Voice Processing

Marchioro Thomas

Project 1 – Linear Prediction Filtering

1 Analysis-Synthesis based on linear prediction

In this part, I've completed the code from the file `lpc_as_toyou.m` to estimate the vocal tract filter $H(z)$ and the excitation $u_g[n]$ for each frame.

The filter $H(z)$ has the form

$$H(z) = \frac{A}{1 - \sum_{i=1}^{p} a_i z^{-i}} \tag{1}$$

and the coefficients a_i are estimated with the autocorrelation method using Levinson recursion, which I had to implement myself.

My implementation of the Levinson recursion is the function `my_levinson` which takes as input the autocorrelation vector `r` of a single frame and the order p of the polynomial $a_1 z^{-1} + \cdots + a_p z^{-p}$ and returns

- `a` the vector of coefficients formatted as $[1, -a_1, \ldots, -a_p]$;
- `e` the estimated error $E^{(p)}$ computed at the last iteration of Levinson recursion;
- `k` the reflection coefficients.

The reason I decided to format the coefficient in that way is that I can efficiently implement the filter using `filter(G, a, input)`, where `G` is the gain, computed as the square root of `e`.

1.1 Results in the time domain

For each frame, I estimate the excitation $u_g[n]$ by inverse filtering the windowed frame and I reconstruct the frame by direct filtering. An example of estimate excitation obtained from this method is shown in Figure 1.

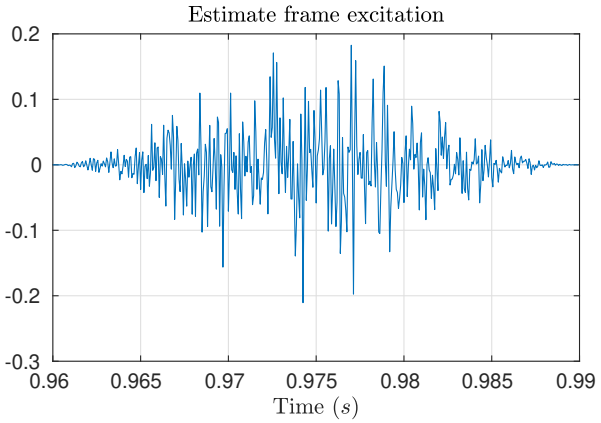

Figure 1: Example of excitation estimated from a single frame.

The reconstruction is accurate enough, as shown in Figure 2, and independent of the order p. This is due to the fact that simply reconstructing the signal in the frequency domain consists in

multiplying and dividing for a given function. In principle, this should lead to a perfect reconstruction, but the application of a filter might lead to the loss of part of the original signal information. Nonetheless, the difference can barely be seen when looking at the complete reconstruction and I couldn't perceive any difference while listening to the original and reconstructed tracks.

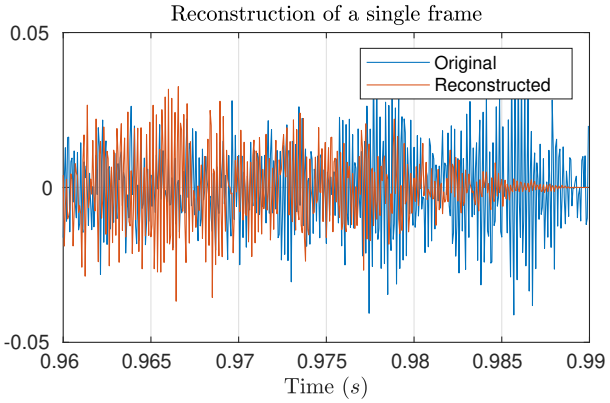

Figure 2: Comparison between an original frame and its reconstruction.

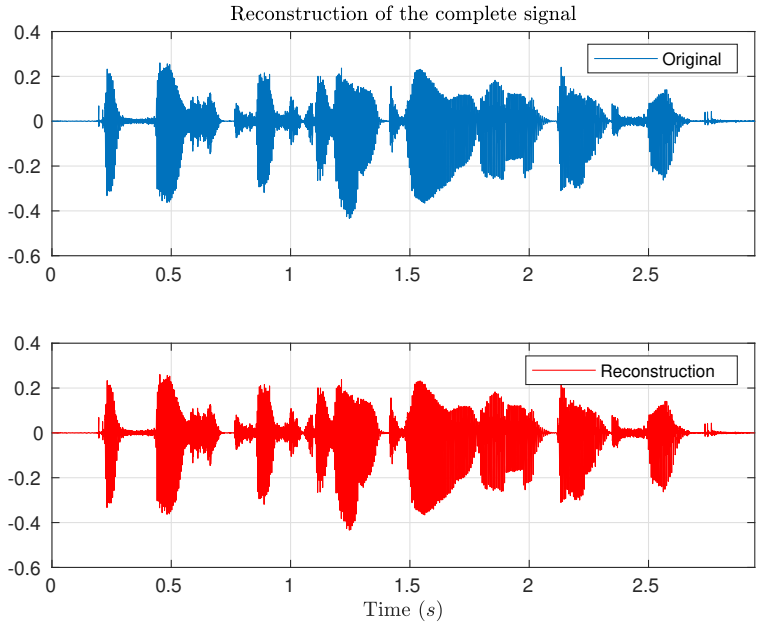

Figure 3: Comparison between the complete track and its reconstruction.

1.2 Analysis in the frequency domain

In the frequency domain, the excitation preserves the symmetry with respect to $F_s/2$, being itself a real signal, as well as the "noisy" component of the speech signal. The vocal tract filter, instead, contains the information of the main frequencies to be emphasized – even if the filter covers all the spectrum up to F_s. It can be seen from Figures 4 and 5 that the same peaks of the vocal tract filter are present in the reconstructed frame, but they are mapped in the range $[0, F_s/2]$. Increasing the order of the filter, the polynomial $1 - \sum_{i=1}^{p} a_i z^{-i}$ contains more poles, hence more frequencies are moved from the excitation to the filter. Therefore, when the order becomes too high, the filter starts "over-fitting" the frame and does not provide anymore a general representation of the vocal tract.

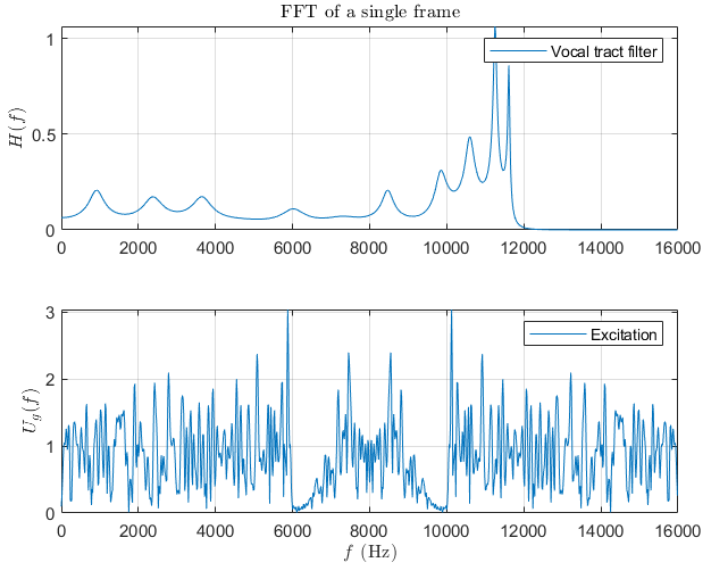

Figure 4: Frequency response of the filter $H(f)$ and FFT of the excitation for a single frame.

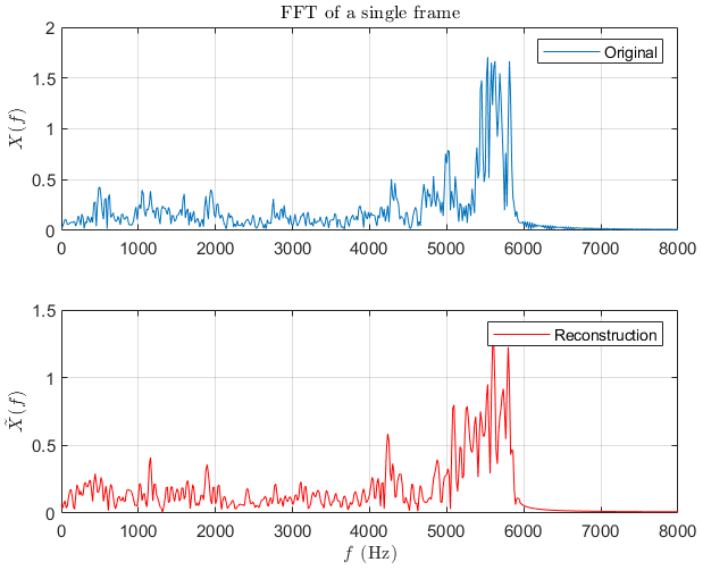

Figure 5: Comparison between the complete track and its reconstruction.

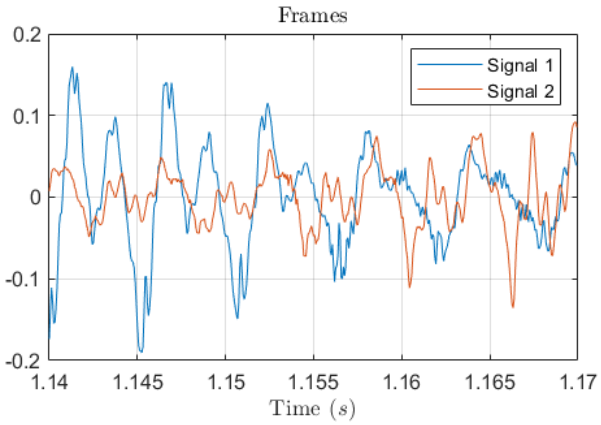

Figure 6: Example of two frames out of sync.

2 Changing the excitation

2.1 Synthetic excitation

In order to simulate a whispering voice, I changed the excitation with Gaussian white noise using the `randn` function, but the result was poor, regardless of filter order. Part of the reason is that the excitation is never completely random, even when whispering, it still has some structure. Thus, I tried to mix the noise with the original excitation and the results slightly improved, albeit more than a whisper it sounds like the voice of a movie villain. I used the same idea to simulate the robot voice, mixing a constant signal with the original excitation, which results in a "flattened" version of the original excitation.

2.2 Excitation estimated from another signal

I recorded myself while saying a random sentence and tried to give my voice to the track `H.22.16k.wav`. In order to do so, for each frame I estimated the vocal tract filter of the lady recorded in the track and used it to filter the excitation from my voice. The result was terrible in general, slightly better for low order of the filter (around $p = 12$). I obtained better result by saying the same sentence from the track, i.e. "the fish twisted and turned on the bent hook". Nonetheless, the voice of the constructed signal still appears distorted and unnatural, the main reason being that – due to my different phase and accent while speaking – the frame-by-frame synthesis applies filter of voiced speech on unvoiced excitation and vice versa. An example of two frames out of sync is shown in Figure 6.

3 Modifying the poles of the filter

The poles of the filter $H(z)$ correspond to frequencies that are emphasized by the vocal tract. In particular, the poles that are closer to the unit circle correspond to the peak frequencies, which in case of voiced speech are the formants. Again, this depends on the order of the filter: if the order of is too high, some frequencies that are not formants and are only due to the speech noise will appear, whereas if the order is too low some formants might be lost. The frequencies correspondent to the most important poles for a filter of order $p = 14$ are shown in Figure 7. Another important characteristic of the filter $H(z)$ is that, since the coefficients of the polynomial at the denominator are real, the non-real poles appear in pairs of complex conjugates. While modifying these poles to change the voice pitch, one must preserve this property of the poles. That is, if one pole is multiplied by a unitary complex number $e^{j2\pi f_{\text{shift}}/F_s}$, its conjugate must be multiplied by $e^{-j2\pi f_{\text{shift}}/F_s}$. Since each pole p_i can be expressed in term of the emphasized frequency f_i as $p_i = |p_i|e^{-j2\pi f_i/F_s}$, then multiplying it by $e^{j2\pi f_{\text{shift}}/F_s}$ means shifting the emphasized frequency to $f_i - f_{\text{shift}}$. With this in mind, I could modify a voice changing it to a higher or lower pitch, depending on the value of

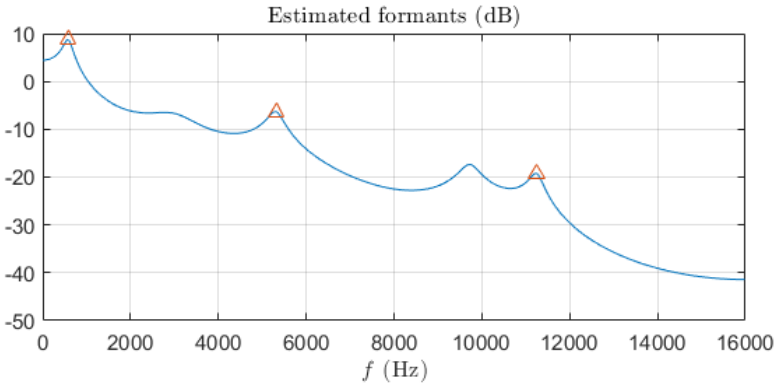

Figure 7: Formants estimated from the vocal tract filter.

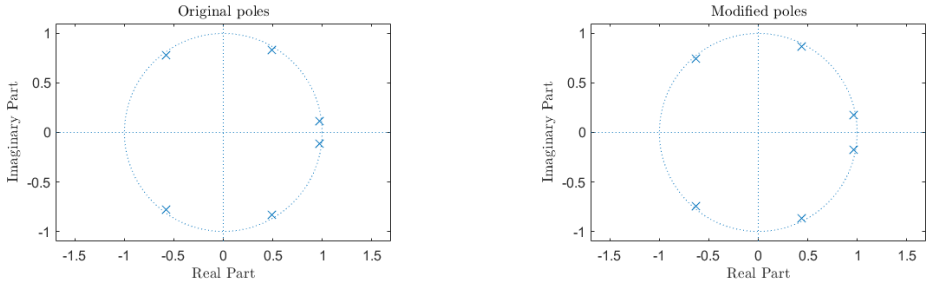

Figure 8: Modification of the poles by shifting the frequency. Conjugate symmetry is preserved.

f_{shift}. It is also important to remember that these frequencies are not the actual frequencies of the formants, they become the formants only after being applied to the excitation, which rescales their position. Thus shifting by $f_{\text{shift}} = 500$ does not imply translating the formants by exactly 500 Hz in the original signal.