Speech Synthesis Coursework

Kirishoban Sanjeevvijay

November 9, 2023

Contents

Abstract

This report explores the process of synthesising vowels using the source-filter model of speech production. A major aim is to estimate the formant structure of selected male and female vowels and generate synthesised vowels by applying a periodic impulse train through the LPC filter obtained from the estimation.

1 Methodology

To initiate the analysis, I commenced by downloading a dataset comprising male and female vowel phoneme samples, focusing on the ’heed’ sound in .wav format, sourced from the designated resource.

Subsequently, I isolated quasi-stationary segments from the selected vowels (producing the ’ee’ sound), each segment 100ms in duration for male and female. These segments would serve as the basis for the model estimation and synthesis portions of the assignment.

In carrying out this task, I employed Matlab, using in-built functions for tasks such as Linear Predictive Coding (LPC), audio input/output, and more from those recommended in the assignment guidelines.[2]

1.1 Model Estimation

I used the lpc Matlab function for auto-regressive (AR) modelling to estimate the LPC coefficients that provide a least-squares error fit to the speech waveform. I then plotted the frequency response of the LPC filter that can be seen in Figure 1 and Figure 3 for the male and female voices, respectively.

To plot the amplitude spectrum of the speech segment in dB, I had to first apply a Fourier transform to convert from a time-domain signal into the frequency domain, then, take the magnitude using the ’abs’ function to take the magnitude of the frequency components (since the FFT result is typically complex), before applying a base-10 logarithmic function and multiplying the result by 20 to convert from linear magnitude to decibels (dB) [1].

Using the aforementioned figures, I estimated the first three formant frequencies of the vowel sounds (as the lowest 3 peaks in the LPC filter’s frequency response correspond to the first 3 formant frequencies). I chose the model orders for the male and female vowel sounds such that the LPC filter would provide a reasonable fit to the formant structure as can be seen in Figures 2 and 4. The chosen male model order (Figure 2) was 18 and for female (Figure 4), was 30.

1.2 Synthesis

Continuing to use Matlab, I generated a periodic impulse train with the same fundamental frequency as the original vowel segment, of 1 second in length for both the male and female vowels. This impulse train served as the simulated periodic source for voiced sounds in the source-filter model of speech production.

The LPC filter, derived from the coefficient estimation, was used to filter the impulse train, resulting in speech waveforms that resembled the original ’ee’ vowel sounds. I then proceeded with testing different AR model orders and segment lengths. These variations allowed me to explore the impact of different modeling choices on the synthesised speech.

2 Results and Discussion

Mean Fundamental Frequency (F0) for Male: 287.00 Hz
Mean Fundamental Frequency (F0) for Female: 308.87 Hz

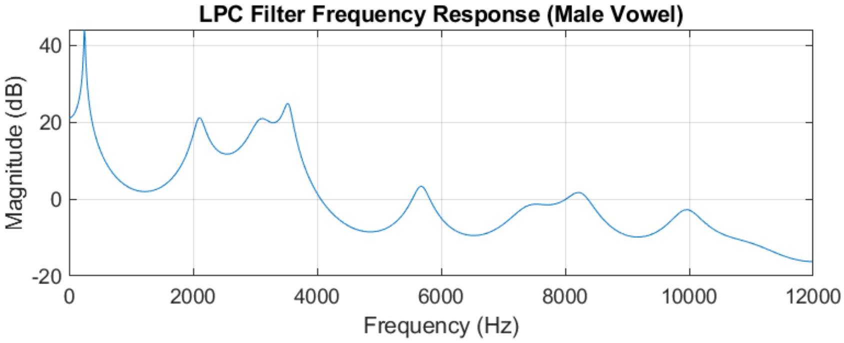

Figure 1: The LPC filter frequency response for the male vowel sound ’ee’ showing magnitude (dB) against frequency (Hz).

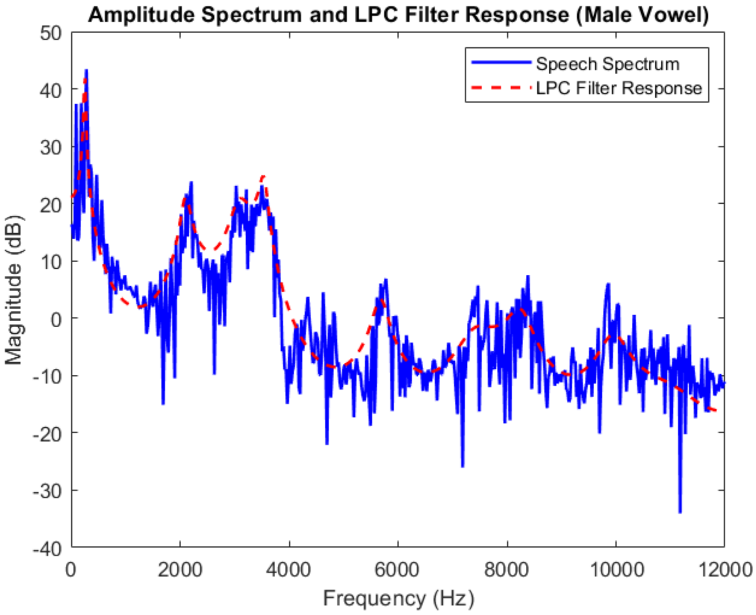

Figure 2: Plot showing the amplitude spectrum of the male speech segment ’ee’ (in dB) and the corresponding LPC filter response as a function of frequency (Hz).

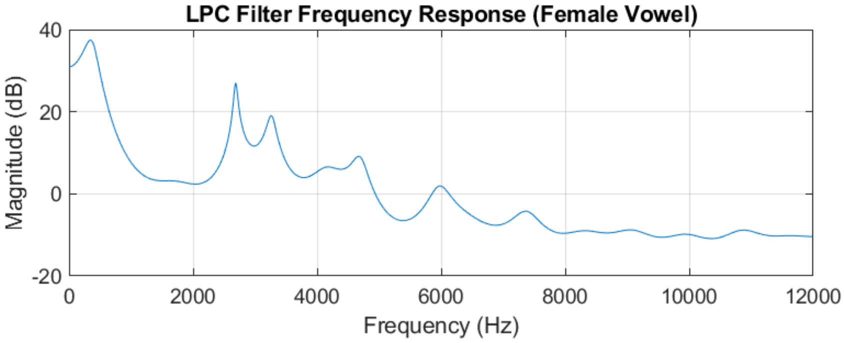

Figure 3: The LPC filter frequency response for the female vowel sound ’ee’ showing magnitude (dB) against frequency (Hz).

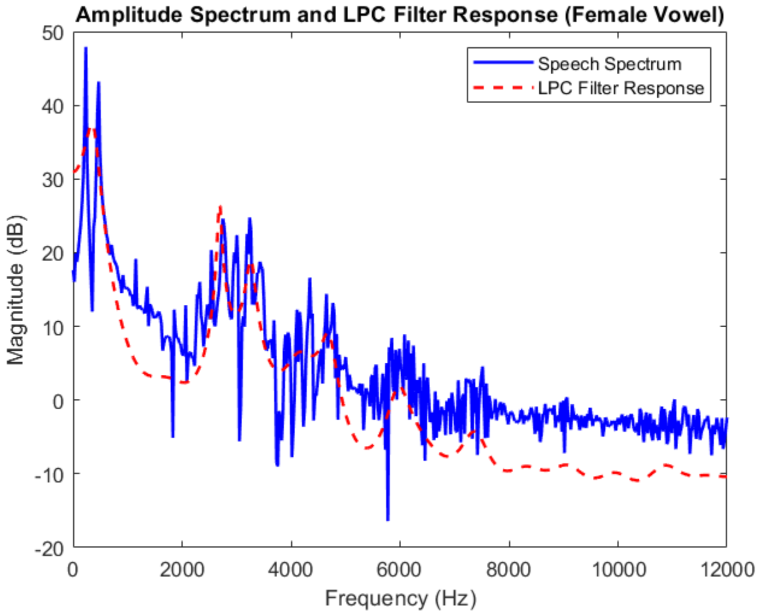

Figure 4: Plot showing the amplitude spectrum of the female speech segment 'ee' (in dB) and the corresponding LPC filter response as a function of frequency (inn Hz).

2.1 Formant Analysis

For the male vowel sound the first three formant frequency estimations are: 258 Hz, 2090 Hz, 3123 Hz. Likewise, the female vowel formant frequency estimations are: 352 Hz, 2677 Hz, 3264 Hz.

The AR model order was varied from 5 to 30 for both the male and female vowels, increasing by increments of 5 each time. Despite such broad variations in AR orders, there was little to no audible difference that could be heard with these variations; the only differences clearly perceivable were those between the higher-pitched female and lower-pitched male vowel sounds.

Although this observation was unexpected, it is certainly interesting that clear vowel sounds were synthesised for both the male and female voices, even at low model orders. This is particularly perplexing as at lower model orders, the LPC filter did not aptly fit the formant structures of the amplitude spectra of the original speech segments.

For each of these model orders tested, the segment length was also changed, going from 70ms to 100ms and lastly, 200ms. The change in segment length from 70ms to 100ms showed little audible difference but the difference from 100ms to 200ms was very apparent. The signal produced with a segment length of 200ms was much clearer and slightly less 'robotic-sounding' than both the 70ms and 100ms segment length audios. This difference was more pronounced in the male audio synthesis than in the female audio synthesis.

2.2 Subjective Assessment

Despite the limited audible variation in vowel speech signals produced, the quality of the synthesised speech cannot be understated. The synthesised audio files for both the male and female voices are clear and audibly recognisable as forming the 'ee' vowel sound (especially when utilising a greater segment length); this is a surprising outcome using even low AR model orders. The main differentiating factor of the synthesised sounds from th original 'ee' sound files is that they sound rather 'robotic' in nature. Drawing from the results of this study, the robotic timbre may become less noticeable if the segment length is increased further using a larger original sample audio file.

2.3 Summary of Main Findings

2.3.1 LPC Coefficient Estimation

Model orders for LPC filter estimation were chosen (18 for male, 30 for female) to ensure a reasonable fit to the formant structure, as evidenced by the frequency response plots (Figures 2 and 4). The first three formant frequencies for male and female vowels were estimated from the LPC filter response.

2.3.2 Speech Synthesis

The synthesis portion involved generating a periodic impulse train and filtering it using the LPC filter. Surprisingly, even at low model orders, clear and recognisable vowel sounds were synthesised. Experimentation with different AR model orders revealed minimal audible differences, however, increasing the segment length was found to improve the quality of the synthesised audio for both the male and female voices, for all AR model orders that were tested. A longer segment length likely allowed for a more accurate representation of pitch variations, formants, and other acoustic features, leading to higher sound quality.

2.3.3 Formant Analysis

The first three formant frequencies for male and female vowels were successfully estimated, as can be seen in section 2.1.

2.3.4 Subjective Assessment

The quality of the synthesised speech was clear and recognisable, improving in said quality as segment length was increased. This continued to be true even with low model orders, although a slight robotic nature was still observed.

2.3.5 Mean Fundamental Frequency (F0)

The male 'ee' vowel exhibited a mean F0 of 287.00 Hz, while the female 'ee' vowel had a mean F0 of 308.87 Hz as mentioned earlier in the results.

2.4 Significance

The ability to synthesise clear vowel sounds with low model orders challenges conventional expectations. Despite variations in AR model orders, the fundamental characteristics of the synthesised vowels remained consistent.

Increasing the segment length improved the quality of the vowel sounds produced further, showing segment length to be a valuable parameter in refining speech synthesis techniques by contributing to enhanced naturalness and clarity in synthesised speech.

2.5 Limitations

This study observed little audible differences across the various AR model orders that were used.

2.6 Future Research Directions

In light of this project's outcomes, several intriguing avenues for future research present themselves. Firstly, expanding the investigation to encompass a broader spectrum of voiced speech sounds (different vowel sounds) would make for an interesting comparison.

Additionally, a more extensive range of AR model orders could help identify thresholds for perceptible differences in synthesised speech, possibly increasing the naturalness of the produced sound.

Comparing the efficacy of alternative modelling methods for voiced speech, beyond only Linear Predictive Coding (LPC), stands as another promising research direction.

Furthermore, delving into how individual differences among various male and female speakers producing the same vowel sounds may influence synthesis outcomes could provide a deeper understanding of the intricacies involved in speech production.

3 Conclusion

To conclude, this study sheds light on the effective synthesis of vowel sounds using LPC, even with relatively low model orders. The consistent quality across varying model orders prompts further exploration into the robustness of LPC in capturing formant structures.

An increase in segment length, however, was found to correlate with an improvement in the quality of the synthesised speech which warrants further exploration into a larger range of segment lengths.

4 Appendix

4.1 Matlab code

The following code was implemented to carry out this assignment in Matlab.[3]

```
% File reading
[y_male, fs_male] = audioread(”heed_m.wav”);
[y_female, fs_female] = audioread(”heed_f.wav”);

% 100ms vowel segments
vowel_segment_male = y_male(0.07 * fs_male : 0.17 * fs_male);
vowel_segment_female = y_female(0.07 * fs_female : 0.17 * fs_female);

% LPC modelling (male)
order = 18;
[A_male, G_male] = lpc(vowel_segment_male, order);

%LPC filter frequency response plot (male)
figure;
freqz(1, A_male, 1024, fs_male);
title('LPC Filter Frequency Response (Male Vowel)');
xlabel('Frequency (Hz)');
ylabel('Magnitude (dB)');

%Plot male vowel amplitude spectrum & LPC filter response
figure;
fft_size = 1024;
speech_spectrum_male = 20 * log10(abs(fft(vowel_segment_male, fft_size)));
f = linspace(0, fs_male/2, fft_size/2);

plot(f, speech_spectrum_male(1:fft_size/2), 'b', 'LineWidth', 1.5);
hold on;
lpc_spectrum_male = 20 * log10(abs(freqz(1, A_male, f, fs_male)));
plot(f, lpc_spectrum_male, 'r--', 'LineWidth', 1.5);

legend('Speech Spectrum', 'LPC Filter Response');
title('Amplitude Spectrum and LPC Filter Response (Male Vowel)');
xlabel('Frequency (Hz)');
ylabel('Magnitude (dB)');

% LPC modelling (female)
f_order = 30;
[A_female, G_female] = lpc(vowel_segment_female, f_order);

%LPC filter frequency response plot (female)
figure;
freqz(1, A_female, 1024, fs_female);
title('LPC Filter Frequency Response (Female Vowel)');
xlabel('Frequency (Hz)');
ylabel('Magnitude (dB)');

%Plot female vowel amplitude spectrum & LPC filter response
figure;
fft_size = 1024;
speech_spectrum_female = 20 * log10(abs(fft(vowel_segment_female, fft_size)));
f = linspace(0, fs_female/2, fft_size/2);

plot(f, speech_spectrum_female(1:fft_size/2), 'b', 'LineWidth', 1.5);
```

```
hold on;
lpc_spectrum_female = 20 * log10(abs(freqz(1, A_female, f, fs_female)));
plot(f, lpc_spectrum_female, 'r--', 'LineWidth', 1.5);

legend('Speech Spectrum', 'LPC Filter Response');
title('Amplitude Spectrum and LPC Filter Response (Female Vowel)');
xlabel('Frequency (Hz)');
ylabel('Magnitude (dB)');

%Pitch detection (autocorrelation)
% [f0_male, t_male] = pitch(vowel_segment_male, fs_male);
% [f0_female, t_female] = pitch(vowel_segment_female, fs_female);

% Autocorr-based pitch (male)
max_lag_male = round(fs_male / 80);
min_f0_male = 80; %Min pitch freq (Hz)
max_f0_male = 400; %Maximum expected pitch freq (Hz)

autocorr_male = xcorr(vowel_segment_male, max_lag_male);
autocorr_male = autocorr_male(max_lag_male+1:end); %only +ve lags

[~, locs_male] = findpeaks(autocorr_male, 'MinPeakDistance', round(fs_male / max_f0_male

pitch_periods_male = diff(locs_male);
f0_male = fs_male ./ pitch_periods_male;

% Filter out invalid pitch values based on expected range
valid_indices_male = f0_male >= min_f0_male & f0_male <= max_f0_male;
f0_male = f0_male(valid_indices_male);

t_male = (locs_male(valid_indices_male) - 1) / fs_male;

% Autocorr-based pitch (female)
max_lag_female = round(fs_female / 80);
min_f0_female = 80;
max_f0_female = 400;

autocorr_female = xcorr(vowel_segment_female, max_lag_female);
autocorr_female = autocorr_female(max_lag_female+1:end);

[~, locs_female] = findpeaks(autocorr_female, 'MinPeakDistance', round(fs_female / max_

pitch_periods_female = diff(locs_female);
f0_female = fs_female ./ pitch_periods_female;

% Filter out invalid pitch values
valid_indices_female = f0_female >= min_f0_female & f0_female <= max_f0_female;
f0_female = f0_female(valid_indices_female);

t_female = (locs_female(valid_indices_female) - 1) / fs_female;

%Mean fundamental freq (f0)
mean_f0_male = mean(f0_male);
mean_f0_female = mean(f0_female);

fprintf('Mean Fundamental Frequency (F0) for Male: %.2f Hz\n', mean_f0_male);
fprintf('Mean Fundamental Frequency (F0) for Female: %.2f Hz\n', mean_f0_female);
```

```
%Synthesis

%Generate periodic impulse train (male)
fundamental_frequency_synthesis_male = mean_f0_male; %using estimated f0 from original
duration_synthesis = 1;
impulse_train_male = impulse_train_generator(fundamental_frequency_synthesis_male, fs_ma

%Filter pulse train using LPC filter determined above
synthesised_speech_male = filter(1, A_male, impulse_train_male);

filename_male = sprintf('male_order%d_length%d.wav', order, 100);
audiowrite(filename_male, synthesised_speech_male, fs_male);
%sound(synthesised_speech_male, fs_male);

%Generate periodic impulse train (female)
fundamental_frequency_synthesis_female = mean_f0_female;
duration_synthesis = 1;
impulse_train_female = impulse_train_generator(fundamental_frequency_synthesis_female, f

synthesised_speech_female = filter(1, A_female, impulse_train_female);

filename_female = sprintf('female_order%d_length%d.wav', f_order, 100);
audiowrite(filename_female, synthesised_speech_female, fs_female);
%sound(synthesised_speech_female, fs_female);


function impulse_train = impulse_train_generator(frequency, fs, duration)
    %Generate a periodic impulse train
    % Calc. sample no.
    num_samples = round(duration * fs);

    % Gen. time vector
    t = (0:num_samples-1) / fs;

    % Gen. impulse train
    impulse_train = zeros(size(t));
    impulse_train(1:round(fs/frequency):end) = 1; %impulse every period
    impulse_train = impulse_train(1:num_samples);
end
```

References

[1] A. Rudiger ¨ G. Heinzel and Max-Planck-Institut fur ¨ Gravitationsphysik R. Schilling. "Spectrum and spectral density estimation by the Discrete Fourier transform (DFT), including a comprehensive list of window functions and some new flat-top windows." In: *Teilinstitut Hannover* 27.2 (2002), p. 10.

[2] Inc. The MathWorks. *LPC Analysis and Synthesis of Speech.* URL: `https://www.mathworks.com/help/dsp/ug/lpc-analysis-and-synthesis-of-speech.html`. (accessed: 09.11.2023).

[3] Inc. The MathWorks. *xcorr.* URL: `https://www.mathworks.com/help/matlab/ref/xcorr.html`. (accessed: 09.11.2023).