Developing a Loudspeaker Beamforming Array to Steer Sound in Three-Dimensions

Sarkis Ter Martirosyan[1] and David Bell[1,2]

[1] *Thomas Jefferson High School for Science and Technology (TJHSST)*
[2] *TJHSST Microelectronics Lab Director*

June 5, 2018

Analyzing the performance of acoustic beamforming has ramifications for radio-frequency beamforming, which is a core component of the budding 5th-Generation Wireless Systems standard. By taking advantage of the constructive and destructive properties of sinusoidal signals in the free-field, beamforming makes it possible to direct radiofrequency signals using static emitters. As beamforming can be applied to both sound waves (with loudspeaker emitters) and electromagnetic waves (with antenna emitters), it is possible to investigate beamforming using acoustic signals, which are slower and more easy to observe. In this article, I develop and explore a system that allows for the beamforming of an arbitrary audio signal in order to solve the problem of personalized sound.

Contents

1 Introduction

The goal of this research was to explore various beamforming algorithms and their performance by developing a mixed hardware-software system for acoustic beamforming. The observations and conclusions drawn from my research on acoustic beamforming are valuable to exploring and solving the personalized sound problem. Furthermore, my acoustic analysis of beamforming algorithms and performance also provides insight on beamforming in telecommunications, as the fundamental underlying mathematics is identical for both electromagnetic and acoustic waves, with the only difference being a change from the speed of sound to the speed of light.

1.1 Problem Definition

To develop a system that uses a loudspeaker array to aim sound at a singular "bright point," a location in the three-dimensional free-field where the signal will be present. All the other points in our field to be "dark points," or points where the signal will be absent (Druyvesteyn and Garas, 1997; Elliott et al., 2012). In this research project, an absence of the signal at a given point is defined as silence at that point.

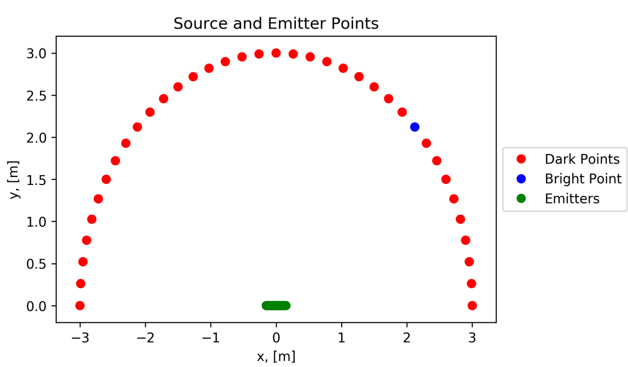

Figure 1

1.2 Mathematical Problem Parameters

The beamforming problem of personalized sound presented in this article is defined by a matrix of M control points $\mathbf{X} \in \mathbb{R}^{3\times M}$ where

$$\mathbf{x}_m = \begin{pmatrix} \rho \cos \frac{\pi(m-1)}{M-1} \\ \rho \sin \frac{\pi(m-1)}{M-1} \\ 0 \end{pmatrix}$$

as the control points are arranged on a semicircle with a radius of ρ meters[1].

Furthermore, a matrix of L emitter points (loudspeakers) is defined as $\mathbf{Y} \in \mathbb{R}^{3\times L}$ where

$$\mathbf{y}_\ell = \begin{pmatrix} (\ell - \frac{L+1}{2})\Delta \\ 0 \\ 0 \end{pmatrix}$$

where Δ is the spacing between loudspeakers in meters[2].

Finally, we can define a frequency-domain representation of a monosignal $\mathrm{s} \in \mathbb{S}_N$, where N is our sampling window[3].

1.3 Complex Mathematical Quantities

The fundamental matrix transformation that defines beamforming is

$$\mathbf{p}(\omega) = \mathbf{Z}(\omega)\mathbf{q}(\omega)s(\omega) \tag{1}$$

where ω is the angular velocity and is defined as $\omega = 2\pi f$, where f is the temporal frequency (Olivieri et al., 2016). The value $s(\omega)$ is defined as the value of the discrete fourier transform bin of the input signal at ω.

The vector of target sound pressures $\hat{\mathbf{p}} \in \mathbb{C}^M$ is the same regardless of what ω we evaluate it at, and is

[1] For this research project $M = 37$ and $\rho = 3$
[2] For this research project $L = 16$ and $\Delta = 0.02$
[3] For this research project $N = 1024$, meaning the signal processing is performed on 1k sized windows

$$\hat{\mathbf{p}} = \begin{pmatrix} 0 \\ \vdots \\ p_B \\ \vdots \\ 0 \end{pmatrix} \tag{2}$$

where p_B signifies the bright point. If the beam was aimed at $90°$, B would be set to 19. The value of p_B is 1, signifying that we want sound to be there, while all the dark points are assigned a value of 0, meaning we don't want sound there.

The vector of loudspeaker weights is $\mathbf{q}(\omega) \in \mathbb{C}^L$ and varies based on the frequency. The aim of the beamforming algorithms detailed in Section 2 is to solve for $\mathbf{q}(\omega)$. In practice, I defined a matrix $\mathbf{Q}$, whose columns are values of $\mathbf{q}(\omega)$ at each frequency of a discrete Fourier transform with N bins on a signal sampled at a rate of 44100 kHz.

The quantities $\mathbf{p}(\omega)$ and $\mathbf{q}(\omega)$ are related by the linear transformation in Equation 1. The elements of $\mathbf{Z}(\omega) \in \mathbb{C}^{M \times L}$ are defined as

$$\mathbf{Z}_{m,\ell}(\omega) = Z(\mathbf{x}_m, \mathbf{y}_\ell, \omega) = \frac{e^{-j\frac{\omega}{c}\|\mathbf{x}_m - \mathbf{y}_\ell\|}}{4\pi\|\mathbf{x}_m - \mathbf{y}_\ell\|} \tag{3}$$

where $j = \sqrt{-1}$ and $\|\cdot\|$ represents the l2-norm operator.

2 Algorithms

In my research project, I explored two widely used beamforming algorithms, Delay and Sum ("DAS") and Pressure Matching ("PM").

2.1 Delay and Sum

Delay and Sum employs the use of constructive and destructive interference to produce unitary pressure at the bright point, and is calculated using this linear transformation

$$\mathbf{q}_{DAS}(\omega) = \mathbf{\Gamma}(\omega)\mathbf{z}_B^\dagger(\omega) \tag{4}$$

where $[\,\cdot\,]^\dagger$ is the complex conjugate transpose and $\mathbf{z}_B(\omega) \in \mathbb{R}^{1 \times L}$ is the row vector of $\mathbf{Z}(\omega)$ corresponding to the bright point (Fink and Prada, 2001). $\mathbf{\Gamma} \in \mathbb{R}^{L \times L}$ is a diagonal matrix of the form

$$\mathbf{\Gamma} = \begin{bmatrix} \gamma_1 & 0 & \dots & 0 \\ 0 & \gamma_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \gamma_L \end{bmatrix}$$

where

$$\gamma_\ell = \frac{16\pi^2\|\mathbf{x}_B - \mathbf{y}_\ell\|}{L}$$

.

2.2 Pressure Matching

Save for Delay and Sum, much of modern beamforming is focused on the optimization of a cost function, referred to as "super-directive beamforming." Pressure Matching is a form of super-directive beamforming, making use of normal equations to solve a least-squares problem (Kirkeby and Nelson, 1993).

All the quantities below have a frequency dependence on ω.

In Pressure Matching, there are two quantities that we are trying to minimize:

1. the complex target field error magnitude $\|\mathbf{e}_{PM}\| = \mathbf{e}_{PM}^\dagger\mathbf{e}_{PM}$
2. the input energy to the emitter (loudspeaker) array $E_\mathbf{q}$

These quantities are defined as follows.

$$\mathbf{e}_{PM} = \hat{\mathbf{p}} - \mathbf{p} = \hat{\mathbf{p}} - \mathbf{Z}\mathbf{q} \tag{5}$$

$$E_\mathbf{q} = \mathbf{q}^\dagger\mathbf{q} \tag{6}$$

The final cost-minimization relationship is defined as

$$\min_{\mathbf{q}} J_{PM} = \min_{\mathbf{q}}(\mathbf{e}_{PM}^{\dagger}\mathbf{e}_{PM} + \beta_{PM}E_{\mathbf{q}}) \quad (7)$$

where β_{PM} is a Tikhnonov regularization parameter. The solution to this minimization problem statement is

$$\mathbf{q}_{PM} = (\mathbf{Z}^{\dagger}\mathbf{Z} + \beta_{PM}\mathbf{I})^{-1}\mathbf{Z}^{\dagger}\hat{\mathbf{p}} \quad (8)$$

2.3 Predicted Algorithm Performance

The two algorithms explored in my research have countering areas of strength and weakness. Delay and Sum has a very low computational complexity and is guaranteed to preserve the integrity of the signal at the bright point. This consistency and speed comes at the price of a broader, less well-directed beam. Pressure Matching, on the other hand, suffers from significantly higher computational complexity and requires intense regularization to preserve the integrity of the original signal. However, despite having these weaknesses, Pressure Matching is able to form an extremely tight and well-directed beam to the bright point, with little peripheral aliasing. Both algorithms should be effective at solving the personalized sound problem and both algorithms should preserve phase integrity at the bright point.

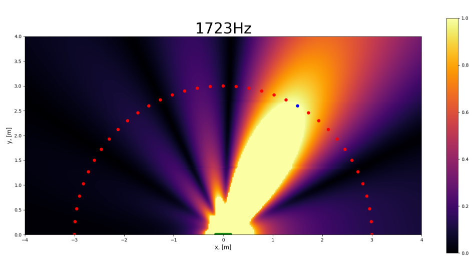

Figure 2: *Pressure Matching*

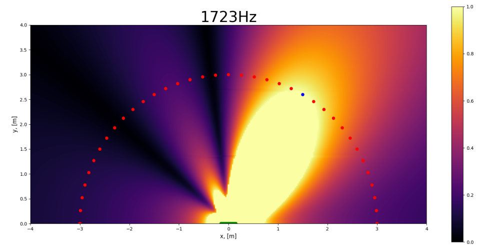

Figure 3: *Delay and Sum*

3 System

To test the performance of the algorithms presented in Section 2, I designed a beamforming system and necessary infrastructure. The system is composed of hardware and software subsystems.

3.1 Hardware

The processing core of the hardware subsystem was a laptop, where all signal processing was performed. The hardware subsystem was able to accept data through a **.wav** file. Then, after processing, the laptop used two USB-hubs and eight USB-audio cards to playback through a 16×1 loudspeaker array.

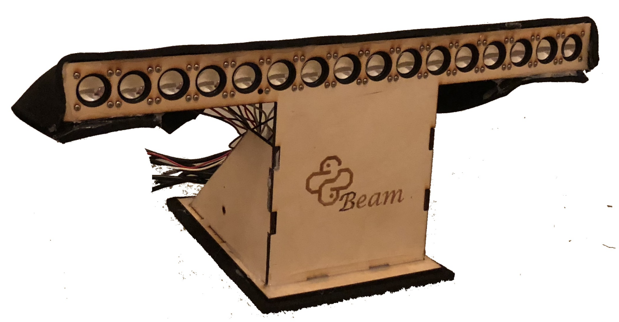

Figure 4: *The* 16×1 *loudspeaker beamforming array used*

3.2 Software

All of the software was written in the Python programming language and made extensive use of the SciPy and NumPy libraries for their highly optimized signal processing and linear algebra routines. The PyAudio library was used to read data from a microphone and to play-back through 16 loudspeakers. Finally, the custom PyBeam library was developed, which contained methods encapsulating the process of generating and applying beamforming filters to an incoming audio signal. The PyBeam library is not only able to process signals and drive a 16×1 array like the one used in testing, but is also able to scale upwards and downwards to any possible array configuration (Ter Martirosyan, 2018).

4 Testing

To determine whether or not the hardware and software developed can beamform successfully, data was collected and checked against the expectations presented in Section 2.3.

To test algorithm performance, synthetic tests were run, with the plots displayed in Figures 2 and 3. These synthetic tests provided valuable insight into the *qualitative* performance of both algorithms. To obtain *quantitative* results, a microphone measured sound pressure and phase at every thirty degrees about the loudspeaker array, starting immediately three meters to the left of the array and rotating all the way to three meters to the right of the array. Sound pressure and phase was measured at each point for both algorithms being aimed straight ahead and thirty degrees right of the vertical as well as a control with no algorithm applied.

5 Results

Most of the testing parameters set forward in Section 2.3 were easily verified, as both algorithms replicated the original signal faithfully at the bright point. However, Pressure Matching took longer to compute than Delay and Sum did and required intense regularization, as expected. The final qualification, however, was more difficult to verify—further testing was necessary to identify if the beamforming system was able to perform "amplitude" beamforming and "phase" beamforming.

5.1 Amplitude Beamforming

Amplitude beamforming refers to the amplitude of the signal at the bright point being significantly larger than the amplitude of the signal at any dark point. Amplitude beamforming is the most well-known form of beamforming, and the only form of acoustic beamforming humans can detect unaided by any equipment. Accomplishing amplitude beamforming is the primary objective of this research project, as that is the only component of beamforming relevant to the problem of personalized sound.

As is evident from the plots in Figure 5, Amplitude Beamforming was successfully achieved. Both algorithms were effective at beamforming sound to the bright point when directed at both $90°$ and $60°$. As predicted, Pressure Matching outperformed Delay and Sum in terms of acoustic contrast between bright and dark points.

5.2 Phase Beamforming

Phase beamforming is another quantifiable metric of beamforming performance. While not particularly prevalent in acoustic beamforming, phase beamforming is vitally important to radio-frequency beamforming. Modern modulation techniques for telecommunications, like Quadrature Amplitude Modulation and Amplitude Phase Shift Keying, communicate data using both amplitude *and* phase. Losing phase integrity essentially renders the data being transmitted nonsense. Since the secondary goal of this research project was to gain intuition on radio-frequency beamforming, exploring phase beamforming makes sense. With both algorithms, the signal at the bright point is expected to be in-phase with the input signal.

The plots in Figure 6 indicate the expected performance and measured performance like the amplitude beamforming plots in Figure 5 along with a 5 centimeter confidence interval. The interval takes into account error in microphone placement ± 5 centimeters from the target radius of 3 meters. From these plots, it is clear that the data mostly falls in the 5 centimeter confidence interval, with the data being in-phase at the bright point, where phase beamforming dictates it should fall.

5.3 Next Steps

That is not to say that the beamformer is perfect. There is significant room for improvement, as while signal directivity was accomplished, the dark points were far from being silent. Possible improvements for my beamformer include:

- Better noise cancellation behind the array.
- Better emitter isolation, so that vibrations in one speaker do not vibrate other speakers or the array.
- Better noise elimination, bringing up acoustic contrast between bright and dark points significantly.

In terms of my testing methodology, it was far from perfect, as all testing was performed in the Thomas Jefferson HS Microelectronics Lab. Preferably, an anechoic chamber or a large empty area would be used, as both spaces have the ability to remove excess acoustic energy. Testing in the microelectronics lab was fraught with sound waves ricocheting off of walls and objects, causing unwanted interference.

Finally, expanding my beamformer to operate in the three-dimensional free-field would enable an even more in-depth analysis of beamforming algorithms and techniques. This extension would mirror the massive-MIMO technology outlined as part of the 5G communication scheme. It is worth noting that this extension involves no new software, only a hardware change from the one-dimensional beamforming array in 7 to a two-dimensional array.

5.4 Conclusions

With the success of amplitude beamforming and phase beamforming verified, that means the beamforming hardware and software system is successful at beamforming an acoustic signal and has promise for solving the personalized sound problem.

Bibliography

Druyvesteyn, WF and John Garas (1997). "Personal sound". In: *Journal of the Audio Engineering Society* 45.9, pp. 685–701.

Elliott, Stephen J et al. (2012). "Robustness and regularization of personal audio systems". In: *IEEE Transactions on Audio, Speech, and Language Processing* 20.7, pp. 2123–2133.

Fink, Mathias and Claire Prada (2001). "Acoustic time-reversal mirrors". In: *Inverse problems* 17.1, R1.

Kirkeby, Ole and Philip A Nelson (1993). "Reproduction of plane wave sound fields". In: *The Journal of the Acoustical Society of America* 94.5, pp. 2992–3000.

Olivieri, Ferdinando et al. (2016). "Theoretical and experimental comparative analysis of beamforming methods for loudspeaker arrays under given performance constraints". In: *Journal of Sound and Vibration* 373.Supplement C, pp. 302 –324. ISSN: 0022-460X. DOI: `https://doi.org/10.1016/j.jsv.2016.03.005`. URL: `http://www.`

sciencedirect.com/science/article/pii/S0022460X16002340.

Ter Martirosyan, Sarkis (2018). *PyBeam*. https://github.com/smtm1209/PyBeam.

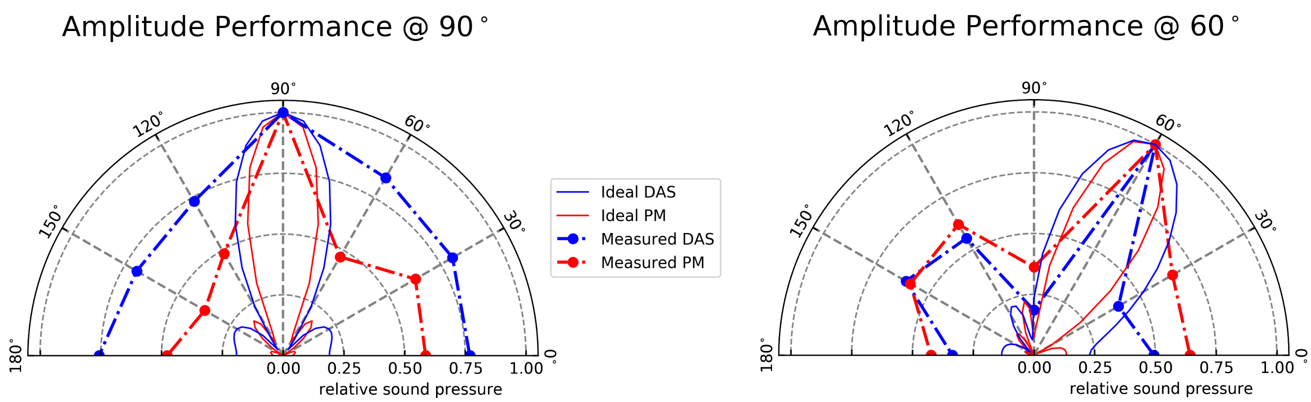

Figure 5: *Amplitudes at* 90° *and* 60°

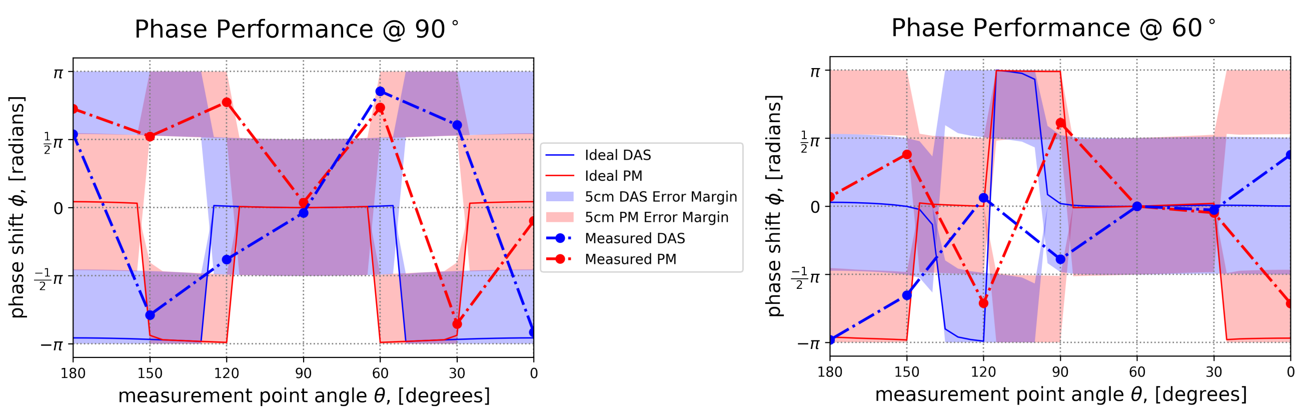

Figure 6: *Phase Shifts at* 90° *and* 60°

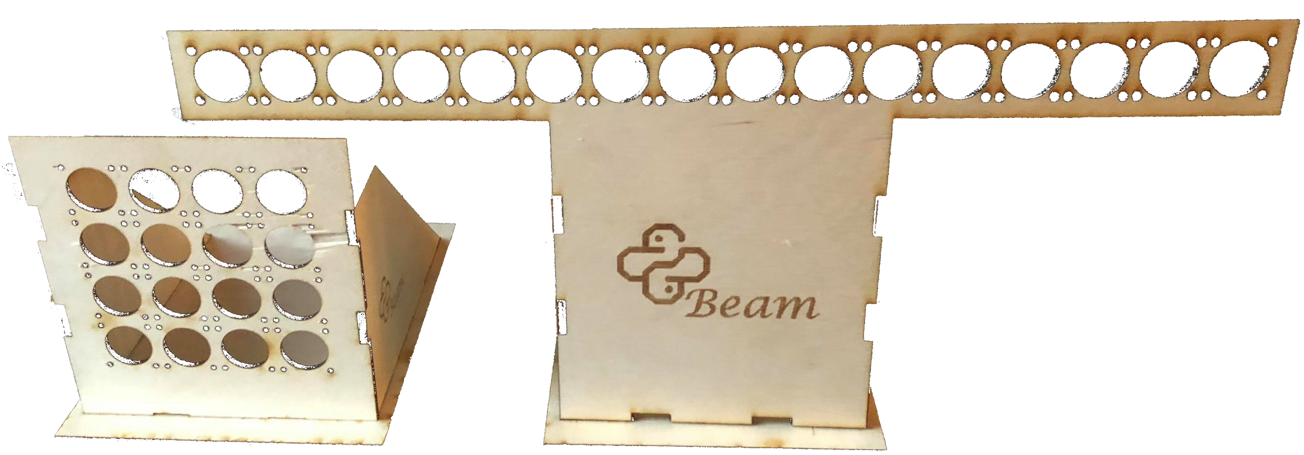

Figure 7: *Example 2D and 1D loudspeaker arrays*