Formant analysis from spectral content

The Praat script *formantSpectrogram.script (Oct.-2023)*

Ton Wempe

Formant frequencies of speech sounds are detected almost exclusively using LPC (Linear Predictive Coding). This achieves an all-pole filter as an approximation of the vocal tract function. In practice, some properties of this approach turn out to be quite problematic. First, the *filter order* (the number of poles) must be predefined, requiring two poles per formant, usually supplemented by two more poles. Furthermore, the frequency range must be limited to the maximum of the area where the main formants occur, the *formant ceiling*. Finally, the overall *spectral slope* caused by the combination of the slopes of the glottal pulse spectrum (on average -12 dB per octave) and of the radiation filter at the lips (+6 dB per octave) must be compensated for, to avoid the resulting slope being attributed to the vocal tract filter. Moreover, the settings for these three requirements are highly dependent on the speaker. Obviously, a set of three average values must be chosen in advance.

A further disadvantage is that an all-pole filter has no zero points so that the parallel filter channel through the nasal cavity is basically not taken into account. This causes that the formants of nasals and laterals are measured less reliably.

Despite all these drawbacks of using LPC for formant extraction, a reasonable compromise is achieved in practice by applying some rules of thumb, different for female and male voices, as proposed in the program Praat[1].

An alternative formant analysis, where no critical settings are required, is that where simply the spectral contents of successive bits of speech are used. Because the fundamental frequency (the 'pitch') affects the accuracy of this analysis, the settings could be made dependent on the local F_0 values, which adjusts the settings to optimal values for each bit of speech.

This method has been applied in the Praat script *formantSpectrogram.script*. It produces an 'F_0-controlled' **spectrogram** of a selected piece of speech, depicted in Praat's "Picture window", along with the marked formant positions. All measured formant frequencies are stored into a "Table" object. (For visual inspection of the spectrogram, it is recommended to take a short piece of speech of, say, a few syllables, since the Picture window cannot be zoomed in).

The term 'F_0-controlled' here means that in both the time domain and frequency domain the window length and spectral bandwidth, respectively, are determined by the local F_0. In the time domain, the window length is equal to 5 periods of the fundamental frequency (if voiced) and in the frequency domain, the spectrum is **smoothed** by a Gaussian window with an adjustable F_0 dependent bandwidth. The default value of this bandwidth is 1.2 x the local F_0 ('pitch') value, which turns out to be a proper value for all types of voices.

[1] *Praat: doing phonetics by computer* (version 6.3.17) by P. Boersma and D. Weenink. [http://www.fon.hum.uva.nl/praat]

This way there will never be vertical period lines or horizontal F_0 harmonics lines and the spectrogram is automatically adapted to female, male and child voices, relying on the 'pitch contour'.

If the local F_0 is "undefined" (i.e., voiceless), the global F_0 median is chosen as the basis for the window length. In these cases, a wide spectral bandwidth is chosen (see below).

Time frames are windowed with a **Hann window**.

The necessary **overlap** is achieved by defining the 'measurement grid' through the parameters "Time resolution of display" (default 5 ms) and "Frequency resolution of display" (default 20 Hz).

If desired, **formants** can be represented in the spectrogram as green dots, whether or not including the voiceless sections. For the voiceless frames, a bandwidth of *4 x global F_0 median* is then chosen. This way, fewer formant marks appear in the noise sections and the spectrogram with formant tracks becomes much quieter.

The formants are determined as follows: From the DFT spectrum of the current time frame, smoothed by the Gaussian smoothing, the positions of the **peaks** are determined via the negative-going zero crossings of the first derivative. The limit for appearing in the spectrogram depends on the maximum positive value of this derivative between the corresponding previous and present peak positions of the spectral function. This achieves a measure of the **steepness** of the leading slope of the peak. Experimentally, a limit of *9 / local bandwidth* appears to be a proper factor. For voiceless frames, taking into account the switch to a higher bandwidth, a limit of *4 / bandwidth* appears to be a useful value.

The formant frequencies of the marked formants are put into the Table "Formants", which can be saved as a "text file". In addition to the formant frequencies, the **q values** are also included, which are derived from the 'steepness' values mentioned above. They are a measure of the 'protrudence' of the peaks, i.e., the 'strength' of the formants. The formant frequencies divided by the q values are related to the bandwidths. These q values more directly reflect the 'strength' of the formants than the reciprocal bandwidths, which cannot be judged without taking into account the formant frequencies.

Compared to the usual LPC formant analysis, this spectral analysis has the following advantages: no settings needed such as 'formant ceiling' and 'order'; less sensitive for global spectral slope; suitable to nasals; good 'coverage' of formant positions on the black areas in the spectrogram.

The default setting values are practically suitable for any voice. Formants are numbered in order from low to high. Of course, the formant numbering here does not necessarily follow the 'linguistic' formant numbering: no prior knowledge is used here.

A disadvantage is that the formant determination by this simple peak detection method is somewhat **noise sensitive**. Therefore, in the start menu, a threshold (concerning the above-mentioned steepness) for displaying the formant dots can be entered for fine tuning to a compromise between displaying weak formants and noise peaks. However, a larger time overlap (i.e., a lower time grid value) is more effective for limiting the noise influence but the processing time of the script may then become quite long.
