Department of Signals & Systems
Chalmers University

Laboratory Exercise 1

Speech Modeling, Analysis, Synthesis & Compression

March 29, 2009

Submitted By

Group 43

Rayyan Ali Qureshi

Department of Signals & Systems
Chalmers University
rayyan@student.chalmers.se

Sohaib Maalik

Department of Signals & Systems
Chalmers University
sohaib@student.chalmers.se

Submitted To

Irene Y. H. Gu

Department of Signals & Systems
Chalmers University
irenegu@student.chalmers.se

Table of Contents

List of Figures

1. Introduction

This lab focuses on how digital speech signals can be synthesized and compressed by using model-based methods. It implies linear predictive coding (LPC) analysis, and then synthesizing the single tone sound (stationary) and a speech sentence (non-stationary) in order to achieve compression. This lab also gives some insight of CELP speech encoder, and basics on pitch estimation methods.

2. Task 1 - Stationary Speech Signals: *Modeling, Analysis and Synthesis*

2.1.Generating a Stationary Speech Signal File

The first step in this task is to record a monotonic speech signal and plot it in the MATLAB.

2.1.1. Plotting of Speech Signal

Figure 1 shows the plot of recorded speech signal 'a'. This recording has been done for 2 seconds at a frequency of 8 kHz using mono channel. In the figure, 8000 samples correspond to 1 second.

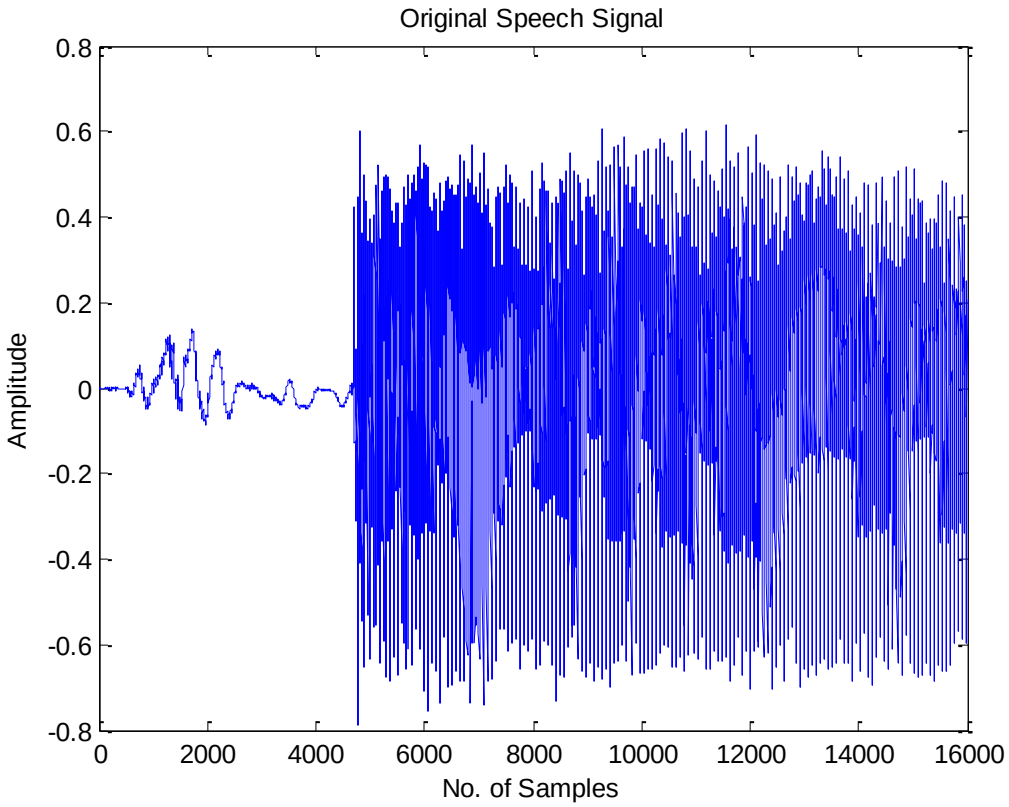

Figure 1: Recorded Speech Signal

2.1.2. Original File

Please see the attached zip file for a wav file named;

Myvowel.wav

2.2.Estimating the LPC Model Parameters

The second step is to apply the linear predictive coding (LPC) model on the speech frame of the stationary sound. The duration of the speech frame is 100ms. The all pole filter should be of order 10 and modeled as using equation given below**;**

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 + \sum_{j=1}^{p} a_j z^{-j}}$$

MATLAB function 'LPC' is used to compute model parameters and the variance (power) of the prediction errors.

2.2.1. Number of Samples in a Frame

As the duration of each speech frame is 100ms and sampling frequency f_s is 8 kHz. So the numbers of samples contained in the frame are;

$$L = Duration\ of\ each\ frame\ \times Sampling\ frequency = 100ms\ \times 8kHz = 800$$

So the numbers of samples contained in a single frame are 800.

2.2.2. Estimated $H(z)$ LPC Parameters

The estimated $H(z)$ LPC parameters are;

a_j, j=1, 2...10, =

-0.4925	-0.1327	-1.2013	0.5653	0.2599	0.4699	-0.0885	-0.1426	0.0709	-0.1378

2.2.3. Variance of the Prediction Error

The variance of prediction errors was found to be 0.0057

2.3.Calculation of Residual Sequence

In this step the inverse filter $A(z)$ has been modeled using equation given below;

$$\hat{e}(n) = s(n) + \sum_{j=1}^{p} \hat{a}_j s(n-j)$$

The original speech signal has been passed through it in order to obtain the residual signal from the LPC predicted speech. The waveform of the original speech signal and the corresponding waveform of the residual sequence $\hat{e}(n)$ have been plotted in Figure 2.

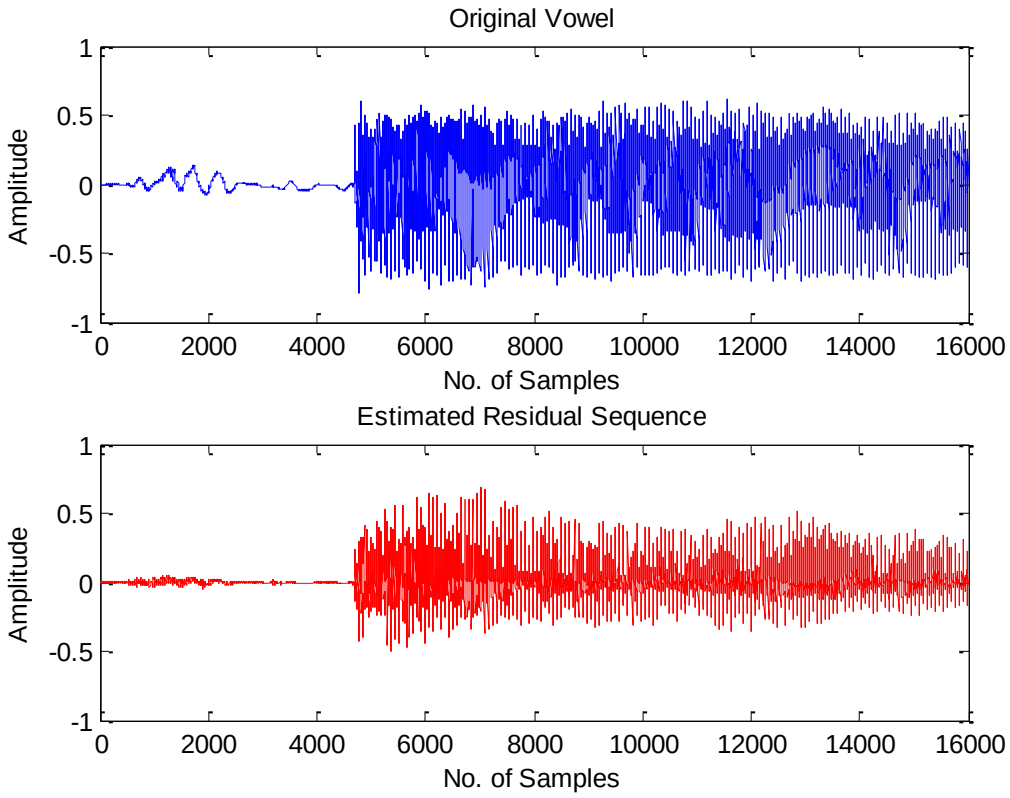

Figure 2: Original Vowel & Residual Sequence

2.4.Re-synthesizing of Speech

The speech can be synthesized by using equation as discussed above under the caption of 'Estimating the LPC Model Parameters'. Replacing the a_j with the estimated $\hat{a}_j$ computed in 2nd step and passing the residual sequence computed in 3rd step through this all-pole filter will result in a re-synthesize speech signal shown in the Figure 3.

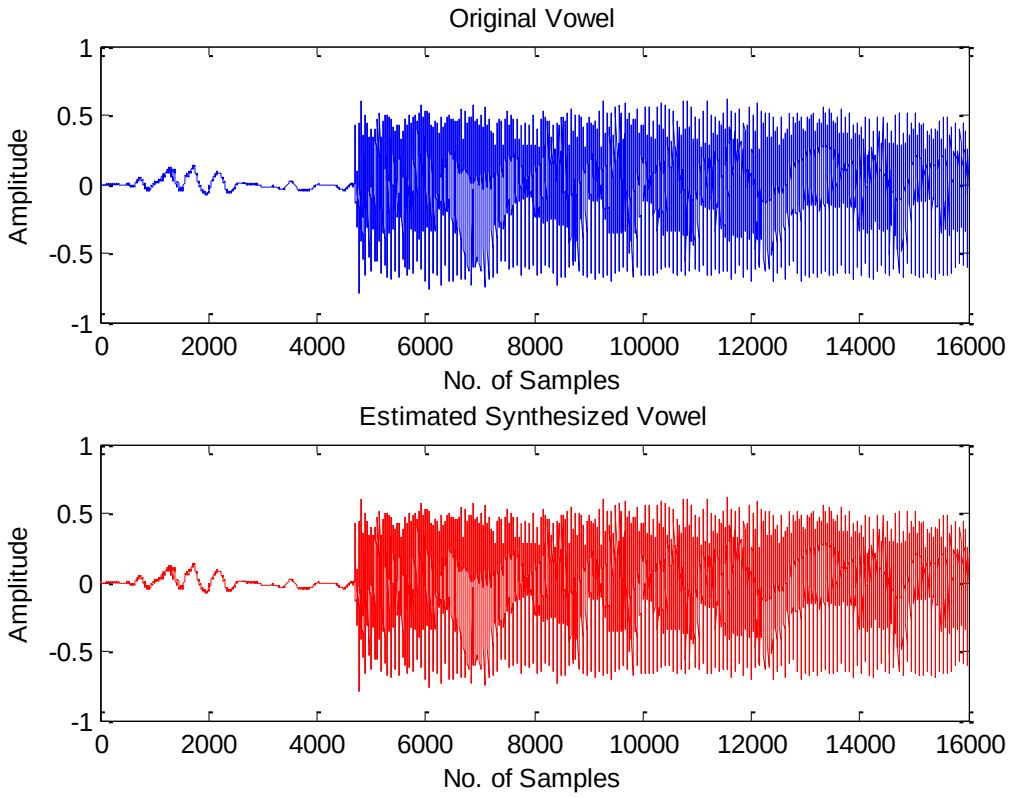

Figure 3: Original Vowel vs. Estimated Synthesized Vowel

2.4.1. Re-synthesized File

Please see the attached zip file for a wav file named;

MyVowel Resynthesized.wav

2.5.Comments

The vocal tract forms the tube, which is characterized by its resonances, which are called 'formants'. LPC analyzes the speech signal by estimating the formants, as these formants are sufficient enough to characterize the speech. The process of removing the formants effects is called inverse filtering, and the remaining signal after the subtraction of the filtered modeled signal is termed to be as residuals. LPC synthesizes the speech signal by reversing the process: using the estimated parameters from the speech sound and the residual to create a source signal, use the formants to create a filter (which represents the tube), and run the source through the filter, resulting in synthesized speech which is close approximation of the original speech [2].

It can be seen clearly from Figure 3 that two waveforms apparently are very much the same. From a human ear's perspective, the difference is in-distinguishable.

3. Task 2 - Non-Stationary Speech Signals: *Modeling, Analysis and Synthesis*

3.1. Generating a Non-Stationary Speech Signal File

A non-stationary speech signal file has been generated and provided. Please see the attached zip file for the file named;

Mysentence.wav

3.2. Block-Based Speech Analysis

The speech signal has been divided into blocks of duration 20ms each in order to do block-based speech analysis of the signal. The LPC analysis is applied to each block, and estimated parameters are stored in a 2D matrix $a(j = 1, \dots, p; i = 1, \dots, TotalBlocks)$, where j is the model order, $p = 10$, and i is the block number.

3.2.1. Total Number of Samples in the Speech Signal

As the duration of each speech frame is 10s and sampling frequency f_s is 8 kHz. So the numbers of samples contained in the frame are;

$$L = Duration\ of\ each\ frame\ \times Sampling\ frequency = 10s\ \times 8kHz = 80{,}000$$

So the total numbers of samples in a speech signal are 80,000.

3.2.2. Total Number of Blocks

The total numbers of blocks in the speech signal are;

$$Total\ Blocks = \frac{Total\ Recording\ Duration}{Time\ Duration\ of\ Each\ Frame} = \frac{10s}{20ms} = 500\ Blocks$$

3.2.3. Sampling Rate

The sampling rate of speech signal is;

$$f_s = 8kHz$$

3.3. Block-Based Estimation of Residual Sequence

In this step residual sequence within each block has been computed. Since the speech signals are not zero outside each block, as for the case of stationary, continuity must be taken under consideration.

The residual speech signal has been observed later on by zooming it, and periodicity is quite evident. The estimation of the period can be automatically achieved if we just introduced a threshold level, and then count the number of times it has been crossed.

3.3.1. Residual Sequence Plot

Figure 4 shows the plot of entire residual sequence $\hat{e}(n)$.

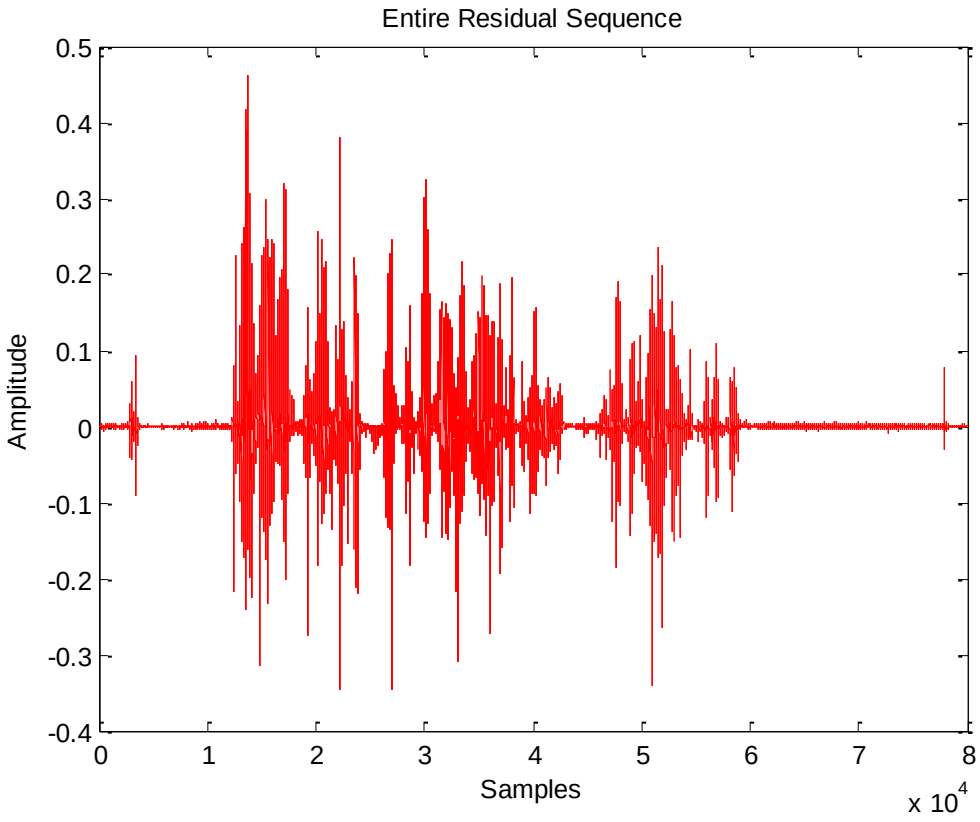

Figure 4: Plot of Entire Residual Sequence

3.3.2. Plotting Two Blocks of Speech & Residual Sequence Each

Figure 5 shows plots of two blocks (40ms) of speech signal and its corresponding residual sequence.

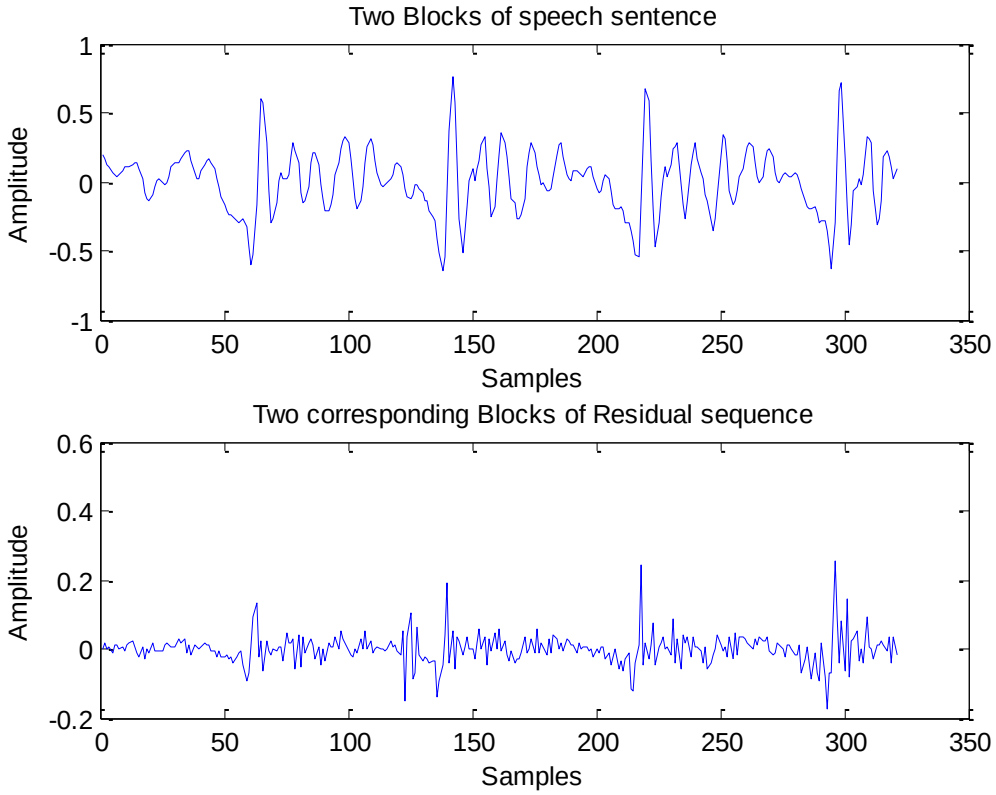

Figure 5: Speech Signal vs. Residual Sequence

3.3.3. Comparison between Speech Signal & Residual Sequence

If we compare the plots shown in Figure4 and 5, it can be clearly observed that pitch period or the fundamental period is easier to observe from residual sequence.

3.3.4. Number of Samples in One Pitch Period

There are 77 samples in one pitch period which has been counted manually.

3.4.Block-Based Speech Re-synthesis

3.4.1. Plot of Original Speech $s(n)$, Re-synthesized Speech $\hat{s}(n)$, and the Residual Sequence $\hat{e}(n)$

Figure 6 shows the plot of the original, re-synthesized and the residual sequence. It can be observed that re-synthesized speech is quite the same as original one except some very minor errors at the edges of each frame.

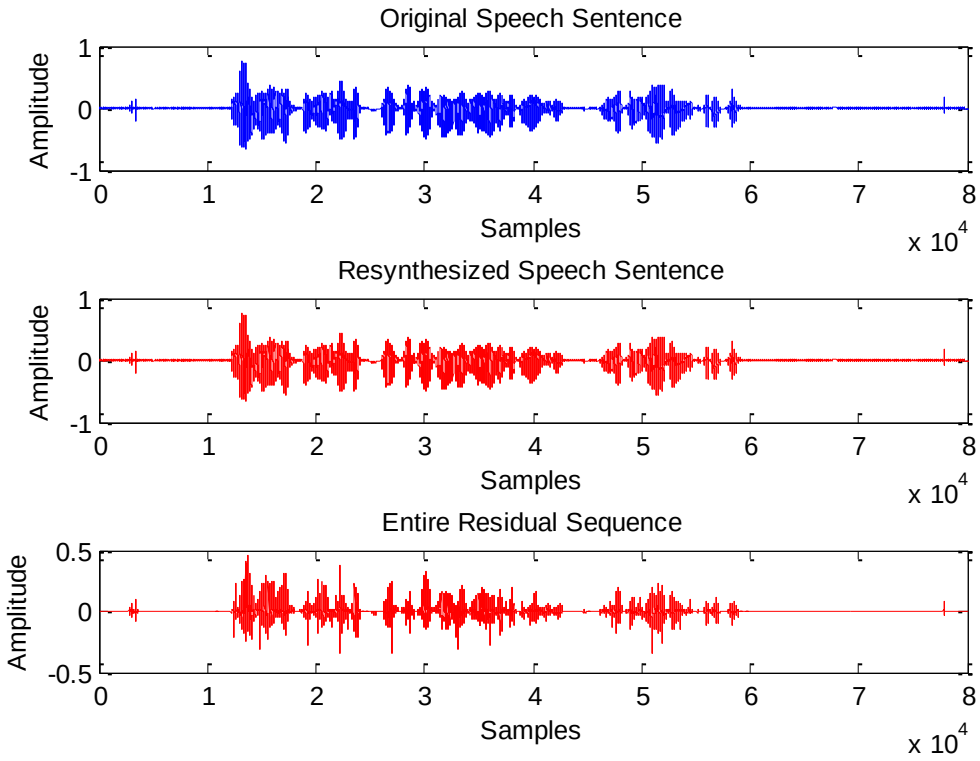

Figure 6: Plot of Original & Re-synthesized Speech and Entire Residual Sequence

3.4.2. Comparison between Original & Re-synthesized Speeches

Original and re-synthesized speech files have been listened to and no difference was found. The speech was same in both the files with almost the same loudness.

3.4.3. Re-synthesized Speech File

Please see the attached zip file for the file named;

Mysentence Resynthesized.wav

4. Task 3 – Re-synthesizing Speech Using 15 Most Significant Residuals/Blocks as the Excitation

4.1. Plot of Original Speech $s(n)$, Re-synthesized Speech $\hat{s}(n)$, and the Modified Residual Sequence $\hat{e}(n)$

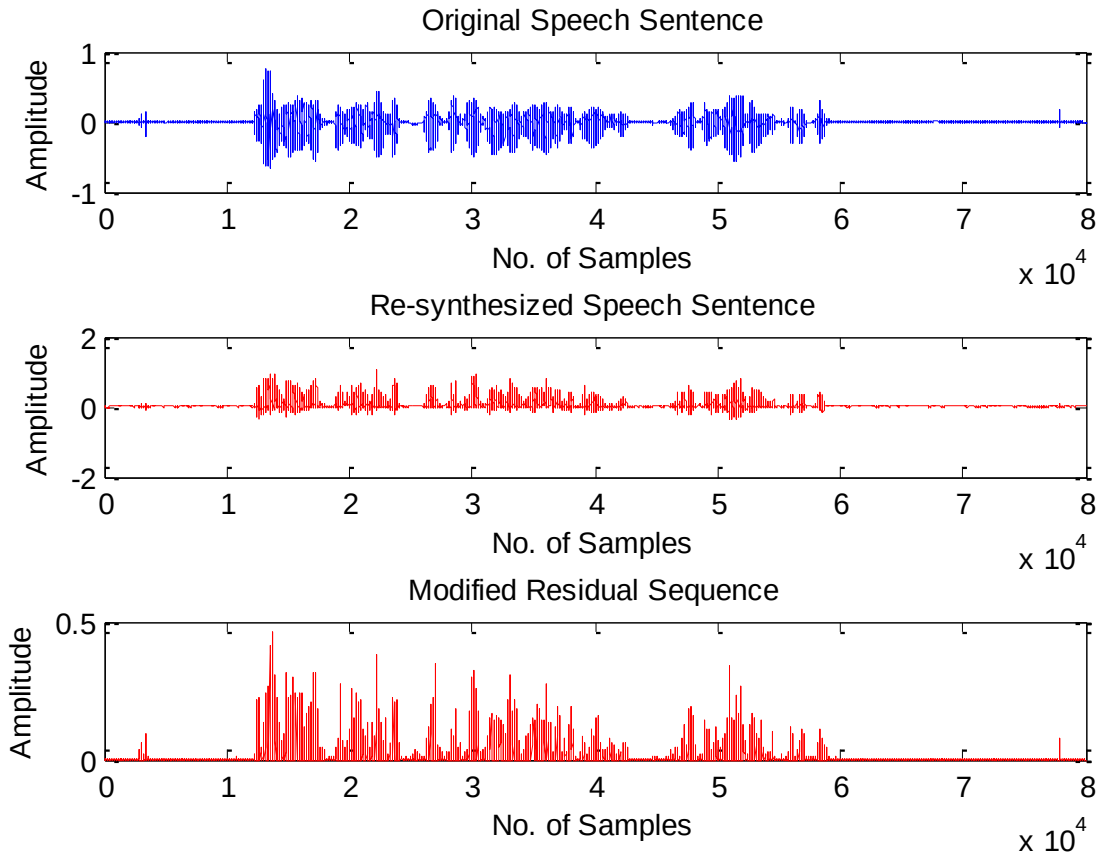

Figure 7: Plot of Original & Re-synthesized Speech Sentence and Modified Residual Sequence

4.2. Comparison between Original & Re-synthesized Speech

Following observations were made after comparing the original speech and the re-synthesized speech sentence generated using 15 most excited samples;

- The voice is definitely a machine generated voice without any emotions or feelings in it.
- Pitch has changed.
- Quality of re-synthesized speech can be improved by increasing the number of prominent or excited residual samples.

4.3. Re-synthesized Speech Sentence

Please see the attached zip file for the re-synthesized speech sentence named as;

Mysentence Re-synthesized.wav

5. Task 4 – Estimating Pitch Periods by using a Cepstrum-Based Method

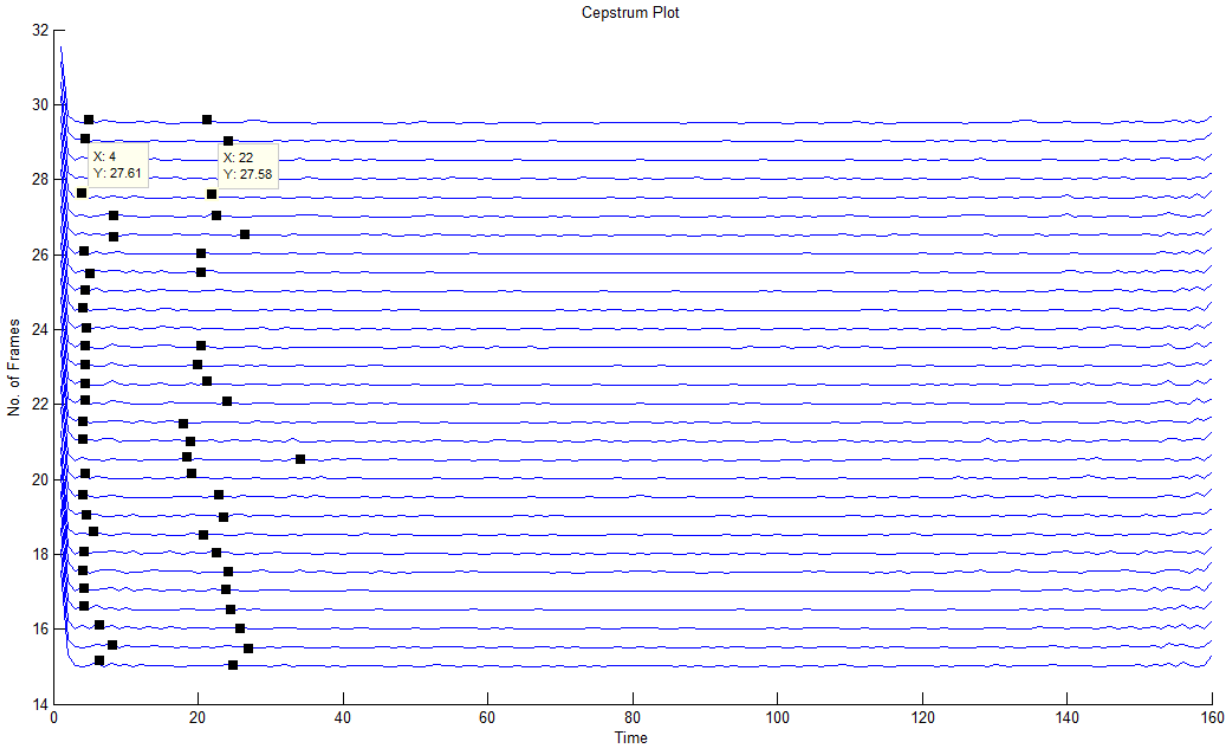

Figure 8: Cepstrum Plot

Figure 7 shows the Cepstrum plot. Black dots indicate the pitch period. The difference of samples between the two peaks is 18 samples so the pitch period is 2.25msec and pitch frequency is 444Hz, as shown in the Figure 8.

6. Task 5 – Discussions

The purpose of this lab was to model and simulate the human speech production system. Initially vocal tract was modeled with all pole filter. Then speech was generated by passing the residual sequence through this filter. As for the residual or excitation sequence, it can be modeled as an impulse train or white noise depending on whether the speech is voice or unvoiced respectively. Summarizing all the tasks in the given lab, in task 1 single tone (monotonic) sound was generated. Considering this sound stationary, LPC analysis was done on it. After that calculated residual sequence was passed through the designed filter for re-synthesizing the sound. **It has been observed that the re-synthesized and the original sound is almost the same.** In task 2 a non stationary speech signal was recorded and LPC analysis was done on it. In order to make the speech signal stationary, the entire speech was divided into number of frames of small duration and block-based processing technique was implied. Filter was modeled and excitation sequence was calculated for each block. As described previously in the report, samples from the previous block were taken into account in order to cater the continuity in the re-synthesized speech. **Again the re-synthesized speech and the original speech was almost the same.** In task 3, while re-synthesizing the speech, model of the residual sequence was implied rather than the original one. **Only 15 prominent residual samples in each block were considered and it was observed that the quality of speech is very bad.** With increasing the number of prominent samples the quality of the speech gets better and vice versa. In task 4, we explored some of the techniques for automatically estimating the pitch period. The technique we used was the Cepstrum Method. Cepstrum method is a technique to estimate the pitch period using non-linear transformation. It is used to separate the energy resulting from vocal cord vibration from the "distorted" signal formed by the rest of the vocal tract. The peak in the Cepstrum graph indicates the presence of pitch or the voiced part. **The main idea behind Cepstrum method is to take inverse Fourier transform of log-magnitude Fourier spectrum**. The reason for this is if we take the Fourier transform of a periodic signal again, it will give peaks corresponding to the peaks in the frequency, thus we can isolate the fundamental period. It can also be interpreted as a de-convolution process. If the input signal is produced by a train of impulses convolved with a filter they are going to be multiplied in the frequency domain, then applying log would transform the multiplication in addition. And applying IFFT again would de-convolve the original signal obtaining the fundamental frequency.

References

[1] Irene Y. H. Gu, Laboratory Exercise 1: *Speech Modeling, Analysis, Synthesis and Compression,* Chalmers University, 2009

[2] http://en.wikipedia.org/wiki/Linear_predictive_coding

Appendix – MATLAB Code

```
close all;
clear all;
clc;
%% TASK 1

% Step 1.1
fs=8000;                            %sampling freq
ts=2;                               %sound duration
CHN=1;                              %num of channel(mono type)
N=2*fs;                             %sound duration
sp=wavrecord(N,fs,CHN);             %recording
%figure;plot(sp);
%wavplay(sp,fs);                    %listening
%wavwrite(sp,fs,'Myvowel');         %Recorded sound written in file
sp=wavread('Myvowel.wav');          %read again
%wavplay(sp,fs);                    %listening again
figure;plot(sp);                    %sound waveform
save Myvowel.mat

% step 1.2
spb=sp(10000:10800);                %picking 100ms segment(L=800samples)
p=10;                               %order of predictor
[coeff,var]=lpc(spb,p);             %lpc applied
coeff(2:end);                       %filter coefficients
var;                                %variance(0.0057)
%figure;plot([coeff,var])           %plotting after lpc

% step 1.3
A=coeff(1:end);
ehat=filter(A,1,sp);                %residual sequence
figure;
subplot(2,1,1);plot(sp,'b');title('Original Vowel');
subplot(2,1,2);plot(ehat,'r');title('Estimated Residual Sequence');

% step 1.4
shat=filter(1,A,ehat);              %resynthezied speech
figure;
subplot(2,1,1);plot(sp,'b');title('Original Vowel');
subplot(2,1,2);plot(shat,'r');title('Estimated Synthesized Vowel');
shat_normalized = shat/max(abs(shat));
%wavwrite(shat_normalized,fs,16,'MyVowel Resynthesized');%saving resynthezied tone
%% TASK 2

% step 2.1
fs=8000;                            %sampling freq
ts=2;                               %sound duration
CHN=1;                              %num of channel
N=10*fs;                            %sound duration
sn=wavrecord(N,fs,CHN);             %recording
%figure;plot(sn);
%wavplay(sn,fs);                    %listening
%wavwrite(sn,fs,'Mysentence');      %Recorded sound written in file
sn=wavread('Mysentence.wav');       %read again
%wavplay(sn,fs);                    %listening again
figure;plot(sn);                    %sound waveform

% step 2.2
p=10;                               %order of predictor
tb=20e-3;                           %20ms block
LS=length(sn);                      %length of speech signal(80,000samples)
L=160;                              %number os samples in 20ms block
Blocks=LS/L;                        %Blocks of 20ms(total=500 blocks)
                                    %dividing the sentence in 500 blocks
snf=[];
for i=1:Blocks
```

```
snb=sn((i-1)*L+1:i*L)';             %sentence divided in blocks of 20ms(160 samples
in each block)
snf=[snf;snb];
end

lpccoeff=[];                        %applying lpc on each block
for i=1:Blocks
    coeff=lpc(snf(i,:),p)           %calculting the parameters for each block
    coeff=coeff(1:end);
    lpccoeff=[lpccoeff;coeff];
end
coefficients=size(lpccoeff);

% Step 2.3
ehat=zeros(1,length(sn));           %Initializing the residual sequence with zeros
for i=1:coefficients(1)
    ehat((i-1)*L+1:i*L)=filter(lpccoeff(i,:),1,sn((i-1)*L+1:i*L));  % Calculating
residual sequence
end
figure;
plot(ehat,'r');title('Entire Residual');%Entire residual sequence
figure;
subplot(2,1,1);plot(sn(13000:13320),'b');title('Two Blocks of speech sentence');
subplot(2,1,2);plot(ehat(13000:13320),'b');title('Two corresponding Blocks of
Residual sequence');
%% Step 2.4
shat_sn=zeros(1,length(ehat));      %initializing resynthesized signal
for j=1:coefficients(1)
    shat_sn((j-1)*L+1:j*L)=filter(1,lpccoeff(j,:),ehat((j-1)*L+1:j*L));  %
Calculating re-synthesized speech
    ehat_block(j,:)=ehat((j-1)*L+1:j*L);
end
figure;
subplot(3,1,1);plot(sn,'b');title('Original speech Sentence');
subplot(3,1,2);plot(shat_sn,'r');title('Resynthesized speech Sentence');
subplot(3,1,3);plot(ehat,'r');title('Entire residual sequence');
%wavwrite(shat_sn,fs,'Mysentence Resynthesized');
%%wavplay(shat_sn,fs);

%% TASK 3

% Step 3.1
ehat =ehat ;% residual sequence from task step 2.3
mod_ehat=[];
ehat_block=abs(ehat_block);
[temp,MSR]=sort(ehat_block,2,'descend');     % Estimating the 15 most significant
excitation pulses
MSR=MSR(:,1:15);

% Step 3.2
mod_ehat_block=zeros(size(ehat_block));      %initializing
for ii=1:coefficients(1)
    mod_ehat_block(ii,MSR(ii,:))=ehat_block(ii,MSR(ii,:));    % Forming mod_ehat
that is modified excitation
    mod_ehat=[mod_ehat mod_ehat_block(ii,:)];
end

modshat_sn=zeros(1,length(mod_ehat));       %initializing
for jj=1:coefficients(1)
    modshat_sn((jj-1)*L+1:jj*L)=filter(1,lpccoeff(jj,:),mod_ehat((jj-1)*L+1:jj*L));
% Calculating re-synthesized speech using modified excitation
    ehat_block(j,:)=ehat((j-1)*L+1:j*L);
end
%wavplay(modshat_sn,fs);
%wavwrite(modshat_sn,fs,'Mysentence Re-synthesized');
figure;
subplot(3,1,1);plot(sn,'b');title('Original speech Sentence');
```

```
subplot(3,1,2);plot(modshat_sn,'r');title('Resynthesized speech Sentence using
modified excitation');
subplot(3,1,3);plot(mod_ehat,'r');title('Modified residual sequence');


%% TASK 4

fs=8000;%sampling freq
ts=2;%sound duration
CHN=1;%num of channel
N=10*fs;%sound duration
sn=wavread('Mysentence.wav');%read again
x = double(reshape(sn,160,length(sn)/160));
x = x';
figure;
hold on
for i=30:59
    x_h = x(i,:).*hamming(160)';%To reduce gibbs effect
    y = [x_h zeros(1,1600)];%padding zeros more than number of samples so its easy
to find peaks
    c = abs(fft(y,160));%Implementing equation 8
    c = log10(c);
    c = ifft(c);
    c = abs(c);
    c = i*0.5 + c;
    plot(c);
end
```