CHAPTER 15

Time-Domain and Frequency-Domain Techniques for Prosodic Modification of Speech

E. Moulines

Ecole Nationale Supérieure des Télécommunications
46 Rue Barrault, 75634 Paris Cedex 13, France

W. Verhelst

Vrije Universiteit Brussel, Faculty of Applied Science, department ETRO
Pleinlaan 2, B-1050 Brussels, Belgium

Contents

Speech Coding and Synthesis
Edited by W.B. Kleijn and K.K. Paliwal

1. Introduction

Speech prosody is generally considered to depend on supra-segmental characteristics such as the temporal variation of pitch, speaking rate, and loudness. At the perceptual level, they are related to speech melody and rhythm. A number of speech processing applications depend on a successful modification of prosodic features, especially of the time-scale and/or the pitch-scale. Shortening the duration of original speech messages, for example, allows for corresponding savings in storage and transmission. Rendering speech at a rate that can be chosen arbitrarily different from the original rate can increase the ease-of-use and the efficiency of speech-reproduction equipment (it can allow, for example, for faster listening to messages recorded on answering machines, voice mail systems, information services, etc., or for rendering speech from dictation tapes in synchrony with the typing speed). For text-to-speech synthesis systems based on acoustical unit concatenation (see chapters 17 and 19), prosodic manipulations are essential in order to adapt the original time scale and pitch scale of the synthesis units to the target prosody.

A number of algorithms for high-quality time-scaling and pitch-scaling have recently been proposed together with real-time implementations on sometimes very inexpensive hardware. Most of these algorithms can be viewed as variations on a small number of basic schemes. It is the main purpose of this chapter to review these algorithms in a common framework based on a simple extension of the short-time Fourier transform (STFT) analysis-synthesis principle.

The chapter is organized as follows. In section 2, a basic speech-production model is reviewed and used to define exactly what is meant by time-scale and pitch-scale modification. Some possible general approaches to the problem are also briefly discussed. Section 3 presents the STFT as a time-frequency representation for the analysis, modification and synthesis of slowly time-varying signals. A fairly general approach is taken in that the STFT analysis and synthesis characteristics (i.e., windowing functions and downsampling factors) are defined as time-varying quantities. This flexibility is one of the key points that are exploited for constructing efficient high-quality time-scaling and pitch-scaling strategies in sections 4.3 and 5.

2. General considerations on time-scaling and pitch-scaling

2.1. A simple model for voiced speech

In discussing the problems of time-scaling and pitch-scaling, we will find it helpful to refer once in a while to a specific model for voiced speech ([1]). It should be stressed, however, that this model will only serve explanatory purposes and is not actually used in the methods presented in this chapter. For methods that do explicitly rely on such models, see for example chapter 4.

According to the source-filter model for speech production (chapter 1), the speech waveform can be seen as the output of a time-varying linear filter driven by an excitation signal $e(n)$. In a widespread engineering approach, this excitation is taken

to be either a sum of narrow-band signals with harmonically-related instantaneous frequencies (voiced speech), or a stationary random sequence (unvoiced speech). Here, we will mainly focus on the voiced speech case.

The time-varying filter in the source-filter model represents the combined acoustical effects of glottal-pulse shape and vocal-tract transmission characteristics; it carries information related to voice quality and articulation. The input-output behavior of the system can be characterized by its time-varying unit-sample response $g(n,m)$ or by the associated time-varying transfer function

$$\sum_{m=-\infty}^{+\infty} g(n,m)\exp(-\mathrm{j}\omega m) = G(n,\omega)\exp(\mathrm{j}\psi(n,\omega)),$$

where $g(n,m)$ is defined as the response of the system at time n to an impulse applied m samples earlier, at time $n-m$. $G(n,\omega)$ and $\psi(n,\omega)$ are respectively referred to as the time-varying amplitude and phase of the system. The non-stationarity of $g(n,m)$ originates from the physical movement of the articulators, which is usually slow compared to the time-variation in the speech waveform. Therefore, $g(n,m)$ can be considered as nearly constant for the duration of its memory, i.e., $g(n,m)$ represents a *quasi-stationary* system.

For voiced speech, the excitation waveform $e(n)$ is represented as the sum of harmonically-related complex exponentials with unit amplitude, zero initial phase, and a slowly varying fundamental frequency $f_k(n) = \omega_k(n)/2\pi = 1/P(n)$:

$$e(n) = \sum_{k=0}^{P(n)-1} \exp\left[\mathrm{j}(\Phi_k(n))\right],$$

$$\Phi_k(n) = \sum_{m=0}^{n} \omega_k(m) = \sum_{m=0}^{n} \frac{2\pi k}{P(m)},$$

where $P(n)$ is the time-varying period of the speech signal and will be referred to as the *pitch* period. Because $P(n)$ is nearly constant around any particular time instant n, the excitation phase $\Phi_k(m)$ may be approximated as

$$\Phi_k(m) \approx \Phi_k(n) + \frac{2\pi k}{P(n)}(m-n) \quad \text{for small} \quad |m-n|.$$

From standard linear-system theory, the voiced speech signal $x(n)$ that is obtained at the output of the system $g(n,m)$ in response to $e(n)$ is given by

$$x(n) = \sum_{m=-\infty}^{+\infty} g(n,m)e(n-m).$$

Assuming that the pitch-period $P(n)$ is nearly constant for the duration of $g(n,m)$, the excitation signal can be approximated by its local harmonic representation to

obtain

$$x(n) = \sum_{k=0}^{P(n)-1} G(n,\omega_k(n)) \exp\left[j(\Phi_k(n) + \psi(n,\omega_k(n)))\right]$$

$$= \sum_{k=0}^{P(n)-1} A_k(n) \exp(j\theta_k(n)). \tag{2.5}$$

The amplitude $A_k(n)$ of the k-th harmonic is equal to the system amplitude at the harmonic frequency $\omega_k(n)$:

$$A_k(n) = G(n,\omega_k(n)).$$

The phase $\theta_k(n)$ of the k-th harmonic is the sum of the excitation phase $\Phi_k(n)$ and the system phase $\psi(n,\omega_k(n))$:

$$\theta_k(n) = \Phi_k(n) + \psi(n,\omega_k(n))$$

$\theta_k(n)$ is often referred to as the *instantaneous phase* of the k-th harmonic. The system phase $\psi(n,\omega_k(n))$ being a slowly varying function of n, the instantaneous phase $\theta_k(m)$ may be approximated, using eq. (2.3), as

$$\theta_k(m) = \theta_k(n) + \omega_k(n)(m \quad n) \quad \text{for small} \quad |m-n|$$

2.2. *Time-scale modification*

The problem with time-scaling a speech signal $x(n)$ of original duration Δ lies with the corresponding frequency shift distortion. It is a common experience that when $x(n)$ is played back at a higher speed than the recording speed, the resulting sound is distorted in that its pitch is raised. Conversely, when the recording is played back at a lower speed the pitch is lowered. Not only pitch is affected but timbre is distorted as well, such that when the playback speed differs significantly from the recording speed comprehension of the messages becomes difficult or even impossible.

The duality between time-scaling and frequency-shifting becomes clear mathematically by considering the signal $y(n)$ that corresponds to an original signal $x(n)$ played at a speed α times higher than the recording speed. Thus, an original time span Δ is played in Δ/α and $y(n) = x(\alpha n)$[1]. From eq. (2.5), we have

$$y(n) = \sum_{k=0}^{P(\alpha n)-1} A_k(\alpha n) \exp(j\theta_k(\alpha n)), \tag{2.9}$$

$$\theta_k(\alpha m) \approx \theta_k(\alpha n) + \alpha \frac{2\pi k}{P(\alpha n)}(m-n) \quad \text{for small } |m-n|$$

[1] Note that αn is not necessarily integer; it should be understood in the sequel that $s(\alpha n)$ corresponds to the n-th sample $s_\alpha(n)$ of the $1/\alpha$-fold band-limited interpolation of $s(n)$.

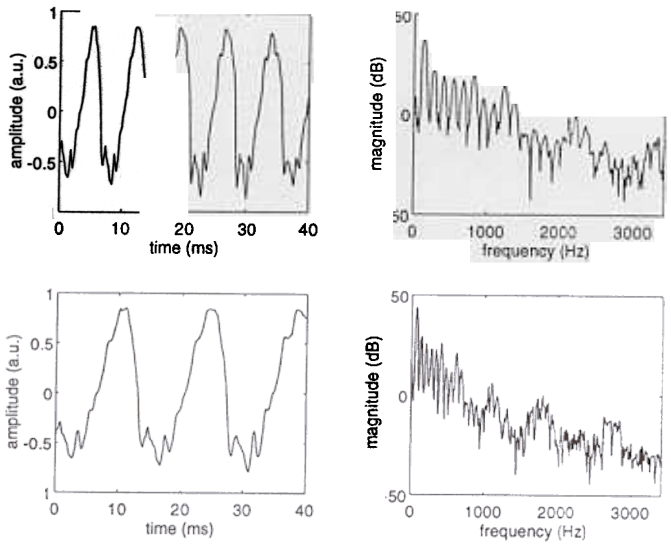


Figure 1. Illustration of the duality between time domain and frequency domain. The upper row shows a 40 *ms* voiced speech segment and its spectrum; the second row illustrates that when this signal is played at half speed it is stretched two-fold in the time domain and compressed two-fold in the frequency domain.

It appears that compression of the time-scale by a factor α does not only compress the time evolution of the pitch $P(n)$ and of the pitch-harmonic amplitudes $A_k(n)$ by that factor, but also compresses the time evolution of the instantaneous phase $\theta_k(n)$, which in turn induces expansion of the instantaneous frequency-scale by α. As illustrated in fig. 1, this means that when a signal is scaled along the time axis by a certain factor, its frequency-domain characteristics (in particular pitch and formant frequencies) are scaled by the inverse of that factor. Thus, such a scaled signal would be perceived as a time-scaled and frequency-shifted version of the original, and this does not correspond to our intuitive expectation related to hearing a time-scaled version of an original acoustic signal. This is because it is our most elementary experience that sound has a time pattern as well as a frequency pattern [2] and that these patterns are relatively independent as they are related to the rhythm and the melody, respectively. We therefore require a different type of time-scaling (one that does not affect the frequency structure of the signal) and we could say that we want the time-scaled version of an acoustic signal to be perceived as the same sequence of acoustic events from the original signal being reproduced according to a scaled time pattern. Time-scale modification should modify the time structure of signal $x(n)$ without altering its frequency structure.

Formally, time-scale modifications are specified by defining a mapping $n \rightarrow n' = D(n)$ (*the time-warping function*) between the original time-scale and the modified time-scale. Often, it is convenient to specify a continuous time-varying *time-*

modification rate $\beta(t)$ from which the time-warping function can be derived as

$$n \rightarrow n' = D(n) = \frac{1}{T}\int_0^{nT} \beta(\tau)\mathrm{d}\tau, \tag{2.10}$$

where T represents the sampling period and $\beta(t) > 0\ \forall\ t$ corresponds to a strictly monotonic warping function. If $\beta(t) = \beta$ is constant, the warping function is linear.

Time-scaling with $\beta(t) > 1$ corresponds to decreasing the apparent articulation rate; with $0 < \beta(t) < 1$ to increasing it. Generally, the time-warping function specifies that *speech events* that occurred at time n in the original signal should take place at time $n' = D(n)$ in the time-scaled signal. With reference to the voiced speech model introduced above, speech parameters should be transformed as follows.

$$\begin{aligned}
&P'(n') = P(D^{-1}(n')) \\
&A'_k(n') = G'(n', \omega_k(n')) = G(D^{-1}(n'), \omega_k(D^{-1}(n'))) \qquad 0 \leq k \leq P'(n') - 1 \\
&\theta'_k(n') = \Phi'_k(n') + \psi\left(D^{-1}(n'), \frac{2\pi k}{P(D^{-1}(n'))}\right) \\
&\Phi'_k(n') = \sum_{m=0}^{n'} \frac{2\pi k}{P(D^{-1}(m))},
\end{aligned} \tag{2.11}$$

where $D^{-1}(.)$ denotes the inverse mapping. These equations express that *i)* the pitch contour of the modified signal is a time-scaled version of the original pitch contour, *ii)* the system function is a time-scaled version of the original system function, *iii)* the instantaneous frequencies at time n' are the instantaneous frequencies of the original signal at the corresponding time $D^{-1}(n')$.

2.3. Pitch-scale modification

In analogy to time-scale modification we can say that the object of pitch-scale modification is to alter the fundamental frequency of speech without affecting the time-varying spectral envelope. Specifying a pitch-scale transformation amounts to defining a time-varying pitch-modification factor $\alpha(n) > 0$ that transforms the pitch contour $P(n)$ into

$$P'(n) = \frac{P(n)}{\alpha(n)}. \tag{2.12}$$

When the pitch-transformation factor $\alpha(n) > 1$ the local pitch-frequency is increased (the pitch-period is shortened); when $\alpha(n) < 1$ the pitch-frequency is decreased. With reference to the speech model, the parameters should be modified according to:

$$P'(n) = P(n)/\alpha(n)$$

$$A'_k(n) = G(n, \alpha(n)\omega_k(n))$$
$$\theta'_k(n) = \Phi'_k(n) + \psi(n, \alpha(n)\omega_k(n))$$
$$\Phi'_k(n) = \sum_{m=0}^{n} \alpha(m)\omega_k(m). \tag{2.13}$$

Note that the amplitudes of the pitch-modified harmonics are samples of the system amplitude function taken at the modified harmonic frequencies. Thus, as opposed to time-scale modification, pitch-scale modification requires the estimation of system amplitudes $G(n, \alpha(n)\omega_k(n))$ and phases $\psi(n, \alpha(n)\omega_k(n))$ at frequencies $\alpha(n)\omega_k(n)$ that are not necessarily pitch-harmonic frequencies in the original signal. Therefore, most pitch-scale modification algorithms explicitly decompose the speech signal in an excitation signal with flat spectral envelope and a time-varying spectral envelope (numerous techniques have been proposed for this purpose. see for example chapter 1).

2.4. Possible approaches to prosodic modification

In the case of speech signals, we have a choice of synthesis models that can be used to produce speech from a set of production parameters. One general approach for time-scaling and/or pitch-scaling of speech could consist of first analyzing the original speech signal to obtain these production parameters, then applying the desired transformation to the production parameters, and synthesizing the corresponding signal. In selecting such an analysis-synthesis model a tradeoff would have to be made between computational complexity and speech quality and it is likely that it will be hard to strike a good compromise with this parametric type of approach. In general, the power and the weakness of a model lies in representing the signal in a compact and simplified way. This gives attractive prospects for speech coding and speech recognition and for speech synthesis by rule. For prosodic modification, however, simplification of the rich acoustic detail from the original speech waveform rapidly leads to a perceived distortion.

In this chapter, we concentrate on non-parametric approaches for prosodic modification. Since sounds are perceived to have frequency-domain features like pitch and timbre that evolve over time, non-parametric approaches make use of a time-frequency representation in which the perceived sound attributes at a given time instant should ideally be represented along the frequency dimension.

3. The short-time Fourier transform and overlap-add synthesis

In search for a good time-frequency representation for prosodic manipulation of speech, we are looking for an analysis method that will reflect how the perceptually important frequency-domain characteristics of the speech signal $x(n)$ evolve over time. If we consider speech as a signal with slowly evolving frequency-domain characteristics (i.e., as a quasi-stationary signal), we can apply a short-time analysis

strategy together with Fourier transformation to obtain the so-called short-time Fourier transform (STFT) as the desired time-frequency representation.

The short-time Fourier transform (STFT) is a basic tool in speech analysis, which have been used for speech synthesis and modifications for many years. Its theory is well understood and efficient implementations are available, based on the FFT algorithm and simple modifications of overlap-and-add synthesis methods [3–6] (a good comprehensive treatment of this concept can be found in [7]). The basic idea is to use a windowing function $w(n)$ to restrict the analysis to short segments of $x(n)$ around the analysis time instants in such a way that $x(n)$ can be considered to have fixed characteristics inside each individual segment. Standard tools for stationary signal analysis, such as the Fourier transform, can then be applied to the individual segments. A conceptual disadvantage of this approach towards time-varying analysis is that the analysis precision will be limited by the windowing operation and non-stationarity; a practical advantage is that short-time analysis works with consecutive, possibly overlapping, signal segments and is easily amenable to on-line processing.

In this section, we present the basic properties of STFT analysis/synthesis techniques that we shall need to understand the speech transformation procedures describe in this paper. The presentation that is given in this chapter is slightly more general than the traditional derivations: in particular, we allow variable analysis and synthesis rates, which prove to be useful in time-scale as well as pitch-scale modifications of speech.

3.1. Analysis

The short-time Fourier transform can be viewed as a way of representing the speech signal in a joint time and frequency domain. As outlined above, it consists of performing a Fourier transform on a limited portion of the signal, then shifting to another portion of the signal and repeating the operation. The signal is then represented by the discrete Fourier coefficients (or, equivalently, the windowed samples) associated with the different window locations.

The successive window locations $t_a(u)$ are referred to as *analysis time instants* (implicitly, these time instants are integer, i.e. correspond to an integer number of samples). Most often, the STFT analysis is performed at a constant rate, i.e. $t_a(u) = uR$. For time-scale and pitch-scale transformation, a non-uniform analysis rate is sometimes more convenient (this is the main point behind the pitch-synchronous methods such as the WSOLA method for time-scaling and the PSOLA method for pitch-scaling, see below). More formally, denote by $x(n)$ the samples of the speech signal and let $h_u(n)$ be the *analysis window*. It is assumed in the sequel that $h_u(n)$ *i)* is centered around the time instant 0, *iii)* is of finite duration T_u and symmetric, and *iii)* is the impulse response of a low-pass filter. The *analysis short-time signal* $x(t_a(u), n)$ associated to the analysis time instant $t_a(u)$ is defined as the product

of the speech waveform and the analysis window centered at $t_a(u)$:

$$x(t_a(u), n) = h_u(n)x(t_a(u) + n).$$

This analysis short-time signal plays a key role in the WSOLA algorithm and the PSOLA algorithm. When needed, we may associate to this short-time analysis signal a short-time analysis spectrum, denoted by $X(t_a(u), \omega)$ and defined as the discrete Fourier transform of $x(t_a(u), n)$

$$X(t_a(u), \omega) = \sum_{n=-\infty}^{\infty} h_u(n)x(t_a(u) + n)\exp(-j\omega n).$$

In many applications of the STFT analysis-synthesis method, the analysis window is a fixed window function, i.e. $h_u(n) = h(n)$. In the more general framework used here, the analysis window may depend

3.2. Modification

The modifications that are to be made to the STFT reflect the transformations we plan to apply to the signal. For speech enhancement or time-varying filtering, the transformation generally consists in multiplying the STFT with another time-frequency function selecting some frequencies of interest or attenuating undesirable disturbances). For pitch-scaling or frequency-shifting, transformations may consist in a frequency-axis compression or expansion, to modify the spacing between the pitch harmonics (see below).

The modification stage consists of the following two steps:

- modify the short-time analysis spectra $X(t_a(u), \omega)$ to produce a stream of *short-time synthesis spectra* $Y(t_s(u), \omega)$,
- synchronize these short-time synthesis spectra $Y(t_s(u), \omega)$ on a new set of time instants, referred to as the *synthesis time instants* and denoted $t_s(u)$.

Note that the first step is simply by-passed in the WSOLA algorithm or in the time-domain PSOLA algorithm.

The stream of synthesis time instants $t_s(u)$ is determined from the stream of analysis time instants according to the desired pitch-scale and time-scale modifications. The number of synthesis time instants needs not be identical to the number of analysis time instants. For non-constant pitch-scale and time-scale modification factors, the synthesis time instants will be generally irregularly spaced, whether or not the analysis rate is constant.

3.3. Synthesis

The last step consists of combining the stream of synthesis short-time signals synchronized on the synthesis time instants to obtain the desired 'modified' signal.

The main difficulty is that, modifying $X(t_a(u),\omega)$ (to achieve the desired prosodic transformation) the result may no longer represent a valid stream STFT in that *a signal which has the modified transform* $Y(t_s(u),\omega)$ *as its STFT may not exist.* Still $Y(t_s(u),\omega)$ would contain the information which best characterizes the signal modification we had in mind, such that a special synthesis formula is required which leads to the correct result if $Y(t_s(u),\omega)$ is a STFT and to a reasonable result otherwise. One such synthesis method uses overlap-addition (OLA). As introduced by Griffin and Lim [8], this procedure consists of seeking the synthetic signal $y(n)$ whose short-time Fourier transform (around time instants $t_s(u)$)

$$\hat{Y}(t_s(u),\omega) = \sum_m f_u(m)y(t_s(u)+m)\exp(-j\omega m)$$

best fits the modified synthesis short-time Fourier transform $Y(t_s(u),\omega)$, in the least-squares sense

$$\sum_u \int_{-\pi}^{+\pi} |Y(t_s(u),\omega) - \hat{Y}(t_s(u),\omega)|^2 d\omega,$$

where the sum is over all time instants $t_s(u)$ for which $\hat{Y}(t_s(u),\omega)$ is defined. $f_u(n)$, which is used in the definition of the short-time spectrum (eq. (3.3)), is referred to as the *synthesis window.* It is usually directly deduced from the analysis window, reflecting the modifications brought to the synthesis short-time spectrum $Y(t_s(u),\omega)$, on which $\hat{Y}(t_s(u),\omega)$ is fitted. The least-squares problem eq. (3.4) can be solved explicitly, by resorting to the Parseval formula; the solution is given in closed form by the following *synthesis formula*

$$y(n) = \frac{\sum_u y_w(u, n-t_s(u))f_u(n-t_s(u))}{\sum_u f_u^2(n-t_s(u))}$$

$$y_w(u,n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} Y(t_s(u),\omega)\exp(j\omega n)d\omega.$$

The synthesis algorithm is similar to weighted overlap-add; the successive short-time synthesis signals are combined, with appropriate weights and time-shifts. The denominator plays the role of a time-varying normalization factor, which compensates for the energy modifications resulting from the variable overlap between the successive windows. The synthesis operation can be simplified if the synthesis windows $f_u(n)$ and the synthesis time instants $t_s(u)$ can be chosen such that

$$\sum_u f_u^2(n-t_s(u)) = 1.$$

For constant synthesis rate $t_s(u) = uR'$ and fixed synthesis window $f_u(n) = f(n)$, a common choice that satisfies this simplifying condition is a Hanning window with 50% overlap between successive segments; some other possibilities are listed in [8].

To see how this OLA synthesis operates, consider the operations of STFT and inverse STFT of a signal $x(n)$ using eqs. (3.1), (3.2), and (3.5) with a fixed analysis window $h_u(n) = h(n)$ of length L and a constant analysis rate $t_a(u) = uR$, where $R < L$ (so that two successive windows overlap).

- To compute $x(uR, n)$, the signal is advanced R points in time and windowed to obtain $x(uR, n) = h(n)x(n + uR)$ (eq. (3.1)). The corresponding short-time analysis spectrum $X(uR, \omega)$ is then obtained by taking the discrete Fourier transform of $x(uR, n)$ towards n (eq. (3.2)).

  To obtain the inverse STFT, the inverse Fourier transform of $X(uR, \omega)$ is computed to recover the windowed segments $x(uR.n)$. This result is windowed again using the synthesis window $f_u(n) = h(n)$ to obtain $h(n)x(uR, n)$ (in absence of modifications, it makes good sense to take the same analysis and synthesis windows; this is no longer the case when considering pitch-scale transformations). As all these segments were positioned around time origin during the analysis, they now have to be delayed to move each one back to its original location along the time axis (i.e., around time uR for segment number u). The result is then obtained by summing all these segments and dividing by a time-varying normalization weight (eq. (3.5)):

$$y(n) = \frac{\sum_u h(n - uR)x(uR.n)}{\sum_k h^2(n - uR)} = \frac{\sum_k h^2(n - uR)x(n)}{\sum_k h^2(n - uR)} = x(n).$$

Thus, the OLA[2] synthesis formula reconstructs the original signal if the analysis short-time spectrum $X(t_a(u).\omega)$ is a valid STFT (as expected, in absence of modification the signal is reconstructed without any alteration). It constructs a signal whose STFT is maximally close to $X(t_a(u), \omega)$ in least-squares sense otherwise.

In several applications, it is required to reconstruct the speech signal from the stream of short-time Fourier transform magnitude functions $|Y(t_s(u), \omega)|$ (see below an application). The least-squares method presented above can also be used (with slight modifications) to this purpose. As previously, the basic idea consists in seeking the synthetic signal $y(n)$ whose short-time Fourier-transform magnitude (around time instants $t_s(u)$) best fits in the mean-square sense the desired time-frequency function $|Y(t_s(u), \omega)|$; in other words, we wish to minimize the following distance:

$$\sum_u \int_{-\pi}^{\pi} \left(|Y(t_s(u), \omega)| - |\hat{Y}(t_s(u), \omega)| \right)^2 d\omega \tag{3.7}$$

where $\hat{Y}(t_s(u), \omega)$ is defined in eq. (3.3). The solution is found iteratively. The iteration takes place as follows. An arbitrary sequence is selected as the first estimate $y^1(n)$ of $y(n)$. We then compute the STFT of $y^1(n)$ and modify it by replacing its

[2] The formula was initially called LSEE MSTFT, which stands for least-squares error estimation from modified STFT [8] but is now usually referred to as (a variant of) the overlap-add (OLA) method.

magnitude by the desired magnitude $|Y(t_s(u),\omega)|$. From the resulting modified STFT, we can obtain a signal estimate using the LS-OLA method outlined above. This process continues iteratively: the $(i+1)$st estimate $y^{i+1}(n)$ is first obtained by computing the STFT of $y^i(n)$ and replacing its magnitude by $|Y(t_s(u),\omega)|$ to obtain $Y^i(t_s(u),\omega)$. The signal with the STFT closest to $Y^i(t_s(u),\omega)$ is found by LS-OLA eq. (3.5). All steps in the iteration can be summarized in the following update equation:

$$y^{i+1}(n) = \frac{\sum_u y^i_w(u, n - t_s(u)) f_u(n - t_s(u))}{\sum_u f(n - t_s(u))^2}, \tag{3.8}$$

$$y^i_w(u,n) = \frac{1}{2\pi}\int_{-\pi}^{+\pi} |Y(t_s(u),\omega)| \frac{Y^i(t_s(u),\omega)}{|Y^i(t_s(u),\omega)|} \exp(j\omega n) d\omega.$$

It has been shown that this iterative procedure reduces the distance eq. (3.7) on every iteration. Furthermore, the process converges to one of the critical point, not necessarily the global minimum of the distance measure. Typically, the convergence is rather slow (using an arbitrary initialization, 100 iterations are necessary to obtain reasonable results). Each iteration step is computationally intensive; it needs to modify the STFT along the frequency dimension (by modifying phase), thus requiring at least one FFT per iteration. This of course precludes the use of such algorithms for real-time processing, even on the fastest hardware.

4. Time-scaling techniques

4.1. OLA time-scaling

It can be noted that with OLA synthesis we are close to realizing time-scale modifications using time-domain operations only. In fact we can see that, by adopting a short-time analysis strategy for constructing $X(t_a(u),\omega)$ and by using the OLA criterion for synthesizing a signal $y(n)$ from the modified representation $Y(t_s(u),\omega) = M_{xy}(X(t_a(u),\omega))$, we will always obtain modification algorithms that can be operated in the time domain if the modification operator $M_{xy}[.]$ works only:

$$\begin{aligned}
Y(t_s(u),\omega) &= X(D^{-1}(t_s(u)),\omega) && \textit{(modification)}\\
y(t_s(u),n) &= x(D^{-1}(t_s(u)),n) && \textit{(inverse Fourier Transform)}\\
y(n) &= \frac{\sum_u f_u(n - t_s(u)) x(D^{-1}(ms),n)}{\sum_u f_u^2(n - t_s(u))} && \textit{(OLA synthesis)}
\end{aligned}$$

In that case we see from the last equation above that the modification is obtained by excising segments $x(D^{-1}(t_s(u)),n)$ from the input signal and repositioning them along the time axis before constructing the output signal by weighted overlap-addition of the segments. However, as illustrated in fig. 2, if we hurry to apply the above formula for realizing a time warp $D(t_a(u))$, poor results will generally be

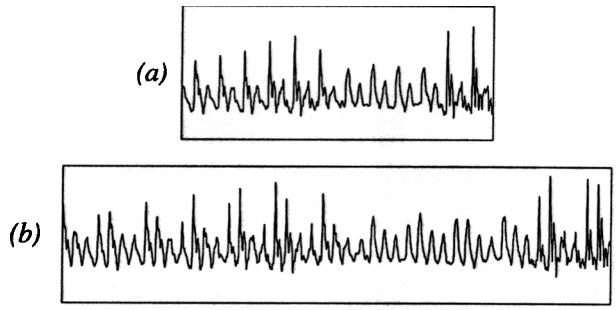


Figure 2. OLA synthesis from the time-scaled STFT does not succeed to replicate the quasi-periodic structure of the signal *(a)* in its output *(b)*.

obtained when using $Y(t_s(u),\omega) = X(D^{-1}(t_s(u)),\omega)$. Short-time analysis causes these problems by constructing a two-dimensional representation $x(t_a(u),m) = h_u(n)x(n + t_a(u))$ in which the two time scales are not independent such that important information about the time structure of the signal $x(n)$ is represented both in ω and in $t_a(u)$. Consider, for example, the case of a periodical signal $x(n) = x(n + P)$. If the window is sufficiently long, each segment $x(t_a(u),m)$ contains several periods of the original. At the same time, this periodic structure also exists among different segments $x(t_a(u)+P,n) = x(t_a(u),n)$ (all segments separated by a multiple of the period P are identical). By arbitrarily repositioning these segments, as in $Y(t_s(u),\omega) = X(D^{-1}(t_s(u)),\omega)$, we destroy the relationship between the time structure inside the segments and the time structure across the segments. In the example shown in fig. 2 this leads to the quasi-periodic structure of the input speech signal *(a)* not being preserved in the time-scaled output *(b)*. (In this example the attempted time-scaling consisted of a reduction of the apparent speaking rate to 60% of the original.)

The phase component $\Gamma(t_a(u),\omega)$ of the complex STFT $X(t_a(u),\omega)$ carries information about the signal's time structure inside the analysis window. A better separation with information on perceptual characteristics in ω and time structural information in $t_a(u)$ is found in the spectrogram $|X(t_a(u),\omega)|$ where each magnitude spectrum shows pitch information in its harmonic structure and formant information in its spectral envelope. Because it contains no phase information, a magnitude spectrum does not specify the precise time structure of the signal segment $x(t_a(u),n)$ nor does it carry information concerning the position of the signal $x(n)$ relative to the window $h_u(n)$. If a sufficiently long window is used (several pitch

periods long), it was shown in [8] that reasonable quality time-scaled signals can be obtained from the time-scaled spectrogram $|Y(t_s(u),\omega)| = |X(D^{-1}(t_s(u)),\omega)|$ by using the iterative magnitude-only least-squares OLA reconstruction (eq. (3.8)).

4.2. Synchronized OLA time-scaling

Since the magnitude only OLA is an iterative procedure that slowly converges to a local optimum it becomes important that a good initial estimate $\Gamma_0(t_a(u),\omega)$ or, equivalently, a good choice for $y^1(n)$ can be proposed. Roucos and Wilgus [9] experimentally studied the convergence of OLA time-scaling and found that for initial estimates like Gaussian white noise 100 iterations were typically required to obtain high quality results. In their effort to find a better initial estimate that would significantly reduce the required number of iterations, Roucos and Wilgus proposed a construction method for a $y^1(n)$ that has by itself already such high quality that further iterations are no longer needed and do not in fact improve the subjective speech quality [9]. This algorithm for time-scaling is called the synchronized overlap-add method (SOLA) and can be described as follows.

It is assumed in this section that the same window $w(n)$ is used at the analysis and at the synthesis stage and that the synthesis time instants $t_s(u) = uR$ are regularly spaced. As discussed in section 4.1, straightforward OLA synthesis from the time-scaled and downsampled STFT $Y(uR,\omega) = X\left(D^{-1}(uR),\omega\right)$ results in a signal

$$y^1(n) = \frac{\sum_u w^2(n-uR)x\left(n-uR+D^{-1}(uR)\right)}{\sum_u w^2(n-uR)}$$

that is heavily distorted, as we illustrated in fig. 2. Interpreted as a possible initial estimate for iterative OLA time-scaling, $y^1(n)$ corresponds to an initial phase $\Gamma_0(uR,\omega)$ that is chosen equal to the actual phase of the individual segments $x\left(D^{-1}(uR),n\right)$. This choice would certainly be fine for the individual segments but, as we discussed in section 4.1, repositioning the segments from their original time position $D^{-1}(uR)$ to the required synthesis time position uR destroys the original phase relations across segments. Roucos and Wilgus noted that this repositioning of segments corresponds to the introduction of a linear phase difference between the individual segments that disrupts the periodical structure of the voiced parts of speech, unless the difference would be equal to a multiple of the pitch period. With their SOLA algorithm [9] they propose to avoid pitch period discontinuities at waveform segment boundaries by realigning each input segment to the already formed portion of the output signal before performing the OLA operation. Thus, SOLA constructs the time-scale modified signal

$$y(n) = \frac{\sum_u v(n-uR+\Delta_u)x(n+D^{-1}(uR)-uR+\Delta_u)}{\sum_u v(n-uR+\Delta_u)}$$

in a left-to-right fashion with a windowing function $v(n)$, and with shift factors Δ_u

that are chosen such as to maximize the cross-correlation coefficient between the current segment $v(n-uR+\Delta_u)x(n+D^{-1}(uR)-uR+\Delta_u)$ and the already formed portion of the output signal

$$y(n;k-1)=\frac{\sum_{l=-\infty}^{u-1} v(n-lR+\Delta_l).x(n+D^{-1}(lR)-lR+\Delta_l)}{\sum_{l=-\infty}^{u-1} v(n-lR+\Delta_l)}$$

SOLA is computationally efficient since it requires no iterations and can be operated in the time domain. As discussed earlier, time-domain operation implies that the corresponding STFT modification affects the time axis only. In case of SOLA, we have

$$Y(uR-\Delta_u,\omega)=X\left(D^{-1}(uR),\omega\right) \tag{4.4}$$

The shift parameters Δ_u thus imply a tolerance on the time-warping function: in order to ensure a synchronized overlap-addition of segments, the desired time-warping function $D(n)$ will not be realized exactly. A deviation on the order of a pitch period should be allowed.

Several alternative synchronization methods can be constructed. In the next section we describe in some detail a particularly efficient synchronized OLA algorithm, called WSOLA (waveform-similarity based overlap-add).

4.3. WSOLA: An overlap-add technique based on waveform similarity

4.3.1. Efficient synchronized OLA time-scaling

From the above discussions we observe that, in order to construct an efficient high-quality time-scaling algorithm based on OLA, a tolerance Δ_u on the precise time-warping function that will be realized is needed to allow a synchronized overlap-addition of original input segments to be performed in the time domain. This tolerance can be used like in SOLA to realize segment synchronization during synthesis $Y(uR-\Delta_u,\omega)=X\left(D^{-1}(uR),\omega\right)$. However, as the Δ_u are not known beforehand, the denominator in the OLA formula (4.2) could not be made constant in that case. A further reduction of computational cost would be possible by using fixed synthesis time instants $t_s(u)=uR$ and a window $v(n)$ such that $\sum_u v(n-uR)=1$. Proper synchronization must then be ensured during the segmentation

$$Y(uR,\omega)=X(D^{-1}(uR)+\Delta_u,\omega).$$

Thus it would seem that a most efficient realization of OLA time-scaling would use the simplified synthesis equation

$$y(n)=\sum v(n-uR)x(n+D^{-1}(uR)-uR+\Delta_u), \tag{4.6}$$

where Δ_u are chosen such as to ensure sufficient signal continuity at waveform segment boundaries according to some criterion. WSOLA [10] proposes a synchronization strategy inspired on a time-scaling criterion.

4.3.2. A waveform similarity criterion for time-scaling

We considered that a time-scaled version of an original signal should be perceived to consist of the same acoustic events as the original signal but with these events being produced according to a modified timing structure. In WSOLA we assume that this can be achieved by constructing a synthetic waveform $y(n)$ that maintains maximal local similarity to the original waveform $x(m)$ in all neighborhoods of related sample indices $m = D^{-1}(n)$. Using the symbol $(=)$ to denote maximal similarity and using the window $w(n)$ to select such neighborhoods, we require

$$\forall m : y(n+m)w(n) \ (=) \ x(n + D^{-1}(m) + \Delta_m)w(n),$$

or equivalently

$$\forall m : Y(m,\omega) \ (=) \ X(D^{-1}(m) + \Delta_m, \omega).$$

Comparing eqs. (4.7) and (4.8) with eq. (4.5), we find an alternative interpretation for the timing tolerance parameters Δ_u as we see that the waveform similarity criterion and the synchronization problem are closely related. As illustrated in fig. 3, the Δ_m in eqs. (4.7) and (4.8) were introduced because in order to obtain a meaningful formulation of the waveform similarity criterion, two signals need to be considered identical if they only differ by a small time-offset[3]. Referring to fig. 3, we need to express that the waveform shapes of segments from the quasi-periodic signal in the middle of the figure are similar at all time instants. Such similarity goes unnoticed in the upper pair of segments because they are located at different positions in their respective pitch cycles. By introducing a tolerance Δ_m on the time instants around which segment waveforms are to be compared, the quasi-stationarity of the signal can easily be detected from the lower pair of segments that was synchronized by letting $\Delta_m = \Delta$. Thus, usage of the requirement of waveform similarity between input and output signals as a criterion for time scaling implies that a synchronization of input and output segments must take place.

As eq. (4.5) can be viewed as a downsampled version of eq. (4.8), we propose to select the parameters Δ_u such that the resulting time-scaled signal

$$y(n) = \sum_u v(n - uR)x(n + D^{-1}(uR) - uR + \Delta_u)$$

maintains maximal local similarity to the original waveform $x(m)$ in corresponding neighborhoods of related sample indices $m = D^{-1}(n)$.

[3] It can be noted that waveform similarity was used to approximate sound similarity. Because two signals that differ only by some time offset sound the same, we also need to declare their waveforms to be similar.

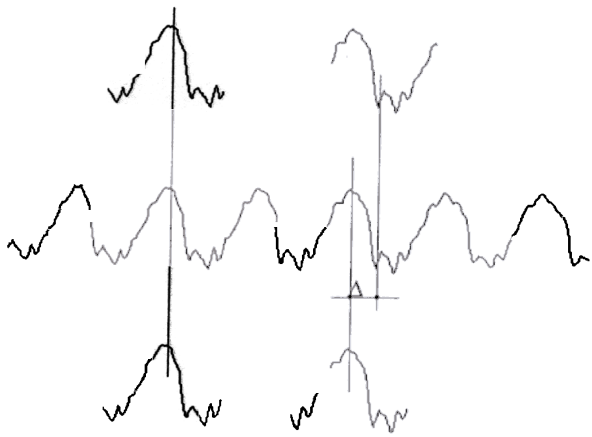


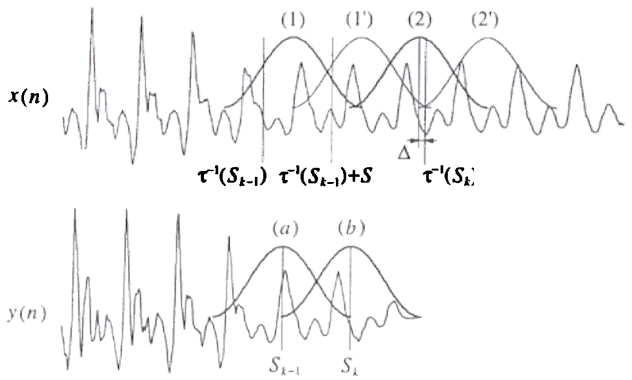
Figure 3. Alternative interpretation of timing tolerance parameters Δ.

4.3.3. WSOLA algorithms

Based on this idea, a variety of practical implementations can be constructed. A common version of WSOLA uses a 20 ms Hanning window with 50% overlap ($R = 10f_s$ with f_s the sampling frequency in kHz) to construct the signal of eq. (4.9) in a left-to-right manner as illustrated in fig. 4 as follows.

Assume the segment labeled (1) in fig. 4 was the previous segment that was excised from the input signal and overlap-added to the output at time instant S_{k-1}, i.e., synthesis segment (a) = input segment (1). At the next synthesis position S_k we need to chose a synthesis segment (b) that is to be excised from the input around a time instant $D^{-1}(S_k) + \Delta_k$, where $\Delta_k \in [-\Delta_{max} \ldots \Delta_{max}]$ is to be chosen such that the resulting portion of $y(n), n = S_{k-1} \ldots S_k$ will be similar to a corresponding portion of the input. As segment (1') overlap-adds with (1) to reconstruct a portion of the original signal $x(n)$, this segment (1') would also overlap-add with segment (a) to reconstruct that same portion of the original in the output signal $y(n)$ (remember that segment (a) = segment (1)). While we can not accept segment (1') as a legal candidate for synthesis segment (b) if it does not lie in the timing tolerance region $[-\Delta_{max} \ldots \Delta_{max}]$ around $D^{-1}(S_k)$, we can always use it as a template to select segment (b) such that it resembles segment (1') as closely as possible and is located within the prescribed tolerance interval around $D^{-1}(S_k)$ in the input signal. The position of this best segment (2) can be found by maximizing a similarity measure between the sample sequence underlying segment (1') and the input signal. After overlap-addition of synthesis segment (b) = input segment (2) to the output we can proceed to the next synthesis time instant using segment (2') as our next template.

WSOLA proposed the criterion of waveform similarity as a substitute for the time-scaling criterion which required that at all corresponding time instants the



'igure 4. Illustration of WSOLA time-scaling.

original and the time-scaled signal should sound similar. Clearly waveform similarity can only be a valid substitute for sound similarity if the similarity can be made sufficiently close. For time-scaling of speech signals, which are largely made up from long stretches of quasi-periodic waveforms and noiselike waveform shapes, it is not unreasonable to believe that a strategy like WSOLA will be able to produce close waveform similarity. In that case, the precise similarity measure selected should not matter too much in that any reasonable distance measure (like cross-correlation, cross-AMDF, etc.) should suffice. As illustrated in figs. 6 and 5, original and WSOLA time-scaled waveforms do indeed show a very close similarity[4]. Also, many variants of the basic technique can be constructed by varying the window function, the similarity measure, the portion of the original $x(n)$ that is to serve as a reference for natural signal continuity across OLA segment boundaries, etc. As many such variants all provide a similar high quality [10], this design flexibility can be used to optimize further the algorithm's implementation for a given target system.

4.3.4. Properties

As a common feature of synchronized OLA algorithms we found that they operate in the time domain by performing a short-time analysis (i.e., a segmentation) and scaling only one of the two time dimensions. They avoid the need for actual frequency-domain computations by synchronization of the segments that are used

[4] As we used the same input signals and the same time-scaling factors for figs. 6 and 2 and for figs. 5 and 1, the reader might be interested in comparing these sets of figs. to get an idea about the effectiveness of the proposed solution.

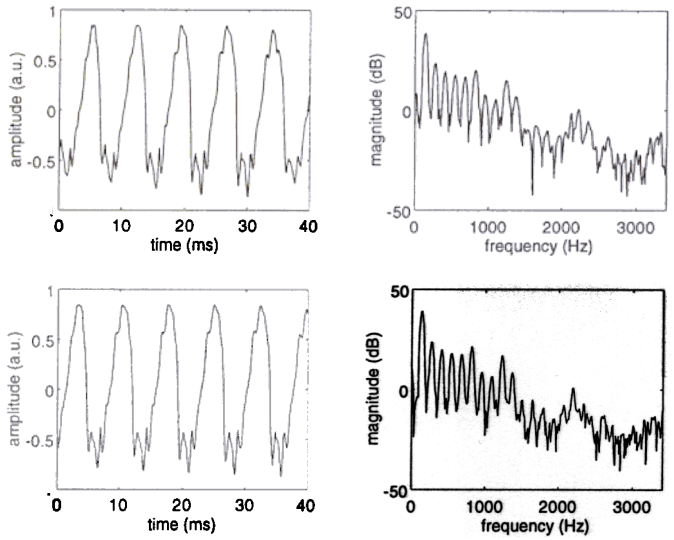


Figure 5. Frequency-domain effects of WSOLA time-scaling. The upper row shows a 40 ms voiced speech frame and its spectrum; the second row illustrates that when this signal is played at half speed using WSOLA no frequency shifting occurs.

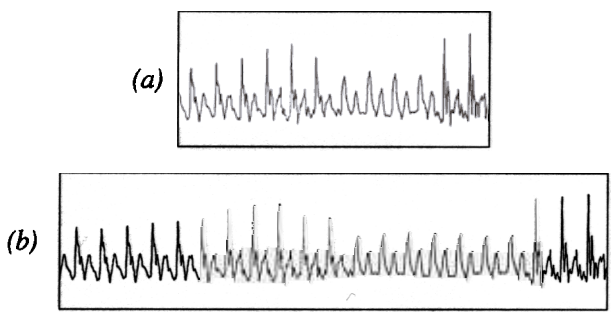


Figure 6. Illustration of an original speech fragment (a) and the corresponding WSOLA output waveform when slowed down to 60% speed (b).

in OLA. Besides high computational efficiency, these methods provide high quality time-scaling because they work with original segments of the input signal. These segments contain rich acoustic detail that would not be easily reproduced with purely model based approaches. Synchronized OLA methods are very flexible in that arbitrary time-warping functions $D(n)$ can be realized (the relationship between corresponding input and output time instants needs not be a linear one, nor even a monotonic one for that matter). Further, these methods are inherently robust since no assumption is made concerning the nature of the signal to be time-scaled (for example, no measures related to a speech-production model are made and modified in this time-scaling). In fact, some useful results have even been shown in experiments on WSOLA time-scaling for digital audio signals [11, 12].

As synchronized OLA algorithms are easily interpreted as automatic waveform-editing methods, we can also easily see that they do have limitations to their capabilities. For one, the structure of signals that can be processed with good success must remain sufficiently simple (synchronization of segments can become a problem if the acoustic waveform is too complex [11]). Fortunately, this requirement is normally satisfied for speech.

Speech is also in another respect an especially well-suited type of signal for being processed in this way. This can best be seen by considering the short-time analysis approach that was used to approximate a time-frequency representation. We already noted in section 4.1 that the time scales inside segments and across segments are not truly independent if both the amplitude and the phase (i.e., the exact waveform shape) of the segments are considered. If synchronization between successive segments succeeds, it is clear that perceptual structures (i.e., features like pitch, timbre, or vibrato that are perceived as frequency-domain objects) will not be distorted if their characteristic period is shorter than the effective length of the segmentation window since no time-scaling takes place inside individual segments. Conversely, if the characteristic period of such structures as vibrato spans several segmentation windows, these structures will be frequency shifted since the distance in time between segments is scaled. Thus the segmentation window should be sufficiently long. On the other hand, if the window length is too long compared to the structure's characteristic period, time resolution will be low and a very coarse approximation to the desired time-warping can result. For general audio signals, this can be a real problem since important structures like vibrato or tremolo can have very low characteristic frequencies, requiring very long segments, while at the same time other structures with high characteristic frequencies can be present whose time patterns evolve very quickly, thus requiring very short segments [11]. Again, for speech signals this problem is not serious since it is suffices to choose the effective window length to be at least one pitch period in order to avoid this type of problems.

Although the quality of processed speech is indeed very high, a slight reverberance can occur when speech is slowed-down by a significant amount. This too is easily explained using the waveform editing interpretation. When slowing down, more samples are needed to construct the output signal than are available in the input signal. Therefore some portions of the input will occur in more than one output

segment and this can cause a reverberation-like effect that will be most noticeable in unvoiced portions of the signal. A possible improvement could be obtained if the segments that are used more than once in the output signal are time reversed whenever they are repeated [13]. This helps to break up the repetitive structure and corresponds to sign-inverting the phase, which is allowed in unvoiced speech but can not be used for voiced segments.

5. Pitch-scaling transformations

Pitch-scaling transformations are generally more difficult to design than time-scaling transformations. As outlined in section 2, pitch-scale modifications require the estimation of the system amplitudes $G(n', \alpha(n')\omega_k(n'))$ and phases $\psi(n', \alpha(n')\omega_k(n'))$, at frequencies not necessarily corresponding to a pitch harmonic in the original signal. This means that it necessitates the extrapolation of information contained in the speech waveform: more specifically, we have to guess the values of the system transfer function at frequencies which have not been observed, because they have not been excited in the original signal. As a result, direct signal editing will not suffice for producing the desired transformation, contrasting with the time-scaling transformation.

To satisfy these requirements, a method aiming at transforming the pitch-scale needs to identify some parts of the production model outlined in section 2. Such an algorithm, will typically operates in several steps, which are described below:

- extract (around each analysis time instant $t_a(u)$) a spectral envelope, modeling the combined transfer function of the supra-glottal cavities and of the glottis. This spectral envelope (roughly speaking, the formant structure), should not be affected by the pitch-scale transformation.
- using this spectral envelope, compute the source component, which approximates the excitation signal $e(n)$ eq. (2.2) around each analysis time instant $t_a(u)$. This source signal carries the informations related to the pitch: it should be modified during the pitch scale transformation.
- synthesize a pitch-modified short-time signal by recombining the spectral envelope and the pitch-modified excitation. Synthesize the final signal by the overlap-add procedure eq. (3.5).

Note that, generally, a pitch transformation induces a corresponding time-scale transformation: compensatory time-scale modification is applied in a fourth, and final step, to restore the original signal duration.

The first non-parametric method to modify the pitch scale was proposed by [14] (see also [15, 16] for more recent references). It was derived from the phase vocoder and follows exactly the scheme outlined above. Around each time instant $t_a(u) = uR$ (where R is a small portion of the window length), a short-time analysis spectrum is obtained by means of eqs. 3.1 and 3.2. A spectral envelope is estimated (using cepstrum analysis) and a flattened short-time excitation spectrum is obtained by multiplying the short-time analysis spectrum by the inverse of the spectral enve-

lope. The pitch information carried by the short-time excitation spectrum is then modified by expanding or compressing the frequency axis, in order to alter the spacing between the pitch harmonics by the required factor. Phase adjustments are then carried out to compensate for the transformation (this point is thoroughly investigated in [16]; see also the section on resampling and its application to pitch-scaling in the appendix). The modified source component is then recomposed with the envelope; the pitch-modified signal is finally obtained by (least-squares) OLA synthesis. This system is known to provide 'reasonable' speech of quality. It is our own experience with this system that the resulting speech is never free from artifacts; in particular, the transformation always introduces some sort of 'reverberant quality', which is typically difficult to eliminate, even by finely tuning the parameters of the transformation. It is also computationally intensive: the explicit use of frequency-domain representation necessitates a direct and an inverse FFT for each short-time signal; since the analysis rate R should be chosen much higher than is required by the perfect reconstruction condition (to avoid too much time-domain aliasing), the computational cost typically requires the use of a digital signal processor for real-time transformation (which is needed in many applications, such as text-to-speech).

In the next section, we introduce the PSOLA framework. It also follows the four steps outlined above, but it does it in an implicit way, that is, the source-filter decomposition and the modification are carried out in a single, simple operation. The PSOLA algorithm shares many features with the WSOLA algorithm. It can be seen as an out-growth of this technique for pitch-scale operation.

5.1. The PSOLA analysis-synthesis framework

The PSOLA (pitch-synchronous overlap add) analysis-modification-synthesis method belongs to the general class of STFT analysis-synthesis method presented in section 3. It proceeds in three steps which are described below.

5.1.1. Analysis

The analysis process consists of decomposing the speech waveform $x(n)$ into a stream of short-time analysis signals, $x(t_a(u), n)$. These short-time signals are obtained by multiplying the signal waveform $x(n)$ by a sequence of time-translated analysis indows (see section 3 above):

$$x(t_a(u), n) = h_u(n)x(n - t_a(u)). \tag{5.1}$$

In the PSOLA context, the analysis time instants $t_a(u)$ are set at a pitch-synchronous rate on the voiced portions of speech and at a constant rate on the unvoiced portions. More specifically, as far as pitch-scaling transformations are concerned, these pitch marks are set (on voiced portions) at the *pitch onset time*, which (loosely) speaking should correspond to the instant of glottal closing (although the

shape of the glottal pulse depends on the type of phonation, the rate of transition from the closed to the open glottis portion is generally slower than that from the open to the closed glottis portion, and thus the main excitation occurs at the instant of glottal closure). For clean speech, reliable estimates of these instants can be obtained, by combining a statistical criterion 'jumping' at the pitch onset time and a pitch detector.

As a statistical criterion, it makes sense to use a measure of the linear dependencies across the successive speech samples; recall that the vocal-tract transfer function can be approximately modeled, within a pitch period, as linear time invariant system. Such system imposes a linear relation on the speech samples when no excitation is present. Thus, when the system parameters are well estimated, the deviation with respect to the linear relationship is small in the closed glottis region and large at the instants of glottal closure. Several works in this direction have been developed recently (see, for example, [17] and the references therein). Another approach consists in using the pointwise Teager operator associated with some kind of band-pass filtering (see [18]).

In either case, the use of a pitch detector as a post-processor is mandatory; it helps to avoid incorrect labeling by maintaining the coherence between the successive detection of the pitch onsets and the local pitch period (this is particularly useful, for example, at the end of a voiced segments or in voiced fricatives or plosives, where the statistical criterion may fail to work). Before concluding this discussion, it should be emphasized that the exact position of the analysis time instant does not play a key role. Modifications of the position of the analysis time instants within the pitch period by a (small) fraction of the pitch period (say 1/10) does not impair the quality of the synthetic speech. On the contrary, it is crucial to maintain an exact pitch synchronicity between the successive pitch marks.

The analysis window is generally chosen to be a symmetrical Hanning window (other windows, like Hamming or Bartlett can be used as well). The window length is chosen to be proportional to the local pitch period $P(s)$, i.e., $T = \mu P(s)$. The proportionality factor μ ranges from 2, for the standard time-domain PSOLA method, to $\mu = 4$, for the frequency-domain implementation. The rationale for using these values is provided later in the discussion.

5.1.2. Modification

it consists of transforming the stream of short-time analysis signals into a stream of short-time synthesis signals, synchronized on a new set of synthesis time instants $t_s(u)$. The synthesis time instants $t_s(u)$ are determined from the analysis time instants $t_a(s)$ according to the desired pitch-scale and time-scale modification. Along with the stream of synthesis time instants, a mapping $t_s(u) \rightarrow t_a(s)$ between he synthesis and the analysis time instants is determined, specifying which short-time analysis signal(s) should be *selected* for any given synthesis time instant.

In the frequency-domain implementation (FD-PSOLA), the short-time synthesis signals are expressed in the frequency-domain, making this scheme computationally less attractive than the standard TD-PSOLA algorithm. Frequency-domain modi-

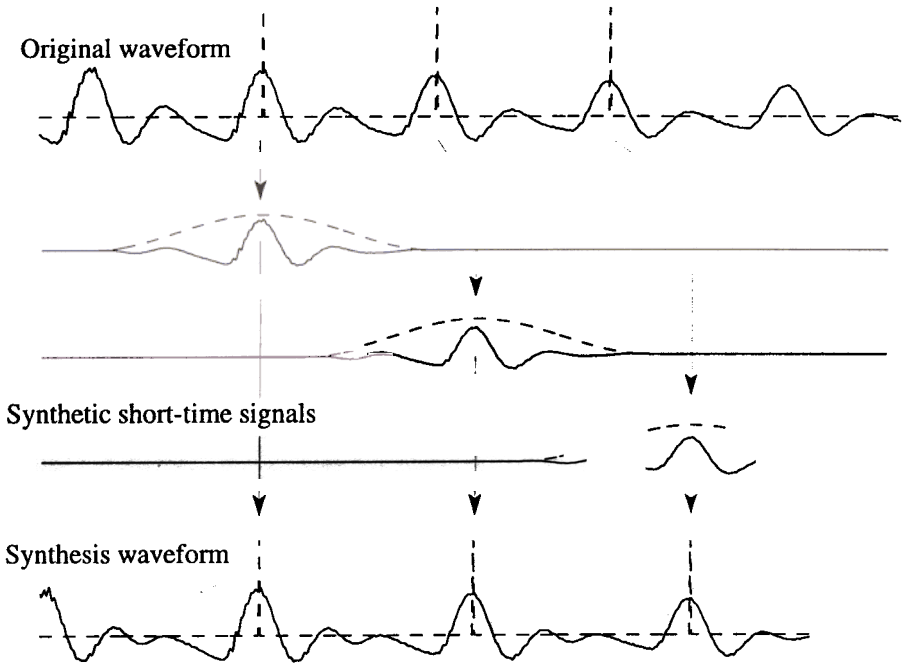


Figure 7. Pitch-scale modification with the TD-PSOLA method. Upper panel: original signal along with the analysis time instants. Middle panel: short-time synthetic signals. Lower panel: pitch-scale modified waveform along with the synthetic time instants. The pitch-scale modification factor is equal to 0.8.

fications closely parallel those proposed in [14], and involves an explicit separation between a spectral envelope component (modeling the vocal-tract transfer function) and a source signal (see section 5.4 for more details).

5.1.3. Synthesis

In the final step, the synthetic signal $y(n)$ is obtained by combining the synthesis waveforms synchronized on the stream of synthesis time instants $t_s(u)$. Least-square overlap-add synthesis procedure eq. (3.5) may be used for this purpose. In the TD-PSOLA algorithm, the synthesis window $f_u(n)$ is equal to the analysis window associated with the analysis time instant $t_a(s)$ mapped with the synthesis tie-instant $t_s(u)$. A different synthesis window must be used in the frequency-domain PSOLA to take into account the inherent change in the time-scale introduced by the modification of the frequency axis (see section 5.4).

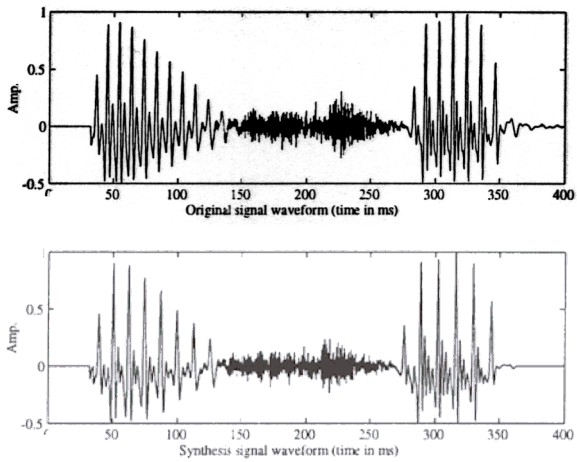


Figure 8. Upper panel: original waveform. Lower panel: pitch-scale modified waveform. The pitch-scale modification factor is 0.8.

5.2. Computation of synthesis time instants

Maybe the only difficulty when implementing a TD-PSOLA pitch transformer is to calculate the synthesis time instants. This does not involve highly abstract mathematical concepts, but rather it is a matter of pure logic. Because the interaction between pitch-scaling and time-scaling transformation factors may be intricate, we have nevertheless chosen a possible solution to handle this problem.

The computation of the synthesis time instants is done in two successive steps. First, the time instants are generated according to the desired pitch-scale and time-scale modification, then each synthesis time instant is associated with one or several analysis time instants.

5.2.1. Pitch-scale modification

Assume for simplicity that the signal is entirely voiced. The analysis time instants $t_a(s)$ are set in a pitch-synchronous way, i.e. $t_a(s+1) - t_a(s) = P(t_a(s))$, in which $P(t)$ is a piecewise-constant pitch-contour function $t \rightarrow P()$,

$$P(t) = P(t_a(s)) \quad t_a(s) \leq t < t_a(s+1). \tag{5.2}$$

The synthesis time instants must also be positioned pitch-synchronously, with respect to the synthesis pitch contour $t \rightarrow P'(t)$. We are left with the problem of finding a series of synthesis pitch marks $t_s(u)$ such that $t_s(u+1) = t_s(u) + P'(t_s(u))$ and $P'(t_s(u))$ is approximately equal to $1/\beta(t_s(u))$ times the pitch in the original

signal around time $t_s(u)$:

$$P'(t_s(u)) \approx \frac{P(t_s(u))}{\beta(t_s(u))}$$

This is easily done recursively: we seek the value of $t_s(u+1)$ that satisfies

$$t_s(u+1) - t_s(u) = \frac{1}{t_s(u+1) - t_s(u)} \int_{t_s(u)}^{t_s(u+1)} \frac{P(t)}{\beta(t)} dt,$$

$$\beta(t) = \beta(t_a(s)) = \beta_s \quad \text{for} \quad t_a(s) \leq t < t_a(s+1).$$

According to this equation, the synthesis pitch period $t_s(u+1) - t_s(u)$ is equal to the mean $1/\beta(t)$-scaled pitch period in the original signal calculated over the time-frame $t_s(u+1) - t_s(u)$.
This integral equation in $t_s(u+1)$ is easily solved because $P(t)$ and $\beta(t)$ are piecewise-constant functions. Pitch-scale modification using TD-PSOLA is shown in figs. 7 and 8.

5.2.2. Time-scale modification

Time-scale modifications are slightly more complicated than pitch-scale modifications because the original signal and the scaled signal do not share the same time-axis. The time-scale modification is specified by associating to each analysis time instant, a time-scale modification factor denoted $\alpha_s > 0$, from which the time-scale warping function $t \rightarrow D(t)$ may be deduced:

$$D(t_a(1)) = 0$$

$$D(t) = D(t_a(s)) + \alpha_s(t - t_a(s)) \quad t_a(s) \leq t < t_a(s+1), \tag{5.5}$$

where $t \rightarrow D(t)$ is a piecewise-linear and strictly monotonic function. Having specified the time-scale warping function, the next step consists of generating a stream of synthesis time instants $t_s(u)$ from the stream of the analysis time instants $t_a(s)$ *while preserving the pitch contour*. As in the preceding case, the analysis time instants $t_a(s)$ are positioned in a pitch-synchronous way, i.e. $t_a(s+1) - t_a(s) = P(t_a(s))$. The target synthesis pitch contour is defined as $t \rightarrow P'(t) = P(D^{-1}(t))$: the pitch in the time-scaled signal at time t should be close to the pitch in the original signal at time $D^{-1}(t)$.

We must now find a stream of synthesis pitch marks $t_s(u)$, such that $t_s(u+1) = t_s(u) + P'(t_s(u))$. To solve this problem, it is useful to define a stream of *virtual time instants* $t'_s(u)$ in the original signal related to the synthesis time instants by

$$t_s(u) = D(t'_s(u)), \quad t'_s(u) = D^{-1}(t_s(u)).$$

Assuming that $t_s(u)$ and $t'_s(u)$ are known, we try to determine $t_s(u+1)$ (and $t'_s(u+1)$), such that $t_s(u+1) - t_s(u)$ is approximately equal to the pitch in the

original signal at time $t'_s(u)$. This can be expressed as:

$$t_s(u+1) - t_s(u) = \frac{1}{t'_s(u+1) - t'_s(u)} \int_{t'_s(u)}^{t'_s(u+1)} P(t)dt,$$

$$\text{with} \quad t_s(u+1) = D(t'_s(u+1)).$$

According to this equation, the synthesis pitch period $t_s(u+1) - t_s(u)$ at time $t_s(u)$ is equal to the mean value of the pitch in the original signal calculated over the time-interval $t'_s(u+1) - t'_s(u)$. Note that this time-interva $t'_s(u+1) - t'_s(u)$ is mapped to $t_s(u+1) - t_s(u)$ by the mapping function $D(t)$. As was the case above, the eq. (5.7) is an integral equation but is easily solved because $D(t)$ and $P(t)$ are piecewise-linear functions.

5.2.3. Combined time-scale and pitch-scale modifications

Assuming as above that the pitch-scale modification and the time-scale modification are defined by two streams β_s and α_s, the eq. (5.7) can still be used after replacing $P(t)$ by $P(t)/\beta(t)$:

$$t_s(u+1) - t_s(u) = \frac{1}{t'_s(u+1) - t'_s(u)} \int_{t'_s(u)}^{t'_s(u+1)} \frac{P(t)}{\beta(t)} dt \tag{5.8}$$

$$\beta(t) = \beta_s \quad \text{with} \quad t_a(s) \leq t < t_a(s+1)$$

Combined time-scale and pitch-scale modifications actually involve no additional complexity. An example of such modifications is given in fig. 9.

5.2.4. Mapping the synthesis time instants to the analysis time instants

The mapping between the analysis and the synthesis time instants is not generally one-to-one. A simple solution consists of taking a weighted average of the two closest analysis short-time signals. Suppose $t_a(s) \leq t'_s(u) < t_a(s+1)$, then

$$y(u,n) = (1 - \alpha_u)x(s,n) + \alpha_u x(s+1,n),$$

$$\alpha_u = \frac{t'_s(u) - t_a(s)}{t_a(s+1) - t_a(s)}.$$

In the simplest implementation, the coefficient α_u is replaced by the integer nearest to it, that is 0 or 1. This latter solution is equivalent to selecting the analysis short-time signal associated with the analysis time instant closest to the virtual time instant $t'_s(u)$. In that case, and when only time-scale modifications are undertaken, the time-domain and frequency-domain PSOLA methods work by eliminating or duplicating analysis short-time signals at a pitch-synchronous rate very much like most standard time-domain pitch-synchronous splicing methods.

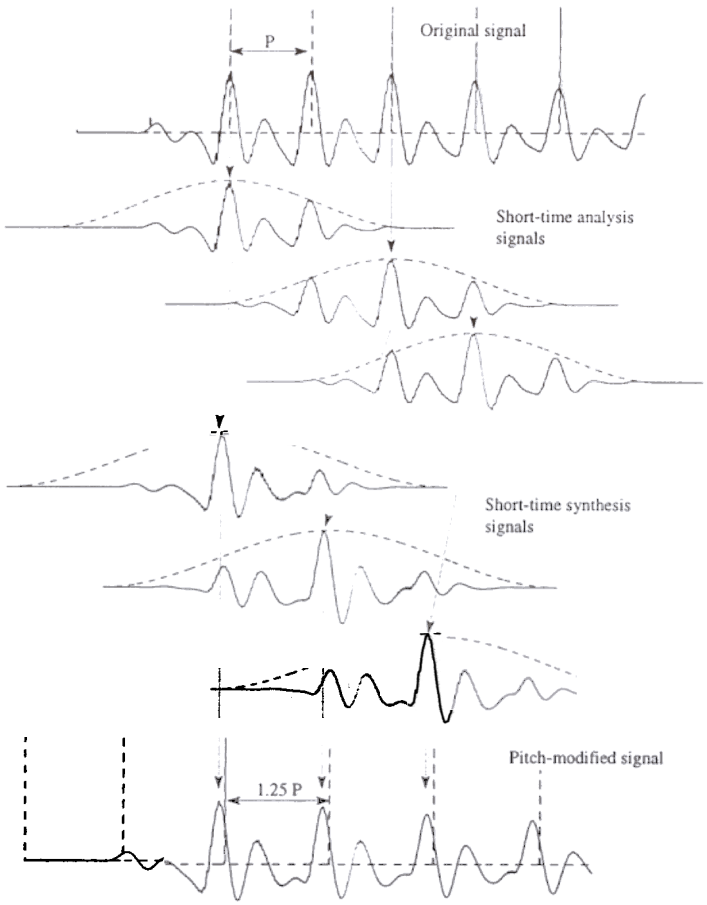


Figure 9. Combined pitch-scale and time-scale modification using the frequency-domain PSOLA approach. The pitch-scale modification factor is 0.8 and the time-scale modification factor is 0.6.

5.3. How does TD-PSOLA do the job ?

Time-domain PSOLA is likely the simplest method that can be imagined of for high-quality pitch-scale modifications of speech signals. The high quality of the obtained for time-scale modification of signals is not really a surprise: the TD-PSOLA operates by 'smoothly' duplicating or eliminating segments of the speech signal at a pitch-synchronous rate. This is the best that can be done, from both theoretical and practical point of view. In particular, it maintains the temporal structure of the original waveform during voicing: TD-PSOLA is *shape invariant* in the sense defined by in [19]; it is worth noting with this respect that the shape-invariant time-scale and pitch-scale transformations based on the sinusoidal representation proposed in this chapter makes use of pitch pulse onset time (which represents the time where the pitch harmonics add coherently, i.e. are in phase), which plays basically the same role than our pitch-synchronous analysis time instants (a more detailed inspection reveals that the two procedures are, despite differences in appearance, nearly identical for time-scale modifications).

The fact that TD-PSOLA is also able to perform high-quality pitch transformations is may be more puzzling. It has long been believed that an explicit decomposition between a source signal, representing the glottal excitation signal, and a spectral envelope, representing the transfer function of the supra-glottal cavities is a pre-requisite for pitch-scale transformation. In fact, TD-PSOLA does such a decomposition, but it does it implicitly, for reasons which are straightforward to understand (a complete theoretical analysis of the TD-PSOLA method is given in [13]). In order to make the explanations as simple and understandable as possible, we will assume in this section a simplistic model for voiced speech signal; the hypotheses we formulate are as follows: *i)* the speech signal is periodic, with an integer pitch period P; the analysis time instant are set at $t_a(s) = sP$ *i)*, the pitch-scale modification factor β is constant and is an integer number, *iii)* the time-scale modification factor is constant and $\alpha = 1/\beta$, *iv)* the analysis windows are all equal to the same prototype window. These hypotheses are of course not accurate, but they are sufficiently representative of the structure of voiced speech to give a clear picture of what TD-PSOLA is doing. Under the preceding hypotheses, the short-term analysis signals are expressed as

$$x(s,n) = h(n)x(n - sP). \tag{5.10}$$

Assume for simplicity that we use the OLA procedure (other synthesis procedures could have been studied as well, at the expense of some mathematical details). Omitting the time-varying normalization factor, we get:

$$y(n) = \sum_s x(s, n - s\beta P) = \sum_s h(n - s\beta P)x(n - sP - s\beta P). \tag{5.11}$$

Since the input signal is assumed to be periodic with period P, $x(m - sP) = x(m)$,

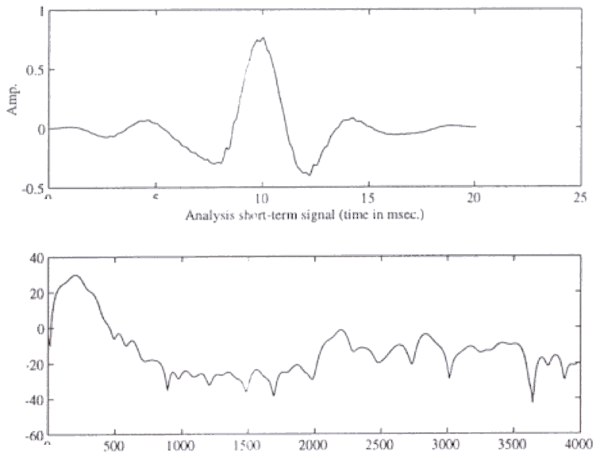


Figure 10. Upper panel: short-time analysis signal. Lower panel: associated short-time analysis amplitude spectrum.

the synthetic signal is expressed as

$$y(n) = \sum_s h(n - s\beta P)x(n - s\beta P).$$

The synthetic signal is thus obtained by replicating, with the period βP the *same* prototype signal, $h(n)x(n)$. Obviously, the synthetic signal $y(n)$ is periodic with period βP (which is exactly what we want!). More interesting is he harmonic structure of this signal: what are the pitch-harmonic amplitudes ? What are the pitch-harmonic phases ? Answers to these questions are straightforwardly obtained by resorting to standard results on the Fourier transform of periodically extended waveforms. By application of the so-called Poisson's formula, we have

$$y(n) = \frac{1}{\beta P} \sum_{k=0}^{\beta P-1} c_k \exp\left(j\frac{2\pi k}{\beta P}n\right)$$

$$c_k = X\left(\frac{2\pi k}{\beta P}\right) \quad \text{with} \quad X(\omega) = \sum_{n=-\infty}^{+\infty} h(n)x(n)\exp(-j\omega n).$$

The above expression means that *the complex amplitudes c_k of the pitch harmonics in the synthetic signal are equal to the values at the pitch-harmonic frequencies $2\pi k/\beta P$ of the discrete Fourier transform (DFT) of the prototype short-time signal* $h(n)x(n)$. In other words, the time-domain PSOLA method *resamples* the Fourier transform of the short-time analysis signal (this is illustrated in figs. 10 and 11).

Thus, the TD-PSOLA method uses the Fourier short-term amplitude spectrum as an implicit estimate of the transfer function of the supra-glottal cavities. This

result is interesting for a number of reasons.

In particular, we can determine which windows $h_a(n)$ are the more appropriate for the TD-PSOLA algorithm. The analysis window should be such that the short-time Fourier transform $X(t_a(s),\omega)$ is a reasonable estimate of the spectral envelope of the current short-term signal. As mentioned above, the length of the analysis window T is proportional to the local pitch period $P(s)$, i.e., $T = \mu P(s)$. For standard window functions, the window's cutoff frequency is inversely proportional to the window-length. For $\mu = 2$, the window's cutoff frequency ($2\pi/P(s)$ for Hamming and Hanning window, $3\pi/P(s)$ for Blackman window) is larger than the spacing between the pitch harmonics ($2\pi/P(s)$): the analysis window does not resolve the individual pitch harmonics. The short-time analysis spectrum $X(t_a(s),\omega)$ is a 'smooth' estimate of the speech-signal spectral envelope: the window main lobe provides a means for interpolating between the pitch harmonic. By contrast, larger values of μ increase the window resolution and reveal the harmonic structure of $X(t_a(s),\omega)$, a property which is undesirable for the TD-PSOLA method: resampling $X(t_a(s),\omega)$ at the synthesis pitch-frequency $2\pi k/\eta P$ is likely to produce audible artifacts due to pitch-harmonic attenuation/cancellation.

The above results also shows some of the shortcomings of the TD-PSOLA method. It is clear that the spectral envelope implicitly used by the TD-PSOLA method does not correspond to the 'true' spectral envelope (i.e. $G(t_a(s),\omega)$ in the speech-production model in 2.2) because of the smearing introduced by the windowing operation $H_a(t_a(s),\omega)$. This discrepancy becomes more acute for high-pitched voices, in which cases the analysis windows are of shorter duration and therefore exhibit broader main lobes. Fortunately, these effects generally do not cause a severe degradation of the quality of the speech output, at least when moderate pitch-scale modifications are used. This is may be because the human ear is not very sensitive to the bandwidth of formants.

5.4 Variations on the PSOLA paradigm

The frequency-domain PSOLA (FD-PSOLA) and the residual-domain PSOLA (LP-PSOLA) approaches are two possible speech modification techniques that can be adapted almost directly from the TD-PSOLA paradigm. These two methods are more flexible than the TD-PSOLA technique because they provide a direct control over the spectral envelope both at the analysis and at the synthesis stage. This is interesting for at least two reasons: first, it is not difficult to guess that it is always possible to extract spectral envelope that are better behaved than the ones provided by a crude short-term Fourier analysis (this is especially true when dealing with female voice... see the discussion above). Better envelopes mean in practice a better quality fo the reconstructed speech output. Second, the control of the spectral envelope at the synthesis stage gives an additional degree of freedom which can be exploited in many different ways. It enables a variety of transformations which could not have been handled by TD-PSOLA: an example is the voice identity transformation, which amounts to transform the voice of one speaker so that 'it

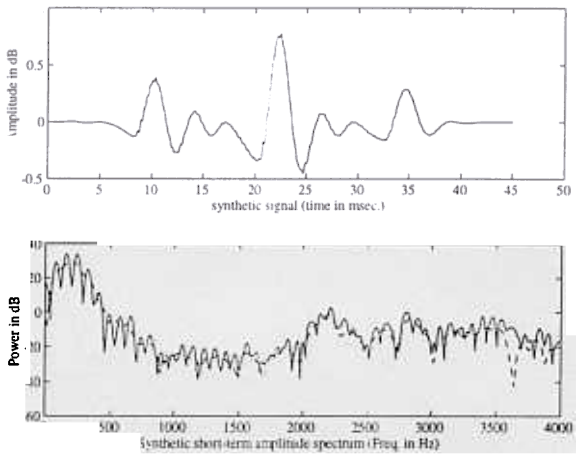


Figure 11. The frequency resampling property: Upper panel: synthetic signal waveform. Lower panel: -solid-line- amplitude spectrum of the synthetic signal -dashed line- amplitude spectrum of the short-term analysis signal. The pitch-modification factor is equal to 0.8.

sounds like it have been uttered by another speaker'

5.5. FD-PSOLA

Historically, the FD-PSOLA technique was the first pitch-synchronous time-scale and pitch-scale modification technique proposed in the literature (see [20]). At that time, FD-PSOLA was primarily thought of as a pitch-synchronous implementation of the speech modification technique proposed in [14]. It was remarked that the pitch synchronicity of the processing brought considerable simplifications in the implementations of the algorithms. In particular, no phase corrections were needed because the linear component of the pitch-harmonic phase rotates by 2π between two successive synthesis instant. More interestingly, it appeared that synchronicity helped to increase significantly the reconstructed speech quality by eliminating much of the artifacts of the original method [14]. In particular, the residual reverberation was almost eliminated, yielding to clearer and crisper synthetic speech.

In practice, the FD-PSOLA technique and the TD-PSOLA technique have much in common. The FD-PSOLA technique differs from the TD-PSOLA technique only in the definition of the short-time synthesis signals for pitch-scale modifications. Prior to overlap-add synthesis, each short-time analysis signal is modified; the modification is carried out in the frequency-domain on the short-time analysis spectrum. The algorithm is an exact replica of the frequency-domain resampling method described in section 5.6. In a first step, the short-term spectrum is split into a global spectra envelope and a source component. The pitch modifications are carried out in a second step on the source signal short-term spectrum by scaling the spacing

between the pitch harmonics by the pitch-modification factor β_s. In the original proposal, the scaling is performed without explicitly extracting the individual pitch harmonics: it is done by linearly interpolating the real and imaginary parts of the complex spectrum. As usual, when the pitch is lowered, the upper frequency band is regenerated either by *spectral folding* or by *spectral copying* (see the appendix). The last step consists in recombining the modified source short-term spectrum and the spectral envelope which is, in the standard implementation, left untouched. It is however likely that the quality of pitch-modification could be improved by taking into account the interaction between the pitch-harmonic structure and the formant frequencies. This kind interaction could be implemented using the FD-PSOLA method, by modifying in an appropriate way the spectral envelope prior recombination (even a simple scaling of the lowest formant seems to suffice to improve the quality).

5.6. LP-PSOLA

The pitch-synchronous time-scale and pitch-scale modification scheme can also be easily embedded in residual-excited vocoders, yielding a method referred to as the linear-predictive PSOLA method (this term is somewhat misleading; the use of a autoregressive model for the spectral envelope is not mandatory; for example, a cepstral deconvolution technique could be used as well). In this scheme, prior to PSOLA processing, the signal is split into a time-domain source component and an spectral envelope (a spectral envelope is estimated for each analysis time instant). Pitch-scale and time-scale modifications are then carried out on the source signal. The output signal is finally obtained by combining the modified source signal with the time-varying spectral envelopes, re-synchronized with the stream of synthesis time instants. A specific (and straightforward) implementation of this scheme using standard linear prediction is given in [13]. An autoregressive model is estimated around each analysis. An excitation signal is then obtained by inverse filtering the speech signal using the analysis filters. Special care should be taken to avoid discontinuities in the speech output, due to a mismatch between the modified excitation energy and the synthesis filter gain. In our implementation, we used a normalized lattice filter (the white-noise gain of the analysis and synthesis filter is equal to one) and we interpolate the reflection coefficients on the sample-by-sample basis. Other models of the spectral envelope and/or other estimates of the model parameters could have been used as well.

Appendix. Pitch-scale modification and resampling

Time-domain and frequency-domain resampling plays a key role in many pitch modification algorithms (time-domain PSOLA is a notable exception). For clarity, we describe in this appendix the basic concepts. More details can be found [4].

The time-domain resampling method described below applies for *constant* and *ra-*

tional sampling-rate conversion factors $\alpha = D/U$. This resampling method consists of *1*) upsampling the original signal by a factor U, *2*) interpolating the upsampled signal by use of a low-pass filter with an appropriate cutoff-frequency, and *3*) downsampling the resulting signal by a factor D by discarding $D-1$ samples out of D. The spectrum of the upsampled signal is a U-folded compressed version of the original spectrum. Alias-free reconstruction requires the cancellation of these images, an operation performed during step 2 above. Downsampling by a factor D, the ideally interpolated signal produces a decimated signal whose frequency contents is concentrated in the interval $[-\pi D/U, \pi D/U]$ when $D/U < 1$ and $[-\pi, \pi]$ when $D/U > 1$.

The time-domain source resampling method is straightforward to implement when one is working with constant rational sampling-rate conversion factors. For time-varying and/or irrational factor, this solution is no longer effective. The STFT analysis/synthesis framework provides a solution to this problem. In the STFT analysis-synthesis framework, sampling-rate conversion is carried out in the frequency-domain. It consists of: *1*) the extraction of a stream of short-term analysis signals at a uniform analysis rate of R samples, i.e. $t_a(s) = sR$ and the computation of the associated short-term spectra, *2*) the interpolation of the complex short-term Fourier spectra at a new set of frequencies, *3*) the synthesis of the rate-modified signal. During step *2*) above, linear interpolation is applied to the real and the imaginary parts of the short-term spectrum. The linearly interpolated short-term spectrum is expressed as

$$\bar{X}(\tilde{t}_s(u), \Omega_k) = (1-\rho(k))X(t_a(u), \Omega_{\tilde{k}}) + \rho(k)X(t_a(u), \Omega_{\tilde{k}+1}), \tag{5.14}$$

$$\tilde{k} = \lfloor k\alpha(t_a(u)) \rfloor, \quad \rho(k) = k\alpha(t_a(u)) - \tilde{k}.$$

Resampling the signal by a factor $\alpha(t)$ causes a local modification of the time-scale by a factor $\beta(t) = 1/\alpha(t)$. To account for this implicit modification of the time-scale, the synthesis short-time spectrum $\bar{X}(\tilde{t}_s(u), \Omega_k)$ is synchronized on the virtual synthesis time instant $\tilde{t}_s(u)$ given by

$$\tilde{t}_s(u) = \int_0^{uR} /\alpha(t)dt \tag{5.15}$$

and the effective length of the short-time modified signal is divided by $\alpha(uR)$. Because of the generally non-integral nature of sample rate change, fractional delays occur between synthesis frames: the virtual synthesis time instants $\tilde{t}_s(u) = D(uR)$ do not correspond to an integer numbers of samples, although the synthesis scheme requires integer synthesis time instants $t_s(u) = \lfloor \tilde{t}_s(u) \rfloor$. To correct for the *fractional delay* $\tilde{t}_s(u) - t_s(u)$, it is necessary to apply a fractional-sample delay correction to each frame, which is applied as a phase correction:

$$\bar{Y}(t_s(u), \Omega_k) = \bar{X}(\tilde{t}_s(u), \Omega_k) \exp\left[-i\Omega_k(\tilde{t}_s(u) - t_s(u))\right] \tag{5.16}$$

The resampled output is finally obtained by a least-squares overlap-add procedure; note that, due to the time-varying nature of the modification, the synthesis rate is

not constant whereas the analysis rate is.
During the resampling process, the cutoff-frequency ω_h of the analysis window is multiplied by a factor $\alpha(t_a(u))$. To avoid artifacts during the synthesis, the bandwidth of the synthesis window should be matched to the modified cutoff-frequency $\alpha(t_a(u))\omega_h$. For standard spectral analysis windows (e.g., Hanning, Hamming, Kaiser), the synthesis window is chosen to be

$$f_u(n) = h_u(n/\alpha(t_a(u))) \tag{5.17}$$

For pitch modification, resampling is performed *without modifying the actual sampling frequency*: the normalized frequency interval $[-\pi, +\pi]$ remains matched with the same 'physical frequencies'. As a consequence upsampling and downsampling operations result *i)* in linear compression-expansion of the frequency axis and *ii)* in linear expansion-compression of the time axis. When applied to speech, the compression-expansion of the frequency axis modifies the spacing between successive pitch harmonics (pitch-scale modification) but also modifies the locations and bandwidths of the formants. This, of course, is undesirable for speech pitch-scale modification. To circumvent this problem, resampling is carried out on the source signal estimated by source/filter decomposition method.

When one increases the pitch of the source ($\alpha = D/U > 1$), the spectrum of the source signal is expanded and the upper frequency part of the source spectrum is simply discarded. By contrast, as mentioned in section 5.6, when one decreases the pitch of the source signal ($\alpha < 1$), its spectrum only occupies a limited portion ($[-\pi\alpha, \pi\alpha]$) of the frequency range. If this source signal is used as is in the synthesis stage, the resulting pitch-scaled speech signal is artificially band-limited. To counter this undesirable side-effect, the missing upper-frequency band of the source signal needs to be regenerated before the final synthesis stage. This can be done either by *spectral folding* or *spectral copying* (see [14, 15]).

The resampling methods present several drawbacks. In the first place, be it by spectral folding or by spectral copying, the technique used to regenerate the upper part of the spectrum is not satisfactory. This point illustrate a primary motivation for substituting a more appropriate technique such as the pitch-synchronous TD-PSOLA method or parametric methods. Second, spectral envelope estimation methods are not 'perfect': they do not completely 'flatten' the spectrum of the source signal. The 'flattened' pitch-harmonic amplitudes present certain fluctuations with respect to the 'ideal' flat envelope spectrum. Time-domain or frequency-domain source resampling techniques linearly warp the frequency axis: the amplitude deviations of the pitch harmonics are also shifted in frequency. Note also that the other details of the source are also shifted from their original spectral location. This is particularly true for the voicing, which is not uniform over the frequencies. The source spectrum of a voiced sound usually exhibits frequency-bands where friction noise dominates the pitch harmonics. The pitch-modification methods based on frequency-domain resampling translate the voiced/unvoiced frequency bands.

References

[1] M. Portnoff, "Short-time Fourier analysis of sampled speech," *IEEE Trans. on Acoust. Speech and Sig. Proc.*, vol. 29, no. 3, pp. 364–373, 1981.

[2] D. Gabor, "Acoustical quanta and the theory of hearing," *Nature*, vol. 159. pp. 591–594, 1947.

[3] R. Crochiere, "A weighted overlap-add method of short-time Fourier analysis/synthesis," *IEEE Trans. on Acoust. Speech and Sig. Proc.*, vol. 28, no. 2. pp. 99–102, 1980.

[4] R. Crochiere and L. Rabiner, *Multirate digital signal processing.* Prentice-Hall, 1983.

[5] J. Allen, "Application of the short-time Fourier transform to speech processing and spectral analysis," in *Proc. Int. Conf. on Acoust. Speech and Sig. Proc.*, pp. 1012–1015, 1982.

[6] S. Nawab and T. Quatieri, "Short-time Fourier transform," in *Adanced topics in signal processing* (J. Lim and A. Oppenheim, eds.), Prentice-Hall, 1988.

[7] J. Deller, J. Proakis, and J. Hansen, *Discrete-Time Processing of Speech Signals.* MacMillan, 1993.

[8] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Trans. on Acoust. Speech and Sig. Proc.*, vol. 32, no. 2, pp. 236–243, 1984.

[9] S. Roucos and A. Wilgus, "High quality time-scale modification of speech," in *Proc. Int. Conf. on Acoust. Speech and Sig. Proc.*, pp. 493–496, 1985.

[10] W. Verhelst and M. Roelands, "An overlap-add technique based on waveform similarity (wsola) for high-quality time-scale modification of speech," in *Proc. Int. Conf. on Acoust. Speech and Sig. Proc.*, pp. 554–557, 1993.

[11] G. Spleesters, W. Verhelst, and A. Wahl, "On the application of automatic waveform editing for time warping digital and analog recordings," in *Proc. Audio. Engineering Society*, p. 11.3, 1994. Preprint 3843.

[12] J. Laroche, "Autocorrelation method for high quality pitch/time scaling." in *Proc. Workshop on App. of Sig. Proc. to Audio and Acoust.*, pp. 200–204, 1993.

[13] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech Comm.*, vol. 9, no. 5, pp. 453–467, 1990.

[14] S. Seneff, "System to independently modify excitation and/or spectrum of speech waveform without explicit pitch extraction," *IEEE Trans. on Acoust. Speech and Sig. Proc.*, vol. 24, pp. 358–365, 1982.

[15] F. Charpentier, *Traitement de la parole par analyse-synthèse de Fourier. Application à la synthèse par diphones.* PhD thesis, Ecole Nationale Supérieure des Télécommunications, 1988. ENST-88011.

[16] E. Moulines and J. Laroche, "Non-parametric techniques for pitch-scale and time-scale modifications of speech," *Speech Comm.*, vol. 16, no. 2, pp. 175–207, 1995.

[17] C. Ma, Y. Kamp, and L. Willems, "A frobenius norm approach to glottal closure detection from the speech signal," *IEEE Trans. on Speech and Audio*, vol. 2, no. 2. pp. 258–265, 1994.

[18] T. Quatieri, C. Jankowski, and D. Reynolds, "Energy onset times for speaker identification," *IEEE Sig. Proc. Letters*, vol. 1, no. 11, pp. 160–162, 1994.

[19] T. Quatieri and R. McAulay, "Shape invariant time-scale and pitch modification of speech," *IEEE Trans. on Sig. Proc.*, vol. 40, no. 3, pp. 497–510, 1992.

[20] F. Charpentier and M. Stella, "Diphone synthesis using an overlap-add technique for speech waveforms concatenation," in *Proc. Int. Conf. on Acoust. Speech and Sig. Proc.*, pp. 2015–2018, 1986.