L19: Prosodic modification of speech

Time-domain pitch synchronous overlap add (TD-PSOLA)

Linear-prediction PSOLA

Frequency-domain PSOLA

Sinusoidal models

Harmonic + noise models

STRAIGHT

This lecture is based on [Taylor, 2009, ch. 14; Holmes, 2001, ch. 5; Moulines and Charpentier, 1990]

Introduction

Motivation

- As we saw in the previous lecture, concatenative synthesis with fixed inventory requires prosodic modification of the diphones to match specifications from the front end
- Simple modifications of the speech waveform do not produce the desired results
  - We are familiar with speeding up or slowing down recordings, which changes not only the duration but also the pitch
  - Likewise, over- or under-sampling alters duration, but also modifies the spectral envelope: formants become compressed/dilated
- The techniques proposed in this lecture perform prosodic modification of speech with minimum distortions
  - Time-scale modification modifies the duration of the utterance without affecting pitch
  - Pitch-scale modification seeks to modify the pitch of the utterance without affecting its duration

Pitch synchronous overlap add (PSOLA)

Introduction

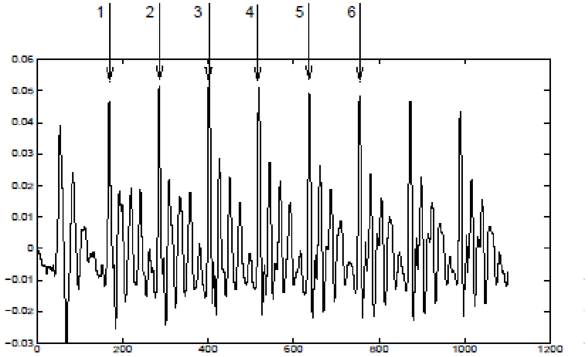
- PSOLA refers to a family of signal processing techniques that are used to perform time-scale and pitch-scale modification of speech
- These modifications are performed without performing any explicit source/filter separation
- The basis of all PSOLA techniques is
  - Isolate pitch periods in the original signal
  - Perform the required modification
  - Resynthesize the final waveform through an overlap-add operation
- Time-domain TD-PSOLA is the most popular PSOLA technique and also the most popular of all time/pitch-scaling techniques
- Other variants of PSOLA include
  - Linear-prediction LP-PSOLA
  - Fourier-domain FD-PSOLA

Time-domain PSOLA

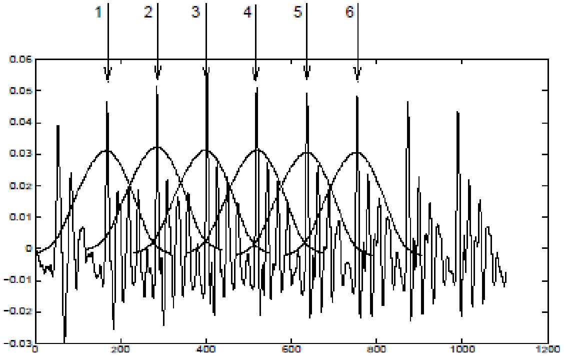
Requirements

- TD-PSOLA works pith-synchronously, which means there is one analysis window per pitch period
  - A prerequisite for this, therefore, is that we need to be able to identify the epochs in the speech signal
  - For PSOLA, it is vital that epochs are determined with great accuracy
  - Epochs may be the instants of glottal closure or any other instant as long as it lies in the same relative position for every frame
- The signal is the separated with a Hanning window, generally extending two pitch periods (one before, one after)
  - These windowed frames can then be recombined by placing their centers at the original epoch positions and adding the overlapping regions
  - Though the result is not exactly the same, the resulting speech waveform is perceptually indistinguishable from the original one
  - For unvoiced segments, a default window length of $10ms$ is commonly used

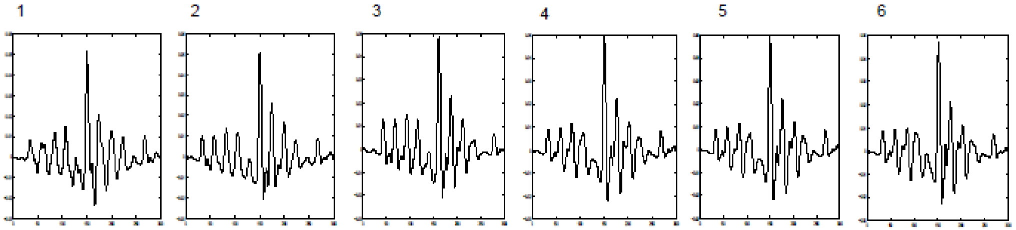
Analysis and reconstruction



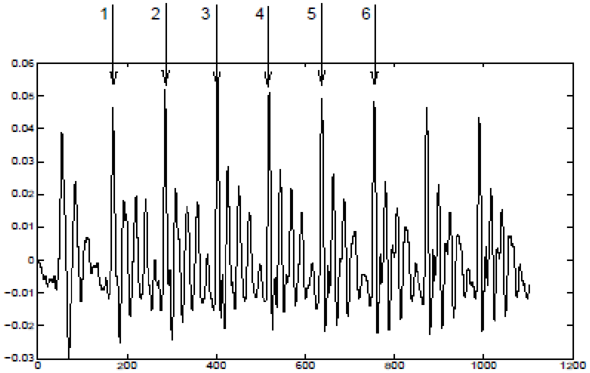
(1) Original speech waveform with epochs



(2) A Hanning window is placed at each epoch



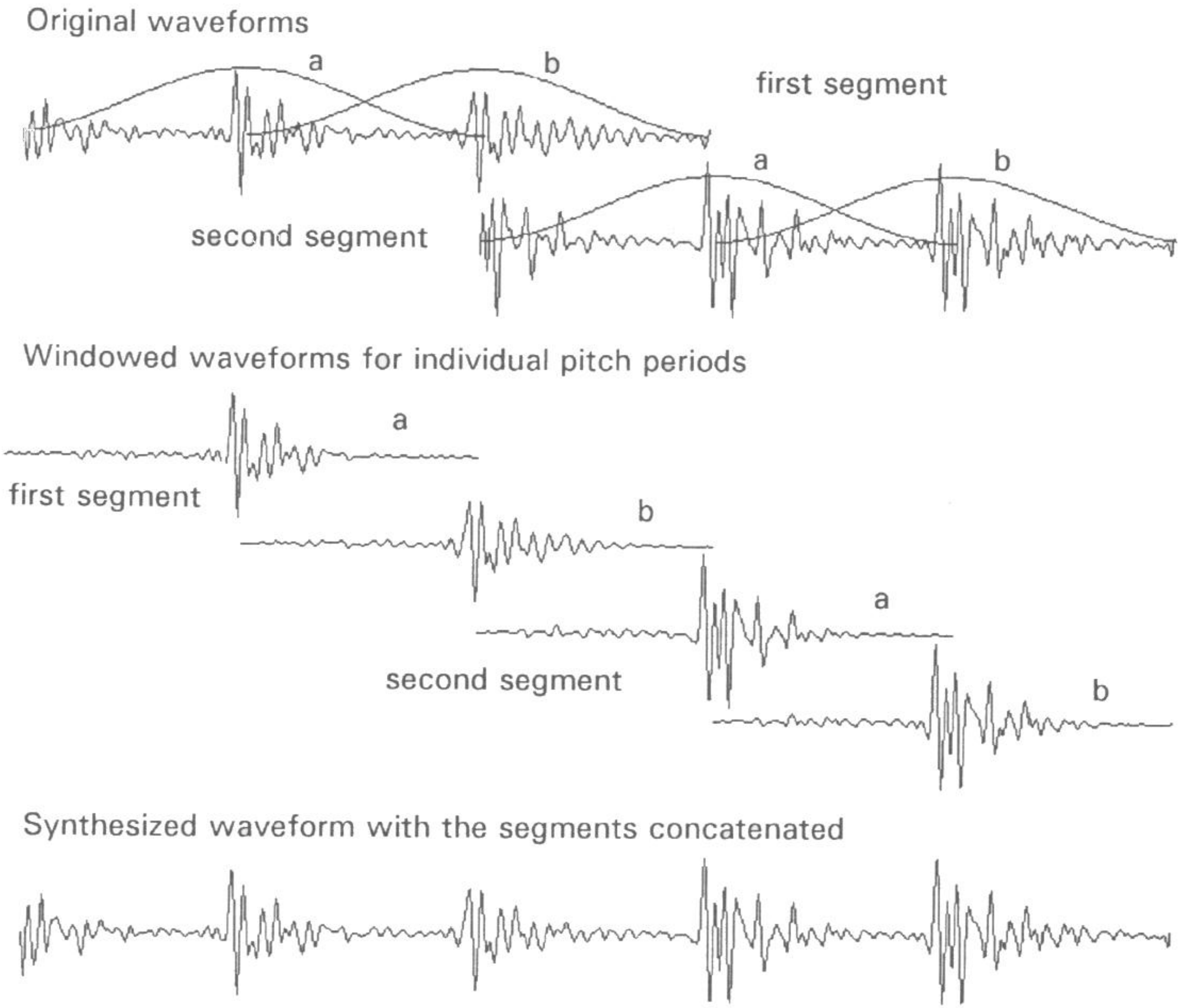
(3) Separate frames are created by the Hanning window, each centered at the point of maximum positive excursion



(4) Overlap-add of the separate frames results in a perceptually identical waveform to the original

[Taylor, 2009]

Merging two segments


Original waveforms
a
b
first segment
a
b
second segment
Windowed waveforms for individual pitch periods
a
first segment
b
a
second segment
b
Synthesized waveform with the segments concatenated

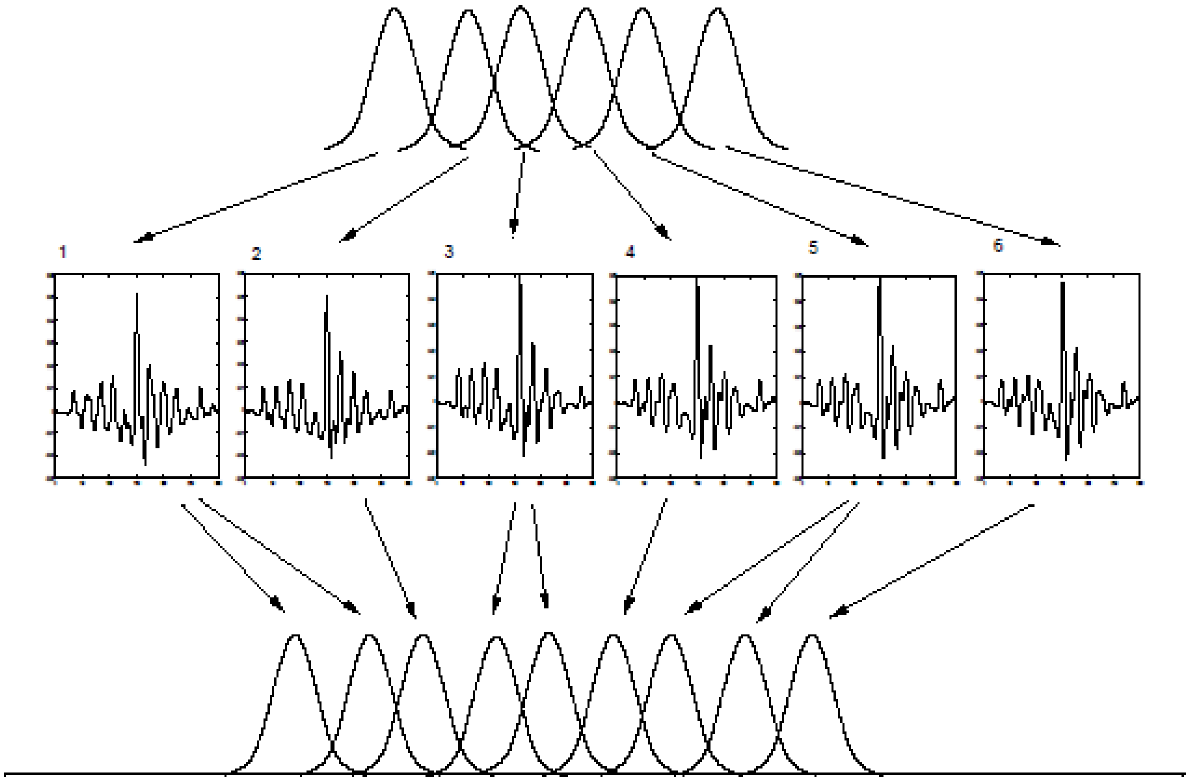
[Holmes, 2001]

Time-scale modification

- Lengthening is achieved by duplicating frames
  - For a given set of frames, certain frames are duplicated, inserted back into the sequence, and then overlap-added
  - The result is a longer speech waveform
  - In general, listeners won’t detect the operation, and will only perceive a longer segment of natural speech
- Shortening is achieved by removing frames
  - For a given set of frames, certain frames are removed, and the remaining ones are overlap-added
  - The result is a shorter speech waveform
  - As before, listeners will only perceive a shorter segment of natural speech

- As a rule of thumb, time-scaling by up to a factor of two (twice longer or shorter) can be performed without much noticeable degradation

Time-scaling (lengthening)

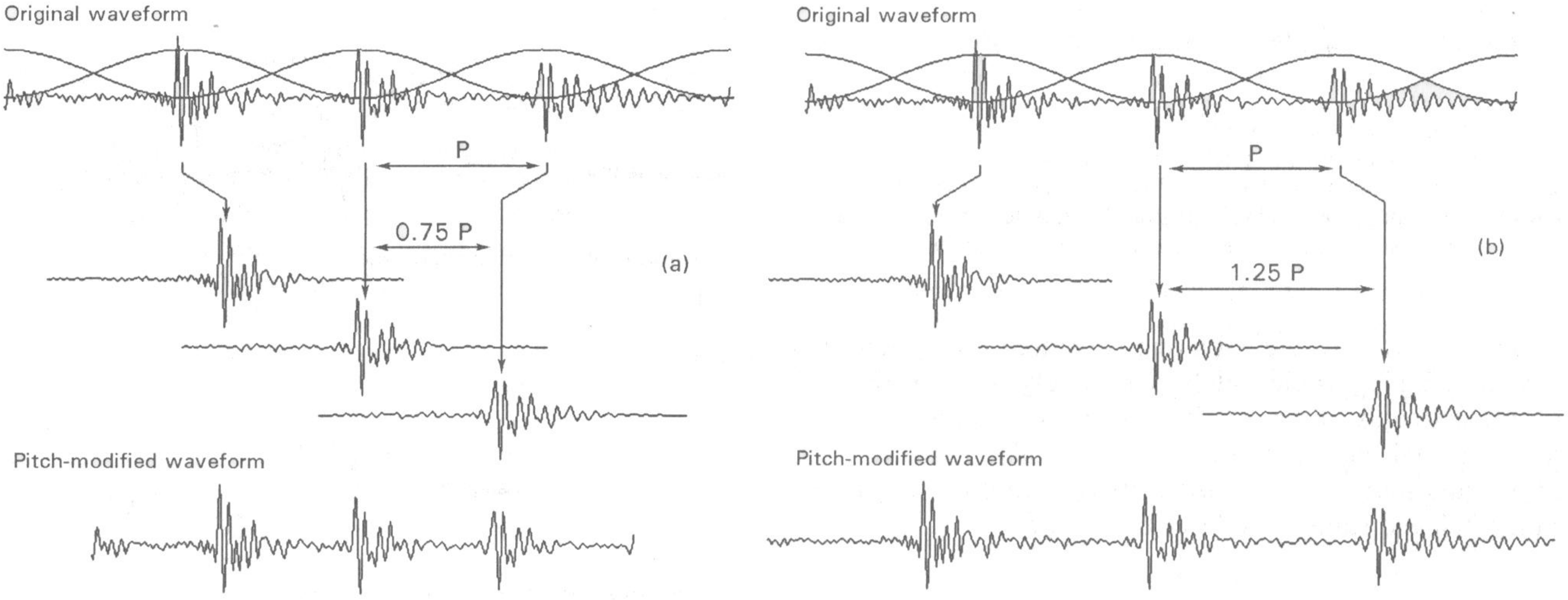


[Taylor, 2009]

Pitch-scale modification

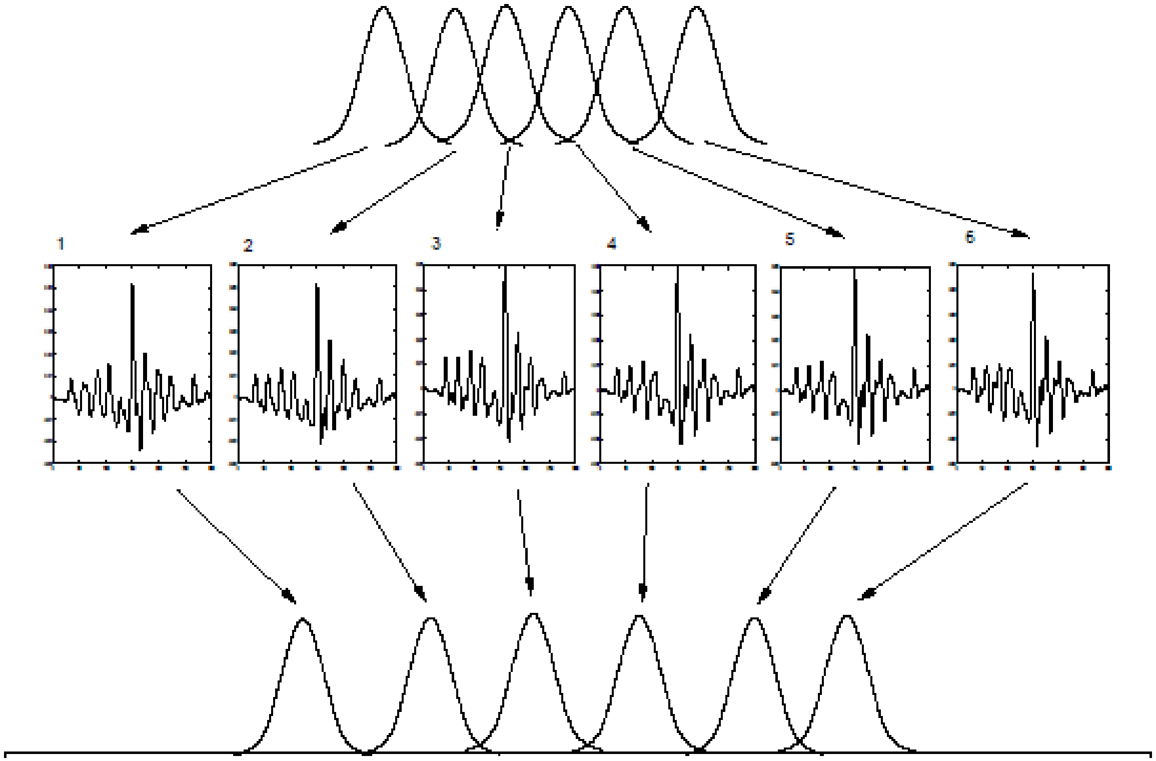
- Performed by recombining frames on epochs which are set at different distances apart from the original ones
  - Assume a speech segment with a pitch of 100Hz (10 ms between epochs)
  - As before, we perform pitch-synchronous analysis with a Hanning window
  - If we place the windowed frames 9ms apart and overlap-add, we will obtain a signal with a pitch of 1/0.009 = 111Hz
  - Conversely, if we place the frames 11ms apart, we will obtaine a signal with a pitch of 1/0.011 = 91Hz
  - The process of pitch lowering explains why we need an analysis window that is two pitch periods long
    - This ensures that up to a factor of 0.5, when we move the frames we always have some speech to add at the frame edges
- As with time-scaling, pitch-scaling by up to a factor of two can be performed without much noticeable degradation

Pitch-scaling



[Holmes, 2001]

Pitch-scaling (lowering)



[Taylor, 2009]

Epoch manipulation

- A critical step in TD-PSOLA is proper manipulation of epochs
- A sequence of analysis epochs $T^a = \{t_1^a, t_2^a \dots t_M^a\}$ is found by means of an epoch detection algorithm
- From this sequence, the local pitch period can be found as

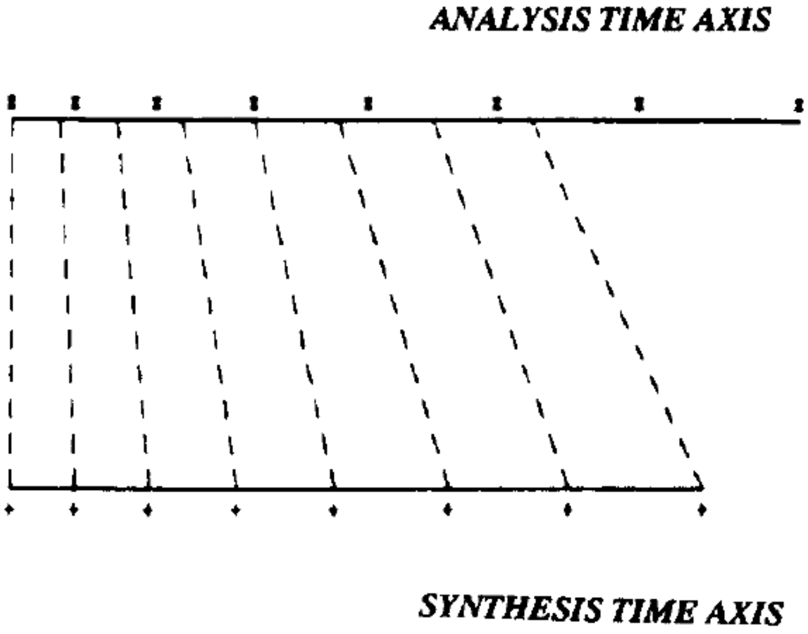
$$p_m^a = \frac{t_{m+1}^a - t_{m-1}^a}{2}$$

- Given the sequence of analysis epochs and pitch periods, we extract a sequence of analysis frames by windowing

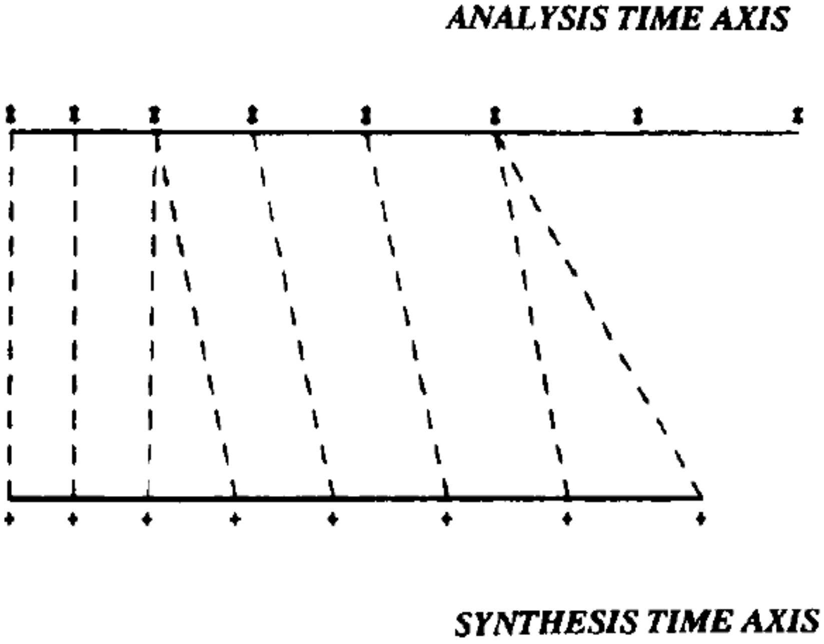
$$x_m^a[n] = w_m[n]x[n]$$

- Next, a set of synthesis epochs $T^s = \{t_1^s, t_2^s \dots t_M^s\}$ is created from the target F0 and timing values provided by the front end
- A mapping function $M[i]$ is then created that specifies which analysis frames should be used with each synthesis epoch

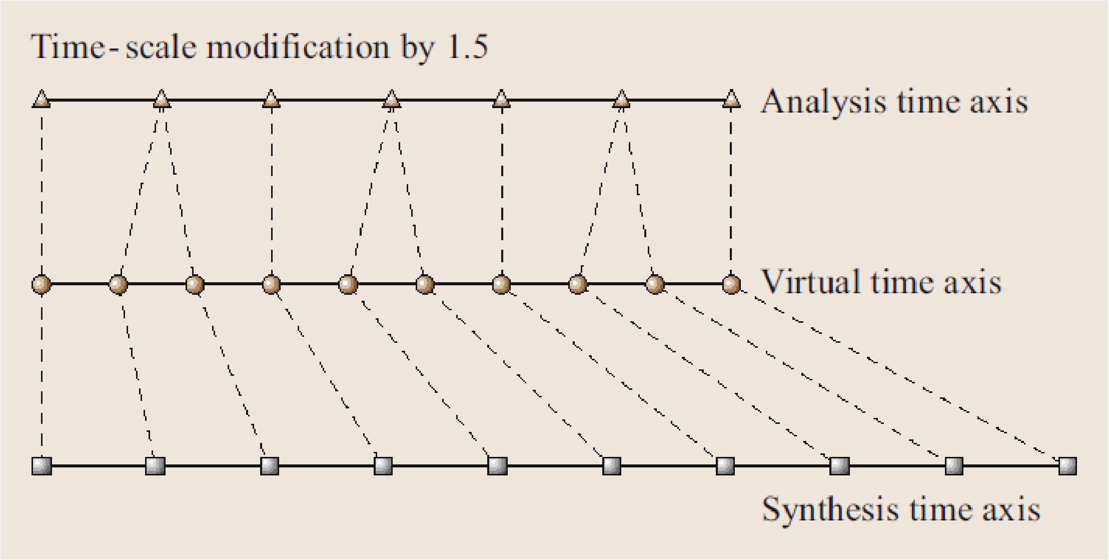
Mapping function $M[i]$ for time-scaling (slowing down)

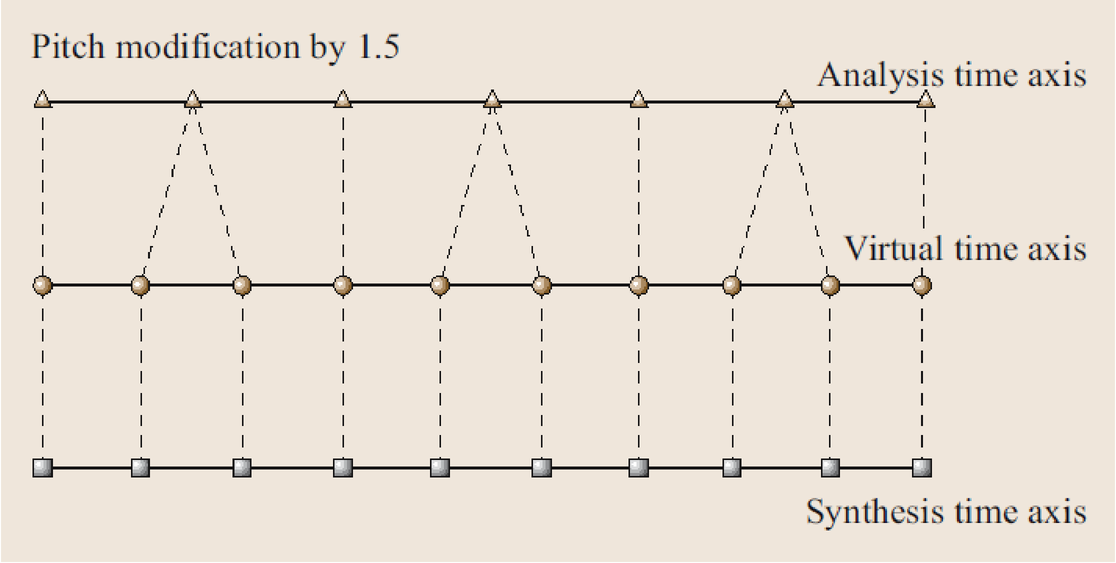


Dashed lines represent time-scale warping function between analysis and synthesis time axes corresponding to the desired time-scaling



Dashed lines represent the resulting pitch-mark mapping, in this case duplicating two analysis ST signals out of six.



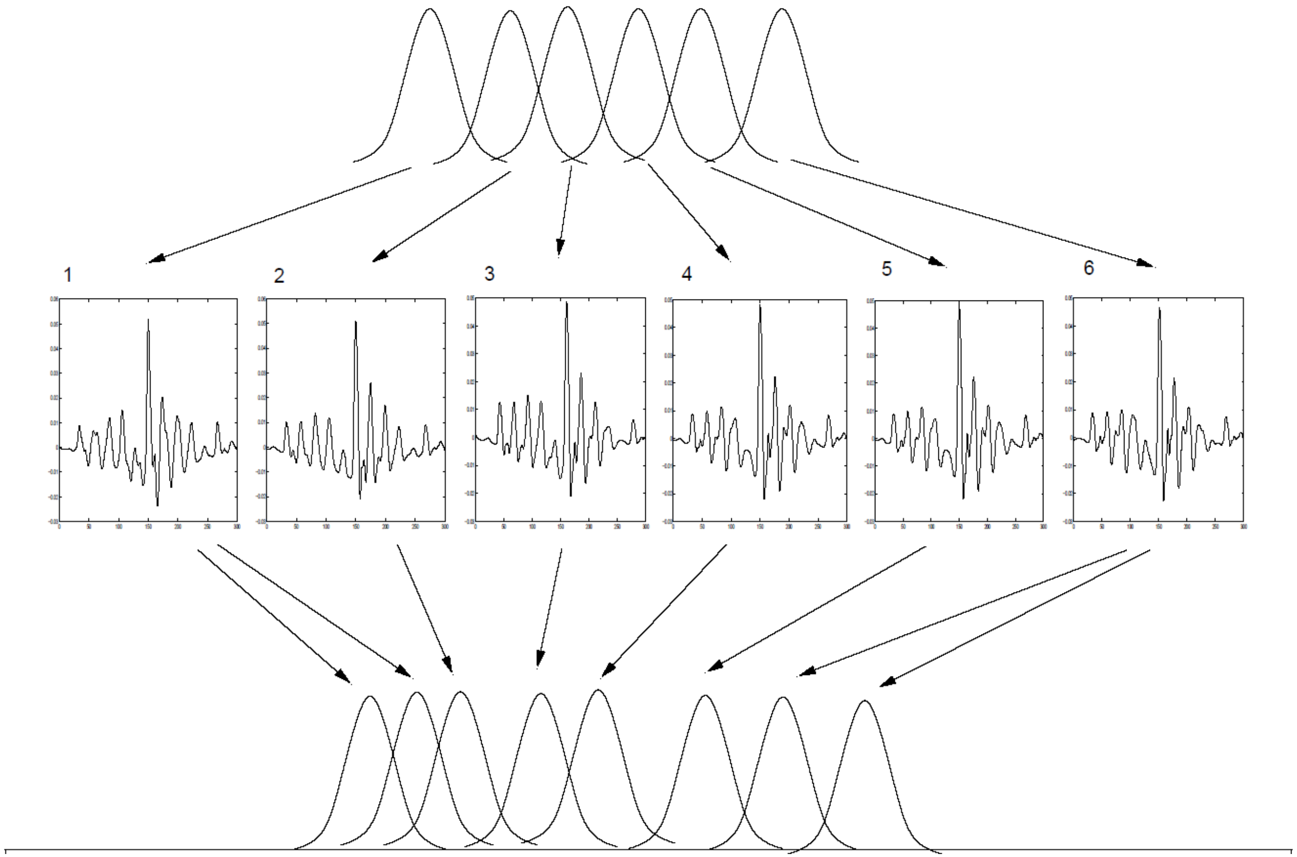


[Stylianou, 2008, in Benesty et al., (Eds)]

Interactions

- Duration modification can be performed without reference to pitch
  - Assume 5 frames of $F0$=100Hz speech spanning 40 ms
  - A sequence with the same pitch but longer (shorter) duration can be achieved by adding (removing) synthesis epochs
  - The mapping function $M[i]$ specifies which analysis frame should be used for each synthesis frame
- Pitch modification is more complex as it interacts with duration
  - Consider the same example of 100Hz and spanning 5 frames, or a total of $(5-1) \times 10 = 40ms$ between t_1^a and t_5^a
  - Imagine we wish to change its pitch to 150 Hz
  - This can be done by creating a set of synthesis epochs $6.6ms$ apart
  - In doing so, the overall duration becomes $(5-1) \times 6.6 = 26ms$
  - To preserve the original duration, we would then have to duplicate two frames, yielding an overall duration of $(7-1) \times 6.6 = 40ms$

Simultaneous time- and pitch-scaling



[Taylor, 2009]

Performance

- Synthesis quality with TD-PSOLA is extremely high, provided that
  - The speech has been accurately epoch-marked (critical), and
  - Modifications do not exceed a factor of two
- In terms of speed, it would be difficult to conceive an algorithm that would be faster than TD-PSOLA
- However, TD-PSOLA can *only* be used for time- and pitch-scaling, it does not allow any other form of modification (e.g., spectral)
- In addition, TD-PSOLA does not perform compression, and the entire waveform must be kept in memory
  - This issue is addressed by a variant known as linear-prediction PSOLA
- Other issues
  - When slowing down unvoiced portions in the range of 2, the regular repetition of unvoiced segments leads to a perceived "tonal" noise
    - This can be addressed by reversing the time axis of consecutive frames
  - Similar effects can also occur for voiced fricatives; in this case, though, time reversal does not solve the problem and FD-PSOLA is needed

Linear prediction PSOLA

Approach

- Decompose the speech signal through an LP filter
- Process the residual in a manner similar to TD-PSOLA
- Convolve the time/pitch-scaled residual with the LP filter

Advantages over TD-PSOLA

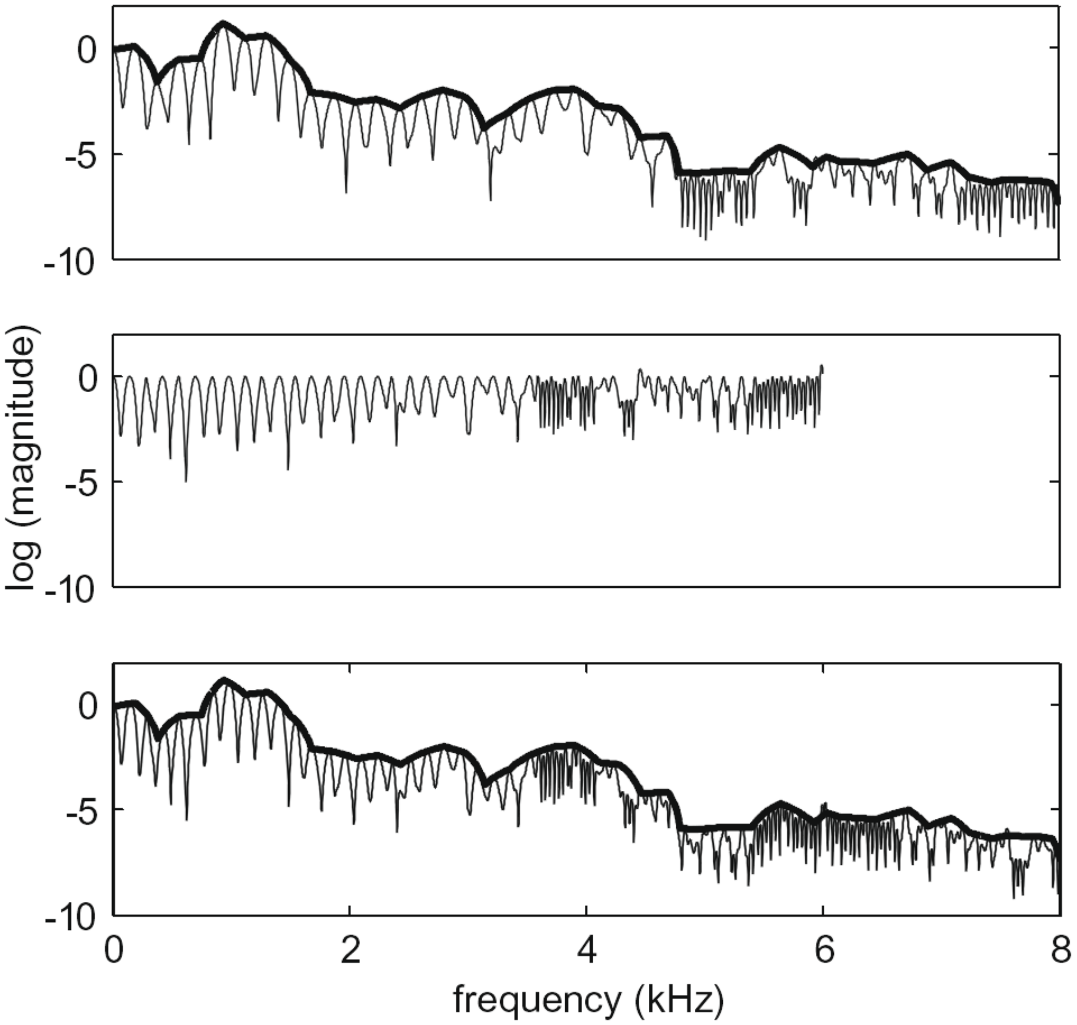
- Data compression
  - Filter parameters can be compressed (e.g., reflection coefficients)
  - The residual can also be compressed as a pulse train, though at the expense of lower synthesis quality
- Joint modification of pitch and of spectral envelope
- Independent time frames for spectral envelope estimation and for prosodic modification
- Fewer distortions, since LP-PSOLA operates on a spectrally flat residual rather than on the speech signal itself

Fourier-domain PSOLA

FD-PSOLA also operates in three stages

- Analysis
  - A complex ST spectrum is computed at the analysis pitch marks
  - A ST spectral envelope is estimated, via LP analysis, homomorphic analysis or peak-picking algorithms (SEEVOC)
  - A flattened version of the ST-spectrum is derived by dividing the ST complex spectrum by the spectral envelope
- Frequency modification
  - Flattened spectrum is modified so the spacing between pitch harmonics is equal to the desired pitch
  - This can be done using either *(i)* spectral compression-expansion, or *(ii)* harmonic elimination-repetition (see Moulines & Charpentier, 1990)
- Synthesis
  - Multiply flattened spectrum and spectral envelope
  - Obtain synthesis ST signal by inverse DFT

Pitch-scaling with FD-PSOLA



[Felps and Gutierrez-Osuna, 2009]

Performance

- FD-PSOLA solves a major limitation of TD-PSOLA: its inability to perform spectral modification
- These modifications may be used for several purposes
  - Smoothing spectral envelopes across diphones in concatenative synthesis
  - Changing voice characteristics (e.g., vocal tract length)
  - Morphing across voices
- However, FD-PSOLA is very computationally intensive and has high memory requirements for storage

Sinusoidal models

Introduction

- As we saw in earlier lectures, the Fourier series can be used to generate any periodic signal from a sum of sinusoids

$$x(t) = \sum_{l=1}^{L} A_l \cos(\omega_0 l + \phi_l)$$

- A family of techniques known as *sinusoidal models* use this as their basic building block to perform speech modification
  - This is achieved by finding the sinusoidal components $\{A_l, \omega_0, \phi_l\}$, and then altering them to meet the prosodic targets
- In theory, we could perform Fourier analysis to find model parameters
  - For several reasons, however, it is advantageous to use a different procedure that is more geared towards synthesis
- If the goal is to perform pitch-scaling, it is also advantageous to do the analysis in a pitch-synchronous fashion
  - The accuracy of pitch marks, however, does not have to be as high as for PSOLA

Finding sinusoidal parameters

- Components $\{A_l, \omega_0, \phi_l\}$ are found so as to minimize the error E

$$E = \sum_n w(n)^2 \big(s(n) - \hat{s}(n)\big)^2 =$$

$$\sum_n w(n)^2 \left(s(n) - \sum_{l=1}^{L} A_l \cos(\omega_0 l + \phi_l) \right)$$

  - which requires a complex linear regression; see Quatieri (2002)

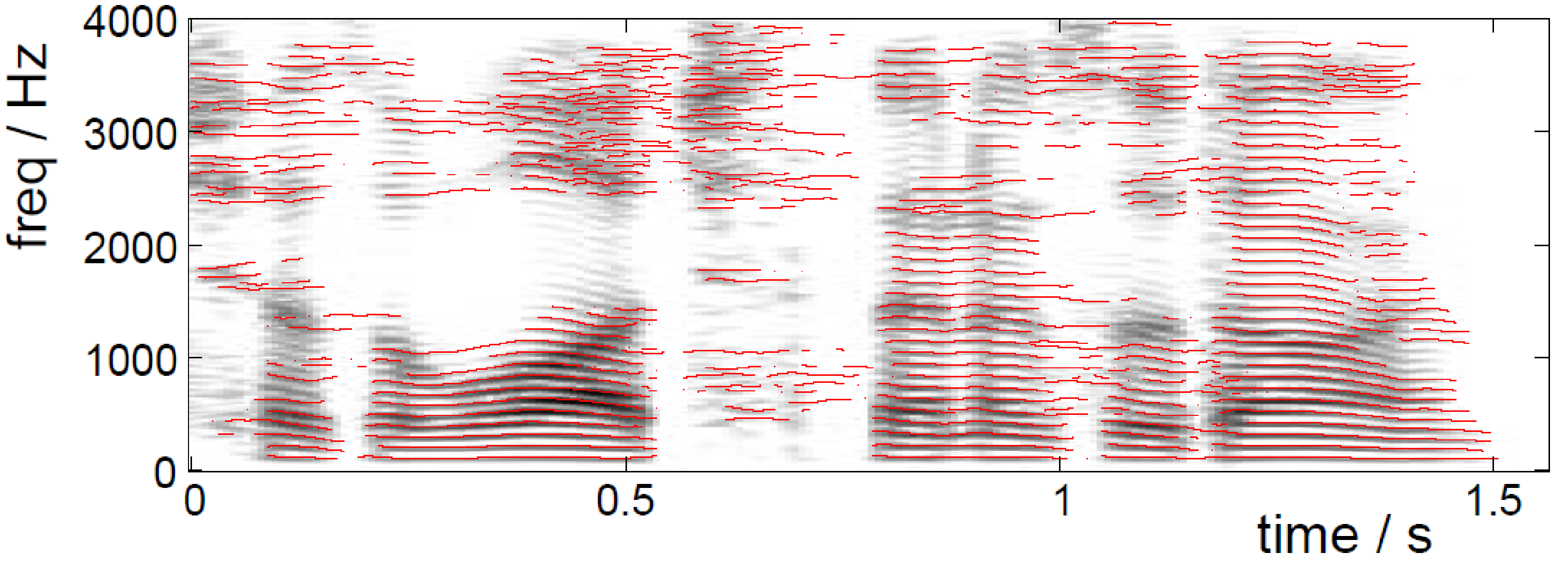
- Why use this analysis equation rather than Fourier analysis?
  - First, the window function $w(n)$ concentrates accuracy in the center
  - Second, this analysis can be performed on relatively short frames

- Given these parameters, a ST-waveform can be reconstructed using the synthesis equation $x(t) = \sum_{l=1}^{L} A_l \cos(\omega_0 l + \phi_l)$
  - An entire waveform can then be reconstructed by overlapping ST segments just as with TD-PSOLA

Modification

- Modification is performed by separating harmonics and spectral envelope, but without explicit source/filter modeling
- This can be done in a number of ways, such as by *peak-picking* in the spectrum to determine the spectral envelope
- Once the envelope has been found, the harmonics can be moved in the frequency domain and new amplitudes found from the envelope
- Finally, the synthesis equation can be used to generate waveforms

Sinewave modeling results

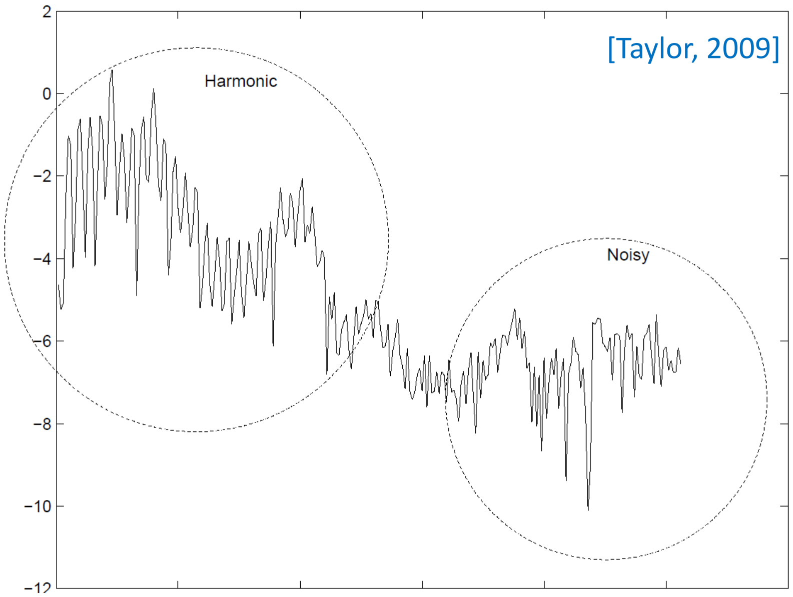


http://www.ee.columbia.edu/~dpwe/e6820/lectures/L05-speechmodels.pdf

Harmonic + noise models

Motivation

- Sinusoidal modeling works quite well for perfectly periodic signals, but performance degrades in practice since speech is rarely periodic
- In addition, very little periodic source information is generally found at high frequencies, where the signal is significantly noisier
  - This non-periodicity comes from several sources, including breath passing through the glottis and turbulences in the vocal tract



[Taylor, 2009]

Overview

- To address this issue, a stochastic component can be included

$$\hat{s}(t) = \hat{s}(t)_p + \hat{s}(t)_r = \sum_{l=1}^{L} A_l \cos(\omega_0 l + \phi_l) + s(t)_r$$

  - where the noise component $s(t)_r$ is assumed to be Gaussian noise

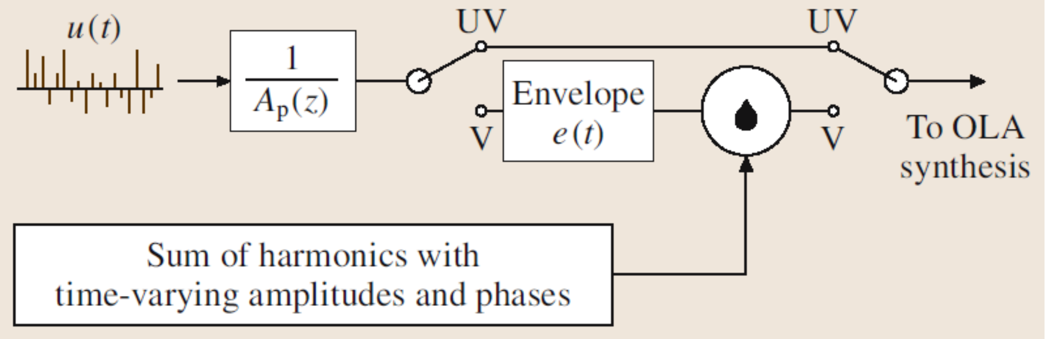
- A number of models based on this principle have been proposed
  - Multiband excitation (MBE) (Griffin and Lim, 1988)
  - Harmonic + noise models (HNM) (Stylianou, 1998)
- Here we focus on HNM, as it was developed specifically for TTS

Harmonic + noise model (HNM)

- HNM follows the same principle of harmonic/stochastic models
- The main difference is it also considers the temporal patterns of noise
  - As an example, the noise component in stops evolves rapidly, so a model with uniform noise across the frame will miss important details
- The noise part in HNM is modeled as

$$s(t)_r = e(t)[h(t,\tau) \otimes b(t)]$$

  - where $b(t)$ is white Gaussian noise
  - $h(t,\tau)$ is a spectral filter applied to the noise (generally all-pole), and
  - $e(t)$ is a function that gives filtered noise the correct temporal pattern



[Dutoit, 2008, in Benesty et al., (Eds)]

Analysis steps

- First, classify frames as V/UV
  - Estimate the pitch in order to perform pitch-synchronous (PS) analysis
    - With HNM, however, there is no need for accurate epoch detection; the location of pitch periods suffices since phases are adjusted later on
  - Using the estimated pitch, fit a harmonic model to each PS frame
  - From the residual error, classify the frame as V/UV
    - Approach: UV frames will have higher residual error than V frames
- For V frames, determine the highest harmonic frequency
  - Approach: move through the frequency range and determine how well a synthetic model fits the real waveform
- Estimate model parameters
  - Refine pitch estimate using only the part of the signal below the cutoff
  - Find amplitudes and phases by minimizing the error E
  - Find components $h(t)$ and $e(t)$ of the noise term

- And finally, adjust phases
    - Since the pitch synchronous analysis was done without reference to a fixed epoch, frames will not necessarily align
    - To adjust the phase, a time domain technique is used to shift the relative positions of waveforms within their frames

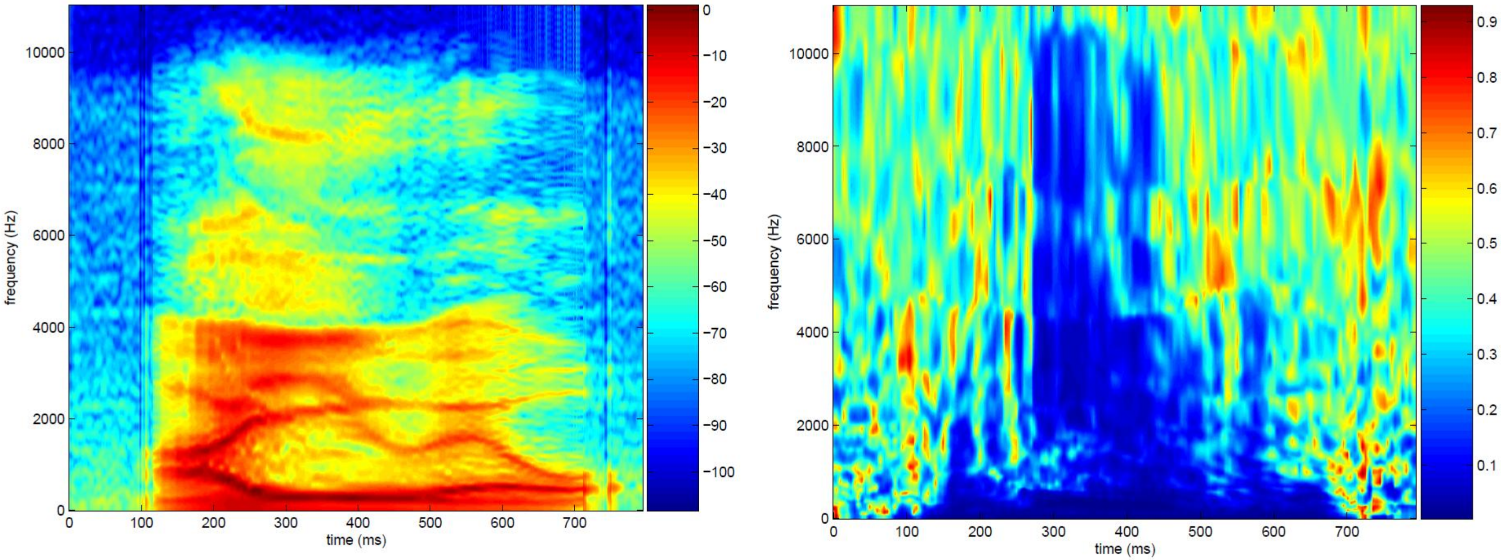
Synthesis steps

- As in PSOLA, determine synthesis frames and mapping $M[i]$
- To perform time-scaling, proceed as with PSOLA
- To perform pitch-scaling
  - Adjust the harmonics on each frame
  - Generate noise component by passing WGN $b(t)$ through the filter $h(t)$
  - For V frames, high-pass-filter the noise above the cutoff to remove its low-frequency components
  - Modulate the noise in the time domain to ensure synchrony with the harmonic component
    - This step is essential so a single sound (rather than two) is perceived
  - Finally, synthesize ST frame by a conventional overlap-add method

STRAIGHT

Overview

– STRAIGHT is a high-quality vocoder that decomposes the speech signal into three terms
  - A smooth spectrogram, free from periodicities in time and frequency
  - An $F0$ contour, and
  - A time-frequency periodicity map, which captures the spectral shape of the noise and also its temporal envelope



[Hawahara, 2007]

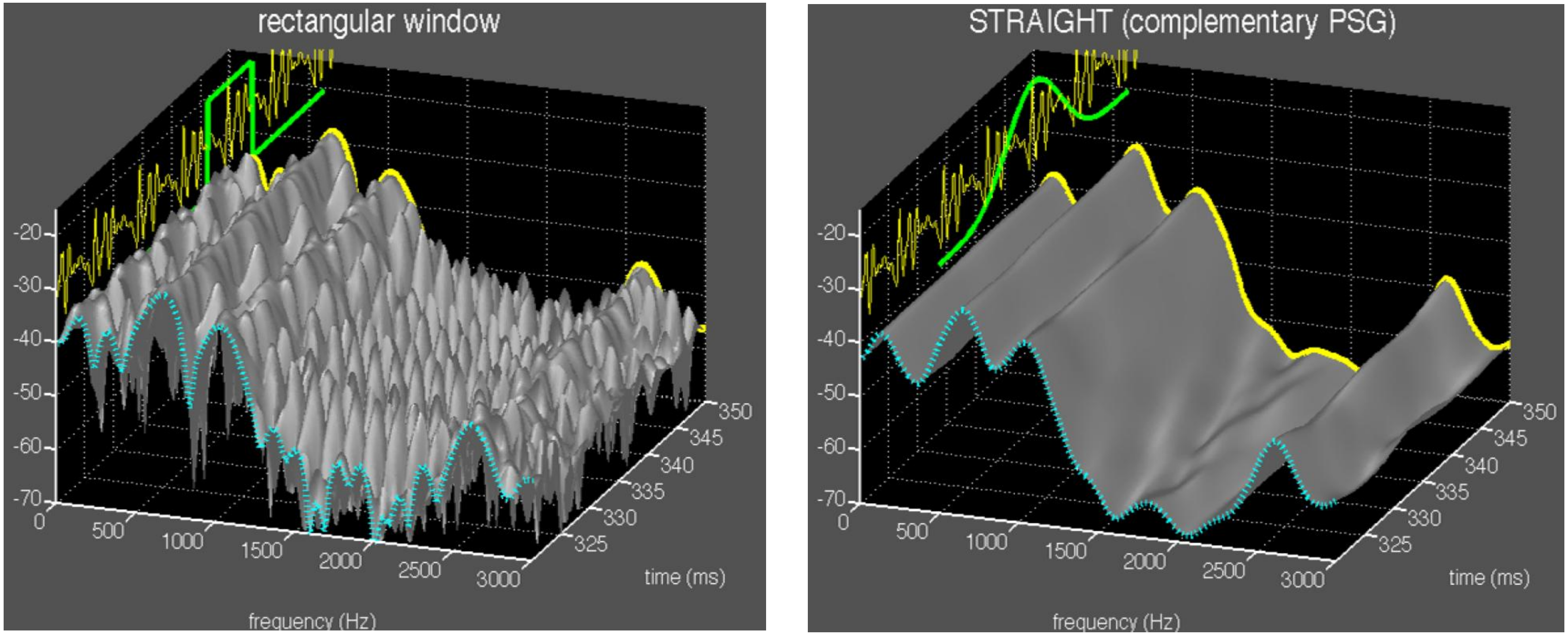
During analysis

- $F0$ is accurately estimated using a fixed-point algorithm
- This $F0$ estimate is used to smooth out periodicity in the ST spectrum using an $F0$-adaptive filter and a surface reconstruction method
- The result is a smooth spectrogram that captures vocal-tract and glottal filters, but is free from $F0$ influences

During synthesis

- Pulses or noise with a flat spectrum are generated in accordance with voicing information and $F0$
- Sounds are resynthesized from the smoothed spectrum and the pulse/noise component using an inverse FFT with an OLA technique
- Notes
  - STRAIGHT does not extract phase information, instead uses a minimum-phase assumption for the spectral envelope and applies all-pass filters in order to reduce buzz timbre

Conventional vs. STRAIGHT spectrogram



[Hawahara, 2002]

Performance

- Prosodic modification with STRAIGHT is very simple
  - Time-scale modification reduces to duplicating/removing ST slices from the STRAIGHT spectrogram and aperiodicity
  - Pitch-scale modification reduces to modifying the $F0$ contour
  - Following these modifications, the STRAIGHT synthesis method can be invoked to synthesize the waveform
- The three terms in STRAIGHT can be manipulated independently, which provides maximum flexibility
- STRAIGHT allows extreme prosodic modifications (up to 600%) while maintaining the naturalness of the synthesized speech
- On the downside, STRAIGHT is computationally intensive