High Resolution Speech F_0 Modification

Tamás Bárdi

Faculty of Information Technology
Péter Pázmány Catholic University, Budapest, Hungary
bardi@itk.ppke.hu

Abstract

The present paper proposes a new algorithm for pitch modification which is convenient for changing the fundamental frequency of speech with so fine resolution that is at least comparable with human pitch perception. Using the proposed method, measurements of just noticeable changes on speech prosody becomes possible. High resolution F_0 manipulation is completed without explicit over-sampling of the signal, our FFT-based fast interpolation technique is used instead. Our algorithm is based on LP-PSOLA method. Although its frequency resolution was enhanced especially for research purposes it is possible that the need will arise from real applications of expressive speech synthesis in the future.

1. Introduction

1.1. Current pitch modification methods

Today the most important application area of speech fundamental frequency (F_0) modification is concatenative speech synthesis. Very popular and widely used pitch modification techniques are the Time-Domain Pitch-Synchronous Overlap and Add (TD-PSOLA) [8], and its variant, the LP-PSOLA method, which works on the Linear Prediction residual error signal [6]. Both of them are used to modify phoneme durations also.

Although the existing pitch manipulation methods have achieved a high level of intelligibility, this problem still attracts the attention of researchers, especially because of the limitations on the modification range. Recent studies investigate the possibility of improving the quality of the resynthetized voice and its natural sounding [5].

Current methods promise reasonable voice quality for not more than 1 octave modification rate, remarking that the quality degradation is noticeable for large scales.

Another known limitation is their frequency resolution: usually pitch period (T_0) can be changed with only integer numbers of samples. Improving this feature was the main goal of the present study.

1.2. Motivations of high resolution pitch modification

One of the most challenging applications of prosody manipulation is emotional or expressive speech synthesis. Synthetic speech that sounds genuinely expressive could be refunding certainly in the computer game industry, and other applications in human-computer interaction are also promising. But there are two problems (at least) in synthetizing speech with prosody that feels really expressive. The first is that variability of pitch in expressive speech is often so large that exceeds the modification range of existing methods. The second is that we do not know really how emotions are expressed by prosody.

From the engineer's point of view, expressive speech would be easier to handle in speech technology, if the effects of affections on prosodic cues could be described with some discrete symbols. Concordantly, there are research efforts to fit the psychological theory of discrete emotions and the categorical perception theory with speech acoustics [7]. But some psychologists agree that the affectivity of speech shows similar behavior to the non-verbal communication and can not be really described with discrete categories [4]. In our mind instead of discrete categories, the exact amounts of prosodic changes are the key indicators of the affective contents of speech. In everyday life our recurrent experience that pretended versus lived emotions can be discriminated from very fine changes in speech melody, remarking that they are highly unintentional. That happens sometimes even if speech sound is the only medium of communication, for example when people talk to each other on the phone.

We have started a research study addressed to recover the role of fine prosodic changes in affective speech, which is the objective of our forthcoming paper. This study involves psychoacoustic experiments for quantitative measuring just noticeable differences on prosodic cues. To provide suitable speech stimuli for these listening tests, special algorithms have been required to manipulate the prosodic features of recorded speech signals. The required algorithms should allow modifying pitch and duration with small or large scale but with very accurately specified amounts. Time resolution of the existing duration modification methods seemed to be suitable for our purposes. But changing the fundamental period with only integer number of samples gives insufficient frequency resolution. Although there are proposed Pitch Determination Algorithms (PDA), providing sub-sample resolution for T_0 [2], earlier there was not appeared any reason to achieve that level on the synthesis side. But is it really so hard to exceed the performance of human pitch perception in frequency resolution?

For a short example, F_0 contour for female speech often reaches 320 Hz. For that pitch - in our experience - listeners who has good ear can reliably notice 1-2 cent difference for sinusoid sounds. (1 cent is one 1200th part of the octave) They also can differentiate 3-4 cents for saw-tooth wave, and 5-6 cents for slightly trembled sinusoid or saw-tooth wave. In comparison, assuming even 48 kHz sampling rate of the speech signal, the 320 Hz F_0 belongs to 150 sample long fundamental period. Decreasing it to 149 samples we get 322.15 Hz, which means 11-12 cents in change. Moreover, fundamental frequency for expressive female speech often goes much higher than 320 Hz, and the sampling rate is less than 48 kHz in most of the implemented speech signal processing applications including speech synthesis systems. The previous calculation for F_0 = 400 Hz sampled at 16 kHz results 44 cent as the smallest possible increase.

Briefly, to begin our above mentioned psychoacoustic experiments we needed a pitch modification algorithm with higher frequency resolution than allowed by any existing method, hence - as our preliminary task - we had to develop one.

1.3. Our extensions to LP-PSOLA method

In this paper we propose an extended LP-PSOLA pitch modification algorithm, which is improved primarily in its frequency resolution. Some improvements in voice quality for large modification rates were also achieved. We extended the traditional LP-PSOLA algorithm with three special signal processing methods which are novel in application for this purpose.

Our first invention is a very simple high-pass filter which helps to suppress the noise effects of phase mismatches which occur especially for large modification rate (1.5 or above).

The second new solution is our special signal interpolation method named to Fractional-lag Time-Delaying (FTD). The third novelty is the Fractional-lag Autocorrelation Function (FACF), which is the interpolated version of the well known Autocorrelation Function (ACF) of discrete-time signals. These two methods are the key of the high resolution fundamental frequency determination and modification, achieving it without over-sampling the speech signal. Both FTD and FACF use Fast Fourier Transformation and their computational costs are less than equivalent over-sampling of the signal.

Section 2 describes our algorithm in detail, taking the focus onto our new solutions.

2. Pitch modification

During the development our algorithm was tested on recorded speech data sampled at 16 kHz. All the sampling frequency dependent parameters like LP order are adjusted to 16 kHz.

Firstly, in our pitch modification system the source speech signal is pre-emphasis filtered with $H(z)=1-0.95z^{-1}$. Then time-varying Linear Predictive analysis filter is applied to get the residual error signal. The filter structure is similar to the one applied in the standard GSM full-rate speech codec controlled by reflection coefficients. The prediction order is 16. Reflection coefficients are computed for 24 ms long Hamming-windows with 20 ms overlapping, and the coefficients are stored for the synthesis filter.

Normally in LP-PSOLA method pitch manipulations are accomplished in the LPC error signal. In our system a very simple first order IIR high-pass filter is applied to the residual error as a weighting filter (detailed in 2.1). All the pitch manipulation is done on the weighting filtered LPC error.

Usually in PSOLA methods pitch modification is done by time shifting the pitch-synchronous Hanning-windows, then overlapping and adding them. Our system does the same, but using our FTD method (detailed in 2.2) time shifting with non-integer number of samples is also allowed. In this way synthesis pitch periods can be adjusted with higher resolution than in traditional PSOLA methods.

Adjusting synthesis pitch with high accuracy needs analysis pitch period determination with the same accuracy. In our system there is an open-loop pitch analysis that searches periods with 1 sample accuracy. It is done by our ACF-based pitch and voicing determination algorithm proposed in [3]. Then closed loop pitch analysis 1 sample around the picked period is done by our FACF method (detailed in 2.3), formally searching for the fundamental period with 0.01 sample resolution. Validation of the effective resolution presented in Section 3.

When pitch epochs are adjusted to fit with the target pitch contour, the signal is filtered with the inverse of our weighting filter. Then LP synthesis filter is applied with the stored reflection coefficients. Finally the target speech signal is delivered by the de-emphasis filter $1/H(z)$.

If no pitch manipulation is done on the weighted residual error, the implemented system reconstructs perfectly the source speech signal.

2.1. Weighting filter

The ideal excitation signal of the vocal tract filter in source-filter models of speech production is the sum of a pulse-train as for the voiced component and some white noise as for the unvoiced component [1]. If the LP residual were forming that ideal excitation, pitch modification could be done on it without any problem. But the noise component in the residual error usually non-white and the voiced component is not a simple pulse-train. In practice the LPC error just resambles to the ideal excitation.

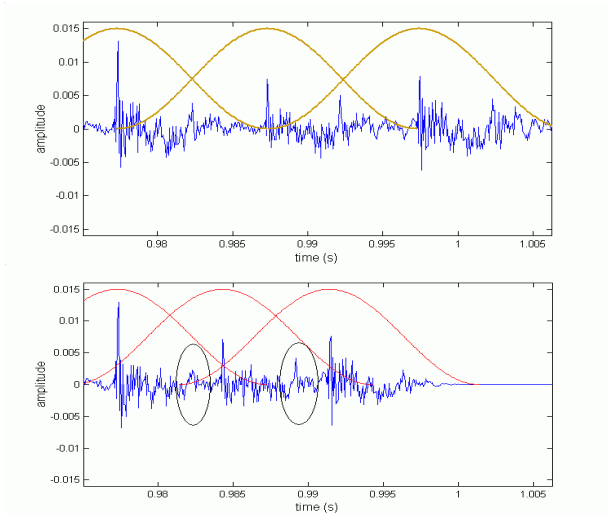

Figure 1: *LPC error before and after shifting pitch-synchronous windows. Phase mismatches caused by disrupted low-frequency component marked with ellipses.*

As in Figure 1 can be seen, LPC error has a low frequency component (close to F_0). Hanning-windows are centered to the pitch epochs. After overlap-and-adding them, to complete pitch modification, that low frequency component is disrupted, and it causes phase mismatches adding some noticeable noise to the resynthetised speech. The significance of phase mismatches is increasing in the case of larger modification rates.

In our system we applied a very simple weighting filter to the residual error, in order to reduce this effect:

$$W(z)=\frac{1-0.95\cdot z^{-1}}{1-0.5\cdot z^{-1}} \qquad (1)$$

This first order IIR high-pass filter is stable, invertible, and easy to implement. Its magnitude frequency response shows about -20 dB attenuation below 125 Hz, increases with 6

dB/octave between 125 and 2000 Hz, and it is approximately flat over 2000 Hz.

Applying *W(z)* to the LPC error the above mentioned low frequency component is suppressed, hence the additional noise effect of phase mismatches on the resynthetized speech is also suppressed. Figure 2 shows the filtered error signal before and after pitch manipulation.

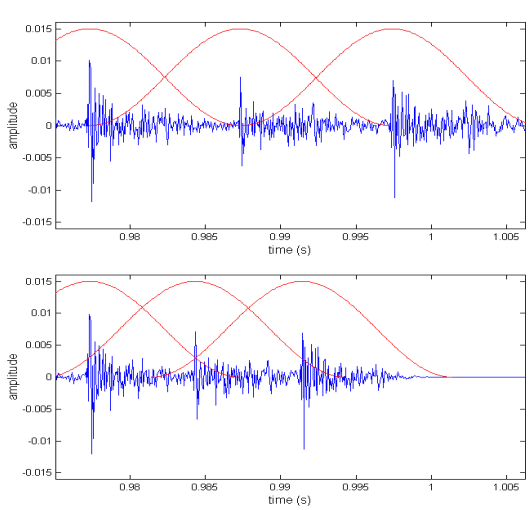

Figure 2: *Weighting filtered LPC error before and after shifting pitch-synchronous windows. Pitch modification produces less phase mismatches.*

2.2. Fractional-lag Time-Delaying (FTD)

In digital signal processing there is a widely used technique for speeding up the computation of convolution via Fast Fourier Transform (FFT) [9]. It is usually done in two steps: expanding input vectors with zeros and then circular convolution is completed using FFT. We used this approach for delaying short-time signals.

Let (x[0], x[1], … x[N-1]) a short-time signal. Then the signal delayed circularly with m sample is:

$$x^{(m)}[n] = x[(n-m) \bmod N] \quad \text{for } n = 0..N-1 \qquad (2)$$

If the delayer filter $d^{(m)}$ is

$$d^{(m)}[n] = \begin{cases} 1 & \text{if } n = m \\ 0 & \text{if } n \neq m \end{cases} \qquad (3)$$

then

$$x^{(m)} = x * d^{(m)} \qquad (4)$$

where * denotes the circular convolution. Transforming it to the frequency domain:

$$X_k^{(m)} = X_k \cdot D_k^{(m)} \quad \text{for } k = 0..N-1 \qquad (5)$$

Doing time-delay in this way $D^{(m)}$ works as a phase rotator applied to X, where the rotation angles are:

$$D_k^{(m)} = e^{-j\frac{2\pi}{N}mk} \qquad (6)$$

Circular convolution using FFT needs $O(N \log(N))$ operations, and simple time-delay as in (2) needs only $O(N)$. But in our way the phase rotation angles can be set to any real number, causing not-integer number of samples delay in the time-domain. Taking care about the well known complex conjugation property of Fourier-transformed real vectors, the rotator coefficients are:

$$D_k^{(t)} = \begin{cases} e^{-j\frac{2\pi}{N}tk} & \text{if } 0 \le k < N/2 \\ \mathrm{Re}\left(e^{-jt\pi}\right) & \text{if } k = N/2 \\ e^{j\frac{2\pi}{N}t(N-k)} & \text{if } N/2 < k < N \end{cases} \qquad (7)$$

where t can be any real number.

Then $x^{(t)}$, the circularly delayed signal with t sample is the inverse FFT of $X \cdot D^{(t)}$.

For any integer t this method results the same as simple delaying in (2). It can be proved that FTD is mathematically equivalent with sinusoidal interpolation of the signal for any non-integer t, but using FFT N new samples are computed simultaneously. So our method interpolates short-time signals efficiently: high accuracy with relatively low computational cost.

Figure 3 shows an example how we use FTD. The first signal is a pitch-epoch centered window, the second is shifted with -40 samples, and third one is shifted with -40.4 samples. The last signal's figure differs from the other two, because the MatLab software simply draws straight lines between the sample points.

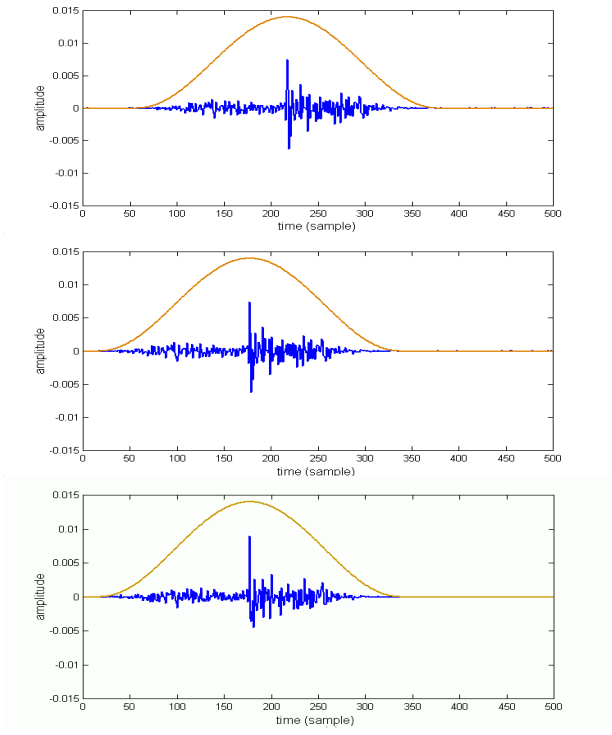

Figure 3: *Time-shifting pitch-epoch centered window. The last signal is shifted with non-integer number of samples.*

2.3. Fractional-lag Autocorrelation Function (FACF)

Using FTD we can get an interpolated version of ACF, defining it for non-integer time-lags. For the short-time signal x on window w at time-lag t the autocorrelation is:

$$r(t) = \sum_{n=0}^{N-1} x^{(t)}[n] \cdot x[n] \cdot w[n] \tag{8}$$

To compute it, actually we do not have to determine explicitly the $x^{(t)}$ signal. Dot product of the vectors can be computed in the frequency domain:

$$r(t) = \frac{1}{N} \sum_{k=0}^{N-1} D_k^{(-t)} X_k^* \cdot XW_k \tag{9}$$

where XW is the FFT of x windowed by w. Consequently, once we have X and XW we can get $r(t)$ for an additional t with computing just the dot-product in (9), with no more FFT or inverse FFT. Also to spare the computational cost, FACF is not normalized to $r(0)=1$. In our application only its maximum place is interesting, which belongs to T_0.

In closed loop pitch analysis, our algorithm uses 12.5 ms long Tuckey-window with 70% flat part. For the example segment of the weighted LPC error, the open loop pitch analysis found the period at 123 samples. As it can be seen on Figure 4, (normalized) FACF between 122 and 124 samples shows very large variability. Its maximum is at period 123.41 samples, but for 122.41 samples the signal is uncorrelated. This example illustrates why over-sampling is used in pitch predictive coders, such as the GSM enhanced full-rate codec.

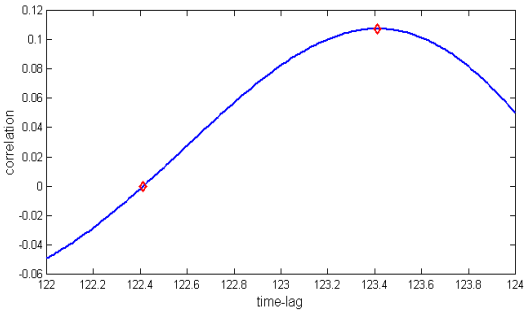

Figure 4: *Zooming to autocorrelation of LPC error. Maximum correlation lag is only 1 sample distance from zero correlation lag (marked).*

3. Discussion

The effective resolution of our FACF-based closed loop pitch analysis method was tested on synthetized signals. The same pitch epoch centered window was put twice in a row with some overlapping to get a short time signal with warranted pitch period. Then some noise, retrieved from unvoiced fricative segment's residual error, was added to it. The maximum place of FACF was compared with the synthetized exact period. Repeating this procedure for all pitch epochs in our test speech data, the estimation error all along was below 0.14 sample beside 0 dB additional noise. This accuracy of estimation limits the accuracy of modification.

There is some benefit of using our weighting filter but moderate indeed. Listeners in subjective test for pitch modification rates less than 1.5 were not able to differentiate speech stimuli whether the weighting filter was used or not. Increasing the modification rate, some metallic brush-scrubbing like noise rise in the resynthetised speech sound, which are slightly but noticeably less for the weighting filtered version. The difference in quality was felt by the listeners to have been compensated with -35 dB white noise added to the weighting filtered version for 1.67 modification rate, and -30 dB for 2.0 modification rate. We think it worth to investigate if inventing a more sophisticated weighting filter more extent improvement in voice quality could be achieved.

The most important feature of our algorithm for us is its frequency resolution. One word long utterances (both female and male speech) were modified in pitch with an eligibly dense sequence of rates, in order to verify the resolution. The sequence is 1.2, 1.2028, 1.2056, 1.2084, 1.2112 and 1.214. Consequently there is about 4 cent difference between neighbors. One of the listeners could notice 16 cent difference or more. The best performance was 8 cent, and no one of them was able to notice 4 cent difference. If the listener is unable to realize the pitch difference between let's say sound A and sound B, and neither between B and C, but realizes it between A and C, then the sounds must be gradually changing in pitch, though the difference is unnoticeable between the neighbors.

Acknowledgements

The author would like to thank to his supervisor György Takács and colleague Gergely Feldhoffer for their help.

4. References

[1] Acero, A., 1998. Source-filter models for time-scale pitch-scale modification of speech. *23rd Int. Conf. on Acoustics, Speech, and Signal Processing*. Seattle.

[2] Bagshaw, P. C.; Hiller, S. M.; Jack, M. A., 1993. Enhanced pitch tracking and the processing of F0 contours for computer aided intonation teaching. *3rd European Conf on Speech Communication and Technology*. Berlin, 1003-1006.

[3] Bárdi, T., 2004. Speech F0 estimation with enhanced voiced-unvoiced classification. *Híradástechnika – Communication.* 59(12), Budapest.

[4] Buda, B., 2001. *A közvetlen emberi kommunikáció szabályszerűségei*. Budapest: Animula.

[5] Cabral, J. P.; Oliveira, L. C., 2005. Pitch-synchronous time-scaling for prosodic and voice quality transformations. *9th European Conf on Speech Communication and Technology*. Lisbon, 1137-1140.

[6] Edgington, M.; Lowry, A., 1996. Residual-based speech modification algorithms for text-to-speech synthesis. *ICSLP 96*. Philadelphia, 1425-1428.

[7] Laukka, P., 2004. Vocal expression of emotion. PhD Thesis. *Acta Universitatis Upsaliensis.* Uppsala.

[8] Moulines, E.; Charpentier, F., 1990. Pitch-synchronous waveform processing techniques for text to speech synthesis using diphones. *Speech Communication* 9(5), 453-467.

[9] Oppenheim, A. V.; Schäfer, R. W., 1989. *Discrete-time signal processing*. Prentice Hall