Earbeamer: A Parallel Beamforming Hearing Aid System

Niket Gupta, CSE, Aaron Lucia, CSE, Nathan Dunn, CSE, and Matteo Puzella, EE

Abstract—Earbeamer is stationary, wall-mounted hearing aid system targeted at the senior citizen population that allows users precise control over the volume of particular individuals within the room. By applying beamforming in parallel over a microphone array, the audio of each identified individual is isolated, and may be attenuated or amplified. Through an Xbox Kinect, the movements of each individual are tracked, ensuring that a conversation is unimpeded regardless of movement within the room.

Index Terms—Beamforming, Microphone Arrays, Acoustics, Hearing Loss, Signal Processing

1 Introduction

Hearing loss is a common problem among the senior citizen population. As we get older, parts of the inner ear that are sensitive to sound begin to atrophy – a process that is often exacerbated by many factors that are commonly found among the elderly, such as diabetes, high blood pressure, and even some chemotherapy drugs. Presbycusis - age related hearing loss - affects about 1 in 3 Americans over the age of 65 [1]. By age 75, this number increases to about 1 in 2.

Its prevalence is concerning, as adequate hearing is a vital requirement for communication. The typical onset of presbycusis coincides with many major social changes in the life of an individual. An individual may be facing retirement, or losing mobility due to age-related ailments.

The loss of these social interactions can compound with hearing loss and have profound effects on cognition. In a study of 2,304 adults with individuals with hearing impairments, those without hearing assistance were 50% more likely to suffer from depression [2]. A separate study found that dementia progressed more quickly among the hearing impaired population than a healthy population, with cognitive performance declining 30 to 40% faster over an equal period of time [3].

1.1 Existing Solutions

The current hearing aids in todays market fall under two categories: analog and digital.

Analog hearing aids pick up sound, amplify the sound, and feed it into the users ear. Analog hearing aids can have certain settings for certain environments if requested to the audiologist [4]. This means that the aid can be adjusted to a specific volume depending on the environment the user is in, whether it be on the highway stuck in traffic or in the house watching television. However, analog hearing aids cannot distinguish between the sounds the user wants to hear and the sounds the user does not want to hear [4].

Analog hearing aids have begun to become obsolete in favor of digital hearing aids. Digital hearing aids contain a microchip that acts as a computer database in order to help the users hearing loss [5] . The digital hearing aid picks up the sound, and converts the analog signal into a digital signal. The ability to convert the signal to digital allows the hearing aid to filter out background noise frequencies and amplify frequencies that are desired, like human speech [5]. The audiologist has more control in adjusting the hearing aid for the user because of the digital conversion. A common complaint with the digital hearing aids is the price tag. The average price of a digital hearing aid ranges from \$1500 to \$3500[6] . Also, the digital hearing aid gathers all sound coming from every direction of the user before any signals are filtered. Therefore, even if the hearing aid is customized to filter out background noise and only amplify human speech, the user does not have control over what conversations he or she will hear.

The current hearing aid market also provides hearing aids that use beamforming. The beamforming hearing aid consists of multiple omnidirectional microphones that form a beam signal [7] . It helps attenuate background noise while focusing toward the target sound. A common polar amplification pattern for a simple beamforming hearing aid is a cardioid. By delaying microphone outputs from the sides and rear of the hearing aid, sounds arriving from those directions can be attenuated, while sounds arriving from directly in front of the user are amplified.

The issue with the beamforming hearing aids is that the sensitivity of the hearing aid drops off at low frequencies, like 1000 Hz [8]. In order for the beamforming hearing aid to accommodate this situation, additional gain must be implemented in the hearing aid to hear low frequencies. However, additional gain comes at the cost of internal noise in the hearing aid, which is unwanted [8].

As mentioned above, all current solutions have a flaw that hampers the users use of their respective hearing aid. Kochkin found that a quarter of individuals who own hearing aids but do not use them cite poor performance amid background noise as the primary reason [9]. Further, as is shown by the polar of the directional hearing in Figure 1, many current directional solutions are dependent upon the listener physically looking at a target to obtain maximum amplification. However, this is often not the case in an actual conversation. There are many instances where a participant in a conversation may not be actively looking at other participants, such as when they are looking at a television,

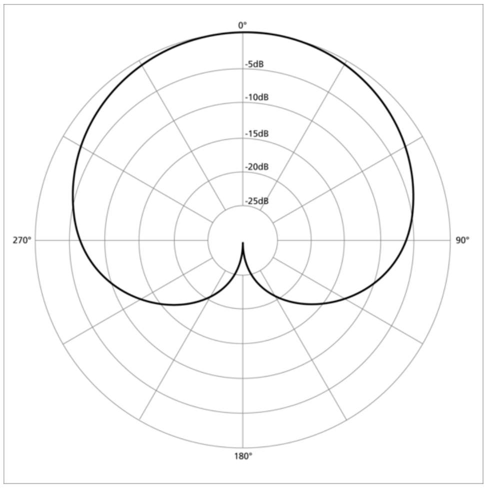


Fig. 1. Cardioid Pattern response for directional hearing aids. 90^o represents the area directly in front of the user

or moving through the room. In these scenarios, a listeners ability to perceive the conversation should not be hindered by head position.

1.2 Requirements

Noting the two above scenarios, we have proposed a hearing aid system that allows an elderly user:

1) To selectively attenuate or amplify nearby human targets within an indoor space, -essentially giving the user the ability to mute or turn the volume up on individuals within the room.
2) To move about the room without negatively affecting the amplification of their conversation

We achieve both of these goals using beamforming through a stationary array of microphones within the room, a process that we will describe in detail in Section II.

1.3 Specifications

The specifications for the hearing aid system are summarized in Table 1:

Specification	Value
Array Width	< 2m
Target Identification Range	> 20ft
Angle of Operation	30^o to 150^o
Maximum Number of Targets	> 4
Beamwidth	$< 15^o$
Bandwidth	1000 4000 Hz
Delay	< 300ms

TABLE 1
Specifications for Device

Size and Range of Device: To formulate our specifications, we made the assumption that the system will be operating within a users home in a space reserved for entertaining guests, such as a family or living room. Considering a 20 x 20 living room, this assumption provides the maximum distance that the system must identify and

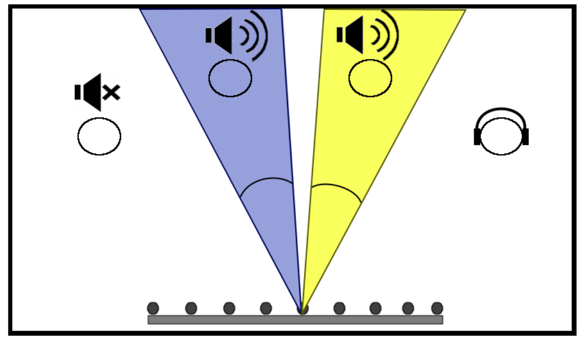


Fig. 2. Desired behavior for our system. A wall mounted device is able to selectively listen to multiple targets within the room for a listener, here wearing headphones

listen to targets, as well as the maximum allowed size of the system.

Within the relatively small space of a living room, it is important to ensure that any potential device does not disrupt the normal daily life of any occupants. We desire a wall-mounted system to leave as much floor space for the occupant as possible. The system should also not overtly draw attention to itself and dominate the room. To accomplish this, we set a device size limit of 2 meters in length, about the size of a large hanging piece of artwork.

Beamwidth: To effectively isolate the output of sound sources, we must be able to encapsulate each target within distinct, non-overlapping beams, as seen in Figure 2 If each target is centered within a beam, then the 3dB beamwidth of that beam must not intrude into the 3dB beamwidth of another beam. We considered a typical living room couch seat as the minimally spaced placement that any two people will be arranged within the room, as seen in Figure 2. If a typical couch cushion is 25 in width, and the couch is 8 from the wall, then the minimum beamwidth is approximately 15^o

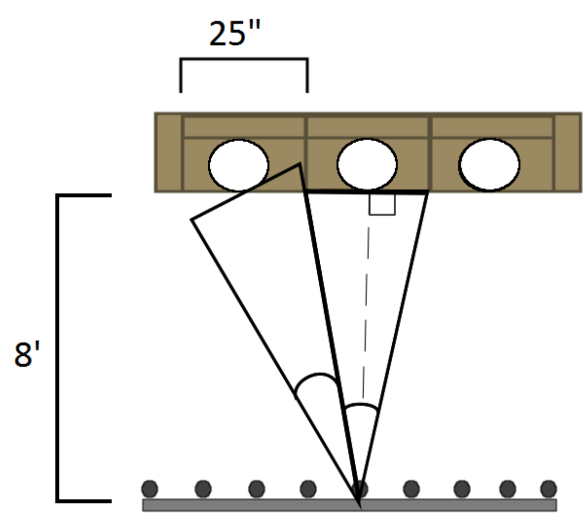


Fig. 3. Finding the minimum beamwidth needed to encapsulate targets within individual beams

Bandwidth: In typical telephony applications, the transmitted human speech spectrum range is about 300 to 3300

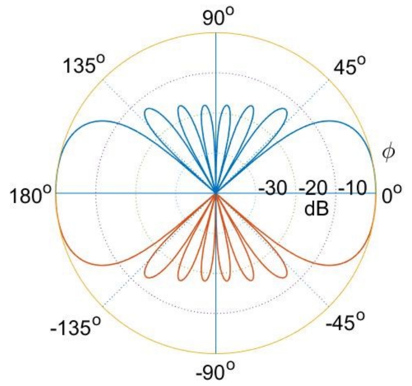


Fig. 4. This is the array response when the elements are steered toward 180^o or 0^o. You cannot amplify one direction without also amplifying the other direction

Hz, in order to maintain intelligibility [10]. However, Section II will show that beamwidth may be gained by sacrificing bandwidth, so we limit ourselves to 600 2400 Hz. This is similar to the 300 2700 Hz bandwidth used for long-distance telephone connections in the 1980s [10].

Delay: To provide seamless conversations, the delay between reception of sound and playback to the user must be minimal. The ITU G-177 specification for VoIP mandates a two-way delay of less than 300 ms [11], so we used this number to provide an upper bound on delay.

Angle of Operation: For a wall mounted system, microphones must be able to target and listen over a wide range of angles to provide adequate coverage for the entire room. For a microphone array parallel to a wall, we would ideally want to isolate and amplify sound over a range of 180^o. However, the nature of our signal processing method, beam-forming, ensures that isolating sound from sources close to the extreme ends of this range is difficult (at $0^o/180^o$, ie. when a source is close to the same wall where the system is mounted).

For reasons that will be expanded upon in Section 2.2 we cannot amplify targets at 180^o without also amplifying sound at 0^o. This is called an "endfire" array response , and may be seen in Figure 4.

We desire for the user to have individual control over each possible source of sound, so we have chosen to limit the angle of operation to 30^o to 150^o to avoid the endfire configuration. For our problem, this is a reasonable limitation, as most targets will be present at some point within the room, not hugging the wall of the device.

2 DESIGN

2.1 Overview

Our approach to addressing this problem harnesses the power of audio beam-forming and mates it with an intuitive user interface. This is enabled by a visual tracking system alongside a mobile app that allows intuitive user interaction for system control. The beam-forming approach has been used several times before and is well documented. Furthermore, our processing algorithm gives us a very low latency so that this system viable for real-time communication.

There is a long history of acoustic beamforming projects in the UMass ECE department [12][13][14][15]. Project Sauron from 2016 left behind a significant amount of valuable hardware, from microphones to a 16 channel ADC. To save on costs, we have aimed to utilize these features within our own project.

However, while using past hardware, we seek to improve upon the performance and usability of previous design projects. One of our aims is to automate the system as much as possible while allowing the user to control only what is relevant, specifically which individuals they would like to isolate and hear. The Microsoft Kinect plays a key role in allowing this automation with its ability to identify and map unique individuals. The mobile app displays this data and allows for user selection. The software program performing the beam-forming processes the audio streams from the beam-forming array and outputs them to the users headset. An overview of this system is shown in Figure 3.

2.2 Microphone Array

The microphone array is the method through which sound is sampled spatially from the environment. Through classes such as ECE 313 and ECE 563, we are familiar with how an analog signal may be sampled in time by a set sampling period. An array of elements displaced by a set distance d samples the same signal with a relative phase shift between elements. By adding the output of each element together, signals from certain directions are added constructively, while signals from other directions are added destructively.

An array of elements with identical radiation patterns can be described by a term called the array factor, which for a one-dimensional linear array of n elements can be written as:

$$A(\phi) = a_0 + a_1 e^{jkd\cos(\phi)} + ... + a_n e^{jnkd\cos(\phi)}$$
$$A(\phi) = \Sigma_n a_n e^{jnkd\cos(\phi)} \quad (1)$$

Where d is the distance between microphones, k is the wavenumber of an incoming wave, phi is the direction of propagation for the sound wave, and a_o a_n are complex coefficients [16]. This sum of complex exponentials completely describes the geometry of the array, with each term representing the relative phase shift resulting from the time that it takes for a wave to propagate from element to element.

The array factor has a dramatic effect on the directivity of the array. For a wave incoming at a direction of ϕ, if each element has an identical power gain of $G(\phi)$, then the gain of the entire array system $G_{tot}(\phi)$ is [16]:

$$G_{tot}(\phi) = |A(\phi)|^2 G(\phi) \quad (2)$$

For example, Figure 6 contains a polar plot of the term $|A(\phi)|^2$ for a linear array of 8 elements, with coefficients $a_0 = ... = a8 = 1/8$, meaning that each element is equally weighted. In this scenario, the maximum power gain occurs when a wave is arriving perpendicular to the linear array, at 90^o, also known as broadside. Intuitively, this is the direction where all microphone inputs add together in-phase.

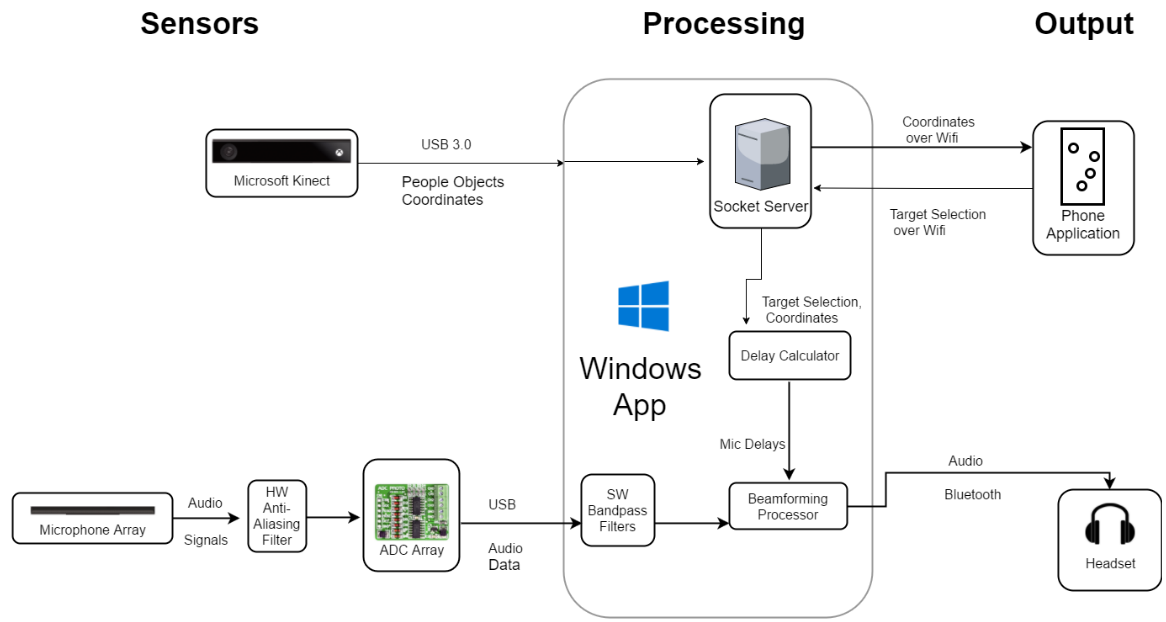


Fig. 5. Caption

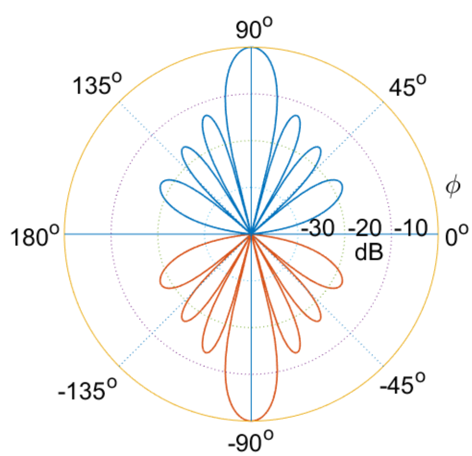


Fig. 6. Array Power gain $|A(\phi)|^2$ for an array of 8 elements, spaced one-half a wavelength apart, with each microphone equally weighted

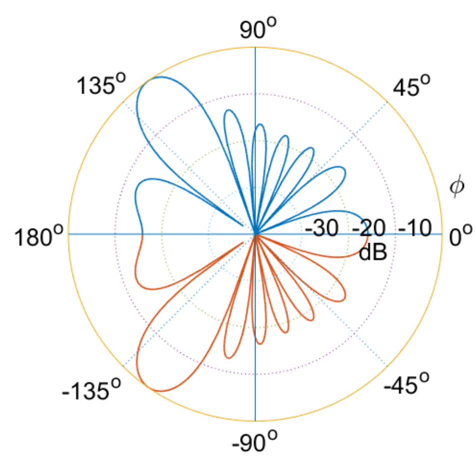


Fig. 7. Array Power gain $|A(\phi)|^2$ for an array of 8 elements, steered towards 125^o

By changing the coefficients $a_0 \ldots a_n$ to a set of complex exponentials, each sampling element provides a phase shift (i.e. a time delay) to the signal that it is sampling. The direction of the main beam in the polar plot of $|A(\phi)|^2$ can be translated to another direction, called the steering angle. Figure 5 displays the array pattern for a beam aimed 35^o from broadside.

From (2), we can see that there are two terms affecting the power gain polar pattern of our linear array:

- $G(\phi)$ determined by the microphone selection
- $|A(\phi)|^2$ determined by the geometry of the microphone elements

By optimizing both of these terms, we can minimize the beamwidth of the array, and meet our specifications

2.2.1 *Microphones*

For microphone selection, we had access to the 16 omnidirectional ADMP510 MEMS microphones used in the SDP 16 beamforming project. As the SDP 16 team noted, this particular model of microphone has a relatively linear frequency response within the frequency band targeted by our system [15].

Given this property of the microphones, we decided to use SDP16s microphones within our own design, to ensure that all frequencies within the targeted band are amplified equally. However, as we are receiving these microphones used, we will need to complete a verification procedure on each microphone, to ensure that it is still functioning after storage. To accomplish this calibration procedure, we will record low, medium and high frequency tones on each microphone, and then play each tone back. If the playback

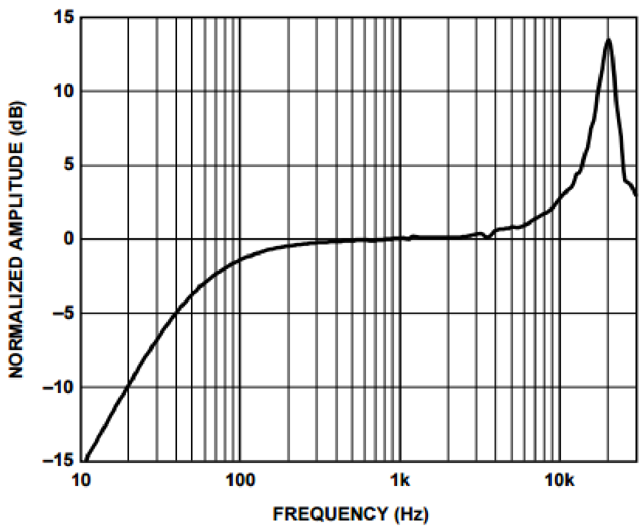


Fig. 8. Frequency response for the ADMP510 microphones. Note the flat response over the targeted band 600 2400 Hz

tone matches the original tone, then we can verify the microphone as functional.

Note that these microphones are omnidirectional, so tones are uniformly amplified in terms of direction. Thus, (2) simplifies to:

$$G_{tot}(\phi) = |A(\phi)|^2 \tag{3}$$

2.2.2 Array Geometry

As (3) shows, the array factor is the only term that determines the directivity of the array. Therefore, in order to optimize the beamwidth, we need to select an optimal microphone geometry.

Orfanidis shows that for a uniform linear array, the 3dB beamwidth may be approximated as [16]:

$$\Delta 3dB = \frac{0.886\lambda}{\sin(\phi_0)Nd}b \tag{4}$$

Where ϕ_0 is the steering angle, λ is the wavelength of tone, N is the number of microphones, d is the microphone distance, and b is a factor dependent on the weighting applied to each microphone.

(4) shows that beamwidth will increase as the beam is steered towards 0 or 180^o, and as frequency decreases. This creates an issue for our beamformer, as it means that different frequencies within the human speech spectrum will produce different beamwidths.

Increasing d or N will decrease beamwidth, but the distance between microphones cannot be increased beyond λL/2, where λL is the largest wavelength within the targeted frequency band. This is the Nyquist criteria for spatial sampling through arrays, analogous to the Nyquist frequency for sampling in time [16]. If the Nyquist criteria is exceeded, then additional beams will appear that are equal in magnitude to the main beam. This is known as spatial aliasing, and an example may be seen in Figure 9a.

With this knowledge in mind, we analyzed SDP16s array. The SDP16 team used a nested array as pioneered by Smith[17], where the targeted frequency band was split into smaller bands, and then subarrays were constructed out of the 16 available microphones. By sharing microphones between subarrays, each subarray could be allocated 8 microphones.

Band	Highest Wavelength to Mic Distance Ratio
600 - 1000	0.617
1000 - 1700	0.69
1700 - 3500	0.72

TABLE 2
Project Sauron Frequency Bands

However, for the SDP16 array, within each band, approximately half of the frequencies would exceed the Nyquist Criteria, as shown in Table 2:

Figure 9a and 9b demonstrate the negative effects of exceeding the Nyquist criteria. As our system implements beamforming in parallel, the spatial aliasing would add even more noise, as the additional beams will each produce an aliased beam.

To correct this issue, we divided the targeted frequency band into octaves, as Smith originally did. For a frequency band [f_L, f_H]:

- For each octave [f_{iL}, f_{iH}], a subarray was created with microphone distance di = $1/(2f_{iH})$, to avoid aliasing
- Each successive subarray had a microphone distance d_i = 2d_i-1, to share as many microphones as possible
- All subarrays were allocated the same number of microphones, to ensure that the array response to each band was identical

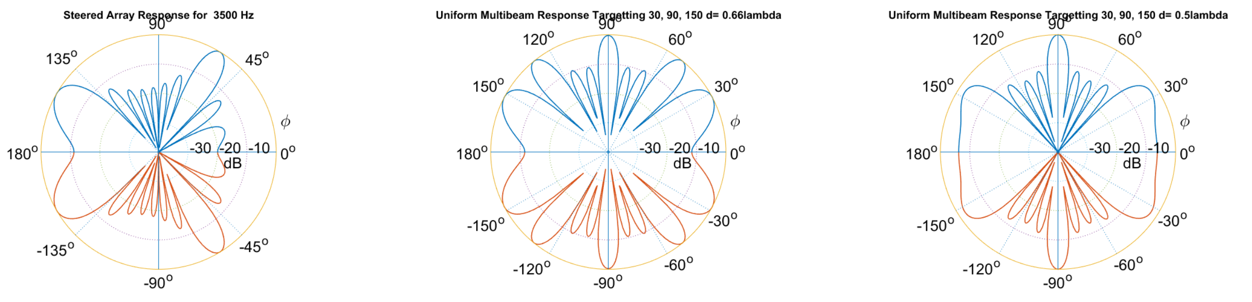
With these requirements, we could create three arrays targeting [600, 1200],[1200, 2400], and [2400, 4800] Hz, and cover almost all of the human speech frequency spectrum. However, by eliminating a sub-band, we could increase the number of microphones allocated to the other bands from 8 to 11 microphones. By performing tests on each team members voice, we found that targeting the octaves [1000 ,2000] and [2000, 4000] produced the most intelligible speech. However, intelligibility is a somewhat subjective quality, and it is worth noting that the majority of the power in human speech is found at frequencies under 1000 Hz.

The array performance is summarized in the table below, for the best and worst cases frequencies:

d/λ	Steering Angle	Beamwidth
$\lambda/4$	90	18.5
	150	36.9
$\lambda/2$	90	9.3
	150	18.5

TABLE 3
Earbeamer Array Performance, with Uniform Weighting

As shown in Table 3, beamwidth suffers when the steering angle is directed towards its maximum angle of 150^o. However, beamwidth in the regions directly in front of the array, from 60 to 120 degrees remains relatively close to the specification. This is likely the best performance we can achieve with our current 16 channel Analog to Digital converter. From Equation 4, the only way to narrow beamwidth



(a) a (b) b (c) c

Fig. 9. The left plot shows the SDP16 arrays response to a 3500Hz signal when the array is steered to 150^o, creating an aliased lobe at approximately 60^o. The center plot shows SDP16s array response when beamforming is performed in parallel, and aimed at 30, 90, and 150 degrees. The spatial aliasing at 135 and 55 degrees adds a significant amount of unwanted amplification to the field. The far right plot shows the highest frequency for the Earbeamer array, pointed at the same locations, with no spatial aliasing

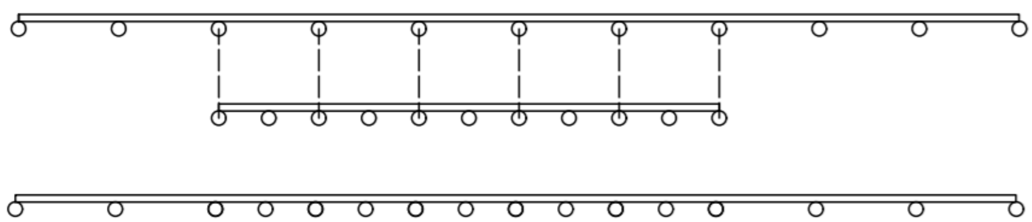


Fig. 10. The Array Geometry for the new Earbeamer array. 16 microphones are shared between a [600,1200] band with d= 7cm, and a [1200,2400] band with d = 14cm

further is to add more microphones, but that would require purchasing an ADC that is outside the range of our budget.

2.2.3 Microphone Weighting

In our particular application of beamforming, it is desirable to have an array response that features a main lobe that is as narrow as possible, and sidelobes that are as small as possible. Beamforming is like an average – we are adding together the outputs of multiple microphones to get a single output. If we apply a higher weight to certain microphones, we can control the overall sidelobe level. Different weighting schemes can be used to accomplish different goals in the array response, but Orfanidis shows that for the narrow beamwidth, low sidelobe problem, the Dolph Chebyshev weighting scheme is the optimal choice. [18]

The Dolph-Chebyshev scheme makes use of Chebyshev polynomials – a sequence of polynomials where the mth polynomal can be defined as:

$$T_m(x) = \begin{cases} \cos(m \arccos(x)) & \text{if } 0 < m \leq 1 \\ \cosh(m \,\mathrm{arccosh}(x)) & \text{if } m > 1 \end{cases} \quad (5)$$

As can be seen in Figure 11, when x is small, $T_m(x)$ is bound between 1 and -1, but when $x > 1$, $T_m(x)$ grows exponentially. Thus, the general idea behind Chebyshev weighting is to define a window such that the sidelobes

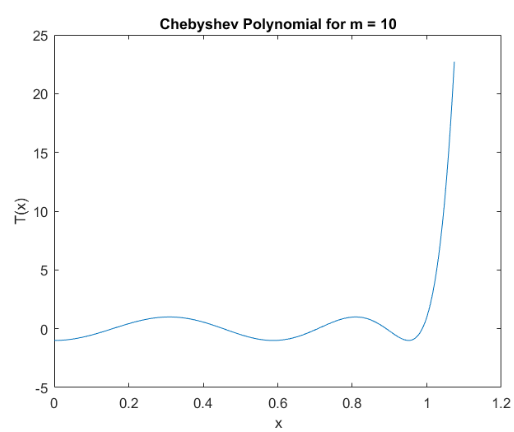


Fig. 11. Plot of the 10th Chebyshev Polynomial

correspond to the $x \leq 1$ section of the polynomial, and the main beam corresponds to the $x > 1$ section.

Using Orfanidis' MATLAB functions, appropriate weights were calculated for our microphone array, with a targeted sidelobe attenuation of -20dB. This produces an array response like the one seen in Figure 12.

The price to pay for the reduction in side lobes is a slight increase in main lobe beamwidth. However, Table 4 shows that the increase was not significant.

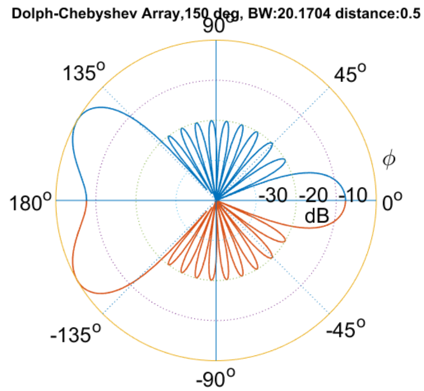


Fig. 12. Earbeamer Array with Dolph Chebyshev weighting, steered towards 150^o, with microphone distance $\lambda/2$

TABLE 4
Comparison of Beamwidth, Uniform vs Dolph Chebyshev

d/λ	Steering Angle	Uniform	Dolph-Chebyshev
$\lambda/4$	90	18.5	20.2
	150	36.9	40.3
$\lambda/2$	90	9.3	10.0
	150	18.5	20.2

2.3 Beamforming Algorithm

Our beamforming algorithm uses a delay-sum technique, where a different time delay is applied to each microphone in the array before combining the signals together. By calculating the delays using the position we wish to target, we can align the phase of the signals sourced from that location causing constructive interference, thus theoretically amplifying the audio only from that location.

$$y[n] = \Sigma_{m=0}^{M-1} x[n - m\tau] \tag{6}$$

The beamforming algorithm was implemented using a pipelined approach in C++ to give us the sample length to perform all our calculations. Using a sample rate of ˜16kHz and a sample size of 1024 samples, each buffer would hold 64ms of audio data, giving us 64ms to filter and perform beamforming over that data. Audio is received on an interrupt from the ADC and waits in a receiving buffer for the beamforming algorithm to use. The algorithm then takes the audio from the 16 microphone channels and splits it into 22 separate audio streams to be filtered, to form the two 11 microphone subarrays. Each of the 22 separate audio streams use 2 rotating arrays of data, to allow a simple indexing method to apply a time shift on the array.

Since one of our goals was to be able to calculate more than one beam of amplification, the beamforming algorithm calculates the delay sum algorithm for each beam selected. Audio level normalization is then applied to the result of each beam so each speaker is amplified evenly, and then the beams are combined into one signal, which can be sent to the pc's audio device.

2.4 Anti-Aliasing Filter and ADC

The purpose of the anti-aliasing filter block is to cutoff sounds after a certain frequency. This filter comes before the Analog to Digital Converter, a device that converts the signal from analog to digital before being sent to the computer.

We have two main requirements for the filter: we desire a sharp attenuation drop at the cutoff frequency, and a delay response that does not interfere with the beamforming algorithm. The latter requirement is because the filter does not affect all frequency components equally; some frequency components may be delayed more than others. Since our beamforming algorithm relies on delaying microphone inputs, unintended delays can damage our ability to recover a desired signal when microphone outputs are added together.

We can evaluate the effect of the filter on the delay of various frequencies by measuring the group delay. Group delay is the rate of change of transmission phase angle with respect to frequency [19]. The goal is to have no change in group delay in the pass band (1-4 kHz).

The highest possible sampling rate we can achieve with our ADC is 16 kHz, or about 1 sample every 62μs. We cannot allow the group delay to exceed this value, or else the audio may become distorted after beamforming.

Through SPICE, we explored and simulated multiple filters for the microphones that applied to our requirements. Rather than building an RC series filter which gives a 3dB drop off at the desired cutoff frequency, it is preferred to have a sharper drop off at the cutoff frequency. There are well documented filter designs that can achieve sharper cutoffs using capacitors and inductors [20]. In order to use these types of filter, we use a method called the Insertion Loss Method. This method is designed to combine capacitors and inductors on a circuit board in order to give the designer more control of filters attributes, like attenuation drop off and group delay.

The two filters that use the Insertion Loss method for sharp attenuation drops are the Butterworth filter and the Chebyshev filter. The Butterworth filter, as shown below in the figure, does not drop off in attenuation until we reach the cutoff frequency. The drop off in attenuation is linear after the cutoff frequency. The Chebyshev filter, a filter built in the same fashion as the Butterworth filter, offers a sharper drop off in attenuation at the cutoff frequency [20]. However, as mentioned before, the greater the change in group delay in our desired passband, the bigger the interference with the beamforming algorithm. We therefore took a look into the linear phase filter. The linear phase filter, another circuit using the Insertion Loss method, can achieve a very good delay response at the expense of a very slow attenuation drop off at cut off frequency.

Before MDR, the initial approach to the hardware filter was to have a cut off frequency at 2.4 kHz, because that is the frequency band we need for the project. A suggestion made by Professor Kelly was to design the hardware filter that achieves a drop off in attenuation around 8kHz , because the target frequency band we want to use is from 0 to 2.4kHz. The group delay does not affect frequencies far away from the cut off frequency. Once the filtered signal is sent to the analog to digital converter, the signal would be sampled at 16 kHz. From there, we can use a software filter to extract only the human speech frequency spectrum.

Since MDR, we have decided to use the simple RC filter to eliminate aliasing, due to its favorable group delay performance. Our final specifications on the printed circuit board are: R = 23.2 ohms and C= 1uF.

The use of the RC filter with the specifications listed above gave us a frequency cutoff of 6.860 kHz. Also, the phase response of the filter will not affect the beamforming algorithm. The recorded biggest change in group delay from 1 to 4 kHz was 8.16us, which showed to have no effect on the beamforming algorithm.

2.5 Xbox Kinect

A key component of our hearing aid system is the ability to identify targets within the room, and determine the location of each target relative to an array of microphones. These coordinates are used to dynamically aim our beamforming algorithm as targets move about the room. To accomplish this, we needed a robust computer vision system with a depth sensor that had adequate range to cover a typical living room. The Microsoft Kinect for the Xbox One fulfilled these requirements.

The Kinect uses a Time-Of-Flight system to gauge depth. An infrared emitter emits pulses of infrared light, and an infrared sensor records when the pulse is reflected back. By recording the time required for the reflection to arrive, the relative distance of a point in space may be calculated. Using this system, the Kinect is able to maintain an effective range of 0.5 to 4.5 meters [21], which more than meets our minimum range specification of 20 feet. Further, the TOF system used in the Kinect for Xbox One has been proven accurate to a granularity of 1mm [22], which is more than adequate resolution for aiming the beamformer.

Fig. 13. Plot of depth data returned from Kinect, showing the output of the MDR demo. For each tracked individual, the user ID and angular displacement is printed

To test and demonstrate the skeleton tracking for our MDR, a program was written to extract coordinates from any target that entered the field of view, calculate the relative angle of that target to the Kinect, and display the angle on a screen. A screenshot of this application is shown in Figure 13

Through the course of this testing, it was found that the angle of view of the Kinect was only about 60^o, as shown in Figure 22 of the Appendix. As a result, placing the Kinect directly behind the microphone array does not allow the target selection for the specified range of 30 to 150 degrees.

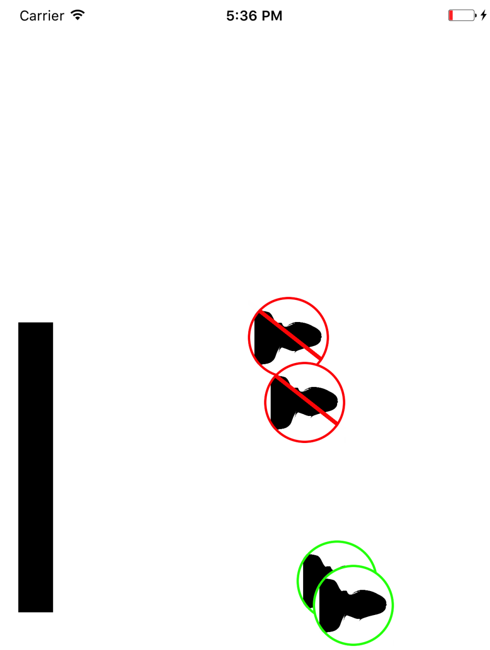


Fig. 14. Graphical interface for the application. Coordinates of identified targets are used to render representations to the screen

To remedy this, we placed the Kinect at an offset from the array, in order to give it a better view of the room. We could then translate the coordinates of the Kinect to a coordinate system based around our array. For a coordinate system that has been shifted horizontally and vertically by (h, k), and rotated clockwise by ϕ, the translated coordinates (x', y') may be found as:

$$x' = (x - h)cos(\phi) + (y - k)sin(\phi) \tag{7}$$

$$y' = -(x - h)sin(\phi) + (y - k)cos(\phi) \tag{8}$$

2.6 iPhone Application

The user interface consists of an iPhone application. The application interface gives the user a visual representation of the targets relative to the Earbeamer system. Each target, represented by an icon as seen in Figure 14, can be tapped by the user, which sends an update to the server to toggle that target's audio.

Communication between the application and computer processing the data is done over a wireless connection using WebSockets. This is a versatile, platform-agnostic protocol that is supported on many platforms. More importantly, it allows two-way communication in an arbitrary manner - either party can send a message to the other at any time.

iPhone application development was done with the Swift programming language, using the Apple Xcode IDE.

Fig. 15. Experimental Set-up for Single Tone Performance

This modern and well-supported environment is designed for interactive applications. Software development techniques learned in the past have been useful in learning this new platform and problem solving during the development process.

The processing computer runs a WebSocket server and listens for connections. When started, the iPhone application attempts to connect to this server and the connection is established for the duration of operation.

The computer sends target data, including a unique ID, and x- and y-coordinates, received from the Kinect at a fixed interval – about once per second – to the iPhone application. The iPhone application sends asynchronous user input updates to the computer when the user taps on a target icon.

The result is an integrated system with a constantly updating display of moving targets. User input is immediately sent to the server and the processing computer's operation immediately accommodates this new configuration, which is heard through the user's headphones. The latency in switching targets is negligible. The iPhone application is also updated to reflect the new status change.

3 Results

The final tests and demo of the system performed well, providing reasonable results so that one could easily tell that they were listening to one person and not the other. However, the audio quality was sometimes limited. Certain improvements can be made in the future to help resolve these issues.

The measured delay of the project was about 250ms, which was below our target of 300ms.

3.1 Verifying Array Response

As shown in Table 3, we were able to mathematically calculate the ideal radiation pattern for the our array, for a variety of frequencies, and determine the 3dB beamwidth in each case.

To determine the real-world performance of the beamforming algorithm, we devised the following testing scenario:

1) For a constant radius 9 feet away from the center of the array, we placed markers every 5 degrees, as shown in Figure 15.
2) The beamforming algorithm was aimed at fixed angle ϕ_0, corresponding to one of the marks.
3) A speaker playing a pure tone was moved sequentially through each of the markers, and the audio recorded during this mark was saved as its own sample.
4) The power in dB for each sample was calculated relative to the sample collected at ϕ_0

We performed this procedure for the best case scenario (4000 Hz), the average case (2000 Hz) as well as the worst case scenario (1000 Hz). Both cases were performed when the array was steered to 60^o, as well as when the array was unsteered at 90^o (broadside).

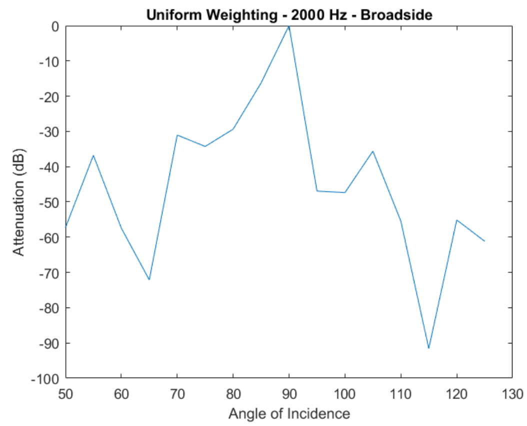


Fig. 16. Experimental Response to 2000 Hz, with Uniform Weighting

Fig. 17. Experimental Response to 1000 Hz, Steered to 60 degrees, with Dolph-Chebyshev Weighting

Figure 16'shows the experimental results for a 2000 Hz signal, when the beamformer is aimed at 90 degrees, and all microphones are uniformly weighted. In this figure, one can see that by the 85^o and 95^o degree marks, the signal attenuation has dropped to at least -20dB. Thus, we can say that for this frequency and this steering angle, the beamwidth is less than 10 degrees.

Figure 17 shows experimental results for a 1000Hz signal, when the array is steered towards 60^o, and Dolph-Chebyshev weights are applied. We can see that due to the fact that the beam is steered away from broadside, and that the microphone distance is one quarter of the wavelength of the target signal, the beamwidth has increased to about 20

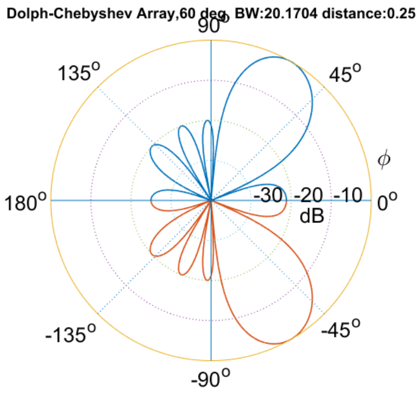


Fig. 18. The theoretical performance of Figure 17

degrees. We can also see the effects of the Dolph Chebyshev weighting, as all of the sidelobes reach a maximum of -20dB. Comparing Figure 17 with 18, we can see that the experimental appears to match the qualities of the theoretical.

3.2 Final Specification Comparison

Specification	Desired	Achieved
Array Size	< 2m	0.914m
Target Identification Range	> 20ft	18 ft
Angle of Operation	$30^o to 150^o$	30^o to 150^o
Maximum Targets	4	6
3dB Beamwidth (Broadside)	15^o	$< 10^o$ (best case) $< 20^o$ (worst case)
Operating Band	1000 - 4000 Hz	1000 - 4000 Hz
Delay	$< 300ms$	250 ms

TABLE 5
Achieved specifications

The final performance parameters for the Earbeamer system are displayed in Table 5. Generally, we were able to achieve our desired level of performance.

Production	1 unit	1000 units
Printed Circuit Board	\$ 59.33	\$ 16.04
Microphones	\$ 9.95	\$ 8.46
Kinect Sensor	\$ 139.98	\$ 139.98
ADC	\$ 700.00	\$ 700.00
Housing	\$ 30.70	\$ 6.20
Unit cost	\$1,090.29	\$ 997.94

Fig. 19. Unit and Estimated Production Costs of the Final EarBeamer Design

Our final cost was calculated as shown in Figure 19. The ADC was the most costly component of our system, but a production design would incorporate a different, much more cost-efficient ADC implementation.

4 Project Management

As each team member has unique professional and educational experiences, we divided the implementation of the project subsystems according to our particular strengths:

- Matteo Puzella is our sole Electrical Engineering major, with a particular interest in microwaves. He is applying his knowledge of filters to our project, as well as contributing to some of the more physical aspects of the project.
- Aaron Lucia has an interest in system design, but also signal processing, and he is applying that to the beamforming algorithm and pipeline design.
- Nathan Dunn has previously completed an REU that involved image processing, so he implemented the Kinect integration. He also completed research into the physics and dynamics of a beamforming array, helping with the design of the microphone array and the techniques of improving the beamforming.
- Niket Gupta has developed an interest in mobile application development through past personal projects, and is working on the user interface with the mobile phone app, as well as figuring out the analog to digital conversion of the microphone signals.

With these particular specialties, each team member was assigned duties as seen in Figure **??**, in order to bring our project to completion. Following our successful MDR, we were able to integrate our subsystems together for CDR, and demoed a working prototype that could achieve parallel beamforming in realtime. After CDR, we had the opportunity to make incremental improvements and optimizations until Demo Day, such as adding Chebyshev weighting to the microphones.

Our success in achieving our objectives can be attributed our commitment to accountability and project management. Each team member had specific, well-defined responsibilities, and we tracked the progression of these responsibilities through weekly team meetings and adviser meetings. In addition to regularly scheduled meetings, the team communicated frequently using a chat application called Slack, with occasional Skype group video calls.

5 Conclusions and Recommendations for Future Work

Through research and project management, we were able to design and build a beamforming hearing aid system that successfully met the specifications we had outlined at the beginning of the project. Having now completed the project we have identified possible areas of improvement and future work.

One of the main drawbacks of our system is latency. Although we met our goal of 300ms of audio latency, the lag from the creation of a sound to its output from a user's headset is still noticeable. During our demonstration, many users reported difficulty comprehending speech, when the output from the headphones did not match the movement of a speaker's lips. Typical commercial quality hearing aids offer latency under 25 milliseconds. [23]

In order to approach the latency of commercial hearing aids, the audio processing software that we wrote would have to be migrated from software to hardware. Software filtering, which is needed to separate the human speech frequency spectrum into our targeted octaves, is computationally intensive. The convolutions required for our FIR software filtering requires 2/3 of the CPU time for our program. Implementing the FIR filters and beamforming algorithms in hardware on an FPGA would significantly reduce the processing overhead.

Further improvements to the beamwidth and audio quality would be possible with another Analog to Digital converter option. For simplicity, we used the ADC of 2016's Project Sauron, which provides a 16 channel, 250 kilo-samples/sec multiplexed conversion. An ADC with more channels would allow us to use more microphones. More microphones would in turn allow us to add another sub-array to the system, without removing microphones from the two existing subarrays. This would allow us to capture a greater portion of the human speech frequency spectrum without sacrificing beamwidth. Audio quality could also potentially be improved by implementing a noise cancellation algorithm.

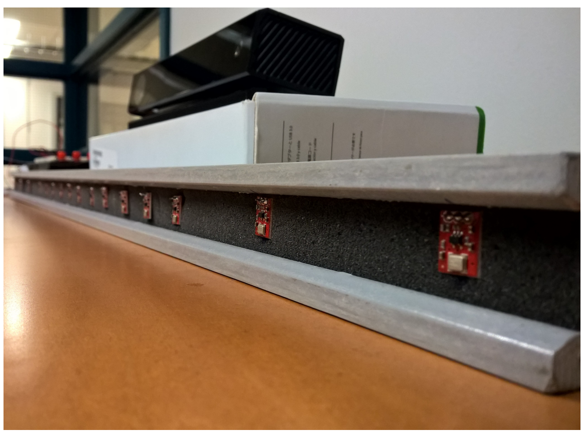


Fig. 20. The final array system and Kinect. Microphones sit on a foam layer, and connect to a PCB inside the housing. When in use, a black cloth sleeve is placed over the array to hide the microphones

6 Acknowledgements

We want to acknowledge the SDP 16 project Sauron team for their advice and use of some of their hardware from last year. We would also like to acknowledge our advisor, Professor Parente, for helping us to stay organized and on schedule, as well as Professors Kelly, Goeckel, Yang, and Tessier for their advice on beamforming applications and planning our project.

References

[1] National Institute on Deafness and Other Communication Disorders. *Age-Related Hearing Loss*. Tech. rep. 97-4235. US Department of Health and Human Services, National Institutes of Health, 2016.

[2] David Myers. "Silently Suffering From Hearing Loss Negatively Affects Quality of Life". In: *American Psychological Association* (Oct. 2016).

[3] FR. Lin et al. "Health ABC Study Group FT. Hearing Loss and Cognitive Decline in Older Adults". In: *JAMA Intern Med* (2013), pp. 293–299.

[4] *Hearing Aids*. URL: https://www.nidcd.nih.gov/health/hearing-aids.

[5] *Digital Hearing Aids*. URL: Retrieved%20from%20http://www.amplifon.com/web/uk/hearing-health/digital-hearing-aids.

[6] Patrick Freuler. *Is there a hearing aid price bubble?* Oct. 2014. URL: https://www.audicus.com/hearing-aid-price-bubble/.

[7] Wayne Staab. *Is Your Hearing Aid a Beamformer?* Apr. 2015. URL: http://hearinghealthmatters.org/waynesworld/2015/beamforming-hearing-aid/.

[8] Thompson Stephen. *Directional Microphone Patterns: They also have disadvantages*. URL: http://www.audiologyonline.com/articles/directional-microphone-patterns-they-also-1294.

[9] S Kochkin. "MarkeTrak V: Why my hearing aids are in the drawer: The Consumer's Perspective". In: *The Hearing Journal* 53 (2000), pp. 34–42.

[10] Jeff Rodman. *The Effect of Bandwidth on Speech Intelligibility*. Tech. rep. Polycom, Inc, Jan. 2003.

[11] *Transmission planning for voiceband services over hybrid Internet/PSTN connections*. G.177. International Telecommunication Union. Sept. 1999.

[12] Hamad Mojahed and Yi Wayne. *Acoustic Beamformer*. Senior Design Project, Advisor Prof. Goeckel. Apr. 2004.

[13] Arash Mashayekhi et al. *Selective Voice Tapping System (SVTS)*. Senior Design Project, Advisor Prof. Goeckel. Apr. 2006.

[14] J.A. Danis et al. *The Acoustic Beamformer*. Senior Design Project, Advisor Prof. Zink. Apr. 2014.

[15] Walter Brown et al. *Sauron Security Final Design Review Report*. Senior Design Project, Advisor Prof. Wolf. Apr. 2016.

[16] Sophocles Orphandis. "Electromagnetic Waves". In: 1st ed. Rutgers University, 2016. Chap. Antenna Arrays, pp. 1089–1090.

[17] R. Smith. "Constant beamwidth Receiving Arrays for Broadband Sonar Systems". In: *Acustica* 23 (1970).

[18] Sophocles Orphandis. "Electromagnetic Waves". In: 1st ed. Rutgers University, 2016. Chap. Array Design Methods, pp. 1144–1156.

[19] *Group Delay*. URL: https://www.microwaves101.com/encyclopedias/group-delay.

[20] David M Poza. *Microwave Engineering*. 3rd. Wiley India Pvt. Limited, 2009.

[21] *Kinect Hardware*. URL: https://developer.microsoft.com/en-us/windows/kinect/hardware.

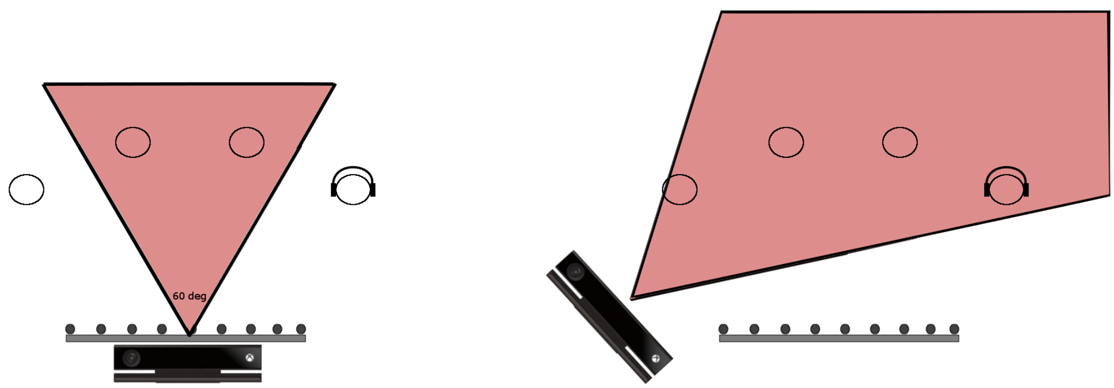
[22] Cecilia Chen. *Verifications of Specifications and Aptitude for Short-Range Applications of the Kinect v2 Depth Sensor*. NASA Glen Research Center. Slide Deck. Sept. 2014.

[23] Justin Burwinkel, Martin McKinney, and Jason Galster. *Acceptable Hearing Aid Throughput Delay For Listeners with Hearing Loss Under Noisy Conditions*. Starkey Hearing Technologies.

APPENDIX A

Category	Deliverable	Assignee
Target Identification	Extraction of Multiple Targets from Kinect	Nathan
Mobile Application	Framework for Communication Between Computer and iOS App Display Graphical Representation of Targets on Screen, using mocked coordinates	Niket
Beamforming	Delay sum Beamforming algorithm with simulated input	Aaron
AD/C	Verify that a given reference tone is digitized through a single channel Obtain verified values of ADC on computer	Matteo/ Niket
Array Design	Create Array Geometry to meet beamwidth goal and avoid spatial aliasing	Nathan
Filter Design	Simulate lowpass filter designs to determine optimal anti-aliasing filter	Matteo

Fig. 21. MDR Deliverables



(a) a

(b) b

Fig. 22. The Angle of View is too narrow to adequately cover a room, so the Kinect must be offset from the microphone array to identify all possible targets

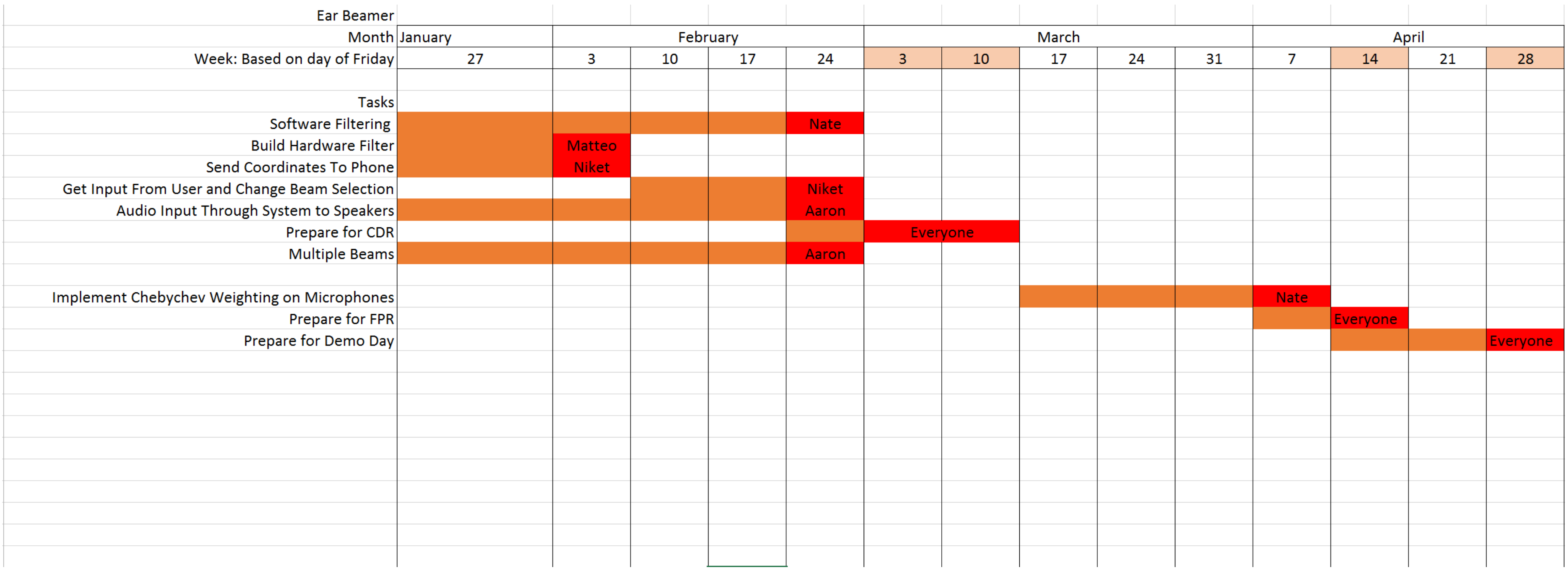


Fig. 23. Gantt Chart for the Remainder of the SDP Projet