A Short Guide to Pitch-marking in the Festival Speech Synthesis System and Recommendations for Improvements

Keywords: *Pitch-marks, Speech Tools, Festival, Speech Synthesis*

CSIR, Pretoria
aby.louw@eng.up.ac.za

Date	Author	Comments	Version
January 2004	Aby Louw	Request for comments and additions	0.1
		.	

Contents

1. Introduction

This document describes the methods used in Festival to extract pitch-marks for use in speech synthesis. The aim of this document is to provide some insight and possible solutions on the problem of pitch-mark extraction for a novice to the field of speech processing.

1.1 Overview of this document

Section 2 gives a definition of pitch-marks as used in the Festival speech synthesis system. Section 3 gives a short description of pitch-mark extraction when electroglottograph signals are available. Section 4 describes in some detail the procedure followed for pitch-mark extraction from a speech waveform. In Section 5 some of the research into automatic pitch-mark extraction is discussed. Finally, Section 6 gives a conclusion.

2. A Definition of Pitch-marks

A pitch-mark (pitch period) is defined as the location of the short-time energy peak of each pitch pulse in a speech signal, in other words, the beginning of a pitch period. This pitch pulse corresponds to the glottal closure instant (GCI). From this definition one can see that an unvoiced speech frame does not have pitch-marks since it has no pitch period. Figure 1 shows two pitch-marks on the short-time speech signal of the vowel /u/.

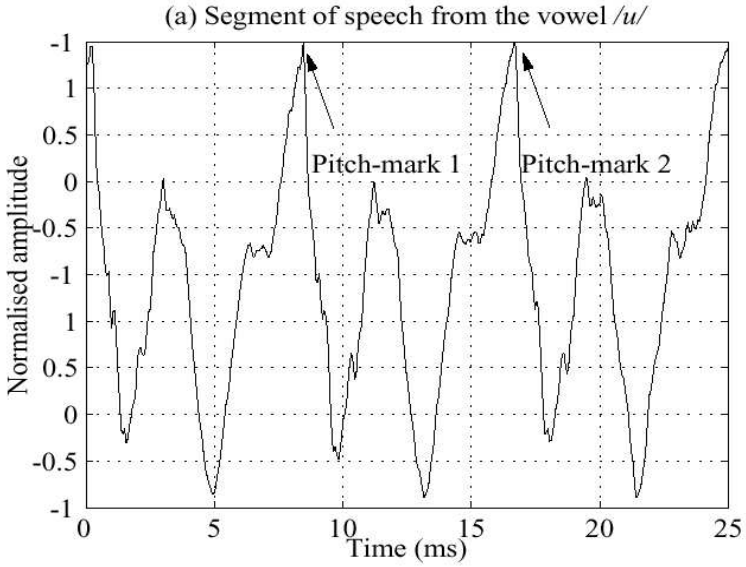

Figure 1: An example of pitch marks on the short-time speech signal of the vowel /u/.

Pitch synchronous speech synthesis algorithms require the beginning location of the pitch period (pitch-mark) for every voiced segment prior to speech synthesis. Festival, in its publically distributed form, currently only supports residual excited Linear-Predictive Coding (LPC) resynthesis [2]. This synthesis technique is pitch synchronous. In other words, it requires information about where pitch-marks occur in the acoustic signal. Pitch-marks are especially important in prosodic modification algorithms that employ a method known as *Pitch synchronous overlap-and-add* (PSOLA) to change the time and pitch scale of a speech signal.

There are two major techniques for acquiring pitch-marks, these are:

- From an electroglottograph signal, and
- algorithms extracting the pitch-marks directly from the speech signal.

3. Pitch-marks from an electrolottograph signal

Where possible, it is better to record speech with an electroglottograph (EGG), also known as a laryngograph, at the same time as the voice signal. The EGG records electrical activity in the glottis during speech, which makes it easier to get the pitch moments, and so they can be more precisely found. Figure 2 shows an example of a speech signal recording with it's accompanying laryngograph signal.

Although extracting pitch-marks from the EGG signal is not trivial, it is fairly straightforward in practice, as The Edinburgh Speech Tools[1] include a program *pitchmark* which will process the EGG signal giving a set of pitch-marks. However it is not a fully automatic process and requires someone to look at the result and make some decisions to change parameters that may improve the result. The parameters and their definitions for the *pitchmark* program are defined in [5].

4. Pitch-marks from a speech signal

If an EGG signal for a speech recording is not available then an alternative is to extract the pitch-marks using some other signal processing function. Finding the pitch-marks is similar to finding the *Fo* contour of a speech signal and, although harder than finding it from the EGG signal, with clean laboratory-recorded speech, such as diphones, it is possible.

Although never as good as extracting pitch-marks from an EGG signal, one can still achieve a fair amount of success in extracting pitch-marks from the raw waveform. The basic program used for the extraction is the same one as used the previous section (Section 3), *pitchmark,* which is part of the Edinburgh Speech Tools distribution. It is more computationally intensive, as it requires rather high order filters. A script, `bin/make_pm_wave` (which is copied by the voice setup process), is used to invoke the program and contains a few variable parameters.

The main parameters in the script are:

- `min` -- the minimum allowed pitch period (for the particular voice), in seconds
- `max` -- the maximum allowed pitch period (for the particular voice), in seconds
- `fill` -- insert and remove pitch-marks according to `min`, `max` and `def` period values. Often it is desirable to place limits on the values of the pitch-marks. This option enforces a minimum and maximum pitch period (specified `man` and `max`). If the maximum pitch setting is low enough, this will ensure that unvoiced regions have evenly spaced pitch-marks.
- `def` -- default pitch period in seconds, used for a guide as to what length pitch periods should be in unvoiced sections. This is usually set to 0.01 seconds.
- `wave_end` -- use the end of a waveform to specify when the last pitch-mark position should be.
- `lx_lf` -- low frequency cutoff (for the high-pass filter).
- `lx_lo` -- order of the high-pass filter.
- `lx_hf` -- high frequency cutoff (for the low-pass filter).

`lx_ho` -- order of the low-pass filter.

This program filters an incoming waveform (with a low-pass and a high-pass filter), then uses autocorellation to find the pitch-mark peaks with the `min` and `max` values specified. Finally it fills (`fill`) in the unvoiced section with the default pitch-marks defined with `def`.

[1] http://www.cstr.ed.ac.uk/projects/speech_tools/

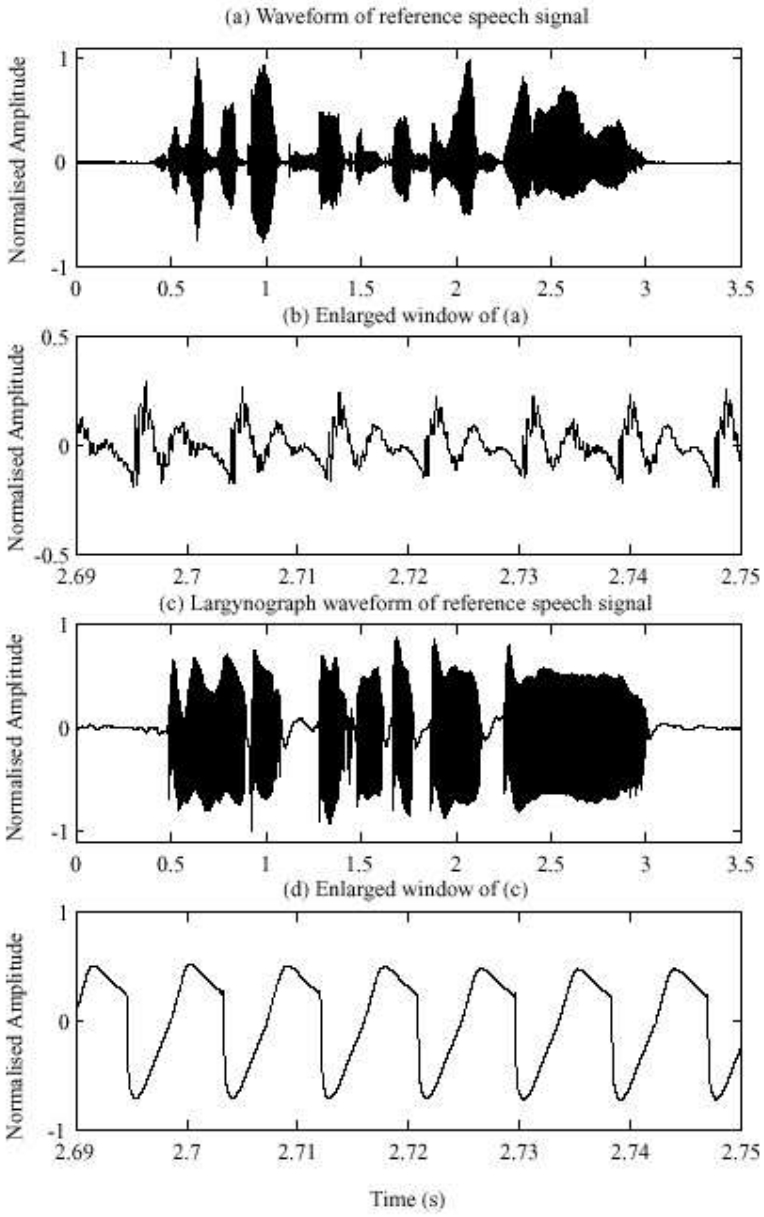

Figure 2: The speech signal "She had your dark suit in greasy wash water all year" with its accompanying laryngograph recording.

The steps involved in finding a speech waveform's pitch-marks are as follows:

1. Remove the `fill` option from the script, so that one can see where the program is finding the pitch-marks.
2. Modify the `min` and `max` values to fit the range of your speaker.
    - For a male speaker values in the range 0.005 and 0.012 (200 to 80 Hz) should be a good starting point.
    - For a female speaker the range of values are usually between 0.0033 and 0.7 (300 Hz to 140 Hz).
3. Run the script on a single file:

```
bin/make_pm_wave wav/awb_0001.wav
```

4. Run another script, `make_pmlab_pm`, on the output of the previous step. This script translates the pitch-mark file into a labelled file suitable for viewing with emulabel[2]:
`bin/make_pmlab_pm pm/awb_0001.pm`

5. Display the pitch-marks with the emulabel program: `emulabel etc/emu_pm awb_0001`

This should produce a number of pitch-marks over the voiced sections of speech. If there are none, or very few it definitely means the parameters are wrong. Figure 3 shows the pitch-marks found for the waveform ``taa taa taa''.

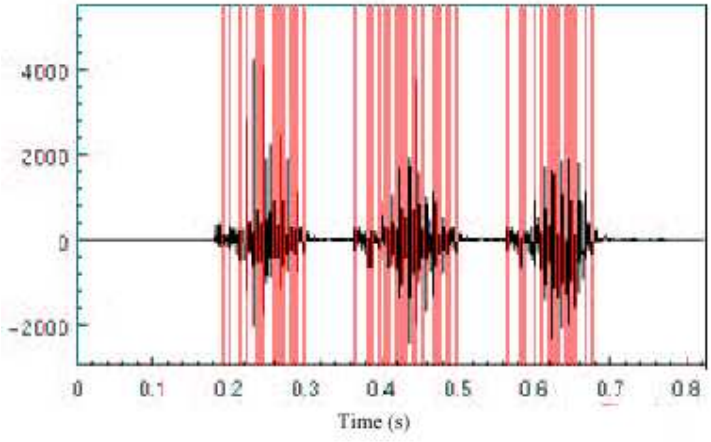

Figure 3: Pitch-marks found on the voiced regions of the waveform “taa taa taa”.

If the high and low-pass filter values `lx_lf` and `lx_hf` are inappropriate for the speakers pitch range one may get either too many, or two few pitch marks. For example if the low frequency cutoff (`lx_lf`) of the example in Figure 3 is lowered, from 200 to 60 only two pitch-marks are in the third vowel as shown in Figure 4.

By zooming in on the first example of Figure 3 one gets Figure 5. The pitch marks should be aligned to the largest (above zero) peak in each pitch period. From Figure 5 one can see that there are too many pitch-marks (effectively twice as many). The pitch-marks at 0.617, 0.628, 0.639 and 0.650 are extraneous. This means the pitch range is too wide. By increasing the `min` size, and lowering the low frequency filter cutoff value (`lx_lf`) one gets the pitch-marks as shown in Figure 6.

From Figure 6 one can see that the pitch-mark doubling as shown in Figure 5 is eliminated, but its now missing pitch-marks towards the end of the vowel, at 0.634, 0.644 and 0.656. The double pitch-mark problem can be lessened by not only changing the range but also the order of the high and low-pass filters (effectively allowing more smoothing). Thus when secondary pitch-marks appear increasing the `lx_lo` parameter often helps. Figure 7 shows the pitch-marks found with the increased `lx_lo` parameter.

[2] The Emu Speech Database System. Speech Hearing and Language Research Centre, Macquarie University. http://www.shlrc.mq.edu.au/main/resources.html

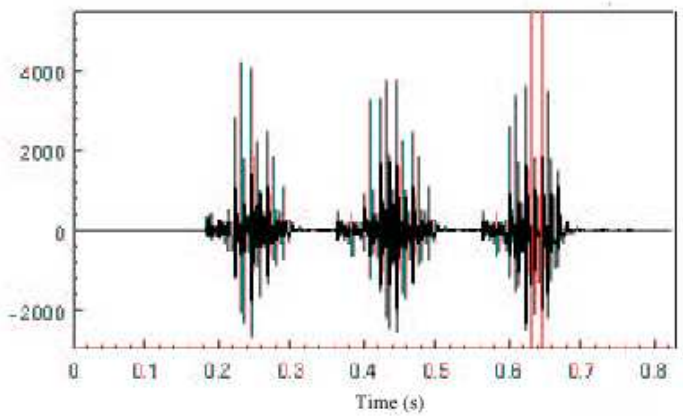

Figure 4: An example of a too low value for the low frequency cutoff (`lx_lf`) value.

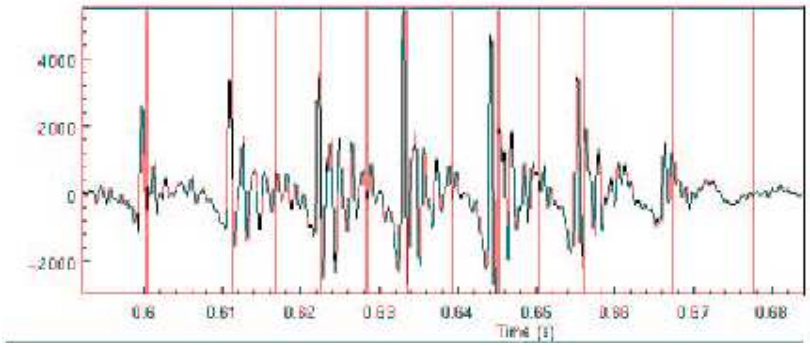

Figure 5: An enlarged version of Figure 3.

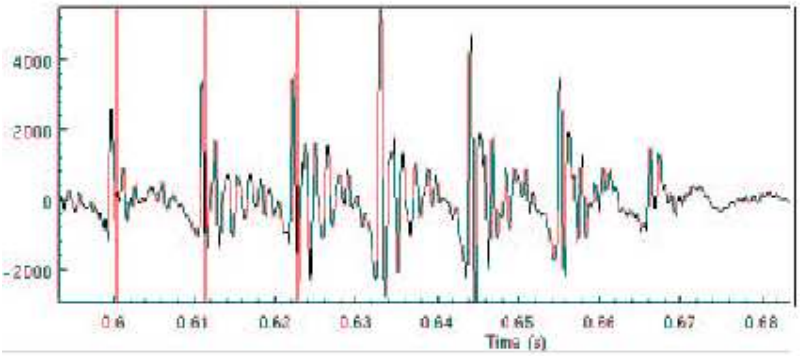

Figure 6: Decreased range and lowered low frequency filter value (`lx_lf`).

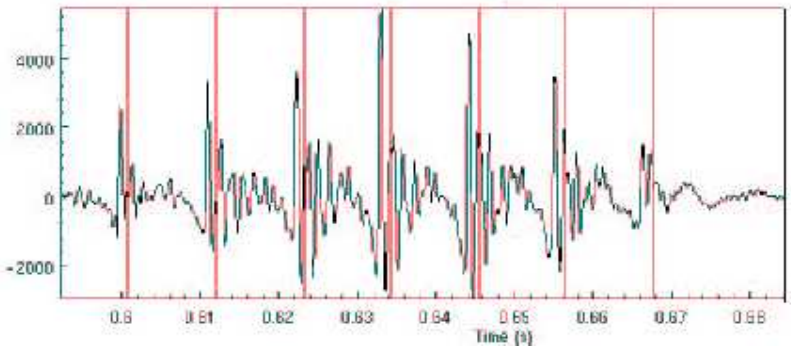

Figure 7: Increased low frequency filter order (`lx_lo`).

The pitch-marks of Figure 7 are satisfactory for that specific file and probably for the whole databases of that speaker. Though it is worth checking a few other files to get the best results. Note the by increasing the order of the filer the pitch-marks move forward. A post-processing step is provided that moves the predicted pitch-marks to the nearest waveform peak. A simple script is provided for this:

```
bin/make_pm_fix pm/*.pm
```

If the pitch-marks are aligning to the largest troughs rather than peaks the signal is upside down. The signal can be inverted with the following script:

```
for i in wav/*.wav
do
   ch_wave -scale -1.0 $i -o $i
done
```

5. Research on automatic pitch-marking

The pitch-marking method in Festival, as described in Section 4, requires some manual intervention. Some of the latest research in pitch-marking has delivered a few algorithms that are fully automatic and achieve acceptable results. These methods differ widely in their implementations and include the following:

- Chen and Kao [1] posed a pitch-marking method based on an adaptable filter and a peak-valley estimation method. The input signals are limited to voiced speech because only the periodic parts are of interest. An adaptable filter, which serves as a bandpass filter, is used to transform voiced speech into a sine-like wave. A Fast Fourier Transform (FFT) is used to transform the time signal to the frequency domain, and the filter's pass band is determined by finding the spectral peak of the fundamental frequency. Consequently, the pass band can be adapted based on the fundamental frequency. The autocorrelation method is then used to estimate the pitch periods on the sine-like wave. In addition, a peak-valley decision method is employed to determine which part of the voiced speech is suitable for pitch-mark estimation. With this technique they achieved a success rate of 97.2%.

- Laprie and Colotte [3] proposed an automatic pitch-marking method whereby the propagation of pitch-marks from one pitch period to another are optimised by means of dynamic programming. Extrema are extracted on regularly spaced speech segments (the size of which is the smallest pitch period under investigation). Then an optimal subset of extrema, which corresponds to pitch-marks, is found by means of a dynamic programming algorithm.

- Sakamoto and Saito [4] proposed an automatic pitch-marking method using the wavelet transform. This method detects discontinuity in the speech waveform which occurs at the glottal closure instant. A 96% detection accuracy was achieved on a performance evaluation.

6. Conclusion

To ease the process of pitch-mark extraction for a novice one has to either provide an automatic method which requires no knowledge of the methods involved, or provide more information on the methods involved so that the novice can make a better informed decision regarding the suitability of the acquired results. Thus, the following three options arise:

1. The default Festival pitch-marking method is not fully automatic and requires manual intervention and affirmation of the results. This may prove to be a formidable hurdle to

a novice in the speech processing field. Thus, it may be advantageous to implement a fully automatic pitch-marking algorithm, such as mentioned in Section 5, which produces acceptable results.

2. Another solution to this problem may be a way of interpreting the results of Section 4 with an analytical method. For instance, the pitch-marks obtained by the method as described in section 4 can be correlated with the *F0* curve of the same waveform. Since the *F0* curve can be obtained from the pitch-marks, one can determine the fitness of the pitch-marks results.

3. Finally, a comprehensive guide can be written, of which Section 4 is an abbreviated version, with step-by-step instructions and examples on how to extract suitable pitch-marks from recorded speech files.

Aviable solution might be a combination of option 2 and 3 above, since the already existing methods in Festival do produce acceptable results.

References

[1] J-H. Chen and Y-A. Kao. Pitch Marking Based on an Adaptable Filter and a Peak-Valley Estimation Method. *Computational Linguistics and Chinese Language Processing,* 6(2):1-12, February 2001.

[2] M. Hunt and D. Zwierynski and R. Carr. Issues in high quality LPC analysis and synthesis. Eurospeech89, v. 2, pp. 348--351, Paris, 1989.

[3] Y. Laprie and V. Colotte. Automatic pitch marking for speech transformations via TD-PSOLA. In *European Signal Processing Conference,* Rhodes, 1998.

[4] M. Sakamoto and T. Saito. An Automatic Pitch-Marking Method using Wavelet Transform. In *Proc. of ICSLP2000*, v.3, pp. 650-653, Bejing, China, October 2000.

[5] P. Taylor and R. Caley and A.W. Black and S. King. *Edinburgh Speech Tools Library, System Documentation.* Centre for Speech Technology, University of Edinburgh, 1.2 edition, June 1999.