston) from the face of the transducer and an edge pulse from the round edge of the face. This helps to picture the ultrasonic radiation as cylindrical direct wave in the main beam with curved edge wave that extends beyond the active transducer region.

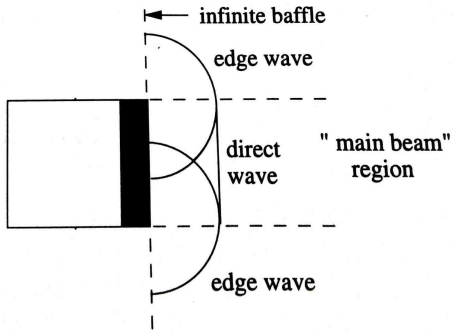


Figure 3: Direct and edge waves for a circular planar transducer [16]

In the far-field (z>>a), the magnitude of the edge wave and the magnitude of the direct wave are relatively equivalent, and the on-axis pressure behavior is summarized to:

$$p(z,\omega) = \frac{-ik\rho c v_0 a^2}{2} \frac{\exp(ikz)}{z}$$

Equation for on-axis pressure of circular piston transducer in far-field region

This equation shows that in the far field, magnitude of the on-axis pressure decays exponentially with distance z.

Compared to the on-axis pressure that is simpler to analytically express, the off-axis pressure is only possible to explicitly obtain in the far-field region of transducer. With the use of spherical coordinate representation, the Rayleigh-Sommerfeld integral is evaluated and reduced to the equation below.

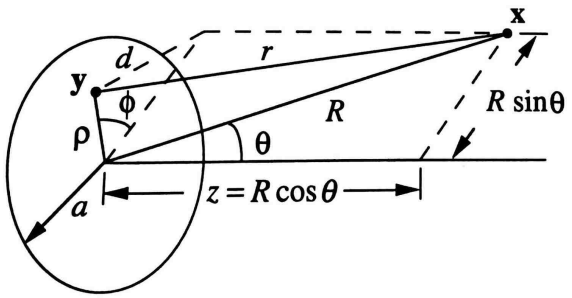


Figure 4: Geometry for calculating off-axis pressure of circular transducer in far-field region [16]

$$p(x,\omega) = -i\omega\rho v_0 a^2 \frac{\exp(ikR)}{R} \frac{J_1(ka\,sin\theta)}{ka\,sin\theta}$$

Equation for off-axis pressure of circular transducer in far-field region:
R=distance from center of transducer plane to the point x
θ =angle between z-axis and line connecting point x and center of transducer
J_1=Bessel function of order one

This characterization of off-axis pressure can be interpreted as a combination of three elements. The first term ($-i\omega\rho v_0 a^2$) shows frequency-dependence, and the second term ($\frac{\exp(ikR)}{R}$) shows spherically spreading wave, while the third Bessel term ($\frac{J_1(ka\,sin\theta)}{ka\,sin\theta}$) shows dependence to the angle from the axis. While the first two terms also

appear in the on-axis response, the angular dependency is unique to the off-axis pressure; moreover the difficulty of analytically modeling a Bessel function in this dependency calls for an alternative approximation method.

In order to address the complexity of analytically modeling the off-axis radiation, multiple modeling methods have been discussed, and one of them is the Gaussian and multi-Gaussian (MG) beam model. A study by Park, Song and Kim in 2006 [17] shows that the ultrasonic pressure radiation can be interpreted as a superposition of two-dimensional Gaussian beams, and when this model was tested for a 5MHz planar transducer, the modelled on-axis and off-axis pressure profiles match the results of Rayleigh-Sommerfeld integral very closely.

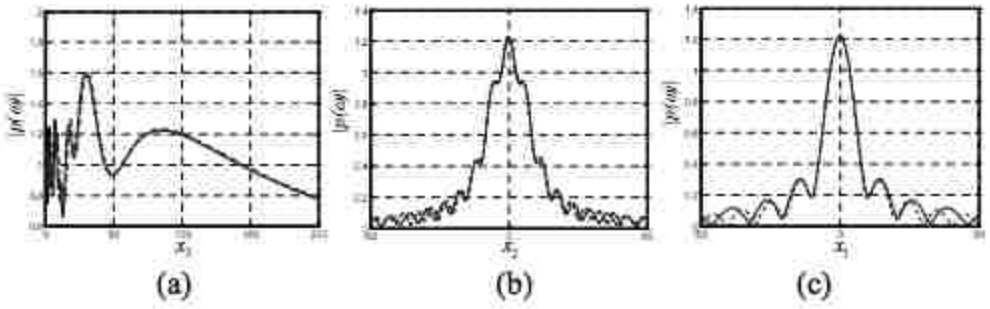


Figure 5: Comparison of (a) on-axis pressure profiles and off-axis pressure profiles (b) along the height direction and (c) along the width direction of 5 MHz, a planar transducer. (solid line: the Rayleigh-Sommerfeld integral, dotted line: the expanded MG beam model) [17]

This multi-Gaussian model has the advantage of being easily calculated and derived analytically. Other studies have also been conducted to address the angular limitation that MG models have in application to phased arrays, such as a proposal to add continuous phase that varies linearly across the face of a transducer by Huang,

Schmerr, and Sedov [18], and have successfully shown the accuracy of MG models in replacing the complex Rayleigh-Sommerfeld integral.

For even simpler representation of the ultrasound pressure field, this MG model can even be reduced into a Gaussian beam. Angular dependency of the radiation magnitude can be characterized by the existence of a large main lobe at small angles, multiple side lobes, and a grating lobe that appears at 90° of the axis. If the transducer directivity characteristic is such that the directivity at the side lobes are negligible, and the transducer array will lay in a plane, as they often are in haptic display application, we can simplify the modeling by concentrating on the main lobe alone. Then, a single Gaussian beam is enough to accurately model the behavior of the off-axis radiation in far-field region.

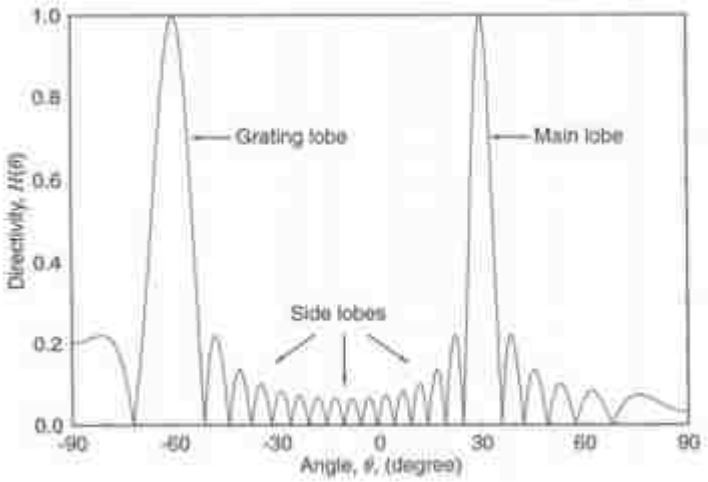


Figure 6: A typical directivity plot of transducer [19]

In order to selectively strengthen the acoustic pressure in a particular spot or pattern, multiple transducers can be integrated to form a phased array. By adding a time delay to individual pulsing transducers in an array, the radiation beam from each array element can have a directed effect to constructively interfere on a point of interest by using Huygen's principle. The required time delay for each element in the array can be easily calculated as:

$$\Delta t = \frac{d}{c}$$

d=distance between source element (each transducer in the array is a point source) and focus point, and c=speed of sound.

The magnitude of pressure at focal point is then the superposition of radiation from individual array element. Due to the directivity of the transducers and the effects of the side lobes, the superposed focal spot in a haptic display is not a single point, and is evaluated on its size, absolute magnitude, relative magnitude to secondary focal spots, and the number of undesired secondary focal spots.

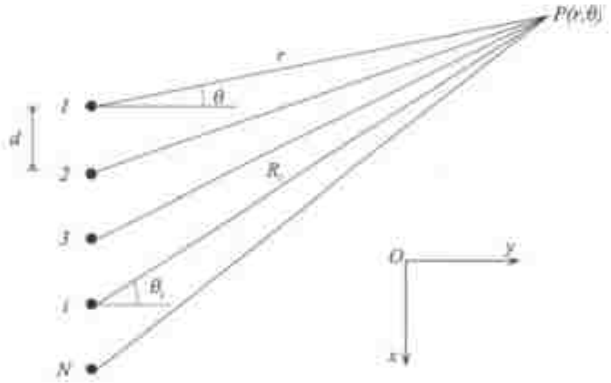


Figure 7: Acoustic waves radiated from an array of simple point sources [19]

Once a selective region of greater acoustic pressure has been formed using a phased array, the radiation pulses then have to be modulated at a lower frequency to stimulate the mechanoreceptors in the skin. Ultrasound frequency range of 20kHz to 20MHz is effective for haptic display application due to its inaudibility and its safety for human body, as it allows harmless multimodal integration between tactile and audible display technologies. But as demonstrated in figure 1, ultrasound on its own has too high of a frequency to stimulate the receptors. Additional pressure modulation of lower frequency around 100-300Hz triggers the Meissner and Pacinian corpuscle in the skin for a neural response, which allows for a simulation of a tactile experience. Both of these mechanoreceptors are also rapidly adapting (RAI and RAII) mechanoreceptors that have transient response to stimulation, allowing for ability to interpret large amount of tactile information. Studies have also shown that varying methods of stimulation such as amplitude modulation (AM), frequency modulation (FM), and pulse-width modulation (PWM) can reproduce different sensations on the skin, such as tickling, itching, pain, temperature, etc. [15]

CHAPTER 3: PRESSURE FIELD SIMULATION

3.1: Building a Pressure Model for Single Transducer

In order to simulate an accurate pressure field map for an ultrasonic haptic display, it is important to first model the radiation pattern of a single transducer accurately. Commercial models of 40kHz ultrasonic transducers are used in prototype versions of the haptic display technology due to their availability and accessibility, but their spec sheets often do not paint a very detailed picture of the radiation field. Thus, it becomes necessary to directly obtain the pressure field data and model the radiation behavior. The task of obtaining detailed pressure field data is accomplished by a low-cost Field Characterization Robot System originally created by Shtarbanov and Wilson [12], as shown figure below. This robot scans a selected 3D space above a radiating transducer and records the location and radiation magnitude at discrete points, using an 40kHz ultrasonic receiver.

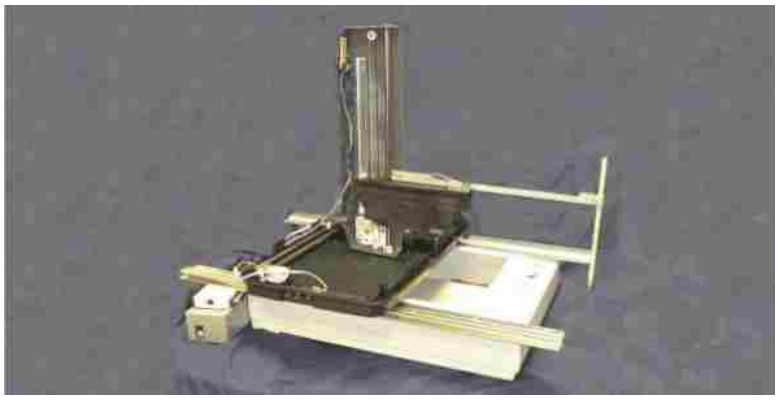


Figure 8: Picture of the Field Characterization Robot System [12]

In this study, a single Murata MA40S4S transducer was activated with a 40kHz pulse wave of 19V amplitude and its radiation pattern in 50cm x 50cm x 50cm space above the transducer surface was studied using MATLAB script [Appendix A]. The same robot and scripts can be used to analyze the radiation pattern of other transducer models.

As discussed in the previous chapter, the scanned field information of approximately 40,000 points was analyzed by dividing it into on-axis pressure behavior along the z-axis (axis normal to the transducer surface) and the off-axis behavior relative to angle from the axis in order to create a simple model. First, 20 points along the z-axis perpendicular to the transducer plane were selected normalized with respect to the maximum pressure at z=0, since this max pressure is subject to the amplitude of applied pulse wave. Then, an exponential curve fit was used to find an analytical function that best models the radiation decay along the axis. The obtained function for the normalized on-axis pressure is:

$$p_1(z) = 1.115436\, exp(-0.052927z)$$

Function for normalized on-axis pressure of Murata MA40S4S transducer

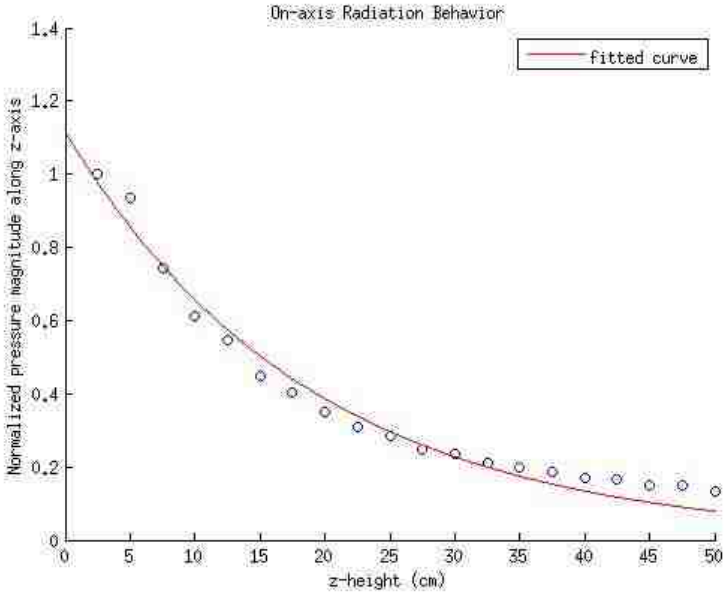


Figure 9: On-Axis Radiation Behavior for Murata MA40S4S transducer. Blue circles indicate scanned pressure magnitudes and the red line indicates an exponential curve fitted to this data set.

This model has a high coefficient of determination of R^2=0.977248, showing that the on-axis radiation of ultrasonic transducers can be represented accurately with a one-dimensional exponential decay equation.

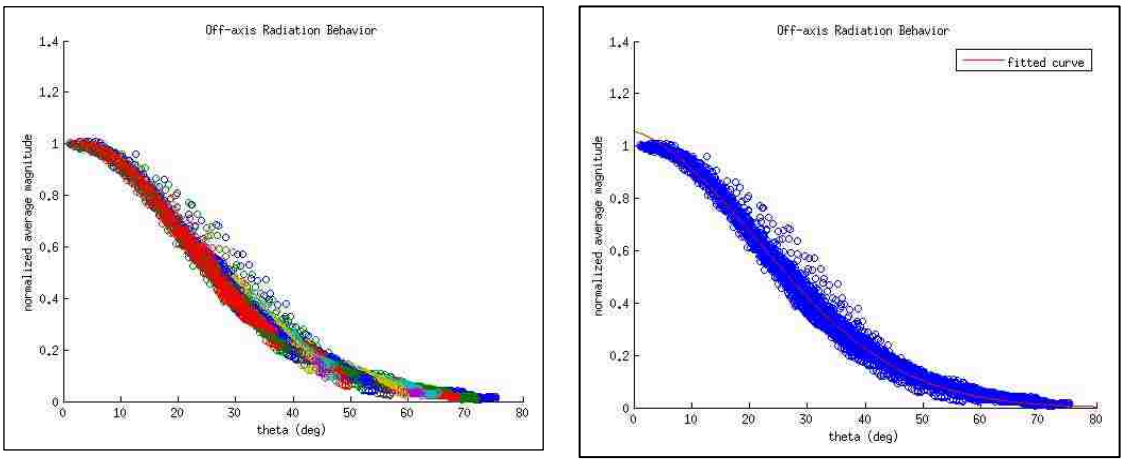
To study the off-axis radiation pattern, the remaining scan magnitudes were categorized by their z-height and the angle θ from the axis using simple geometry. Then, multiple magnitudes with same z-height and θ were averaged out to create a more consistent understanding. These averaged magnitudes were then normalized with respect to the on-axis pressure at the same z-height to isolate only the angle-dependent component of the off-axis behavior. All of these normalized magnitudes can be seen in figure 10.

This data set is fitted to a single Gaussian function as it is sufficient to model the main lobe behavior of the transducer. The obtained function for directivity is as shown below:

$$p_2(\theta) = 1.071272\, exp\left(-\left(\frac{\theta + 4.108851}{35.458904}\right)^2\right)$$

Function for directivity of Murata MA40S4S transducer

This directivity model also has a high coefficient of determination of R^2=0.991386, echoing the studies that proposes Gaussian functions as a good analytical alternative to the Bessel function form of angular dependency.



(a) (b)

Figure 10: Directivity Behavior for Murata MA40S4S transducer. Different colors in (a) indicate different z-heights, while (b) compares the scanned pressure magnitudes in blue with the fitted Gaussian curve in red.

With these two models of on-axis behavior and directivity, the acoustic pressure at point above the surface of the transducer can be expressed as following:

$$p(z, \theta) = p_0 \, p_1(z) \, p_2(\theta)$$

Function for acoustic pressure magnitude at a given point

In this expression, the pressure can be separated into three parts, each independent of each other. p_0 is a constant max pressure that can be obtained at z=0, θ=0at a given input pulse signal (which is characterized by frequency ω and amplitude A). $p_1(z)$ is the z-height dependent on-axis pressure that is modelled by exponential decay. $p_2(\theta)$ is the angle-dependent directivity that is modelled by a Gaussian function.

Despite the simplicity that this model provides, it is important to note its limitations. First of all, the on-axis behavior function $p_1(z)$ is focused on characterizing the exponential trend of far-field radiation, and is not as accurate in the near-field region where the behavior is more complex. Secondly, this model assumes that the $p_1(z)$ factor spreads evenly spherically, and fails to model the behaviors in high θ region accurately. This flaw is noticed when observing the trend of angles where the directivity falls to one-half (-6dB), one-fifth (-14dB), and one-tenth (-20dB) at different z-heights, as shown in figure 11. The expected trend would be a constant of directivity angle across all z-heights, but we see that the desired trend is lost at higher θ region.

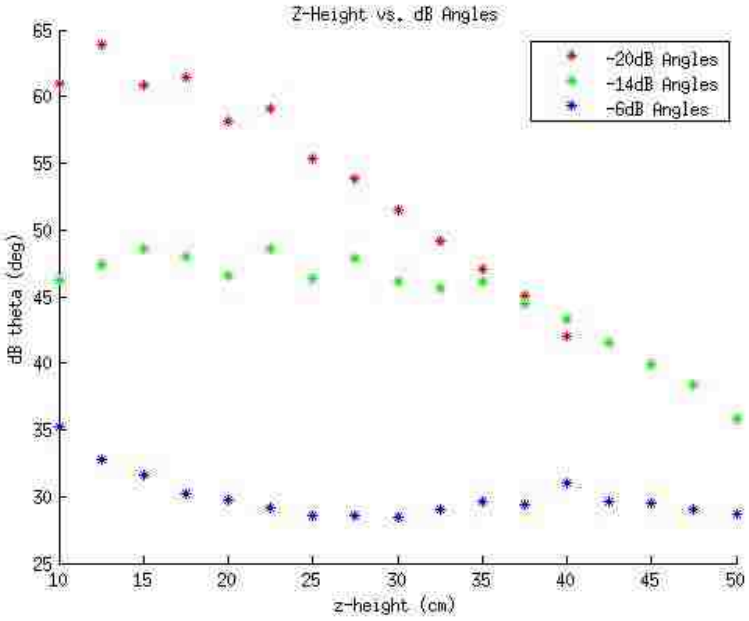
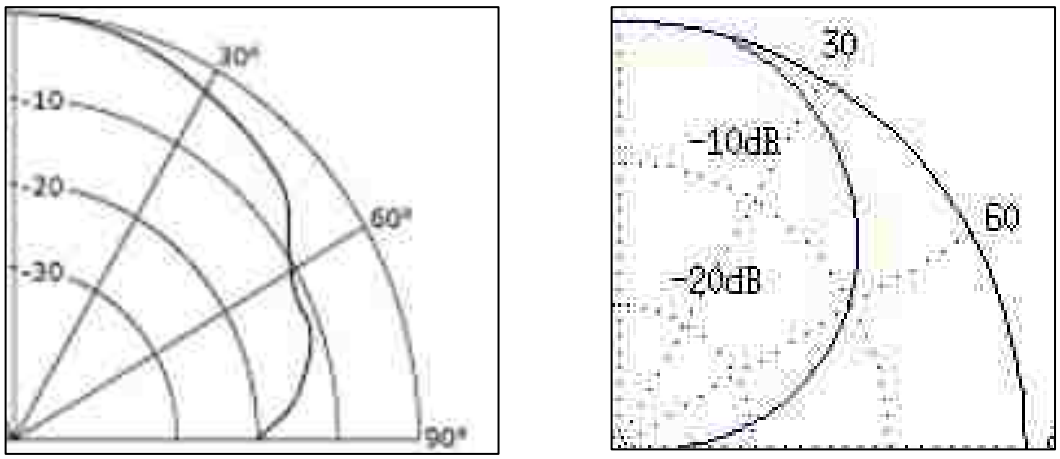


Figure 11: z-height vs. dB Angles for Murata MA40S4S transducer. Each asterisk indicates the angle at which the directivity falls to a particular dB level (indicated by its color) at a specific z-height.

The model above has additional limitations in high θ region due to its focus on the main lobe of radiation. The single Gaussian model of $p_2(\theta)$ shows good accuracy in central region, but does not take into consideration the effects of the side lobes, which weigh much heavily in the high θ region. Figure 12 below compares the polar directivity plots from the MA40S4S spec sheet and the from the single Gaussian model of $p_2(\theta)$, and we can observe that they differ for angles higher than 40°.



(a) (b)

Figure 10: Polar Directivity Plot for Murata MA40S4S transducer from (a) spec sheet [20] and (b) single Gaussian model of directivity.

Despite these limitations, the proposed model is still useful for the application of ultrasonic haptic display. The near field limit is heavily dependent on the diameter of the transducer ($\propto D^2$) and for most expected haptic display usage, the z-height of the desired focus point would be >10 cm while the diameter is usually <2 cm. This ensures that our area of interest is in the far-field region where the exponential model of $p_1(z)$ is valid. Furthermore, the dimension of the whole transducer array is usually smaller than the z-height, restricting the area of interest to low θ region. This θ restriction could be further relaxed if the array is created with much smaller transducers; for such endeavor, the use of a high power CMUT array has been suggested [21] [22]. Thus, the simplified model of transducer pressure radiation is sufficient for accurately simulating ultrasonic haptic display field.

3.2: Simulating a Phased Array

Using the radiation model of a single transducer as detailed above, it is now possible to build a lightweight simulator that maps the pressure field of a phased array. The main goal of the simulator is to be modular and easy to vary the physical characteristics of the array as well as different complexity of radiation models, while maintaining a fast processing speed over large simulation surface. This modularity and speed allows the simulator to have a strong scalability, as to be flexible under further development of tactile display technology. The simulator in Appendix B is built in MATLAB, and takes advantage of MATLAB function modularity.

General process of the simulator is as follows:

1. The user defines parameters such as z-height of the plane to be simulated, number of transducers in the array, transducer diameter, transducer model name, resolution of simulation, and coordinates for the focal point.

2. The simulator creates an array of transducer objects, which has information on the transducer locations. Appendix B includes functions to create random, linear, hexagonal and square/rectangular arrays.

3. The simulator creates an array of distances between each transducer center and the focal point, as well as the time delay to be applied in the signals of each transducer to introduce a phase difference.

4. For each discrete point in the simulation plane (obeying the user-defined resolution), pressure is calculated as a superposition of radiation from each transducer in the array. In other words, this point pressure is the RMS of time-dependent function $P(t)$, where $P(t) = Re(\Pi - wt)$ and the complex sum $\Pi = \sum_{n=1}^{N} p_n e^{i\varphi_n}$ is over N number of transducers. In this definition, p_n is the magnitude of pressure that follows the model described in Part A of this chapter and $\varphi_n = d_n k + \Delta t_n \omega$ is the phase of pressure where d_n is the distance between the n-th transducer and the focal point, k is the wavenumber, and Δt_n is the time delay applied to the n-th transducer.

5. The simulator creates a contour plot of the simulated pressure for the user.

Figure below shows samples of contour plots for different array shapes at target z-height of 10cm above the surface of the transducer using Murata MA40S4S transducer model. Each of the different arrays have been simulated on a 10cm x 10cm xy-plane with a focal point in center of the plane, and show some of the unique characteristics that each array shape contributes to the pressure radiation pattern. These characteristics, as well as some other parameter variations to optimize for both the strength of pressure and focality are discussed in the next part of this chapter.

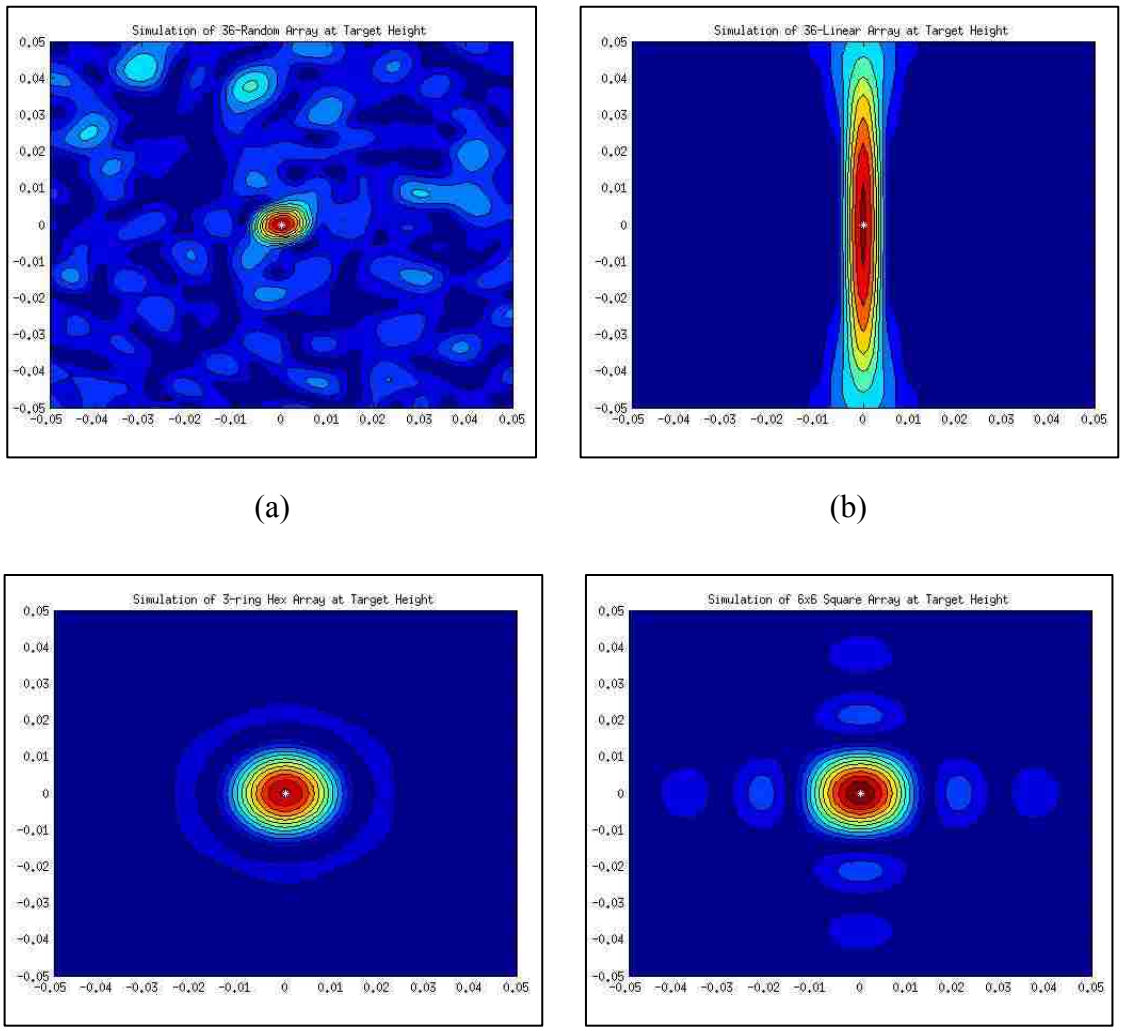


Figure 11: Simulated Pressure Contour Plot for Murata MA40S4S transducer at z-height of 10cm. Each of the arrays have been directed to focus on the center of the plane, indicated by a white point. (a) is for random array of 36 transducers, (b) for horizontal array of 36 transducers, (c) for a heaxagonal / honeycomb array of 37 transducers, and (d) for a 6x6 square array.

3.3: Parameter Optimization

This simple simulator can now also be used to study the effects of different physical parameters of an ultrasonic phased array in order to build a display that best suits specific needs of the user. In the following study, simulated pressure field at a set z-height is evaluated against three different criteria to determine the strength and accuracy and the granularity of a haptic display array. First criterion is the focal radius, which is defined as the beam radius around the max focal point where the point pressures start falling below a set threshold. Next criterion is the peak normalized intensity in the field, which is at the center of targeted focal point unless a critical flaw in the design. Third criterion is the number of sub-focal peaks, which is the number of other local intensity peaks that are above the threshold of the peak intensity. A well-designed display should create a focused pressure spot with a small beam radius that is still detectable on the palm, a strong peak intensity, and a miniscule number of sub-focal peaks that can be mistaken with the main focal point. MATLAB script used to evaluate these three criteria is included in Appendix B.

These criteria of focal radius, peak intensity, and number of sub-focal peaks were used to study the characteristics of each of four array shapes: random, linear, hexagonal, and square. Because arrays with strong symmetry can vary in its ability to create a well-focused pressure point depending on the location of the focal point, the three criteria were evaluated for both a center focal point and an off-center focal point. Table below outlines the result of the study:

For a center focal point at [x=0, y=0, z=10cm]				
	36 Transducer Random Array	*36 Transducer Linear Array*	*37 transducer Hexagonal Array*	*6x6 Square Array*
Focal Radius (mm)	3.7	3.6	9.8	10.1
Peak Normalized Intensity	8.3806	8.4326	13.1136	12.6290
Number of Sub-Peaks (threshold= 50%)	5	0	1	1
For an off-center focal point at [x=4.6cm, y=3.2cm, z=10cm]				
	36 Transducer Random Array	*36 Transducer Linear Array*	*37 transducer Hexagonal Array*	*6x6 Square Array*
Focal Radius (mm)	4.0	22.5	9.0	8.0
Peak Normalized Intensity	9.5266	9.2998	24.2389	23.6878
Number of Sub-Peaks (threshold= 50%)	1	0	0	0

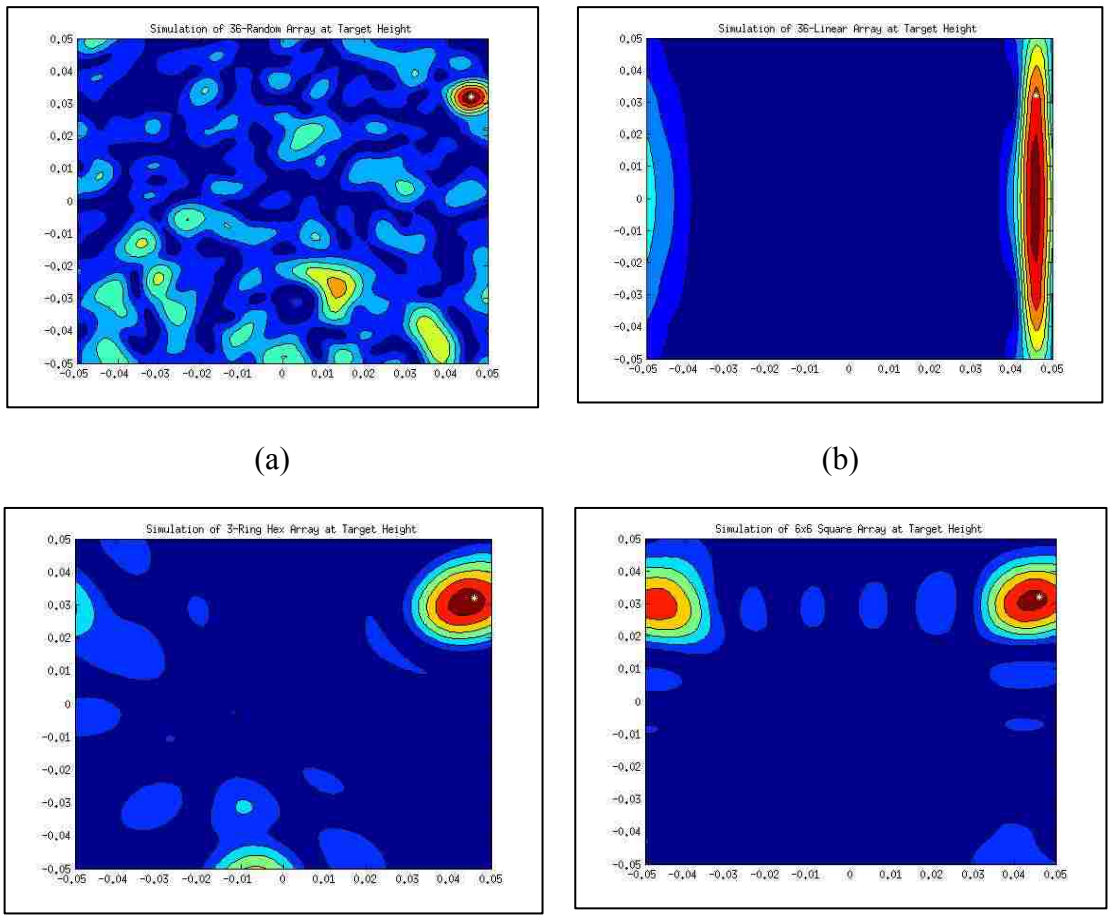


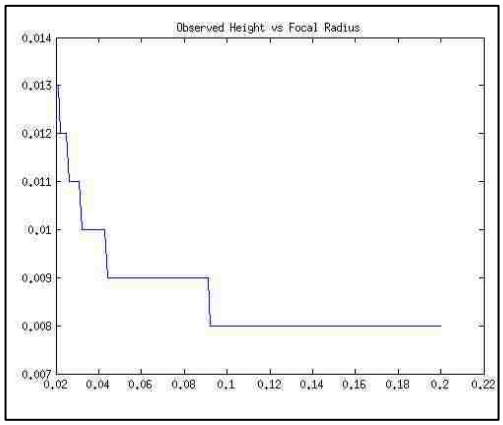
Figure 12: Simulated Pressure Contour Plot for Murata MA40S4S transducer at focal z-height of 10cm. Each of the arrays have been directed to focus on a top right corner focal point, indicated by a white point. (a) is for random array of 36 transducers, (b) for horizontal array of 36 transducers, (c) for a hexagonal / honeycomb array of 37 transducers, and (d) for a 6x6 square array.

There are a couple interesting trends to notice in this set of data. First of all, the array with randomly placed transducers seems to face least change in focal radius and peak intensity between a center focal point and an off-center focal point, mainly due to its lack of symmetry. But it creates less than desirable number of sub-peaks as shown in figure 12-a. Linear array also seems to create constant peak intensity between two points, but has a large focal radius for the off-center point. Moreover, figure 12-b shows

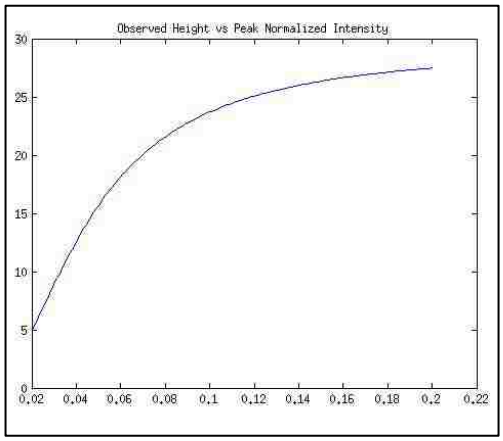
that the concentrated pressure created by the linear array has a large oval shape, and its strongest pressure is not at the focal point. This shows that the linear array is only effective at creating focal points along its axis.

Both the hexagonal and the square array shows promising results as they create strong peak intensity and little sub-peaks in both the center and off-center focal point simulations, while their focal radii are comparable. Hexagonal array has the advantage of being the most-packed structure for circular transducers, but is more complex than the square array to model and manufacture. The square array is highly scalable and easy to design, and is thus a better array shape to manufacture. For this reason, additional parameters such as observation z-height, number of transducers, and inter-element spacing will be simulated with a 6x6 square array at center focal-point.

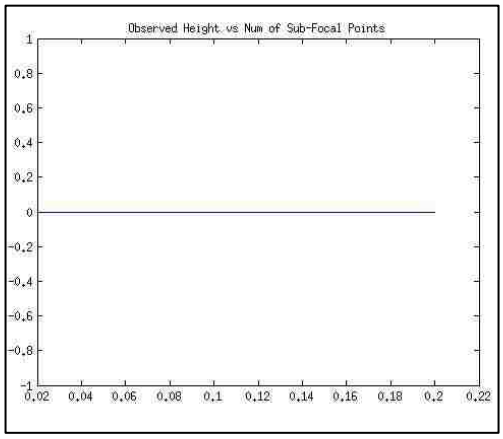
Figure 13 below shows the evaluation of pressure field observed at different z-heights (in range of 2cm-20cm above surface of the array plane) for a square array that is focused at a fixed z=10cm height. We can observe that the focal radius decreases in steps as z-height increases and starts to plateau around the focal height of 10cm, while the peak intensity increases rapidly then starts to slow down around the same height. The number of sub-focal peaks remains 0 along the observed z-range, confirming that the desired concentration of pressure can be achieved at the focal height and above.



(a)

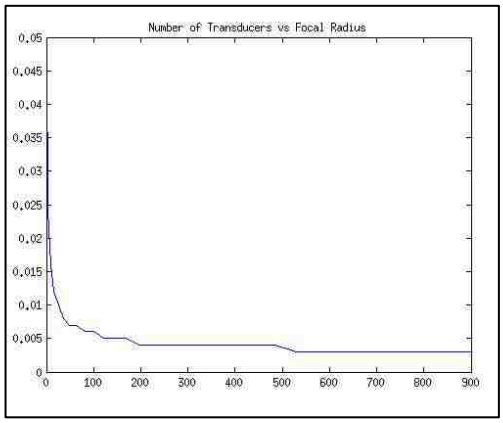


(b)

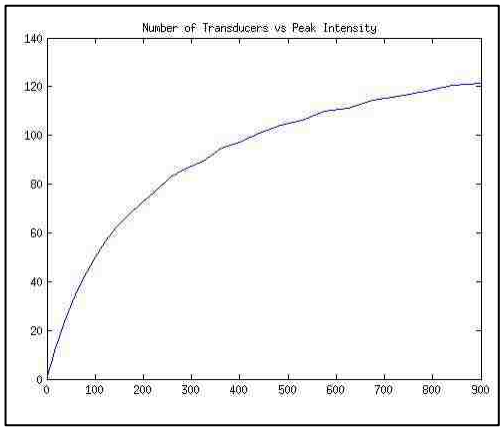


(c)

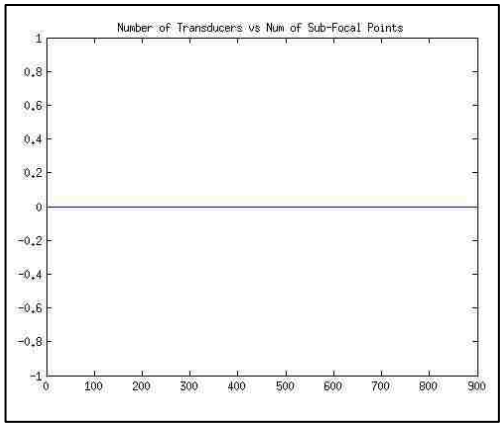
Figure 13: Evaluation of (a) focal radius, (b) peak normalized intensity, and (c) number of sub-focal points (with threshold of 0.5) with respect to the observation height (in m). The simulated pressure field is for a 6x6 square array with focal point at [x=0, y=0, z=10cm]



(a)



(b)

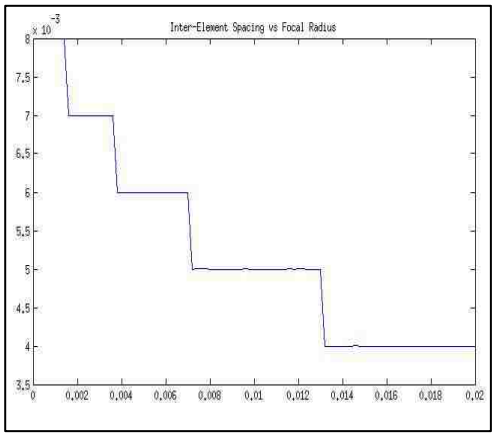


(c)

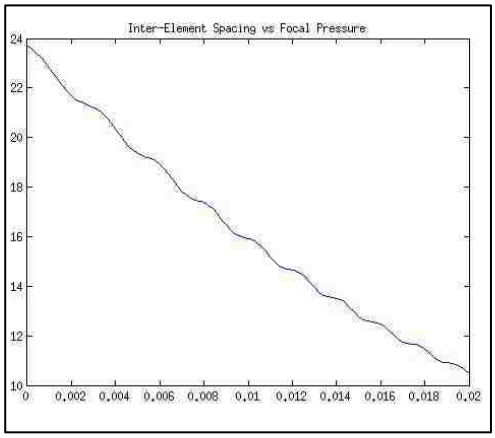
Figure 14: Evaluation of (a) focal radius, (b) peak normalized intensity, and (c) number of sub-focal points (with threshold of 0.5) with respect to the number of transducers in a square array. The simulated pressure fields were at focal height of z=10cm.

To study the effects of the array size, pressure field at focal height were simulated for square arrays of 1x1 size to 30x30 size, and their characteristics were mapped against the number of transducers in figure 14 above. General trends of the focal radius and peak intensity are very similar to that of figure 13, and shows that there is a diminishing return rate as the array size increases, as the focal radius stabilizes around 3mm and the peak intensity grows at a slower rate.

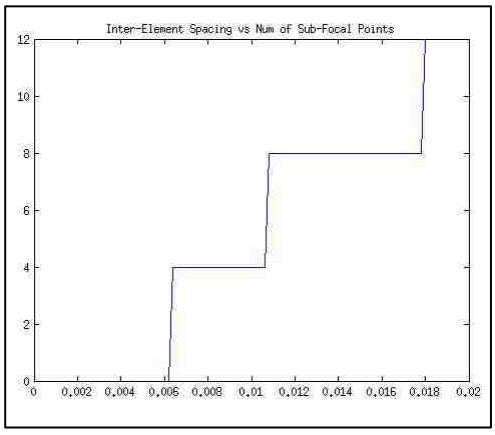
Additionally, a uniform spacing between the transducers in a square array can also affect the intensity and the focality of the pressure field. In their publication, Wooh and Shi discusses the idea of utilizing inter-element spacing to improve the beam directivity in an ultrasonic phased array without producing additional side lobes and grating lobes that can interfere with the main lobe. Simulating the pressure field over a 6x6 square array at focal height with a spacing s between transducers that varies between 0 to $2\lambda \approx 2$cm yields a result that is shown in figure 15.



(a)



(b)



(c)

Figure 15: Evaluation of (a) focal radius, (b) peak normalized intensity, and (c) number of sub-focal points (with threshold of 0.5) with respect to the inter-element spacing (in m). The simulated pressure fields are for a 6x6 square array with focal point at [x=0, y=0, z=10cm].

Figure 15 shows that while the focal radius and peak radiation intensity decay as inter-element spacing increases, there is a point where multiple secondary peaks reach half the magnitude of the main focal peak. This implies that about s=6mm, right before strong sub-peaks start appearing, is the best spacing to obtain a smaller focal radius without losing the definition of the focal point. The conclusion also agrees with the theorized critical inter-element spacing [19] $s_{cr} = \frac{\lambda}{1+sin\theta_s}$ where θ_s is the maximum desired steering angle between the edge of the array and the focal point. For this 6x6 array, this angle $\theta_s = tan^{-1}\left(\frac{3D+2.5s}{z}\right) \approx 25°$.

Array shape, observation height, array size, and inter-element spacing are only a small portion of the wide set of physical and wave parameters that can alter the focality and intensity of an ultrasonic haptic display. The simplified model and MATLAB simulator developed in this chapter provide a method to study these effects quickly and thus can be used to design a phased array that best fits the needs of the display user. Enhancing the design of tactile displays could further expand the possibilities of integrating this technology in future products.

CHAPTER 4: IMPLEMENTATION AND RESULT

To demonstrate the capabilities of ultrasonic phased arrays as tactile display as well as compare the results against the simplified model of transducer radiation pattern, a 6x6 array of Murata MA40S4S transducers was built in lab. As shown in figure 16, the transducers were placed without inter-element spacing to maximize the pressure intensity for a small array and create a focal point pressure that is above the skin threshold of about 8KPa. [21]. The necessary 40kHz signal to excite the transducers were created with Digilent Zybo Zynq-700 ARM/FPGA SoC Board with a modulation of 200Hz to stimulate the mechanoreceptors and appropriate time delay for each of the transducers in the phased array. This 3.3V digital signal was then amplified to a higher voltage through a bootstrap gate drive circuit using IRS2103 half-bridge driver and a pair of NMOS switches to increase the magnitude of pressure. The system is described in details in the operation order below.

1. The ARM core in Zybo board creates time delay information for each transducer based on array location information and user input of the focal point location. These delays are sent to the FPGA in Zybo board and generates the 40kHz pulse with 200Hz modulation. The C++ code for the ARM core and FPGA code in HDL Verilog can also create signals for a dynamically shifting point that switches between multiple focal point locations.

2. The driver circuit shown in the middle of figure 16 amplifies the pulse signal to 20-35V amplitude. A schematic of a single driver circuit is also shown in figure 17 [23]. This bootstrap gate driver circuit is efficient in high frequency and power, and gives the user the flexibility to choose a wide range of pressure magnitude

3. The amplified signal is fed to the 6x6 array of ultrasonic transducers to create a pressure focal point that can be sensed with the palm.

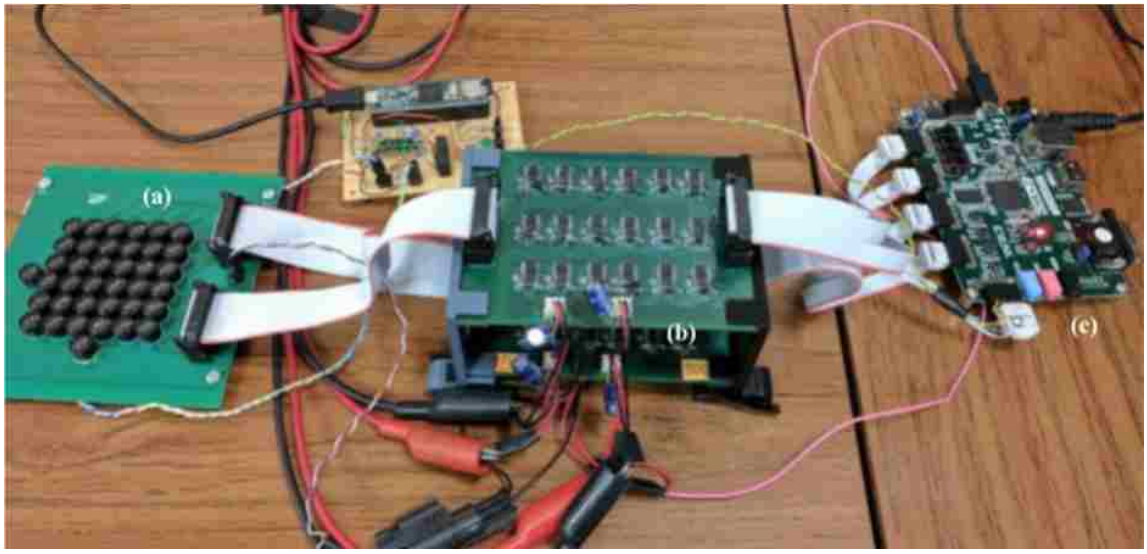


Figure 16: A picture of the ultrasonic haptic display prototype system. Marked with an (a) is the 6x6 array of transducer, (b) indicates the drive circuit board with 18 circuits on each board, and (c) labels the Digilent Zybo board

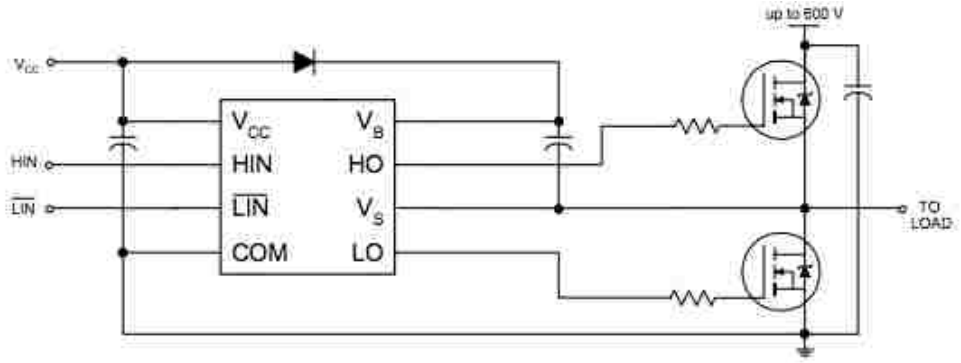
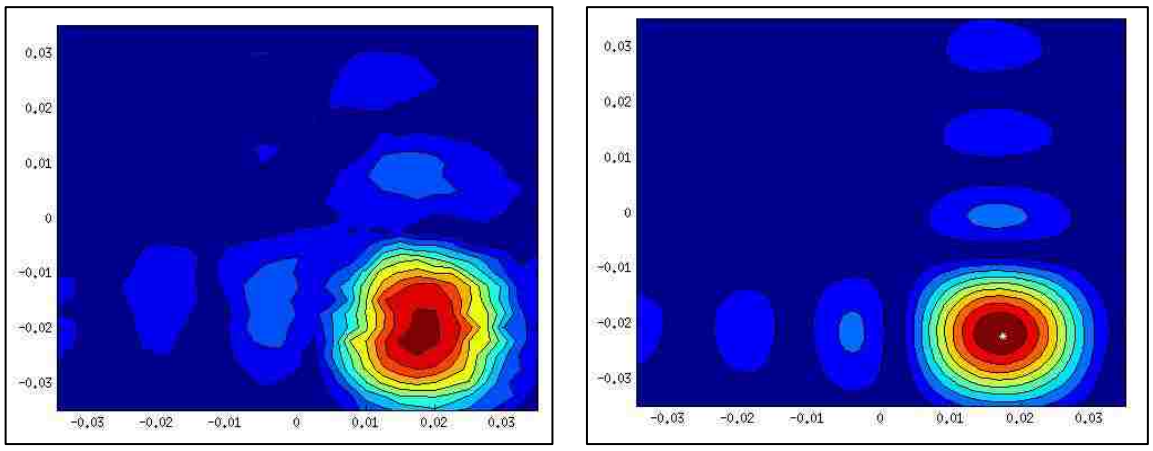


*Figure 17: A schematic of the bootstrap gate drive circuit used to amplify the haptic display signal. For this application, $V_{cc} = 12VDC$, $HIN = \overline{LIN} = 3.3V$ digital signal from Zybo Board, and the NMOS switches were connected to 20-35V DC power sour*ce. [23]

The sensation at the focal point created using this system can be described as being comparable to a gentle air stream with a circular / dome shape of approximately 5mm-10mm radius for any point in the 7cm x 7cm plane above the array at z-height of 10cm.

Scanning and mapping the pressure field of this prototype using the Field Characterization Robot System yields the result shown in figure 18. For this figure, an off-center focal point of [x=1.75cm, y=-2.25cm, z=10cm] was chosen and compared with the simulated pressure field at the focal height.

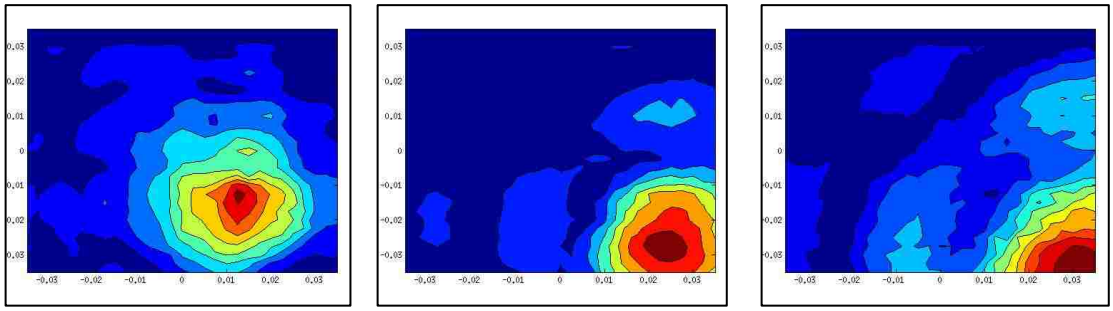


(a) (b)

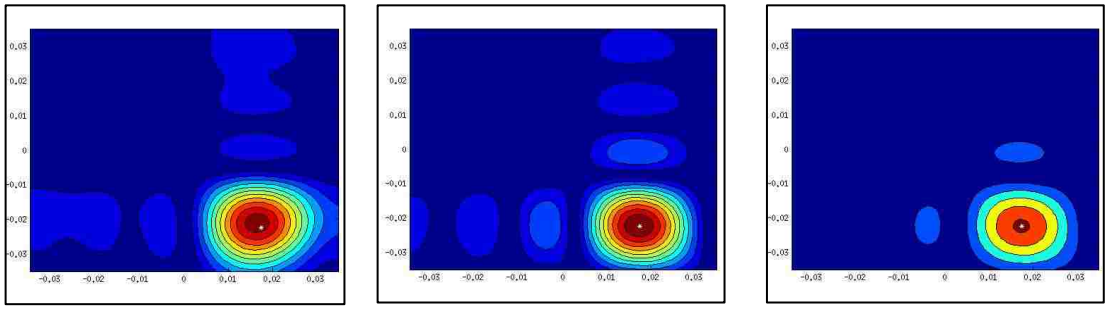
Figure 18: Pressure field of a 6x6 square tactile display array at focal height of z=10cm. Both the (a) scanned and (b) simulated field maps are for a 7cm x 7cm plane and for a focal point at [x=1.75cm, y=-2.25cm, z=10cm].

We can observe that the simulated pressure field mirror the scanned field very closely, and accurately describes the radiation pattern of the main beam as well as the sub-peaks. The scanned field has a focal radius of 10.5mm while the simulated field has a focal radius of 9mm, and both fields do not have a sub-peak that is greater than the threshold of 50%.

On the other hand, the scanned and the simulated fields start to have more discrepancy as we begin to observe at z-heights that is not at the focal height. As shown in figure 19, while the simulated field consistently maintains a main beam around the focal point at z-heights both below and above the focal height, the scanned field shows a distinct shifting of the peak intensity. Both show that the focal radius increases and other sub-peaks emerge; thus it can be concluded that the simulator is most accurate at the focal height.



(a)



(b)

Figure 19: Pressure field of a 6x6 square tactile display array at non-focal height of z=5cm, z=1.5cm, and z=2cm (from left-to-right, respectively). Both the (a) scanned and (b) simulated field maps are for a 7cm x 7cm plane and for a focal point at [x=1.75cm, y=-2.25cm, z=10cm].

CHAPTER 5: CONCLUSION

As the augmented and virtual reality industry continues to grow, it becomes increasingly important to develop a tactile display technology that can seamlessly integrate with the currently popular visual VR products. Ultrasonic haptic display that uses phased array structure to create a concentrated pressure field and stimulate the mechanoreceptors in the skin is a promising field that offers a new direction to offer an immersive multimodal experience to the users in entertainment, medical, and educational fields. To create an efficient array for tactile display application, it is essential to be able to model and simulate the radiation pattern of the phased array before building a physical device, and study both the focality and the intensity of the created focal points to evaluate its capacity to create a sharply defined tactile image.

This study demonstrates the possibility to create an accurate and simple simulator that models an ultrasonic transducer radiation behavior as a combination of an exponential on-axis behavior and a single Gaussian off-axis behavior. It also shows some physical parameters that can be varied to change the focality and intensity of the tactile image, such as array shape, observation height, array size, and inter-element spacing. Simulations show that the hexagonal / honeycomb and square shaped arrays yield the best combination of intensity, focality, and flexibility. When conducting additional simulations with a square array, chosen for its modularity and manufacturability, observation at focal height is recommended, while linearly increasing the array size leads to diminishing return of focality and intensity and adding

a strategic critical inter-element spacing between the transducers can lead to a better focality.

A physical prototype of a 6x6 square array haptic display was built using as seen in figure 16 to compare the real generated pressure field against that of a simulated field. The scan of a field with an off-center focal point shows that the simulator can accurately predict the shape, focal radius, and number of significant sub-peaks accurately at the focal z-height, but starts to show discrepancies as you move away from the focal z-height. Such difference can be attributed to the simplifications in the directivity model for high θ region and a lack of consideration of the radiation's traveling medium (air).

Various modifications could be applied to the current pressure field simulator to increase its accuracy against the real-life haptic display pressure field. One of it can be to create a multi-Gaussian directivity model as opposed to the single-Gaussian model proposed in this paper to more closely predict the behavior of side lobes as discussed in [17]. This may also help to resolve the dB angle discrepancy discussed in chapter III. Further studies of sound pressure behavior as it travels through air can also help to better model the added noise and loss of definition at larger z-heights.

Additionally, current hardware prototype of the haptic display can also be improved to elevate the state of ultrasonic tactile technology to a commercially viable product. Principal improvement can be to replace the current 40kHz ultrasound

transducers with a smaller and mass producible transducer. One prospective replacement is using capacitive micro-machined ultrasonic transducer (CMUT) array to create fine display granularity by creating a haptic ‘pixel’ [21] [22], as a single transducer could be less than 1mm in size. Intensity of radiation can also be increased by introducing methods to reduce the energy loss between the display and the skin surface, which can include beamforming methods or acoustic foam filters. The driver circuit can also be enhanced by reducing signal noise and spikes, while also consuming less power.

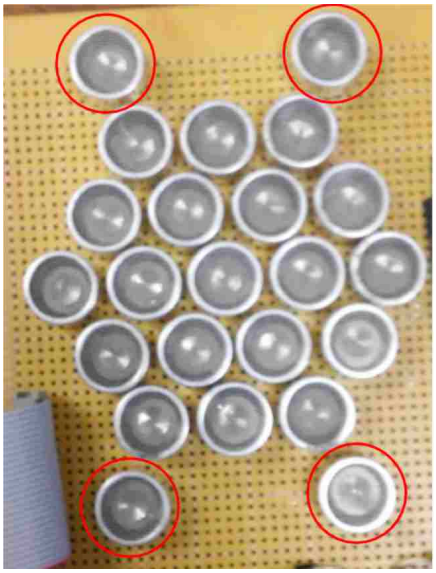


Figure 20: Picture of a 2-ring hexagonal tactile display with 4 sensors (marked with red circles) to detect simple user movement.

While it is crucial to continue to improve the focality and intensity of the haptic display, expanding its applications will also boost the technology’s relevance to the public. Adding a layer of dynamic interactivity to the display using sensors will open up a two-way communication channel between the user and the display, which can be

achieved by using same ultrasound transducers as the display panel. For example, a rudimentary left & right swipe sensor was added to a small hexagonal tactile display as seen in figure 20. The transducers indicated with red circles were used as sensors that detected a change in the radiation field above it as the display user's palm reflected off the radiation. This allowed the device to identify the direction of the hand's movement as they used the tactile display- such information can, for instance, be used to change the page in a handheld e-reader or change the song in a vehicle if tactile displays were to be incorporated. A more sophisticated sensor was also developed by Konings in [14] to also show that a very small number of sensors can be sufficient to detect the location of the tactile display user's finger and greatly increase the data capacity of the sensors.

REFERENCES

[1] V. Chouvardas, A. Miliou and M. Hatalis, "Tactile displays: Overview and recent advances", *Displays*, vol. 29, no. 3, pp. 185-194, 2008.

[2] Y. Ikei, K. Wakamatsu and S. Fukuda, "Texture presentation by vibratory tactile display-image based presentation of a tactile texture", *Proceedings of IEEE 1997 Annual International Symposium on Virtual Reality*, 1997.

[3] K. Tsukada and M. Yasumura, "ActiveBelt: Belt-Type Wearable Tactile Display for Directional Navigation", *UbiComp 2004: Ubiquitous Computing*, pp. 384-399, 2004.

[4] H. Kajimoto, N. Kawakami, S. Tachi and M. Inami, "SmartTouch: electric skin to touch the untouchable", *IEEE Computer Graphics and Applications*, vol. 24, no. 1, pp. 36-43, 2004.

[5] T. Hoshi, M. Takahashi, T. Iwamoto and H. Shinoda, "Noncontact Tactile Display Based on Radiation Pressure of Airborne Ultrasound", *IEEE Transactions on Haptics*, vol. 3, no. 3, pp. 155-165, 2010.

[6] Keisuke Hasegawa, Hiroyuki Shinoda: Aerial Display of Vibrotactile Sensation with High Spatial-Temporal Resolution using Large-Aperture Airborne Ultrasound Phased Array, Proc. IEEE World Haptics Conference 2013, Oral, pp.31-36, Daejeon, Korea, April 14-18, 2013.

[7] S. Inoue and H. Shinoda, "A pinchable aerial virtual sphere by acoustic ultrasound stationary wave", *2014 IEEE Haptics Symposium (HAPTICS)*, 2014.

[8] S. Inoue, Y. Makino and H. Shinoda, "Active touch perception produced by airborne ultrasonic haptic hologram", *2015 IEEE World Haptics Conference (WHC)*, 2015.

[9] Y. Makino, Y. Furuyama, S. Inoue and H. Shinoda, "HaptoClone (Haptic-Optical Clone) for Mutual Tele-Environment by Real-time 3D Image Transfer with Midair Force Feedback", *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, 2016.

[10] T. Carter, S. Seah, B. Long, B. Drinkwater and S. Subramanian, "UltraHaptics", *Proceedings of the 26th annual ACM symposium on User interface software and technology - UIST '13*, 2013.

[11] "Ultrahaptics - A remarkable connection with technology", *Ultrahaptics*, 2017. [Online]. Available: https://www.ultrahaptics.com/.

[12] A. Shtarbanov and M. Wilson, "Senior Design Final Report", Lehigh University.

[13] R. Alvarez and H. Park, "Final Report: Visuo-Tactile Display", Lehigh University.

[14] A. Konings, "Design of an Acoustic Target Detector for Ultrasonic Tactile Displays", Lehigh University.

[15] L. Gavrilov, "The possibility of generating focal regions of complex configurations in application to the problems of stimulation of human receptor structures by focused ultrasound", *Acoustical Physics*, vol. 54, no. 2, pp. 269-278, 2008.

[16] L. Schmerr Jr., *Fundamentals of Ultrasonic Nondestructive Evaluation*. Boston, MA: Springer US, 1998, pp. 159-166.

[17] J. Park, S. Song and H. Kim, "Calculation of Radiation Beam Field from Phased Array Ultrasonic Transducers Using Expanded Multi-Gaussian Beam Model", *Solid State Phenomena*, vol. 110, pp. 163-168, 2006.

[18] R. Huang, L. Schmerr and A. Sedov, "Modeling the radiation of ultrasonic phased-array transducers with Gaussian beams", *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 55, no. 12, pp. 2692-2702, 2008.

[19] S. Wooh and Y. Shi, "Optimization of Ultrasonic Phased Arrays", *Review of Progress in Quantitative Nondestructive Evaluation*, pp. 883-890, 1998.

[20] Murata Manufacturing Co., Ltd., "MA40S4S Ultrasonic Sensors", *Murata.com*, 2017. [Online]. Available: http://www.murata.com/en-us/products/productdetail?partno=MA40S4S.

[21] V. Chouvardas, M. Hatalis and A. Miliou, "Design and simulation of a tactile display based on a CMUT array", *International Journal of Electronics*, vol. 99, no. 10, pp. 1351-1363, 2012.

[22] V. Chouvardas, M. Hatalis and A. Miliou, "Optimizing the design of a tactile display based on a capacitive micromachined ultrasonic transducer array", *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 26, no. 5, pp. 448-456, 2012.

[23] International Rectifier, "IRS2103(S)PbF Datasheet", *Infineon.com*, 2017. [Online]. Available:

https://www.infineon.com/dgdl/irs2103.pdf?fileId=5546d462533600a40153567
62b71279f.

APPENDIX A: RADIATION PATTERN CHARACTERIZATION SCRIPT

Following MATLAB script was written and run with MathWorks,Inc.'s MATLAB ®, R2012b edition.

```
clear all; close all; clc;

%Import scan file of single transducer
filename='log_19v_Murata.csv'; thres=0.1;
M=csvread(filename,1,0);
Data=M(4:size(M),:);
Data(:,5)=[];
noise=min(Data(:,4));
Data(:,4)=Data(:,4)-noise;

% Define start, end, and resolution for eaxh axis
x_start = M(1,1); y_start = M(1,2); z_start =
M(1,3);
x_size = M(2,1); y_size = M(2,2); z_size =
M(2,3);
x_res = M(3,1); y_res = M(3,2); z_res = M(3,3);
total_z_points=round(z_size/z_res)+1;
total_xy_points=length(Data)/total_z_points;

%Initialize arrays
theta_mag=zeros(total_xy_points,
total_z_points*4);
mag_z=zeros(total_z_points,2);

%Find center
[max,center_index]=max(Data(:,4));
center_x=Data(center_index,1);
center_y=Data(center_index,2);
center_z=Data(center_index,3);

 for i=1:size(Data,1);
    x=Data(i,1);
    y=Data(i,2);
    z=Data(i,3);
    mag=Data(i,4);
    z_point=(z/z_res)+1;

      %Record magnitudes at z-axis points
      if (x==center_x) && (y==center_y)
         mag_z(z_point,1)=z;
         mag_z(z_point,2)=mag;
      end

    col_index=(z_point-1)*4+1;
    row_index=floor(i/total_z_points)+1;

    %Calculate angle theta from z-axis
    theta=atand(sqrt((x-center_x)^2+(y-
center_y)^2)/z);
    theta_mag(row_index,col_index)=theta;
    theta_mag(row_index, col_index+1)=mag;
 end

for j=1:total_z_points
   col_index=((j-1)*4)+3;

   %Normalize off-axis magnitudes wrt to on-
axis magnitude at z-height
   for k=1:total_xy_points

theta_mag(k,col_index)=theta_mag(k,col_index
-1)/mag_z(j,2);
   end
end

%Evaluate the dB angle behavior
figure
average=[];
avg_matrix=[];
twentydb=zeros(total_z_points-1, 2);
fourteendb=zeros(total_z_points-1,2);
sixdb=zeros(total_z_points-1,2);

for l=4:total_z_points-1
z_interest=l*z_res;
z_interest_index=((z_interest/z_res)*4+1);
theta_mag_plane=theta_mag(:,z_interest_index)
;
theta_mag_plane(:,2)=theta_mag(:,z_interest_in
dex+2);
theta_mag_plane=sortrows(theta_mag_plane);
avg_plane=[];

for k=2:total_xy_points
   last_row=size(avg_plane,1);

   if
theta_mag_plane(k,1)~=theta_mag_plane(k-1,1)
     avg_plane=[avg_plane;
theta_mag_plane(k,1),theta_mag_plane(k,2)];

   elseif k~=2
```

```
avg=(avg_plane(last_row,2)+theta_mag_plane(
k,2))/2;
      avg_plane(last_row,2)=avg;
      avg=round(avg*10)/10;

       %20dB angles
       if (avg==0.1)
          twentydb(l,1)=z_interest;
          twentydb(l,2)=avg_plane(last_row,1);
       end

       %14dB angles
       if (avg==0.2)
          fourteendb(l,1)=z_interest;
          fourteendb(l,2)=avg_plane(last_row,1);
       end

       %6dB angles
       if (avg==0.5)
          sixdb(l,1)=z_interest;
          sixdb(l,2)=avg_plane(last_row,1);
       end
    end
end

average=cat(3,average,avg_plane);
avg_matrix=[avg_matrix; avg_plane];

%plot off-axis magnitudes vs. theta across all
planes
scatter(avg_plane(:,1), avg_plane(:,2));
hold on
end

title('Off-axis Radiation Behavior');
xlabel('theta (deg)');
ylabel ('normalized average magnitude');

twentydb(~any(twentydb,2),:)=[];
fourteendb(~any(fourteendb,2),:)=[];
sixdb(~any(sixdb,2),:)=[];
mag_z=mag_z(2:size(mag_z,1),:);
mag_z_nor=(mag_z(:,2)/mag_z(1,2));

%plot on-axis radiation pattern
figure
scatter (mag_z(:,1), mag_z_nor, 'b');
title ('On-axis Radiation Behavior');
hold on

%Look for on-axis model as exponential
function
[curve_fit_z,gof_z]=fit(mag_z(:,1),mag_z_nor,'
exp1');
plot(curve_fit_z, 'r');
xlabel('z-height (cm)');
ylabel('Normalized pressure magnitude along z-
axis');
fit_eval_z=gof_z.rsquare;
alpha_z=curve_fit_z.a;
beta_z=curve_fit_z.b;
fprintf('\n The exp fit function is f= %f *exp(%f
* x). \n',alpha_z,beta_z);
fprintf('\n The gof_z is= %f \n',fit_eval_z);

%plot dB angle graph
figure
scatter(twentydb(:,1), twentydb(:,2),'r*');
hold on
scatter(fourteendb(:,1), fourteendb(:,2),'g*');
hold on
scatter(sixdb(:,1), sixdb(:,2),'b*');
title('Z-Height vs. dB Angles');
xlabel('z-height (cm)');
ylabel('dB theta (deg)');
legend('-20dB Angles','-14dB Angles','-6dB
Angles');

figure
scatter(avg_matrix(:,1),avg_matrix(:,2));
hold on

%Look for directivity model as Gaussian
function
[curve_fit,gof]=fit(avg_matrix(:,1),avg_matrix(:
,2),'gauss1');
plot(curve_fit,'r');
title('Off-axis Radiation Behavior');
xlabel('theta (deg)');
ylabel ('normalized average magnitude');

fit_eval=gof.rsquare;
alpha=curve_fit.a1;
beta=curve_fit.b1;
gamma=curve_fit.c1;
fprintf('\n The gaussian fit function is f= %f *
exp(-((x- %f)/ %f)^2. \n ',alpha,beta,gamma);
fprintf('\n The gof is= %f \n',fit_eval);

%plot polar directivity graph
figure
ang=0:-0.01:-pi/2;
ang_deg=ang*(180/pi);
expected_directivity=1.071272 .* exp(-
((ang_deg+4.108851)./ 35.458904).^2);
```

```
expected_directivity_db=mag2db(expected_dire
ctivity);
expected_directivity_db(:,(1:30))=0;
%correcting for dB>0 error
polar(ang+pi/2,expected_directivity_db+60);
title('Directivity plot in dB');
```

APPENDIX B: PRESSURE FIELD SIMULATION SCRIPT

Following MATLAB script was written and run with MathWorks,Inc.'s MATLAB ®, R2012b edition. Each custom functions used in the simulator are indicated with their function call name in bold.

MAIN BODY OF SIMULATOR:

```
close all; clear all; clc; close all;

%Define observation height, and transducer
characteristics
z_measure = 100e-3;
xdcr_diameter = 10.5e-3;
xdcr_radius = xdcr_diameter/2;
model_name='murata';

%Define simulation plane size (square plane)
XYmin = -50e-3;
XYres = 1e-3;
XYmax = 50e-3;
X = [XYmin:XYres:XYmax];
Y = X;

%Define target focal point
target_x = 17.5e-3;
target_y=  -22.5e-3;
target_z = 100e-3;
target=[target_x, target_y, target_z];

%Define the shape and size of array
spacing=0e-3;
%xdcr_array=create_lin_array (36,
'horizontal',xdcr_radius);
%xdcr_array=create_rand_array(36,xdcr_radius
);
%xdcr_array=create_hex_array(3, xdcr_radius);
xdcr_array = create_square_array_spacing(6, 6,
xdcr_radius,spacing);
plot_array(xdcr_array, target);

xdcr_array = gen_delays(xdcr_array, target);
distances = gen_distances(xdcr_array, X, Y,
target_z);

P = zeros(length(Y),length(X));
for x=1:length(X);
   for y=1:length(Y);
      %For each point in simulation plane,
calculate RMS pressure
      P(y,x) = get_pressure_point(xdcr_array, ...
         [X(x),Y(y),z_measure], ...

reshape(distances.xy(y,x,:),1,length(xdcr_array)
), ...

reshape(distances.xyz(y,x,:),1,length(xdcr_array
)),model_name);
   end
end

%Evaluation of pressure field starts here
XY_specs = [XYmin, XYres, XYmax;
         XYmin, XYres, XYmax];
focal_thres=0.5;

%Find peak intensity and focal radius
[focal_r, peak_p]=measure_radius(P,
XY_specs, focal_thres);
local_peaks=imregionalmax(P);
num_sub_peaks=-1;

%Find sub-peaks
[local_peak_j,local_peak_i]=find(local_peaks=
=1);
peaks_ji=zeros(length(local_peak_j),2);
peaks_ji(:,1)=rot90(local_peak_j);
peaks_ji(:,2)=rot90(local_peak_i);

for k=1:size(peaks_ji,1)
  point_peak=P(peaks_ji(k,1),peaks_ji(k,2));
   if point_peak>=focal_thres*peak_p
      num_sub_peaks=num_sub_peaks+1;
   end
end

%Plot pressure field
figure
contourf(X,Y,P);
hold on
plot(target_x, target_y, 'w*');
title('Simulated Pressure Field');
figure
mesh(X,Y,P);
```

CREATE_LIN_ARRAY:

```
function lin_array=create_lin_array(num_xdcrs,
orientation, radius )

d=2*radius;
z=0;

if strcmp(orientation, 'vertical')
   if mod(num_xdcrs, 2)
      y=-d*(floor(num_xdcrs/2));
   else
      y=-d*((num_xdcrs/2)-0.5);
   end
   x=0;
   for i=1:num_xdcrs
      lin_array(i)=create_xdcr(radius,[x,y,z]);
      y=y+d;
   end

elseif strcmp(orientation, 'horizontal')

   if mod(num_xdcrs, 2)
      x=-d*(floor(num_xdcrs/2));
   else
      x=-d*((num_xdcrs/2)-0.5);
   end
   y=0;
   for j=1:num_xdcrs
      lin_array(j)=create_xdcr(radius,[x,y,z]);
      x=x+d;
   end
else
disp ('/n Please give a valid orientation');
end
end
```

CREATE_RAND_ARRAY:

```
function rand_array = create_rand_array(
num_xdcr, radius )
%Creates an array with random locations of
transducers (no overlaps)
%within a given X & Y boundaries

i=2;
x_limits=0.5*[-num_xdcr*radius,
num_xdcr*radius];
y_limits=x_limits;
z_plane=0;
first_x=(x_limits(2)-
x_limits(1))*rand+x_limits(1);
first_y=(y_limits(2)-
y_limits(1))*rand+y_limits(1);
rand_array(1)=create_xdcr(radius, [first_x,
first_y,z_plane]);

while i<=num_xdcr
   center_x=(x_limits(2)-
x_limits(1))*rand+x_limits(1);
   center_y=(y_limits(2)-
y_limits(1))*rand+y_limits(1);
   rand_array(i)=create_xdcr(radius, [center_x,
center_y,z_plane]);
   for j=1:length(rand_array)-1
      compr_center=rand_array(j).center;
      distance=sqrt((compr_center(1)-
center_x)^2+(compr_center(2)-center_y)^2);
      if (distance<=radius*2)
        break
      else if (j==length(rand_array)-1)
            i=i+1;
            break
      end
   end
end
end
```

CREATE_HEX_ARRAY:

```
function
xdcr_array=create_hex_array(num_rings,
radius)

   d=2*radius;
   theta_factor=60;
   theta=90;
   x=0;
   y=0;
   z=0;
   xdcr_array(1)=create_xdcr(radius, [x,y,z]);

   if num_rings>=1
      for i=2:7;
         x=d*cosd(theta);
         y=d*sind(theta);
         theta=theta-theta_factor;
         xdcr_array(i)=create_xdcr(radius,
[x,y,z]);
      end
   end
   theta_factor=30;
   theta=120;
   if num_rings>=2
      for i=8:19;
         if mod(i,2)==0
```

```
            d=2*radius*sqrt(3);
         else
            d=4*radius;
         end
         x=d*cosd(theta);
         y=d*sind(theta);
         theta=theta-theta_factor;

xdcr_array(i)=create_xdcr(radius,[x,y,z]);
      end
   end
   theta_factor=20;
   theta=130;
   if num_rings>=3
      for i=20:37;
         if mod(i,3)==1
            d=6*radius;
         else
            d=radius/sind(10);
         end
         x=d*cosd(theta);
         y=d*sind(theta);
         theta=theta-theta_factor;

xdcr_array(i)=create_xdcr(radius,[x,y,z]);
      end
   end
end
```

CREATE_SQUARE_ARRAY_SPACING:

```
function sqr_array =
create_square_array_spacing(num_col,
num_row, radius, s)

d=2*radius+s;
z=0;

if mod(num_row,2)
   y=d*floor(num_row/2);
else
   y=d*((num_row/2)-0.5);
end


for j=1:num_row
   index_base=(j-1)*num_col;

   if mod(num_col,2)
     x=-d*(floor(num_col/2));

   else
     x=-d*((num_row/2)-0.5);
   end

   for i=1:num_col

sqr_array(index_base+i)=create_xdcr(radius,
[x,y,z]);
      x=x+d;
   end

   y=y-d;
end

end
```

CREATE_XDCR:

```
function xdcr_struct=create_xdcr(radius, center)

   xdcr_struct = struct('radius', radius, 'center',
center, 'delay', 0, ...
      'amp', 1);

end
```

PLOT_ARRAY:

```
function array_plot = plot_array(xdcr_array,
target_point)

figure
for h=1:length(xdcr_array)
   hold on
   r=xdcr_array(h).radius;
   x=xdcr_array(h).center(1);
   y=xdcr_array(h).center(2);
   th=0:pi/50:2*pi;
   xunits = r*cos(th)+x;
   yunits = r*sin(th)+y;
   plot (xunits, yunits, 'r')
end
plot(target_point(1), target_point(2), 'b*');
hold off
end
```

GEN_DELAYS:

```
function xdcr_array = gen_delays(xdcr_array,
target_point)

   speed_sound = 343;

   distances=zeros(1,length(xdcr_array));
   %Find distance between each of the xdcr
center to the target
```

```
    for i=1:length(xdcr_array)
        del_x = xdcr_array(i).center(1) -
target_point(1);
        del_y = xdcr_array(i).center(2) -
target_point(2);
        del_z = xdcr_array(i).center(3) -
target_point(3);

        distances(i) = sqrt((del_x^2) + (del_y^2) +
(del_z^2));
    end

    prop_times = zeros(1,length(xdcr_array));
    for i=1:length(xdcr_array)
        prop_times(i) = distances(i)/speed_sound;
    end

    max_prop_time = max(prop_times);
    for i=1:length(xdcr_array)
                xdcr_array(i).delay =
max_prop_time - prop_times(i);
    end
end
```

GEN_DISTANCES:

```
function  distances = gen_distances(xdcr_array,
X, Y, z)

    distances_xy =
zeros(length(Y),length(X),length(xdcr_array));
    distances_xyz =
zeros(length(Y),length(X),length(xdcr_array));
    [X2,Y2] = meshgrid(Y,X);

    for i=1:length(xdcr_array)

        distances_xy(:,:,i) = sqrt( (Y2-
xdcr_array(i).center(2)).^2 + ...
            (X2-xdcr_array(i).center(1)).^2);

        distances_xyz(:,:,i) = sqrt(
(distances_xy(:,:,i).^2) + ...
            ((z - xdcr_array(i).center(3))^2) );
    end

    distances = struct('xy', distances_xy, 'xyz',
distances_xyz);
end
```

GET_PRESSURE_POINT:

```
function Pressure =
get_pressure_point(xdcr_array, point, dist_xy,
dist_xyz,model)

    x = point(1);
    y = point(2);
    z = point(3);

    speed_sound = 343;
    freq = 40e3;
    period = 1/freq;
    w = 2*pi*freq;

    lambda = speed_sound/freq;
    wave_num = (2*pi/lambda);

     delays=cat(2,xdcr_array.delay);
     amplitudes=cat(2,xdcr_array.amp);

    mag=gen_mag(z, dist_xy, amplitudes,
model);
    phi = (dist_xyz.*wave_num) + (delays.*w);
    complex_p = complex(mag.*cos(phi),
mag.*sin(phi));
    sum_complex_p = sum(complex_p);

    time = 0:period/10:5*period;
    Pressure_time =
abs(sum_complex_p)*cos(angle(sum_complex_
p)-w.*time);
    Pressure = rms(Pressure_time);

    test_var = 1;
end
```

GEN_MAG:

```
function magnitude = gen_mag( z_value,
dist_xy, amp, model_name )

if strcmp(model_name,'murata')
    %Model built by Directivty_Calculation.m
analysis on log_19v_murata.csv magnitude
measurement

%Pz=a*exp(b*z) w/ R^2 value of 0.977248
    alpha_z=1.115436;
    beta_z=-0.052927;
    Pz=alpha_z*exp(beta_z*z_value);

%Directivty=a*exp(-((theta-b)/c)^2) w/ R^2
value of 0.991386
```

```
    alpha=1.071272;
    beta=-4.108851;
    gamma=35.458904;
    theta=atand(dist_xy./z_value);
    dir=alpha.*exp(-((theta-beta)./gamma).^2);
    magnitude=Pz.*dir.*amp;

else
    error('Please specify a valid model name');

end
end
```

MEASURE_RADIUS:

```
function [F, max_P] = measure_radius(field,
field_specs, focal_thresh)

close all;

    %get field specs
    x_res = field_specs(1,2); y_res =
field_specs(2,2);
    x_points = ((field_specs(1,3) -
field_specs(1,1)) / x_res) + 1;
    y_points = ((field_specs(2,3) -
field_specs(2,1)) / y_res) + 1;



    max_P=max(max(field));
    [center_j, center_i] = find(field == max_P);

    radius_data = zeros(1,4);
    radius_data_discrete = zeros(1,4);
    threshold = focal_thresh*max_P;

    for i=center_i:1:x_points% cols pos
        temp_p = field(center_j,i);
        if (temp_p < threshold)
            radius_data_discrete(1) = i-1-center_i;
            temp = linear_interp(field(center_j,i-1),
...
                field(center_j,i), threshold);
            radius_data(1) = (i-1-
center_i+temp)*x_res;
            break;
        end
    end

    for i=center_i:-1:1% cols neg
        temp_p = field(center_j,i);
        if (temp_p < threshold)
            radius_data_discrete(2) = center_i-i+1;
            temp = linear_interp(field(center_j,i), ...
                field(center_j,i+1), threshold);
            radius_data(2) = (center_i-(i+1)-
temp)*x_res;
            break;
        end
    end

    for j=center_j:1:y_points% rows pos
        temp_p = field(j,center_i);
        if (temp_p < threshold)
            radius_data_discrete(3) = j-1-center_j;
            temp = linear_interp(field(j-
1,center_i),...
                field(j,center_i), threshold);
            radius_data(3) = (j-1-
center_j+temp)*y_res;
            break;
        end
    end

    for j=center_j:-1:1% rows neg
        temp_p = field(j,center_i);
        if (temp_p < threshold)
            radius_data_discrete(4) = center_j-j+1;
            temp = linear_interp(field(j,center_i), ...
                field(j+1,center_i), threshold);
            radius_data(4) = (center_j-(j+1)-
temp)*y_res;
            break;
        end
    end

    %compute the radius of the beam
    cnt = 0; radius = 0; radius_discrete = 0;
    for i=1:4
        if (radius_data(i) ~= 0)
            cnt = cnt+1;
            radius = radius + radius_data(i);
            radius_discrete = radius_discrete +
radius_data_discrete(i);
        end
    end

    if (cnt == 0)
        radius = 0;
        radius_discrete = 0;
    else
        radius = radius/cnt;
        radius_discrete =
round(radius_discrete/cnt);
    end

    F = radius;
```

```
end

function pos = linear_interp(val1, val2,
val_target)

   %val1 is assumed to be in linear position 0,
and val2 in linear
   %position 1
   if (val1 == val2)
      pos = 0;
   end
   if (val1 < val2)
      pos = (val2-val_target)/(val2-val1);
   else
      pos = 1 - (val_target-val2)/(val1-val2);
   end
end
```

AUTHOR BIOGRAPHY

Hyo Won Park was born in Gumi, South Korea, on February 25, 1994, as the second daughter of Ju Il Park and Kyung Hwa Kim. After receiving a high school degree from American School Foundation of Monterrey in 2012, she attended Lehigh University to obtain a Bachelor of Science degree in Electrical Engineering with the highest honors in May, 2016.

During her study in Lehigh University, she was involved in research opportunities in Lehigh's Display Research Lab, as well as completing a short internship with LG Electronics México in summer of 2013.