(2EC502) Digital Signal Processing

Special Assignment

Topic: Hearing AID for Impaired People

Submitted by: Arjit Singh (21BEC010)
Dhruv Shah (21BEC030)

Submitted to: Prof. Rutul Patel

Abstract

Hearing aids are devices used by hearing impaired people to compensate for hearing losses. Digital hearing aids are better in performance over analog hearing aids, since analog hearing aids work as amplifiers. Digital hearing aids are programmable to adjust the gain value, enhance signal to noise ratio, reduction in noise, feedback cancellation etc. In this project, the simulation of the simple digital hearing aid is developed using MATLAB programming language. The implementation of this configurable digital hearing aid (DHA) system includes noise reduction filter, frequency-dependent amplification and amplitude compression. Through this project there is- reduction of white Gaussian noise, increase in the gain for frequencies which were difficult to hear, and shape the amplitude to prevent any of the frequencies from becoming too loud.

Introduction

Hearing Disorders happen normally as the age increases, which causes a decrease in normal functionality of the ear. Hearing disorders diminished sensitivity to the sounds normally heard. Deafness and speech perception are two categories of hearing losses. The audible frequency range for human ears is 20 Hz to 20 kHz and human hearing is most sensitive in the range of 1 kHz to 4 kHz [3]. Hearing is measured in decibels. In all frequencies 0 to 20 dB is the normal hearing range.

A hearing aid is a small electronic device that you wear in or behind your ear. It makes some sounds louder so that a person with hearing loss can listen, communicate, and participate more fully in daily activities. It can help people hear more in both quiet and noisy situations. However, only about one out of five people who would benefit from a hearing aid uses one. It has three basic parts: a microphone, amplifier, and speaker. The hearing aid receives sound through a microphone, which converts the sound waves to electrical signals and sends them to an amplifier. The amplifier increases the power of the signals and then sends them to the ear through a speaker.

There are two categories of hearing aid systems: Analog and Digital hearing aid system. Analog hearing aids are not much different from linear amplifiers and generally do not provide any noise-cancellation mechanism. Whereas digital hearing aids contain a very advanced degree of signal processing that offer significant environmental noise reduction. The analog hearing aids offer a generalized solution to the hearing impairment while we know that everyone's hearing characteristics are unique and therefore there should be specialized solutions according to the hearing impairment of the individuals. The digital circuits are more flexible than analog circuits. They can be precisely programmed to match the patient's individual hearing loss, sometimes at each specific frequency/pitch. This signifies the use of human audiogram. To compensate for the frequency dependent hearing loss, hearing aids can be fit to comply with an individual's audiogram so that different gains are applied to different frequency bands. Digital hearing aids can be operated with very little battery power, usually in mW.

Hearing loss is generally classified as mild, moderate, severe, or profound. The quietest sounds or softest intensity levels of sounds that can be perceived by people suffering from different hearing losses are summarized in Table 1:

Category	Softest Intensity Level
Mild Loss	25 – 40 dB
Moderate Loss	40 – 70 dB
Severe Loss	70 – 95 dB
Profound	95 dB or more

Table1. Hearing Thresholds associated with various categories of hearing losses

Methodology

Below is a block diagram for the MATLAB implementation of Digital Hearing Aid System. The input speech signal takes the form of a human voice. A Noise Addition filter adds noise into the clean input signal after which it is passed through a Noise Reduction System to suppress any noise. Then the filtered input signal passes through the Frequency Shaping System which modifies the spectral content of the speech according to the listening convenience of the hearing impaired. Finally, the amplified speech passes through Amplitude Compression System to ensure the overall gain of the amplified speech according to the listening comfort of the hearing impaired, before producing an adjusted output speech.

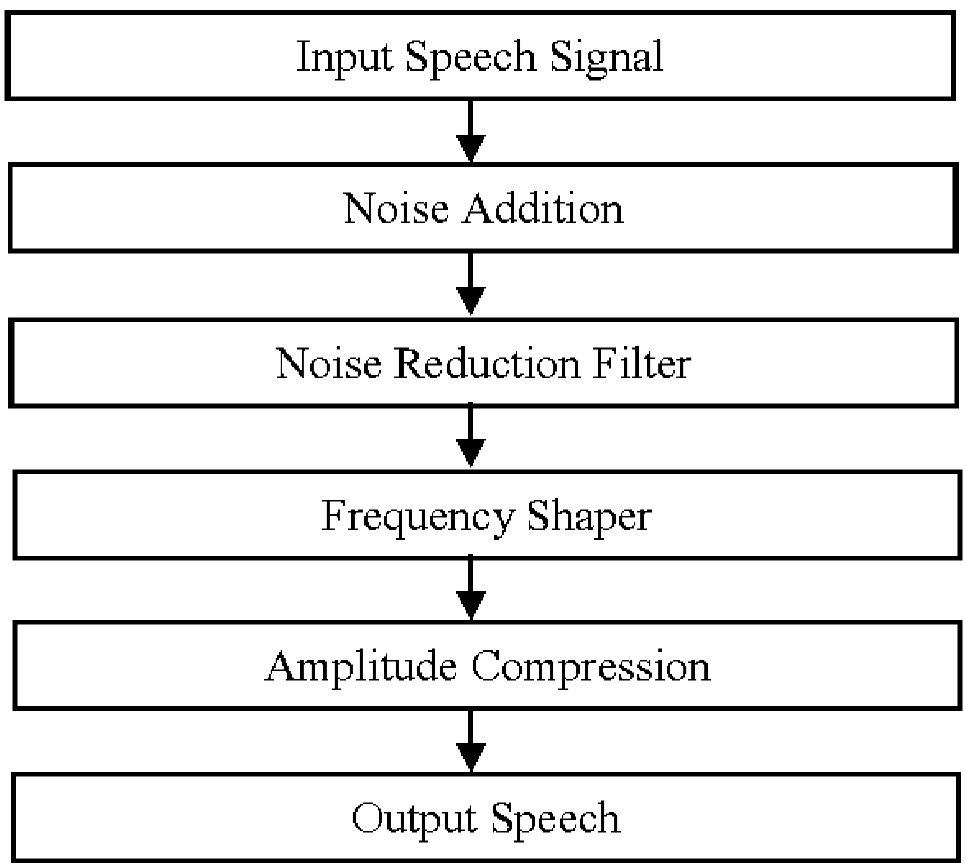


Fig. 1. Block diagram representation of the hearing aid system.

Let us look at each block in depth:

A. Noise Addition

Since the speech signal input for this system is pure, noise is introduced to simulate a real-world scenario. MATLAB functions are utilized to introduce Additive White Gaussian Noise (AWGN) and random noise into the input speech signal of this system. Noise (AWGN) possesses an equivalent amount of power per Hertz within a designated frequency band, exhibiting a consistent and uninterrupted frequency spectrum. It is composed of all frequencies with equal intensity and its probability density function is normal (Gaussian).

B. Noise reduction Filter

Following this, the noisy signal is denoised by passing it through this filter.
There are two techniques for signal denoising:

1. Using Wavelet filter function:
Wavelets are nonlinear functions and do not remove noise by low pass filtering like many traditional methods. For wavelets the amplitude, instead of the location of the Fourier spectra, differs from that of the noise. This allows for thresholding of the wavelet coefficients to remove the noise. If a signal has energy concentrated in a small number of wavelet coefficients, their values will be large in comparison to the noise that has its energy spread over many coefficients. These localizing properties of the wavelet transform allow the filtering of noise from a signal to be very effective.

2. Using Low Pass filter:
Low pass filtering approaches, which are linear time invariant, can blur the sharp features in a signal and sometimes it is difficult to separate noise from the signal where their Fourier spectra overlap. While linear methods trade-off suppression of noise for broadening of the signal features, noise reduction using wavelets allows features in the original signal to remain sharp.

Also, upon calculating the MSE (Mean Square Error) in both cases, it was found that wavelet approach results in very less MSE as compared to low pass filter approach. Thus, the wavelet filter function is chosen using Matlab's predefined function ‘wdenoise’.

C. Frequency Shaper

People with hearing loss often have different levels of loss at different frequencies, which leads to sounds being perceived as distorted relative to a normal-hearing individual because the various frequency components are not weighted in the expected manner. Thus, applying a uniform gain does not generally correct for hearing loss. Therefore, instead of amplifying the entire incoming signal, we need to enhance speech only in the hard-to-hear frequency bands for a particular hearing impaired. Research shows that most hearing-impaired people have difficulties hearing high frequency signals, so the frequency shaper needs to apply high gain at those frequencies to correct for the loss of hearing.

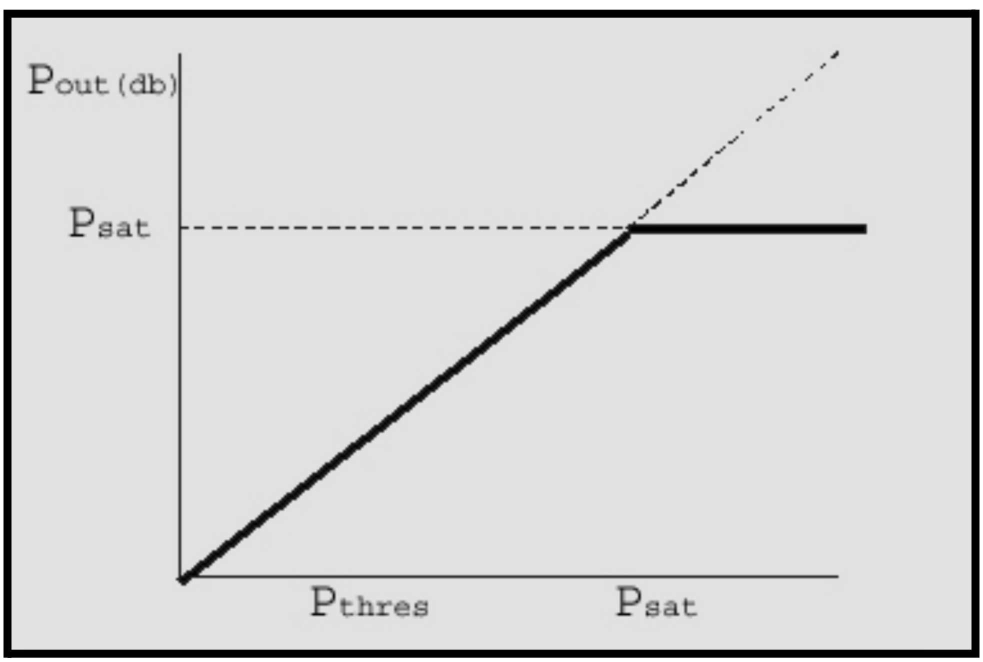
The filter applies a gain greater than one to the frequencies that the user has difficulty hearing. As one of its parameters, the filter takes in a vector of frequencies, determined by an audiologist, that defines the user's hearing characteristics. For each range, the frequency shaper applies a certain gain based on the user's specific hearing loss. Thus, our frequency shaper is completely configurable to any user.

D. Amplitude Compression

A normal hearing range extends from approximately 0 dB to 120 dB, where 0 dB is the Threshold of Hearing and 120 dB is the Threshold of Pain. Discomfort usually begins to occur around a saturation level of about 90 dB of sound. Hearing loss compresses the range of hearing, raising the Threshold of Hearing and typically lowering the Threshold of Pain. For example, a person with moderate hearing loss would have a Threshold of Hearing around 40 - 70 dB and a Threshold of Pain around 100 db.

The basic function of a hearing aid is to make sounds audible, yet not uncomfortably loud for the user. The Frequency Shaper amplifies the sound in the hard-to-hear frequency regions. During this process, some sounds may exceed a certain level and add to the listening discomfort of the user. Therefore, the hearing aid must imply some amplitude compression mechanism to control overall gain of the speech amplification system according to the listening comfort of the hearing impaired.

The job of Amplitude Shaper is to process the input signal sample-by-sample and ensure that output power does not exceed a given Saturation-Level Psat, as shown in figure below. For example: Person who has a moderate hearing loss and has a Saturation-Level (Psat) of 75 dB. So, the Amplitude Shaper must check, bit by bit, that output power does not exceed the given saturation level, Psat, which in this case is 75 dB. Output power is set to Psat for the levels above Psat.



Pseudo Code

1. Initially, a clean speech signal is taken as input.
2. To simulate real-world conditions, (AWGN) noise is intentionally introduced in the input signal.
3. Wavelet-based denoising is then applied to reduce the introduced noise.
4. A frequency shaping filter is used to enhance a set of difficult to hear frequencies according to the patient's needs using a band pass filter.
5. Amplitude compression techniques are employed to suppress the signal's gain to the max threshold value provided according to the user's need.
6. Final Adjusted signal is obtained.

Implementation and Simulation

The code, which is written in MATLAB, imports the input wave signal and retrieves its number of bits and sampling frequency. The signal is subsequently subjected to random noise and Additive White Gaussian Noise (AWGN) before undergoing processing through a variety of MATLAB functions to generate an output that is audible to the individual with hearing loss.

A speech signal sample is chosen from the predefined built-in audio file that is supplied by MATLAB. The file 'handel.mat' is imported into the code using the load command and is played for a duration of 10 seconds.

Consider an example, a patient suffers moderate hearing loss which is characterized by:

- Threshold of pain at 70 dB.
- Moderate hearing loss at medium frequency range of 500 – 2400 Hz. The required gain is 40 dB on this frequency range.

Also, the SNR (signal to noise ratio) of the noisy signal is taken to be 15 dB.

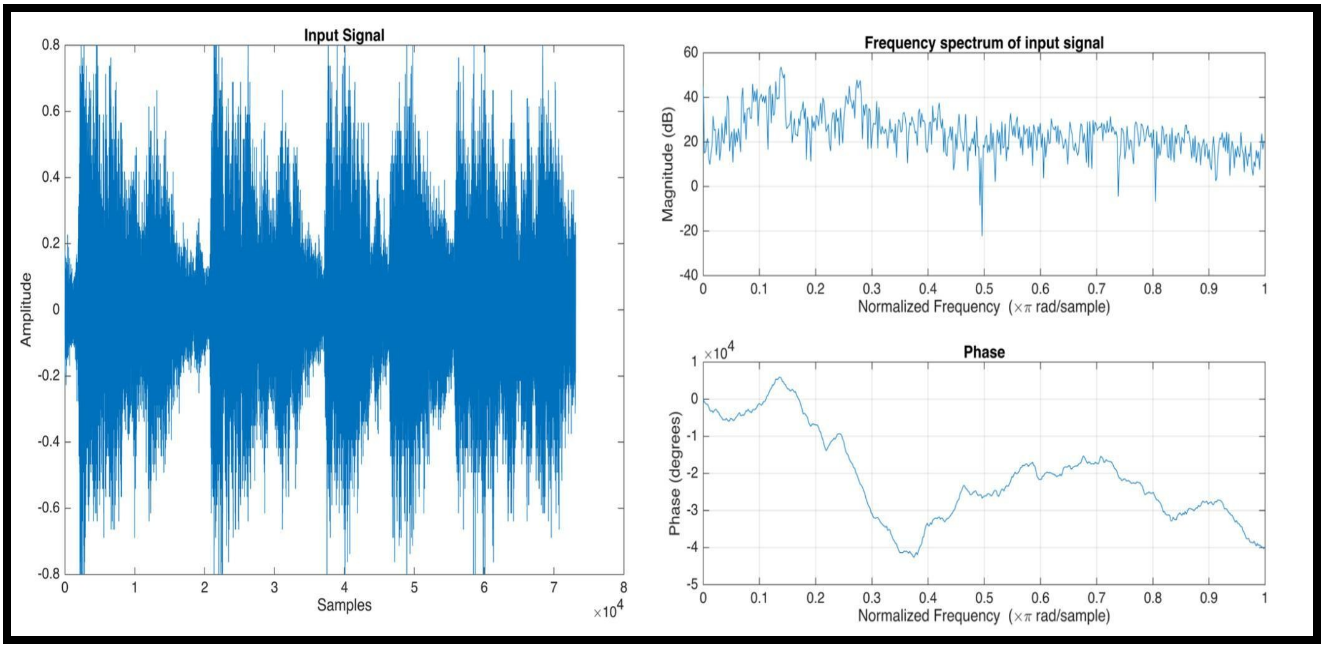
MATLAB Code:

```
% Loading a built-in audio file – handel.mat from MATLAB
clc; clear all; close all
load handel.mat
fs=Fs;

y = y(:, 1); original_sound = y;
figure,plot(y); title('Input Signal'); xlabel('Samples'); ylabel('Amplitude');

disp('Playing Input Sound')
sound(y,fs); pause(10);
figure; freqz(y)
title('Frequency spectrum of input signal')

%Adding Noise to the input signal, SNR = 15 dB
yy = awgn(y,15);  noi = yy;
figure,plot(yy); xlabel('Samples'); ylabel('Amplitude'); title('AWGN induced Signal');

disp('Playing Noisy Signal');
sound(yy); pause(10);
figure; freqz(yy)
title('Frequency spectrum of noisy signal')
mse1 = mean((original_sound(:)-yy(:)).^2)

% Wavelet Based Method for Denoising of the signal
x =
wdenoise(yy,'DenoisingMethod','bayes','ThresholdRule','Soft','NoiseEstimate','LevelIndependent',8,'wav
elet','sym8');
figure,plot(x); xlabel('Samples'); ylabel('Amplitude'); title('Denoised Signal');

disp('Playing denoised sound');
sound(x,fs); pause(10);
figure; freqz(x)
title('Frequency spectrum of denoised signal')
mse2 = mean((original_sound(:)-x(:)).^2)

% Frequency shaper using band pass filter, Gain_db = 40;
T = 1/fs; len = length(x);
p = log2(len); p = ceil(p); N = 2^p;
f1 = fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',400,500,2400,2500,80,0.5,80,2*fs);
hd = design(f1,'equiripple'); freqz(hd);
```

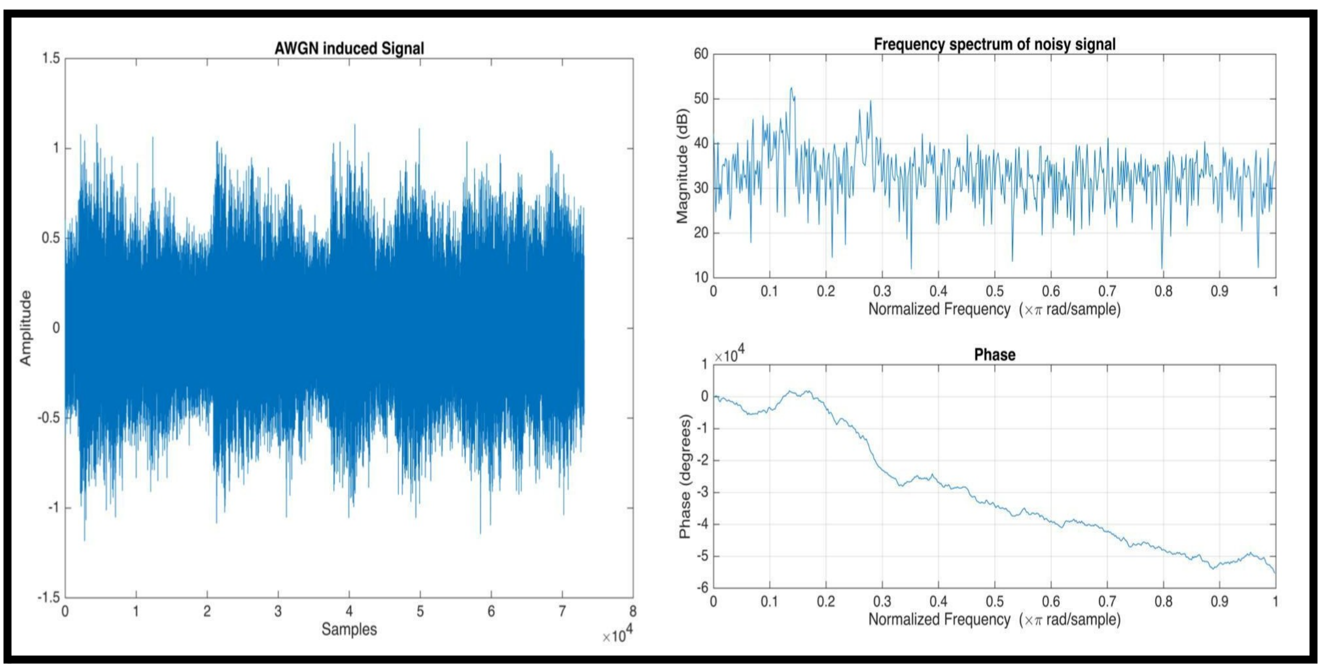
```
y = filter(hd,x); y = y*100;
figure; plot(y); xlabel('Samples'); ylabel('Amplitude'); title('Frequency shaped signal');
disp('Playing frequency shaped sound');
sound(y,fs); pause(10);
figure; freqz(y)
title('Frequency spectrum of frequency shaped signal')
mse3 = mean((original_sound(:)-y(:)).^2)

% Amplitude Shaper/Compression, P_sat_db = 80
out1 = fft(y); phase = angle(out1); mag = abs(out1)/N;
[magsig,~] = size(mag); threshold = 1000;
out = zeros(magsig,1);

for i = 1:magsig/2
        if (mag(i) > threshold)
        mag(i) = threshold;
                mag(magsig - i) = threshold;
        end
        out(i) = mag(i)*exp(j*phase(i));
        out(magsig-i) = out(i);
end
outfinal = real(ifft(out))*10000;

figure; plot(outfinal); xlabel('Samples'); ylabel('Amplitude'); title('Amplitude shaped signal')

disp('Playing amplitude shaped...');
sound(outfinal,fs); pause(10);
figure; freqz(outfinal)
title('Frequency spectrum of Amplitude shaped signal')
mse4 = mean((original_sound(:)-outfinal(:)).^2)

% Spectrograms

load handel.mat
figure;
subplot(2,1,1);
specgram(noi);
title('Spectrogram of Original (Noisy) Signal');


subplot(2,1,2);
specgram(outfinal);
title('Spectrogram of Adjusted Signal');
```

Results

1. An input speech signal – 'handel.mat' is loaded into the code. It is free of noise. The time and frequency response of the input signal are shown in the figures below respectively.



2. Now to simulate a real-life situation, AWGN with SNR of 15 dB is added to the input speech. The time and frequency response of the noisy signal are shown in the figure below respectively.

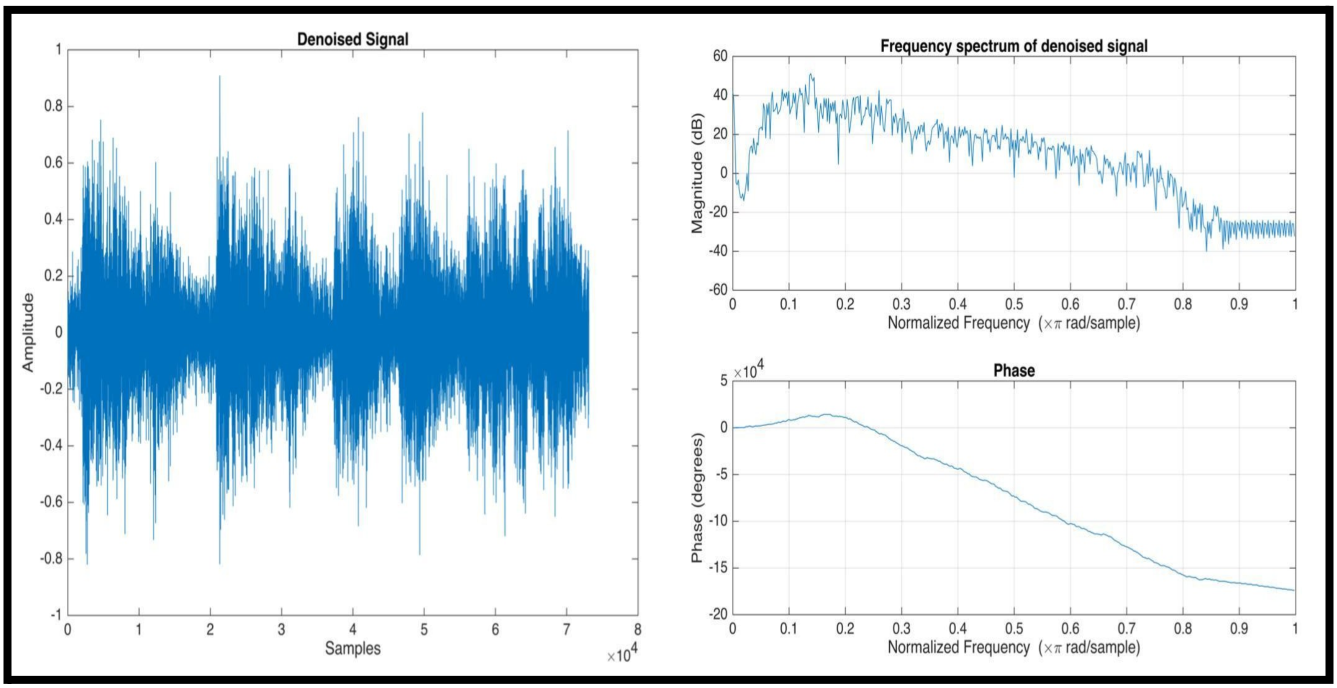


Here, MSE (Mean Square Error) is calculated between original sound and noisy signal (yy) as shown:

```
mse1 = mean((original_sound(:)-yy(:)).^2)
```

```
mse1 = 0.0314
```

3. The MATLAB Wavelet toolbox's 'wdenoise' function effectively reduces the presence of AWGN in the incoming noisy signal, achieving significant noise removal. The time and frequency response of the denoised signal are shown in the figure below respectively.



Here, MSE (Mean Square Error) is calculated between original sound and denoised signal (x) as shown:

```
mse2 = mean((original_sound(:)-x(:)).^2)
```

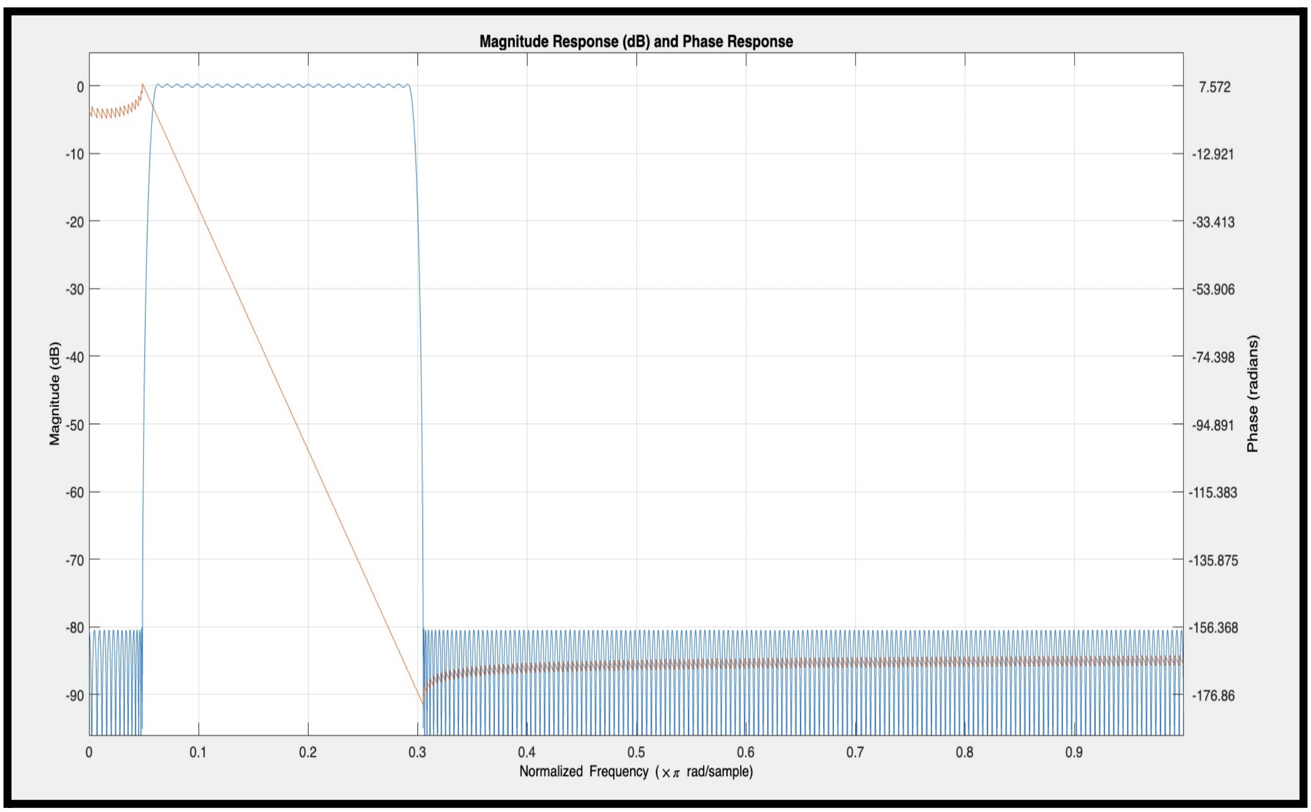
```
mse2 = 0.0154
```

Upon comparing the values of mse1 and mse2, it becomes evident that mse2 is approximately half of mse1. This observation confirms the effectiveness of noise reduction achieved through the wavelet approach. It is worth noting that the wavelet approach is significantly superior to the low-pass filter approach in noise reduction.

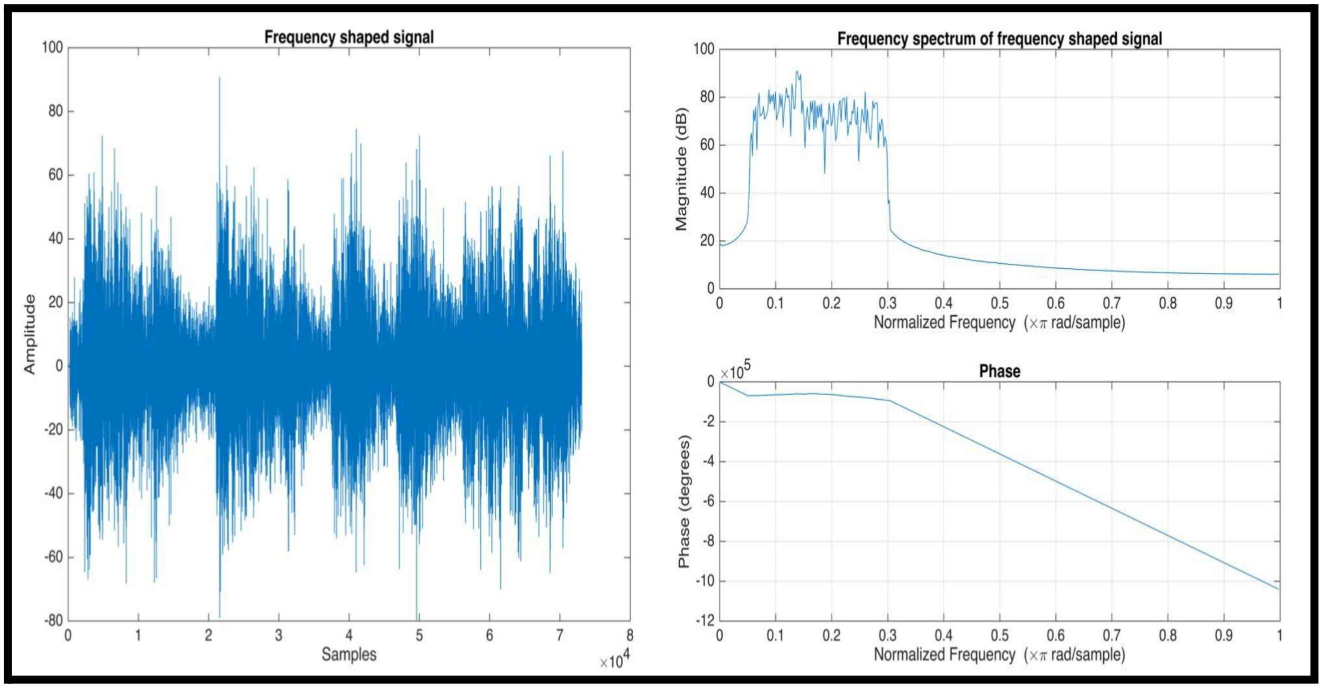
4. Subject 1, an individual with moderate hearing loss in the medium frequency range of 500 to 2400 Hz requires a 40 dB gain within this frequency range. To address this, the power of denoised signal in this range needs to be increased, and this is achieved by passing the signal through a bandpass filter which is a frequency shaper function and has a required gain of 40 dB, modified on specific frequency range as per the user requirements.

Frequency shaper function which has the following parameters and its output as shown below:

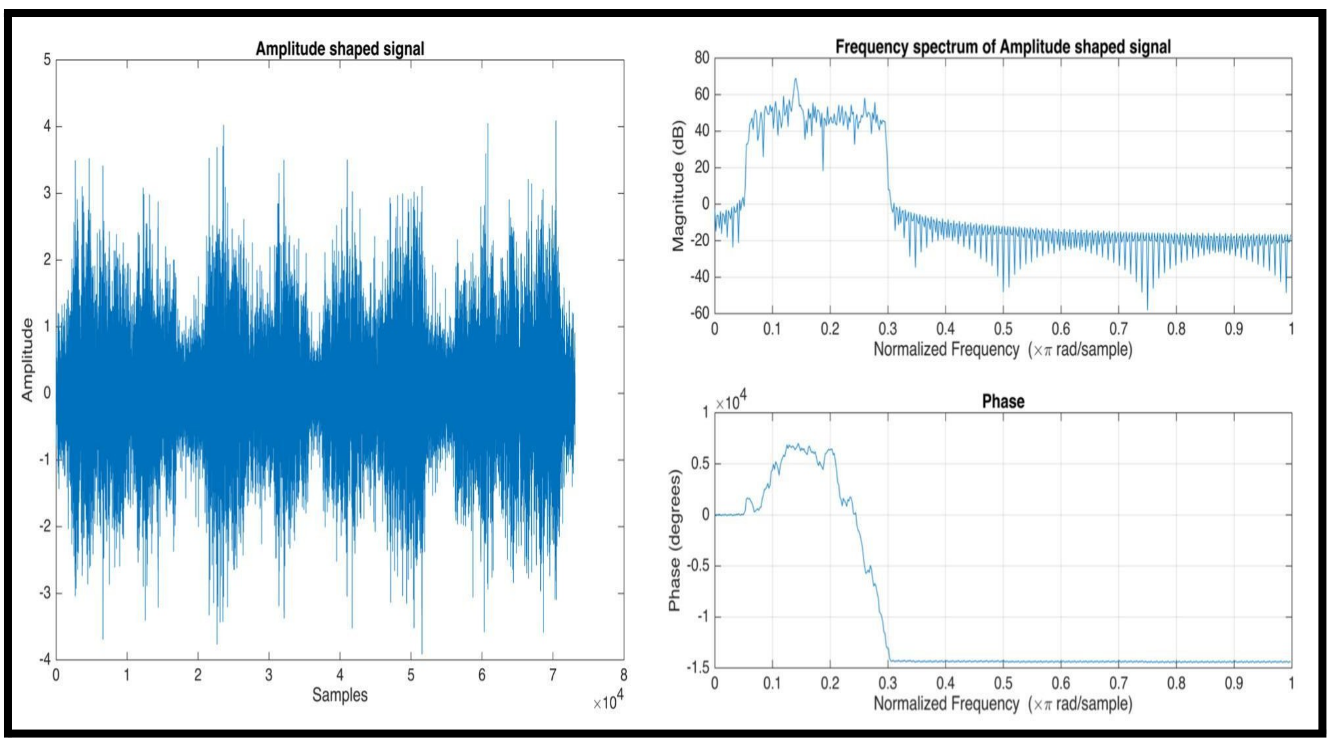
```
f1 = fdesign.bandpass('Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',400,500,2400,2500,80,0.5,80,2*f
hd = design(f1,'equiripple');
y = filter(hd,x);
freqz(hd);
```



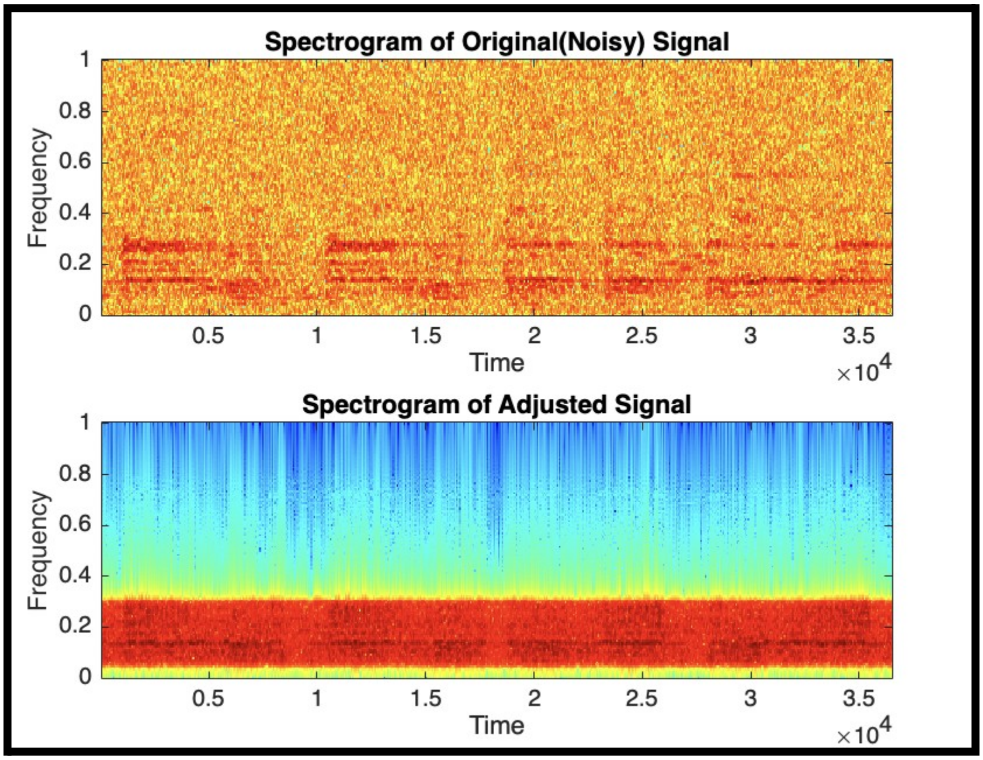
Following this, the time and frequency response of the frequency shaped signal are shown in the figure below respectively.



5. Now, Amplitude shaping filter reduces the power of frequencies whose magnitude exceeds that of the saturation value. Here the saturation value is 80 dB. Comparing the graphs of the original signal and the amplitude shaped signal, we saw that the amplitude of the noise in the signal was noticeably reduced.



We also compared the spectrograms of the Original (noisy) signal and the final adjusted signal and saw that the adjusted speech signal was stronger and more recognizable to the hearing-impaired individual.



Conclusion

Hearing aid technology has witnessed remarkable advancements in the past decade, primarily driven by the introduction of digital signal processing (DSP) within the hearing aid industry. As new technologies continue to emerge and evolve, they aim to address the unmet needs of consumers.

In the context of implementing a digital hearing aid system using MATLAB, sound processing undergoes digitization. It has been observed that by employing various functions such as denoising, frequency shaping, and amplitude limitation on speech signals, it is possible to make the signal more suitable for individuals with hearing impairments. This digitalization facilitates precise signal analysis and filtering, allowing signals to be processed in one or more frequency channels. Ultimately, the digital signal is converted back to its analog form.

The advantages of utilizing digital hearing aids are manifold and include the potential for improving the quality of life. These benefits encompass enhanced sound quality, increased comfort during listening, better communication in noisy environments, improved speech intelligibility in group conversations, and increased flexibility, particularly for individuals with progressive hearing loss.

References

[1] G. J. Proakis and G. D. Manolakis, Digital Signal Processing; principles, algorithm, and application, 3rd edition, Prentice Hall, New Jersey, 1996.

[2] Navdeep Kaur and Dr. Hardeep Singh Ryait., “Study of Digital Hearing Aid Using Frequency Shaping Function”, International Journal of Engineering Research and Technology (IJERT), Vol.2, Issue 5, May 2013.

[3] Zaman, S. S. Maghdid, H. Afridi, S. Ullah and M. Zohaib, "Enhancement of Speech Signals for Hearing Aid Devices using Digital Signal Processing," 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 2020.

[4] S. E. Voss, “Method and device to optimize an audio sound field for normal and hearing-impaired listeners”, 2006.