SIGNAL PROCESSING EE 229

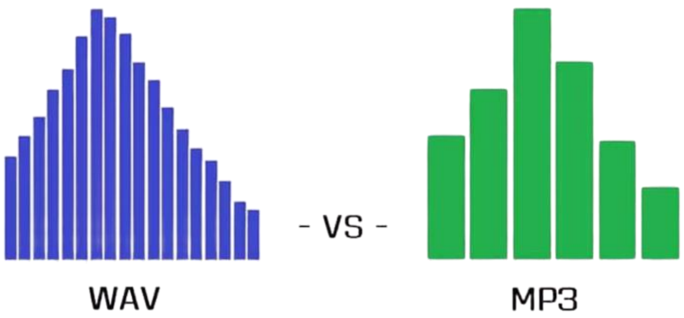
PAVITPAL SINGH – 23B3967

VINAY AGARWAL -23B3914

In this project, we try to understand how to use impulse response to produce output for an input file.

Glossary:

- Stereo file: This is an audio file that has sound amplitude data stored in two channels,i.e. one for right and one for left.
- Input file: This is an audio file. We want to know how it will sound in a given room or surroundings. In our case, it is the audio file BheegiRegular.wav..
- Convolution: This process allows us to take the source-emitted sound signal (input file) and outputs the left- and right-ear received signals. Since .wav files consist of amplitude vs time stamps (or n), we do discrete signal convolution of the input file with the RIR(Room Impulse Response) file.
- .wav: WAV files are uncompressed and lossless, meaning they retain the original audio and are considered high quality. WAV files are often used for studio recordings, audio mastering, and publishing audio for TV, film, radio, and commercials. They are also suitable for editing because they contain more data. However, WAV files are more extensive and may not be ideal for streaming or downloading.



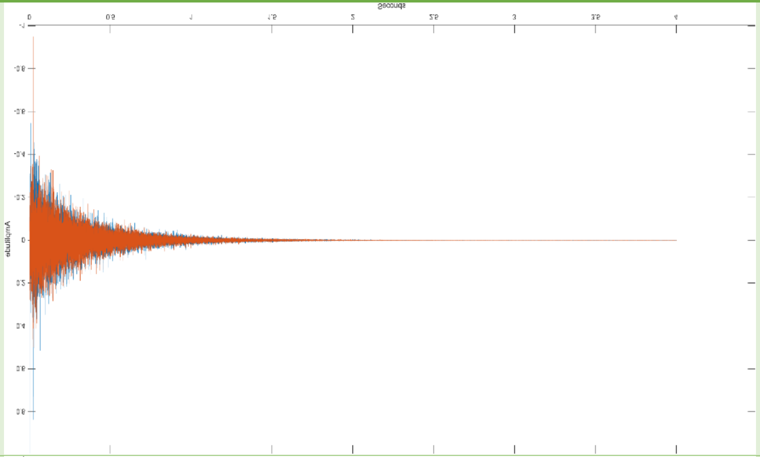
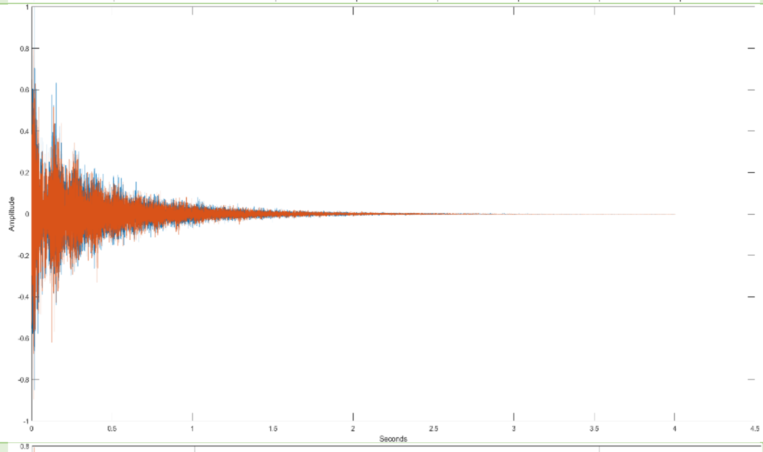
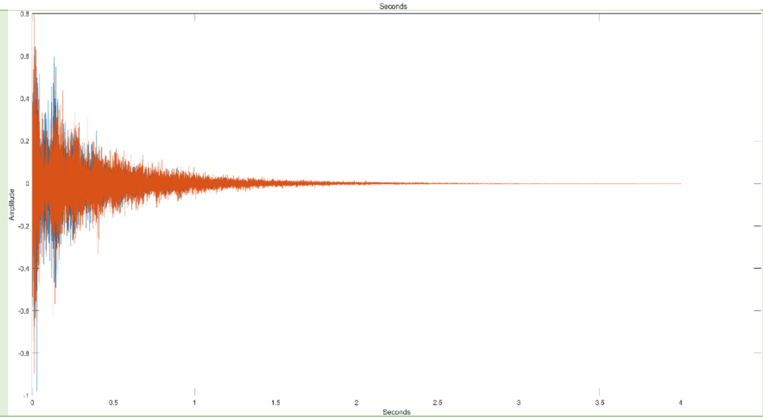
We are going to use Scilab to perform calculations. The Scilab code is explained below:

```
clear; close; clc;

//input wav file
[inp, fs_inp] = wavread("BheegiRegular.wav");


//loading 2 channel RIR
rir = wavread("RIR-file-name");

//SINGLE CHANNEL CONVOLUTION FUNCTION
function[output] = fun_conv(inp1,rir1)
    inp_len = length(inp1);
    rir_len = length(rir1);
    out_len = inp_len + rir_len - 1;
    output = zeros(1,out_len);
    for n = 1:out_len
        s = 0;
        for k = 1:rir_len
            if (n-k+1 >0 & n-k+1 <= inp_len) then
                s = s + rir1(k)*inp1(n-k+1);
            end
        end
        output(n) = s;
    end
endfunction


//obtain RIR for left channel
rir_left = rir(1,:);
//obtain RIR for right channel
rir_right = rir(2,:);

//obtain convolved signal for left channel
out_left = fun_conv(inp,rir_left);
//obtain convolved signal for right channel
out_right = fun_conv(inp,rir_right);

//obtaining stereo sound by combining two channels
out = [out_left;out_right];
out = out/max(abs(out));  //normalization

//playing convolved signal
playsnd(out,fs_inp);

//writing convolved signal
wavwrite(out,fs_inp,"Output-file-name.wav")
```

This is a very time-consuming process. Therefore, we used the input file BheegiRegular-part.wav, and the output generated is output1.wav. We only calculated the first second of the output file.

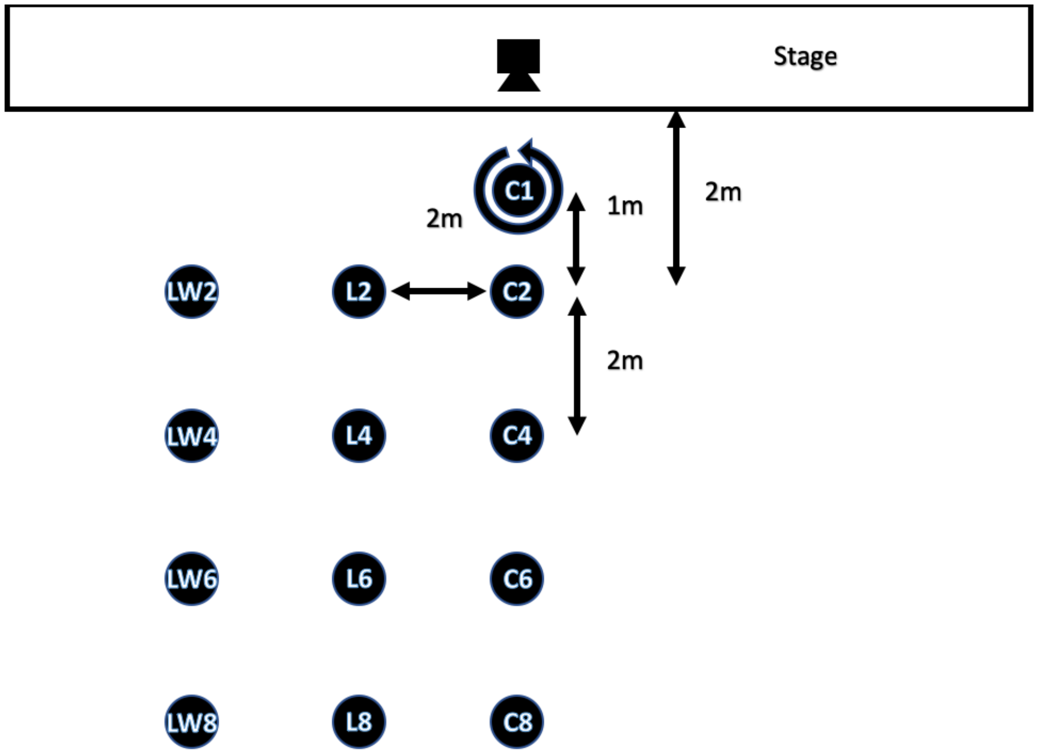
We want to experiment with more impulse response files, and using the above method would not be possible due to the longevity of the code's processing time. We have used the Scilab inbuilt function conv() for further convolutions. This function implies FFT (Fast Fourier Transform). The algorithm is based on Fourier transforming the input file and impulse response into the frequency domain, multiplying the transformed files and then applying inverse Fourier transform on the product to get output. The algorithm first breaks the input into chunks for faster calculations. This algorithm is many times faster than the above code.

We have convolved the BheegiRegular.wav with various room impulse response files that were provided to us. The results are tabulated below.

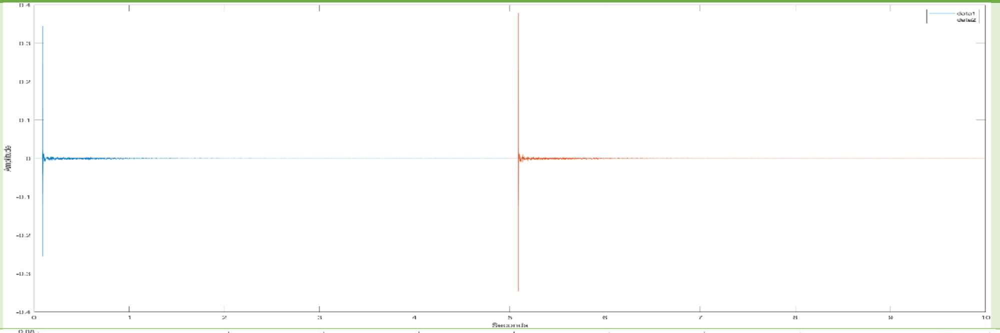
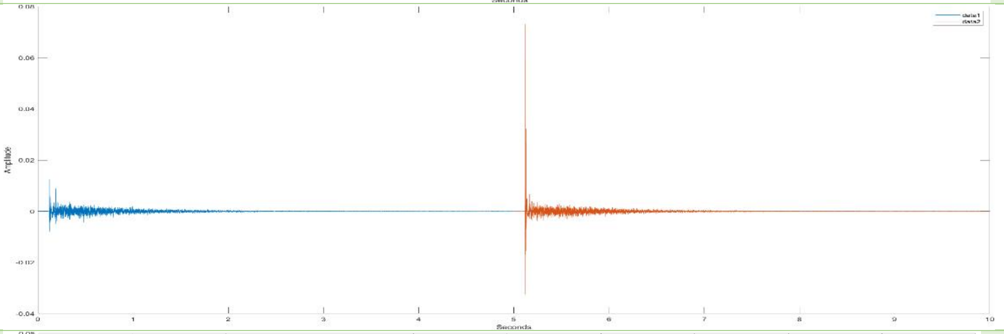
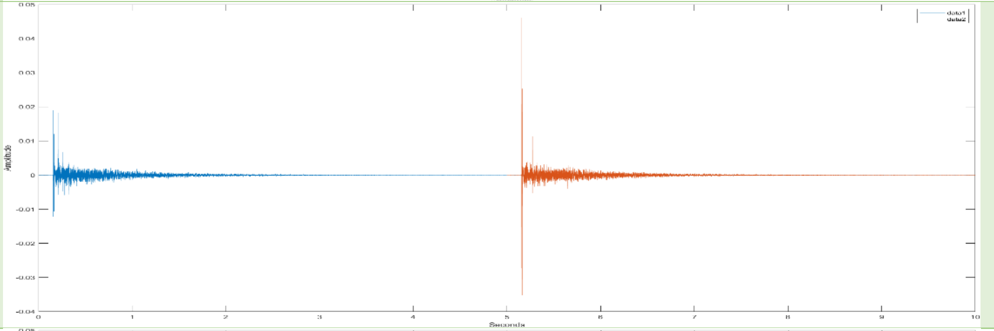
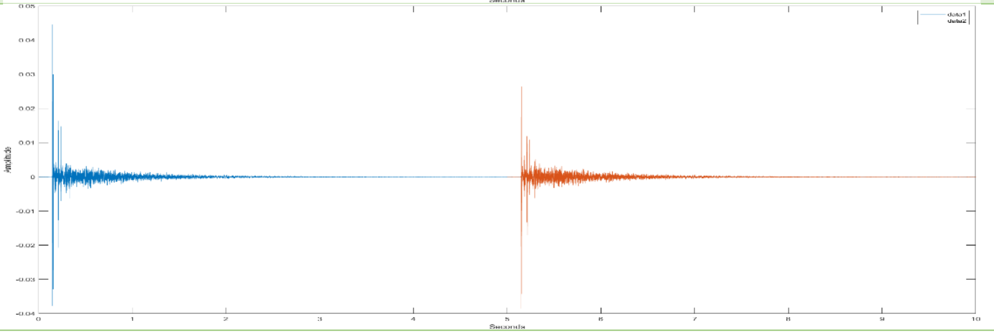
RIR file	Impulse Response plot(orange=> channel 2, blue=> channel 1)	Output file	Observations
Five_Columns_Long_16k.wav		Output_five_columns.wav	This impulse response is relatively smoothly attenuated. In the output file, we can hear each word of the song, and it gets damped smoothly. We cannot hear any distinct echo.
long_echo_hall_16k.wav		Output_echo_hall.wav	The plot shows that impulse response consists of distinct echoes attenuated over time. The output file shows that each word of the song and its echo can be heard. The output file feels like being in a large room where consecutive echoes are heard.
parking_garage_16k.wav		Output_parking.wav	Even though the impulse response plot is similar to the previous one, we can see that the first echo is relatively more dominant.

We have seen how different impulse responses can give different outputs and how Room Impulse Responses can be used to simulate being in particular surroundings. Till now, we have worked with impulse responses, which were similar for both channels. Now, we will work with impulse responses, which are different for two channels. We will use the dataset [360° Binaural Room Impulse Response (BRIR) Database for 6DOF spatial perception research (zenodo.org)](zenodo.org).

The data consists of acoustic data captured for 13 locations per the following setup.



We will first convolve BheegiRegular.wav with RIR.wav files for some locations and analyse the result. We have not used the normalising function: `out = out/max(abs(out));` to see the effect of distance between source and listener.

Position of listener	Impulse Response plot (blue=> channel 1 and orange-> channel2) Channel 2 is displaced by 5 seconds for better comparison	Output	Observations
C1		C1m_output.wav	Impulse responses of the C1 position are of maximum magnitude among the examples and the left and right responses are also of similar magnitudes.
LW2		LW2_output.wav	The right channel impulse response is evidently of larger magnitude, and the position of LW2 explains that. The output file also simulates the LW2 position.
LW8		LW8_output.wav	The right channel is of larger magnitude in this case as well, but the relative difference is less than in LW2. This is explained by the fact that in LW8, the source is more towards the front than in LW2.
C8		C8_output.wav	The two channels are of similar magnitude, but their magnitude is less than that of C1. This is due to the distance of the listener from the source.

We have seen how impulse responses can virtually take us in a room/ setting. Till now, we have worked only with stereo impulse response. Now, we will work with BRIR and HRTF files.

- HRTF: Humans determine a sound source's location in three fundamental ways. These are called sound localization cues. The first is an interaural level difference, or ILD. If a sound is louder in the left ear than in the right ear, you will naturally perceive the sound to originate from the left side. The second localization cue is an interaural timing difference or ITD. When a sound originates from the right side, for example, the sound will reach the right ear slightly before it reaches the left ear. There is also a third sound localization cue – HRTF. HRTF stands for Head-Related Transfer Function. We are talking about the effect that the listener's head has on the signal. Our subconscious awareness of the impact of the head, outer ears, and shoulders on sounds around us opens the door to more precise localization. If a sound comes from the left side, it will not only be louder overall in the left ear, but the high frequencies will also be attenuated or reflected before they reach the right ear. That will result in a slightly darker sound quality in the right ear. The shape of the pinnae also plays into this, filtering sound differently depending on the angle at which the sound arrives. Therefore, sounds from behind the listener will undergo a slightly different transfer function than sounds from in front of the listener.

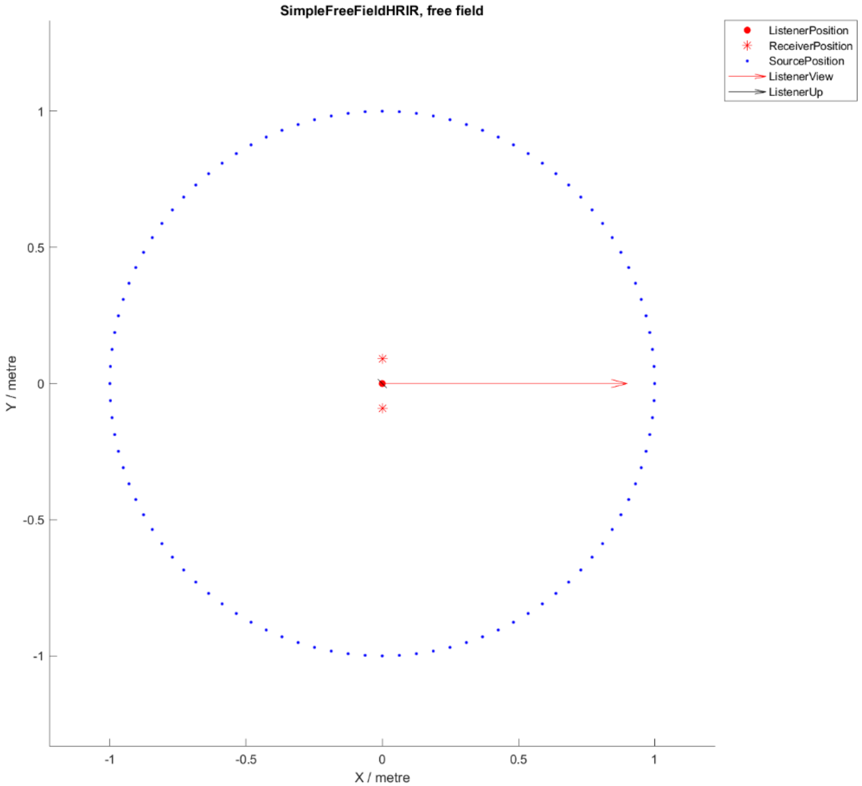
- BRIR: This is an impulse response recorded using binaural microphones. These special microphones simulate a human head's average size, density, and shape. Other microphones even add an artificial torso to capture the cues the shoulders and chest provide in localizing a sound source. They may be as simple as shown below:

3Dio's FS range of binaural mics rely solely on their prosthetic ears to simulate the human hearing system.

- SOFA file: SOFA is a format for storing spatially oriented acoustic data like head-related transfer functions (HRTFs) and binaural room impulse responses (BRIRs).

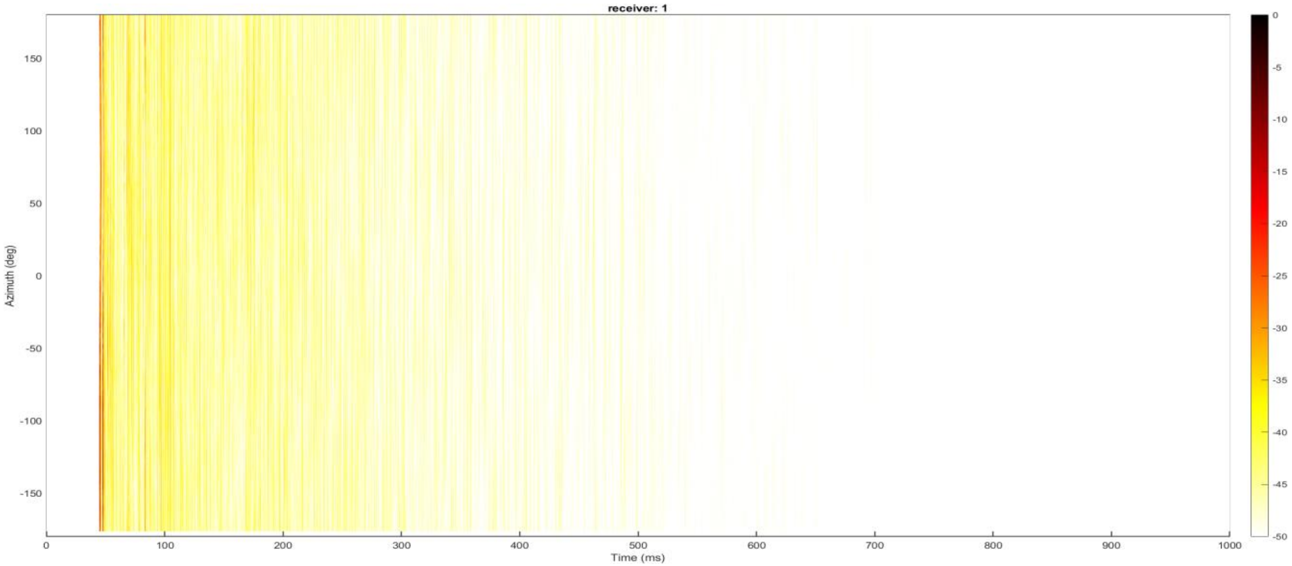
We will use the [360° Binaural Room Impulse Response (BRIR) Database for 6DOF spatial perception research (zenodo.org)](#) and load the .sofa file for the L6 receiver. We will use MATLAB for this part and have added a toolbox ([SOFA Toolbox for Matlab/Octave](#)) to work with .sofa files. Here are our observations from the .sofa file.

This is a [SimpleFreeFieldHRIR](#) type of sofa file. This file has 100 HRTFs measured by rotating the binaural microphone by 3.6° each time. The HRTFs are stored as FIR (Finite Impulse Response). Each of these 100 files has linked metadata containing the location and direction vectors. Hence, these files (their data and metadata) can be processed to study the surroundings and generate various parameters.



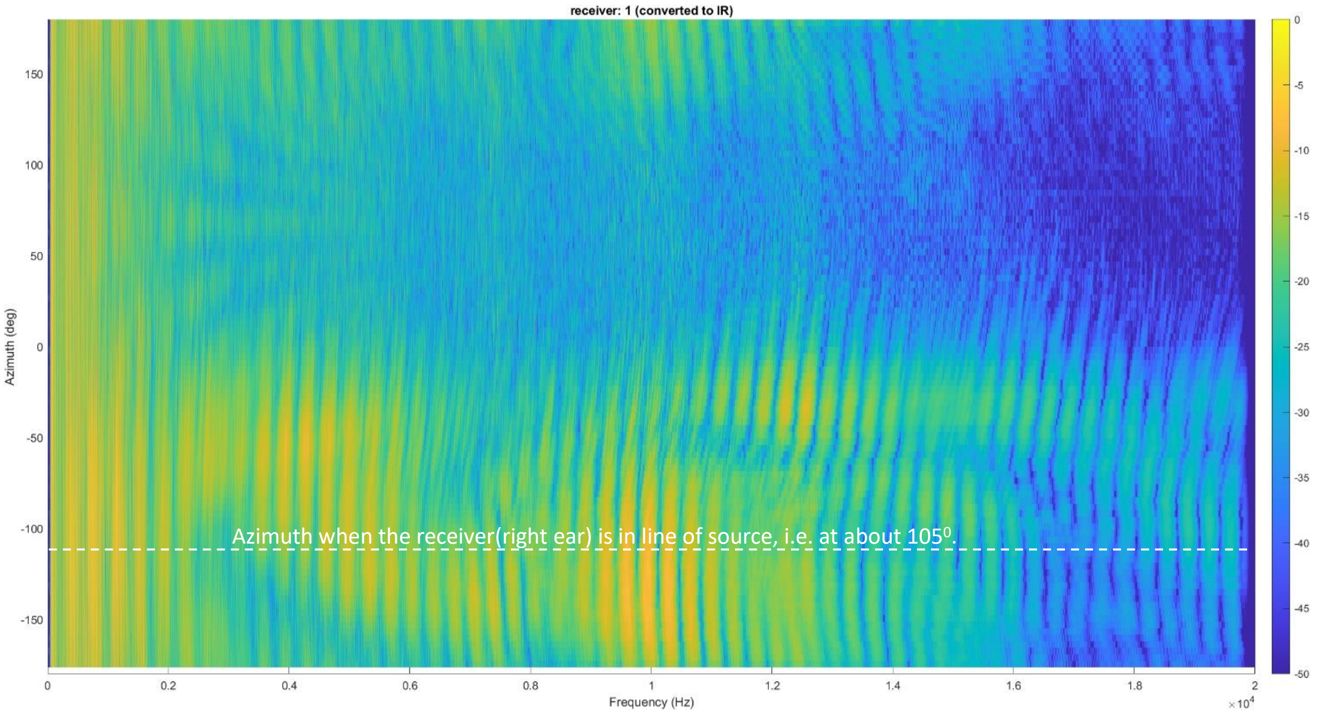
Plot showing various angles at which the impulse response was measured.

Receiver position represent the binaural microphone setup.



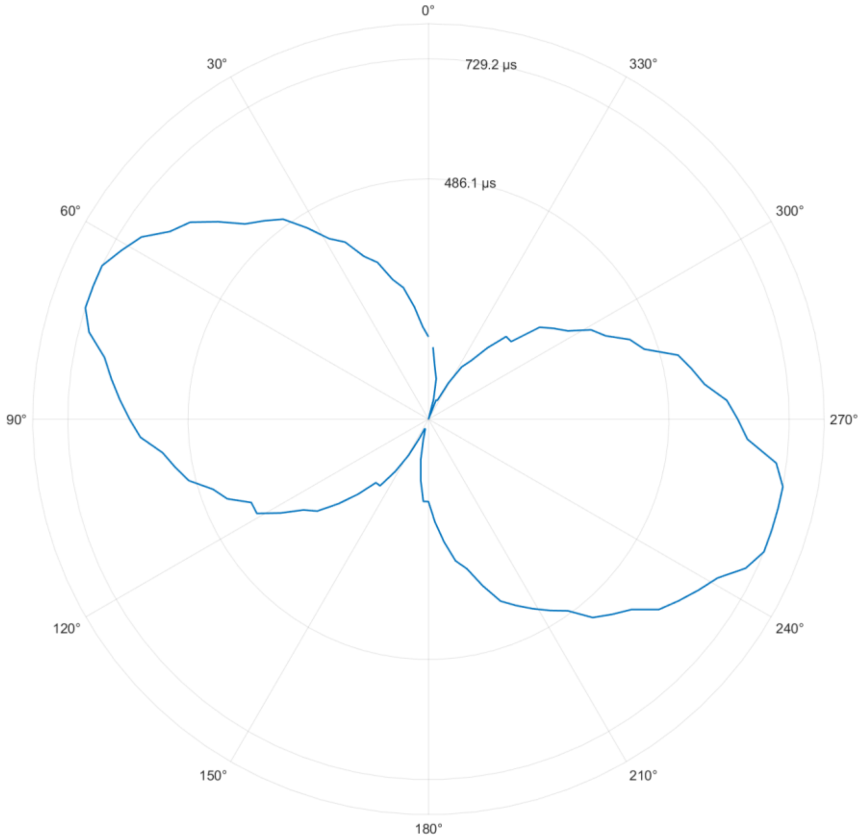
This is Energy Time Curve for receiver 1 in the horizontal plane.

The energy decreases with time as the impulse dies.



This is Magnitude spectra in the horizontal plane.

We can see higher frequencies getting attenuated as the receiver turns away from the source.



This is plot of Interaural time delays.

We can see that delay is zero for azimuth of about -15^0 which is the angle between listener front sight and source. Hence this sofa file can be used to calculate the azimuth of the source with respect to receiver.

Sources:

[Surround Sound In Headphones? | HRTF & Binaural Audio Explained – Audio University (audiouniversityonline.com)]

[Bacila, B. I., & Lee, H. (2019). 360° Binaural Room Impulse Response (BRIR) Database for 6DOF Spatial Perception Research. Presented at the Audio Engineering Society Convention 146, Dublin, e-Brief 513]

[SOFA (Spatially Oriented Format for Acoustics) - Sofaconventions]