

Southern University of Science and Technology

Speech Signal Processing

# Lab 6 Report

11510478 郭锦岳

Question 1

In question 1, the task is to resynthesize the signal using autocorrelation method with linear prediction coefficients. This need frame by frame calculation of the coefficients and error signal. Then, according the linear assumption of the vocal system, and the assumption of autocorrelation method that s(m) is only non-zero in 0<m<L-1.

Code:

function [ error,synth ] = lab6q1( input\_file, frame\_size, frame\_shift, order )

%LAB6Q1 use LPC

signal = audioread(input\_file);

error = zeros(1,length(signal));

synth = zeros(1,length(signal));

for n\_hat = 1:frame\_shift:length(signal)-frame\_size

frame = signal(n\_hat:n\_hat+frame\_size-1);

frame = frame.\*hamming(frame\_size);% windowing

%calculate a\_k

a = lpc(frame,order);% p-th order lpc

a = a(2:end);

%calulate e(n)

error\_frame = filter([1,a],1,frame);% frame error

error(n\_hat:n\_hat+frame\_size-1)=error(n\_hat:n\_hat+frame\_size-1)+error\_frame';% adding to total error

synth\_frame = filter(1,[1,a],error\_frame);

synth(n\_hat:n\_hat+frame\_size-1)=synth(n\_hat:n\_hat+frame\_size-1)+synth\_frame';% adding to total signal

end

subplot(3,1,1);plot(signal);title('original signal s[n]');

subplot(3,1,2);plot(error);title('error signal e[n]');

subplot(3,1,3);plot(synth);title('resynthesized signal s\_{hat}[n]');

end

After generating the frame error and resynthesized signal, we need to use overlap addition to add it on the whole error and resynthesized signal.

Another point to mention is the hamming window of the frame signal. This is to reduce the side effect of the error signal, since the error is larger near m = 0 and m = L.



Figure 1, frame size = 320, frame shift = 80, p = 12

The result shows a pretty good similarity between the resynthesized signal and the original one. When listening to the resynthesized one, the words can be clearly recognized, and the tones and voices are remained. The only difference is a little bit saw could be heard.

An interesting fact is that the error signal seems to have more similarity, and when listening to the error signal, it is even clearer than resynthesized signal.



Figure frame size = 320, frame shift = 80, p = 128

But when p increases, the error signal shows less similarity, and the resynthesized signal becomes better.

Question 2

In question2, the task is to compare the spectrum similarity of autocorrelation method. According to page 7 in slide 7, H(z) is the reconstruction of spectrum without excitation signal. Therefore, it should show the same formants with the spectrum obtained by DFT.

Code:

function [ ] = lab6q2( input\_file, frame\_start, frame\_size, order )

%LAB6Q2

signal = audioread(input\_file);

frame = signal(frame\_start:frame\_start+frame\_size-1);

%calculate a\_k

a = lpc(frame,order);% p-th order lpc

a = a(2:end);

%calulate e(n)

error\_frame = filter([1,-a],1,frame);% frame error

spec\_frame = 10\*log(abs(fft(frame,1024))); % frame spectrum

spec\_frame = spec\_frame(1:end/2); % get a half

lpc\_frame = 10\*log(abs(freqz(1,[1,a],1024)));

error\_spec = 10\*log(abs(fft(error\_frame,1024)));

error\_spec = error\_spec(1:end/2); % adjusting axis



Figure 3 frame = 6000, frame size = 640, p = 12

In the third picture, the blue line is the spectrum of fft, and the orange line is the spectrum of lpc when p = 12. We can see that the formants are in the same position, showing that both method can obtain the same correct answer.

Since the order of the lpc method directly determines the number of poles, which is the same as the number of peaks, increasing the order can have a better result. As shown below, the curves has more same values when the order comes to 64.



Figure frame = 6000, frame size = 640, p = 64