Sinusoidal Modelling and Synthesis

Johannes Luig

Amir Rahimzadeh

Seminar Work for "Algorithmen in Akustik und Computermusik II, SE"
June 2008

1 Introduction

Sinusoidal modelling and Synthesis (SMS) is a signal analysis and synthesis framework where a speech- or music signal is represented as a sum of sinusoids each with time-varying amplitude, frequency and phase.

The goal is modelling a signal by a reduced set of parameters yielding a more compact representation of the data.

Once a time-frequency representation of the signal is available it is usefull in different ways. The reduced amount of data is of interest for transmission or storage purposes. Also various digital audio effects like pitch-shifting, time stretching and sound morphing could easily be applied using the time frequency data.

It is common to analyze time varying signals framewise. A frame is a segment of the signal ranging from a few milliseconds up to 0.5 seconds depending on the application. These frames are analyzed in the frequency domain applying a Fast Fourier Transform (FFT). The relevant components are found by applying a peak picking algorithm to the spectra and the corresponding amplitudes, frequencies and phases are extracted. Then the individual analysis points on the resulting time-frequency map are connected to form tracks.

For the syntesis the values between the analysis points have to be interpolated in order to get a timegrid that satisfies the original sampling rate. Finally the sound can be synthesized using a bank of oscillators which is fed with the parameters extracted before namely, the corresponding amplitudes, frequencies and phases.

The following section recalls the basics of sinusiodal modeling and frame-by-frame signal processing, before a basic SMS algorithm will be presented in section 3. In section 4, improvements of the basic algorithm concerning frequency estimation, track continuation will be explained and finally a signal analysis method that overcomes the time- vs. frequency resolution tradeoff of plain FFT analysis for deterministic signals is presented.

2 Basics

2.1 Why Sinusoids?

In general the goal of modelling a signal is to reduce redundancy and to get a more compact representation of the data. There are different techniques to model a time series and it depends on the signal which technique to apply.

Sinusoids are especially suited for modelling sounds with harmonic content. Most natural acoustical sounds exhibit this attribute and the reason for this sinusoidility can be found in the way of the sound production.

Human voice production system consists of two fundamental parts working together, namely the voice chords (the excitation source) and the pharynx with mouth and nasal cavaties acting as acoustical filter. During voiced parts of speech the vocal chords are opening and closing at a certain frequency (the "fundamental frequency", f_0) modulating the airstraim coming from the lungs. The harmonic overtone structure results from the structure of the pharynx which can be seen as a open tube in a simplified way, letting develop all overtones $f_1 - f_n$ being integer multiples of the fundamental f_0.

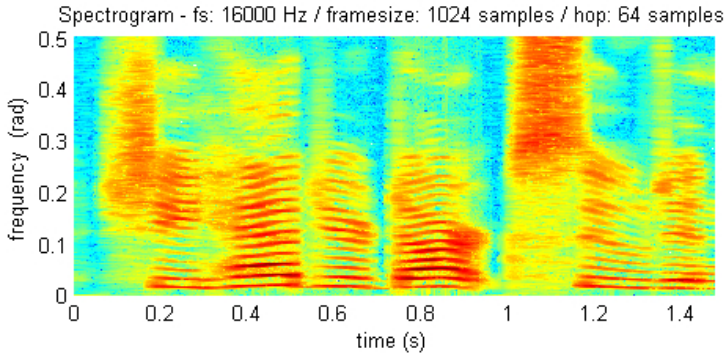

Figure 1: Spectrogram of human speech (1.5 seconds)

Sounds of musical instruments show a comparable spectral sound structure due to the sound production mechanism. There is an (constant or singular) excitation to a mass spring system which vibrates in the frequencies possible due to the limitations of the physical system (mass, stiffness, length).

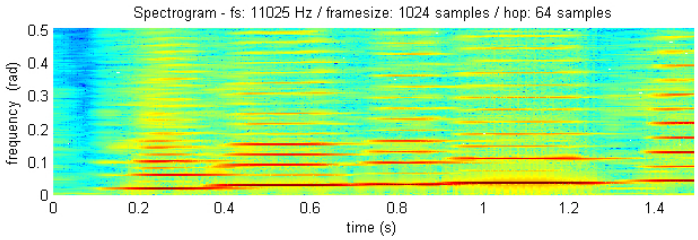

Figure 2: Spectrogram of a clarinet sound (1.5 seconds)

2.2 Sinusoidal Modeling

The basic idea of time series modelling is to represent a signal by a reduced set of parameters which in the case of sinusoidal modelling are the frequencies, amplitudes and phases of a set of sinusoids. The resulting signal can be written as

$$s(n) = \sum_k P_k(n) = \sum_k a_k(n) \cos(\phi_k(n)) \ . \tag{1}$$

A good measure for the quality of the estimate is to calculate the mean squared error which is the squared difference between the original signal and the signal model averaged over time:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{2}$$

The question is how to choose the appropriate model parameters in order to minimize this error function. The answer can be found in the spectrum of the signal. As can be seen from the spectrograms (Fig. 1 and 2), most energy in voiced speech and harmonic sounds is concentrated on a few quasi-stationary sinusoids. In a single spectrum these sinusoids appear as peaks. That means that the appropriate model parameters to minimize the error can be drawn directly from the spectral representation of the sound. Nevertheless there are certain aspects in the analysis of a sound that influence the accuracy of estimated parameters which will be discussed in the following sections.

2.2.1 The Analysis Window

Speech and musical sounds exhibit spectral content that is varying quickly. Therefore analysis has to be taken on a short time basis on segments of the original waveform. Therefore signal is multiplied with a window function limiting the support of samples. This has two effects. On the one hand noise is introduced if the segment length is not an integer multiple of the fundamental period of the signal causing pure sinusoids widen up in the spectrum.

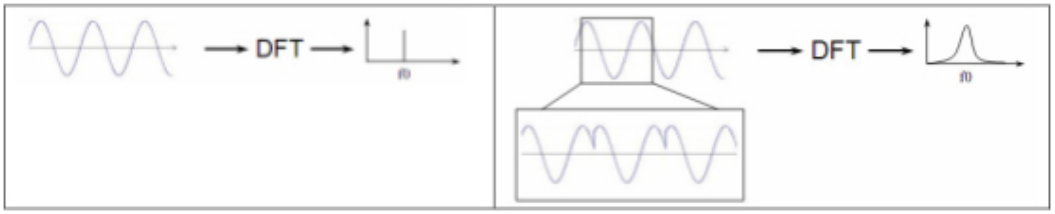

Figure 3: DFT of a pure infinite sinusoid in theory (left) and of a segmented sinusoid with segment length $\neq k \cdot T_0$.

On the other hand, multiplication in the time domain results in convolution in the frequency domain. Hence, a pure sinusoid is smeared in the frequency domain due to the convolution with the DFT of the window function which is a sinc-function.

The sidelobes of the sinc-function are the reason for the widening of the sinusoid causing FFT channel cross-talk. The heigth of sidelobes can be reduced by using tapered windows which apply a weight to the time series resulting in clear peaks in the spectrum.

The most commonly used windows are called Rectangular, Triangular, Hamming, Hanning, Kaiser, Blackman and Chebyshev. They differ mainly in two aspects namely the width of the mainlobe and the heigth of the sidelobe with respect to the main lobe which are measured in FFT bins and dB respectively.

The rectangular window has the narrowest mainlobe (2 bins, needed for good frequency resolution) but also the highest sidelobes ($-13dB$) of all window functions causing FFT cross-channel talk reducing the ability to distinguish two sinusoidal components lying close together in the frequency domain. The Hamming window for comparison has a wider main lobe, 4 bins,

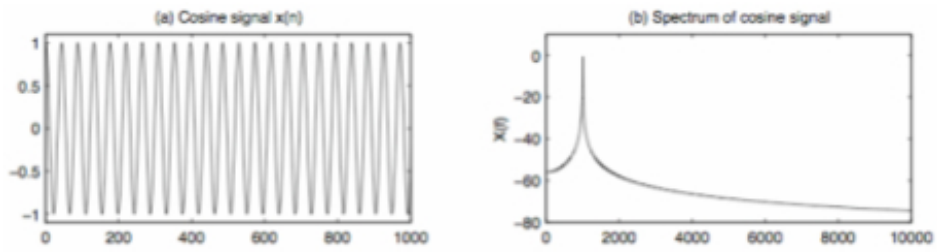

Figure 4: Cosine signal (left) and corresponding spectrum (right).

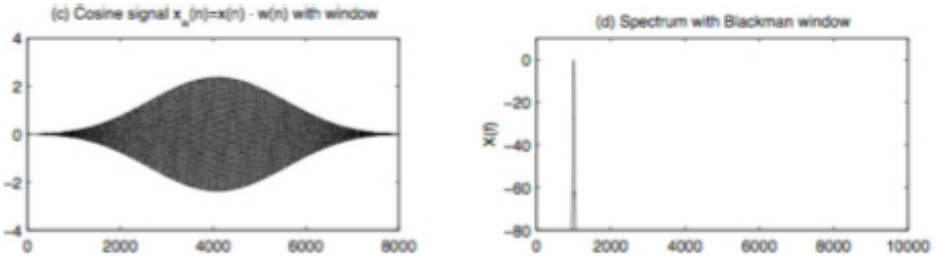

Figure 5: Windowed cosine signal (left) and corresponding spectrum (right).

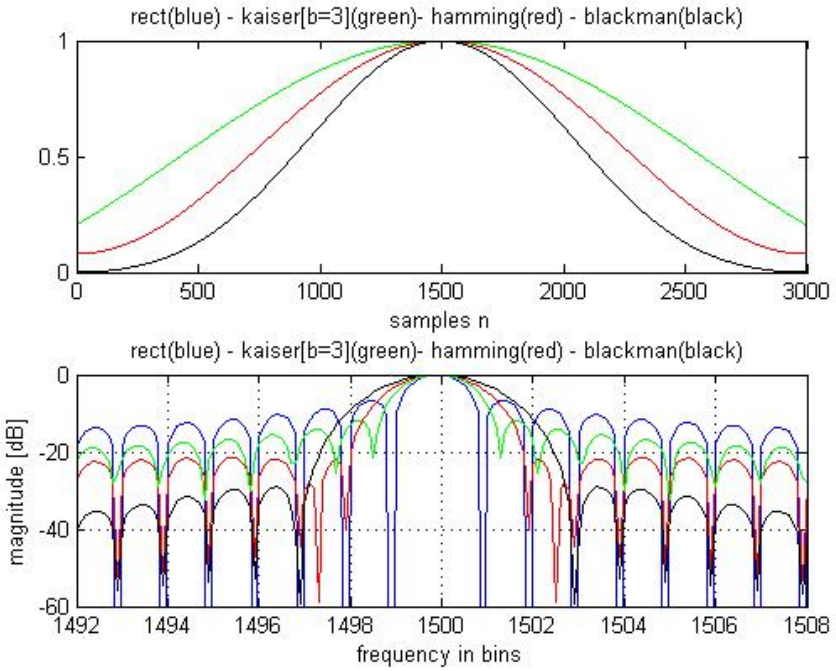

Figure 6: Comparison of different window types both in time and frequency domain.

and the highest side-lobe is $42dB$ down. So the shape of the window is of large importance and should be chosen according to the application.

2.2.2 The Window Length

Once a window type is chosen, it is easy to calculate how many samples are needed to achieve a desired frequency resolution. To "resolve" two sinusoids separated in frequency by Δ Hz, we need (in noisy conditions) two clearly discernible main lobes; (i.e., like in Fig. 7)

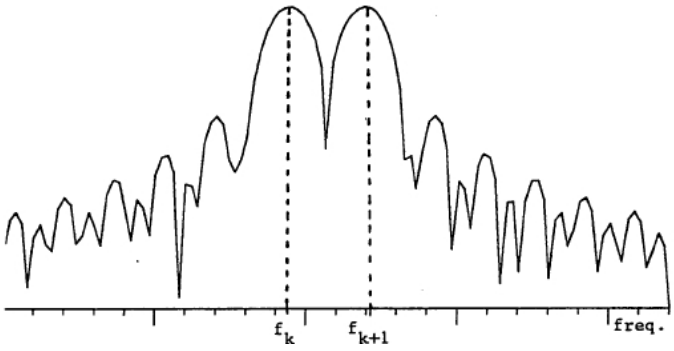

Figure 7: Spectrum of two clearly separated sinusoids. (from [2])

To obtain the separation shown (main lobes meet near a zero-crossing), we require a main-lobe bandwidth Bf in Hz such that $B_f \leq \Delta$. In more detail, we have

$$B_f = K\frac{f_s}{M} \tag{3}$$

$$\Delta = f_2 - f_1 \tag{4}$$

where K is the main-lobe bandwidth (in bins), f_s the sampling rate, M is the window length, and f_1, f_2 are the frequencies of the sinusoids. Thus, we need

$$M \geq K\frac{f_S}{\Delta} = K\frac{f_s}{f_2 - f_1} \ . \tag{5}$$

That means that resolving two harmonics f_k and f_{k+1} of a fundamental f_1 with a frequency relation of $f_1 = f_{k+1} - f_k = \Delta$ requires a bandwidth $B_f \leq f_1$ and hence a number of samples $M \geq K \cdot f_s/f_1$.

The ratio between sampling frequency and fundamental- or difference frequency can also be interpreted as period ratios since

$$\frac{f_s}{f_1} = \frac{T_1}{T} = P \tag{6}$$

where P is the period in samples.

More generally, that means to resolve any 2 sinusoids we need at least K periods of the difference frequency $|f_2 - f_1|$ under the window ($M \geq KP$).

A last but important thing to mention concerning the length of windows is the difference between even and odd length windows. An odd length window is centered around the middle

sample while a even length one has no sample value at this position. Moreover for phase detection purposes a zero phase window is often preferred which is attained most naturally by using a symmetric windows with one sample at time origin, hence an odd-length window.

3 The McAulay-Quatieri Algorithm

3.1 Overview

In 1986, Robert McAulay and Thomas Quatieri [1] presented an approach to speech analysis and synthesis based on a sinusoidal representation, which should be designated as the "basic sinusiodal model" in many following publications.

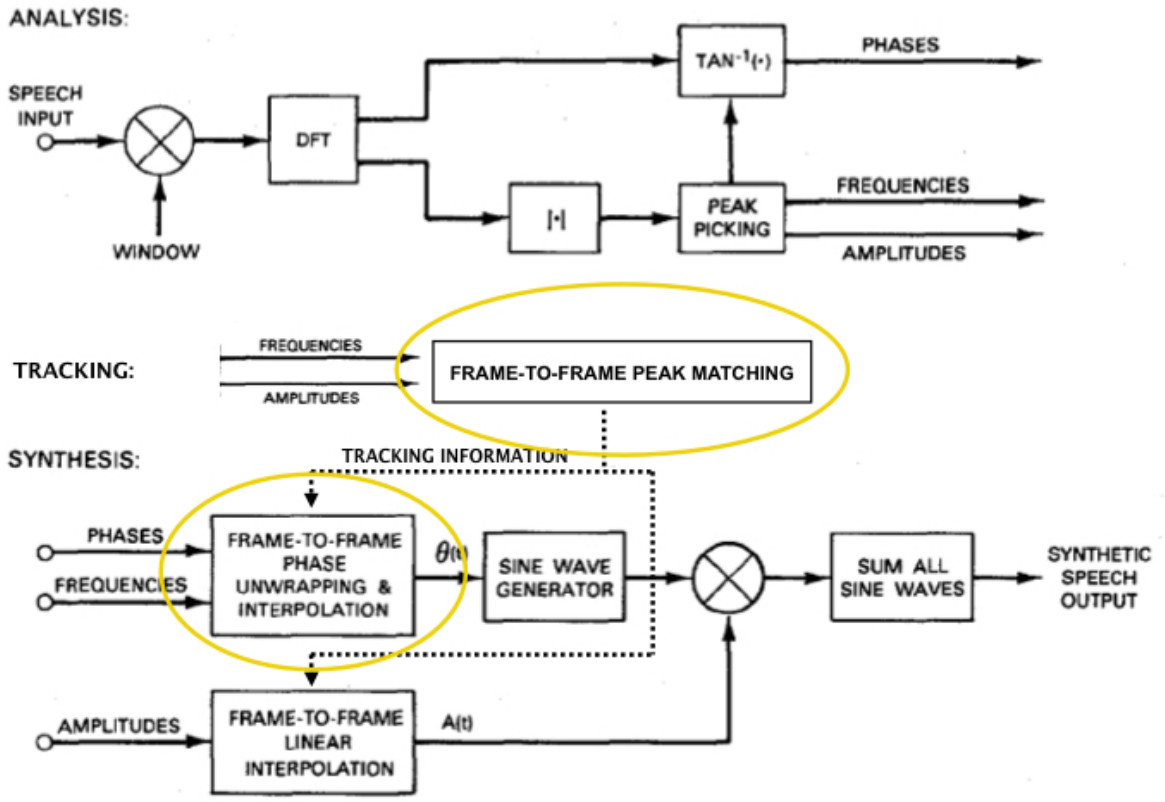

Figure 8: Schematic of the McAulay/Quatieri algorithm. The marked blocks are of special relevance and will be discussed in detail. (from [1], extended)

The main idea behind this method is to estimate the sine wave components of a sound by extracting the amplitudes, frequencies and phases from its Short-Time Fourier Transform (STFT) using a simple peak-picking algorithm. Therefore, narrowband components (peaks in the magnitude spectrum) are tracked, before a cubic function is used to unwrap and interpolate the phase in a "smooth" way.

3.2 Tracking

3.2.1 The Concept

Since the number of existing partials will hardly be constant from frame to frame, matching corresponding peaks in successive frames means more than just sorting the peaks by frequency. In practice, there are spurious peaks that come and go due to the effects of sidelobe interaction, the peak locations change as the pitch changes; not to mention transitions between voiced and unvoiced time segments.

In order to account for rapid variation in the partial peaks, McAulay and Quatieri introduced the concept of "birth" and "death" of sinusoidal components.

3.2.2 The Method

To be able to decide whether a partial track is continued or aborted, a *matching interval* around the current frequency is defined to avoid "jumps" in the partial track contour. This section describes the process of matching each frequency in frame k to some frequency in frame $k+1$, which consists of three steps.

To avoid ambiguities, a frequency ω is specified by an additional frame index (in the superscript) and a track index (in the subscript). In this notation, the task is then the assignment $\omega_n^k \rightarrow \omega_m^{k+1}$.

First Step: Binary Check First of all, the following frame $(k+1)$ is browsed for peaks within the predefined matching interval around ω_n^k. If this is not the case, the current track is declared "dying"; whereas otherwise, the peak with the "closest" frequency is marked as a *candidate match* and the remaining steps are executed.

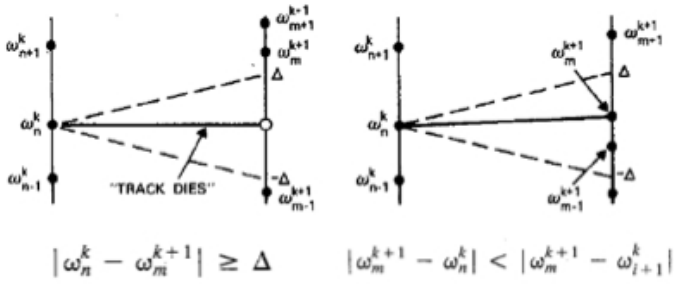

Figure 9: Two possible results of step 1: Dying (left) and continuing track (right) (from [1])

Second Step: Candidate Verification Once a tentative match ω_m^{k+1} has been found, it has to be verified that the is no better match in the current frame k. If this is true, the algorithm checks whether more than one frequency lies within the matching interval. If so, a definitive match is made by choosing the frequency with the smallest Euclidean distance; if not, the track "dies" (see Fig. 10).

Third Step: Anyone left? When a successor has been found for each frequency in the current frame k, the following frame $k+1$ is checked for "un-matched" frequencies. Since

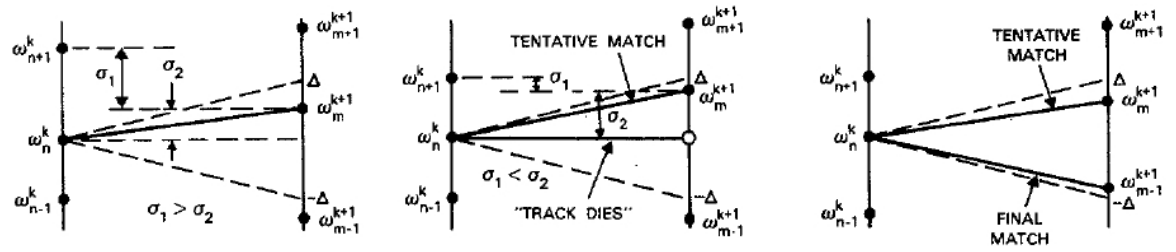

Figure 10: Left: the best match has initially been found. Center: no other frequency within matching interval. Right: Lucky boy. (from [1])

such a frequency has no predecessor in frame k, it marks the begin of a new partial track, as depicted in Fig. 11.

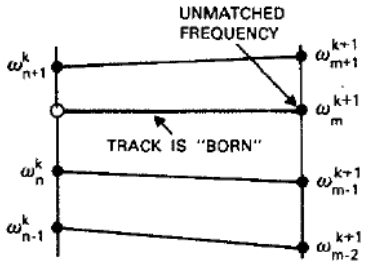

Figure 11: "Birth" of a new partial track. (from [1])

3.3 Parameter Interpolation

The straight-forward synthesis approach

$$\tilde{s} = \sum_{l=1}^{L(k)} \hat{A}_l^k \cos\left[n\hat{\omega}_l^k + \hat{\theta}_l^k\right] \tag{7}$$

(where L indicates the total number of detected peaks in frame k) leads to discontinuities at the frame boundaries due to the time-varying nature of the parameters.

Since a simple overlap-and-add synthesis system is only suitable when using a very short hop size (i.e., a high frame rate), McAulay and Quatieri introduce the following approach to smoothly interpolate the parameters.

3.3.1 Amplitude Interpolation

An easy and sufficient solution to the interpolation problem for the amplitudes is doing it the linear way:

$$\hat{A}_l(n) = \hat{A}_l^k + \frac{\hat{A}_l^{k+1} - \hat{A}_l^k}{S}\ n\ , \tag{8}$$

where $n = 0, 1, \ldots, S-1$ represents the sample index within the frame.

3.3.2 Frequency and Phase Interpolation

This approach does not work for interpolation of the remaining two parameters, however, since the measured phase is always obtained in a "wrapped" form, (i.e., modulo 2π). The strategy proposed by the authors is to perform phase unwrapping and to formulate a cubic phase interpolation function. This action performs frequency interpolation at the same time, since the derivative of the unwrapped phase equals the instantaneous frequency.

A third-order polynomial describing the evolution of the phase track over time and its derivative write to[1]

$$\tilde{\theta}(t) = \zeta + \gamma t + \alpha t^2 + \beta t^3 \ , \tag{9}$$

$$\dot{\tilde{\theta}}(t) = \gamma + 2\alpha t + 3\beta t^2 \ , \tag{10}$$

which yield at the frame boundaries (inserting 0 and T for the time variable t)

$$\tilde{\theta}(0) = \zeta = \hat{\theta}^k \ , \tag{11}$$

$$\dot{\tilde{\theta}}(0) = \gamma = \hat{\omega}^k \ , \tag{12}$$

and

$$\tilde{\theta}(T) = \hat{\theta}^k + \hat{\omega}^k T + \alpha T^2 + \beta T^3 = \hat{\theta}^{k+1} + 2\pi M \ , \tag{13}$$

$$\dot{\tilde{\theta}}(T) = \hat{\omega}^k + 2\alpha T + 3\beta T^2 = \hat{\omega}^{k+1} \ . \tag{14}$$

As already mentioned, the measured phase $\hat{\theta}^{k+1}$ has to be unwrapped. This means, an integer factor M has to be determined in a way that the resulting phase evolution over time is as smooth as possible. The ideal case, of course, would be a constant frequency and thus a linear phase; a criterion for the choice of M could be formulated as

$$f(M) = \int_0^T \left[\ddot{\tilde{\theta}}(t, M)\right]^2 dt = \min \ , \tag{15}$$

minimizing the frequency change in the interval $[0, T]$. Solving 13 and 14 for α and β leads to

$$\begin{bmatrix} \alpha(M) \\ \beta(M) \end{bmatrix} = \begin{bmatrix} \frac{3}{T^2} & \frac{-1}{T} \\ \frac{-2}{T^3} & \frac{1}{T^2} \end{bmatrix} \begin{bmatrix} \hat{\theta}^{k+1} - \hat{\theta}^k - \hat{\omega}^k T + 2\pi M \\ \hat{\omega}^{k+1} - \hat{\omega}^k \end{bmatrix} \ , \tag{16}$$

and the value minimizing 15 is obtained through

$$m = \frac{1}{2\pi}\left[(\hat{\theta}^k - \hat{\omega}^k T - \hat{\theta}^{k+1}) + (\hat{\omega}^{k+1} - \hat{\omega}^k)\frac{T}{2}\right] \ . \tag{17}$$

The desired factor M is then chosen as the nearest integer to m. Having performed all this algebra, we can finally write down the cubic phase interpolation function as

$$\tilde{\theta}(t) = \hat{\theta}^k + \hat{\omega}^k t + \alpha(M) t^2 + \beta(M) t^3 \ . \tag{18}$$

A typical set of phase interpolation functions is depicted in Fig. 12.

[1] In the following equations, the track index l is omitted for convenience.

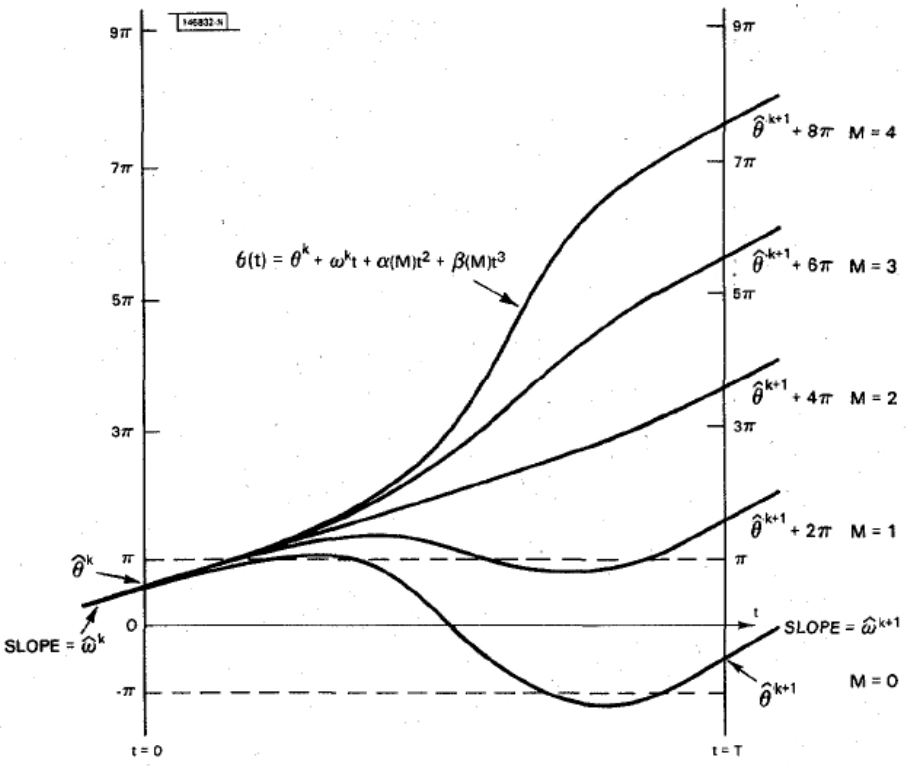

Figure 12: The smoothest interpolation function results for $M = 2$. (from [1])

3.3.3 Final Result

The final synthetic waveform is given by

$$\tilde{s}(n) = \sum_{l=1}^{L(k)} \tilde{A}_l(n) \cos\left[\tilde{\theta}_l(n)\right] \tag{19}$$

where $\tilde{A}_l(n)$ is the result of 8 and $\tilde{\theta}_l(n)$ is computed by solving 18.

4 Improvements

4.1 Improving Frequency Resolution

The frequency estimation accuracy of the analysis stage is an important factor for the resulting audio quality when synthesizing sound from their parameter representation. Unfortunately the bins obtained from the STFT are only accurate within +/- half a bin due to the sampled nature of discrete spectra:

$$res_{STFT} = \frac{f_s}{N} \qquad \text{with} \quad \begin{array}{lll} f_s & ... & \text{sampling frequency} \\ N & ... & \text{frame length in samples} \end{array} . \tag{20}$$

The simplest strategy to increase the frequency resolution is to increase the number of samples per frame by adding zeros at the end of the frame which is called *zero-padding*. The computational load increases as the number of samples increases; the computational complexity for an N-point FFT calculation using the radix-2 approach is given by

$$\mathcal{O}(\frac{N}{2} \cdot \log_2 N) \tag{21}$$

where N is the length of the transform. Therefore high zero-padding factors necessary to meet the required frequency resolution are a very inefficient way to increase precision. The example below shall demonstrate that:

$$\begin{aligned} f_s &= 22050Hz && \dots \text{sampling frequency} \\ T &= 10ms && \dots \text{frame length} \\ N &= 0.01s \cdot 22050\tfrac{1}{s} = 221 && \dots \text{number of samples per frame} \\ res_{FFT} &= \tfrac{22050Hz}{221} \approx 100Hz && \dots \text{frequency resolution without zero-padding} \end{aligned}$$

Let the desired frequency resolution $res_{FFT,des}$ be $1Hz$, i.e., $N_{des} = 22050samples$. The resulting zero-padding factor Z is then computed by

$$Z = \frac{N_{des}}{N} = \frac{22050}{100} \approx 220 \ . \tag{22}$$

Another more efficient way to increase the resolution is to use small zero-padding factors together with an interpolation scheme for finding the real maximum which is lying somewhere between two bins. One of these methods is parabolic interpolation.

4.2 Parabolic Frequency Interpolation

Parabolic interpolation is a very simple interpolation method which reduces the computational load drastically compared to zero padding. A parabola is fit through the highest three samples of a peak to estimate the true peak location and height.

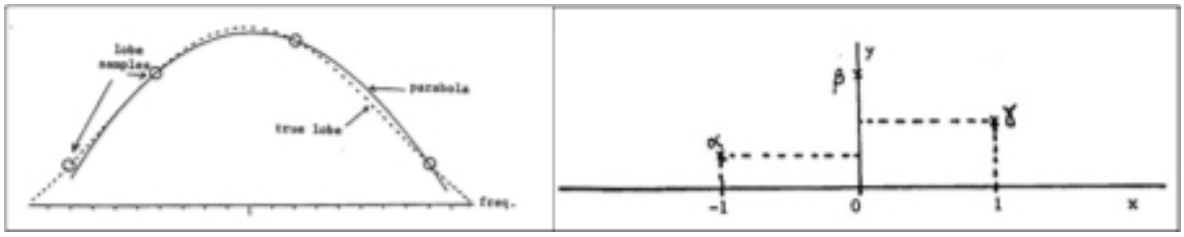

Figure 13: Left: comparison of true lobe and fitted parabola. Right: the parabola is fit through the three surrounding samples (from [2])

To describe the parabolic interpolation strategy, let's define a coordinate system centered at $(\beta, 0)$, where β is the bin number of the spectral magnitude maximum. A general parabola of the form

$$y(x) = a(x - p)^2 + b \tag{23}$$

is desired, such that

$$y(-1) = \alpha, \quad y(0) = \beta, \quad \text{and} y(1) = \gamma \tag{24}$$

where α, β, γ are the values of the three highest samples surrounding the true maximum.

Solving the parabola for p the peak location gives:

$$p = \frac{1}{2}\frac{\alpha - \gamma}{\alpha - 2\beta + \gamma} \ . \tag{25}$$

The estimate of the true peak location (in bins) will be

$$k^* = k_\beta + p \ , \tag{26}$$

the peak frequency in Hz is $\frac{f_s k^*}{N}$. Using p, the peak height estimate is then

$$y(p) = \beta - \frac{1}{4}(\alpha - \gamma)p \ . \tag{27}$$

Alternatively, $y(p)$ can be computed separately for the real and imaginary parts of the complex spectrum to yield a complex-valued peak estimate (magnitude and phase).

According to [2], maxima found using dB magnitude for the interpolation are about twice as accurate as using linear magnitude values. One question remains, if there is an optimal nonlinear compression to the magnitude spectrum when using quadratic interpolation to find peak locations.

4.3 Time-Frequency Reassignment

4.3.1 Motivation

The Short-Time Fourier Transform (STFT), which is in most cases used as the basis for a signal representation in the time-frequency domain, suffers from what is known as the *Heisenberg uncertainty principle*: the fact that we have a trade-off between time and frequency resolutions. On one hand, a good time resolution requires a short windowing function; on the other hand, a good frequency resolution requires a narrow-band filter, i.e., a long windowing function.

Choosing a long window but keeping the hopsize small is a reasonable attempt to cope with this problem, but still does not resolve the issue of temporal "smearing" introduced due to the window length.

4.3.2 Background

The reassignment method uses the partial derivatives of the short-time *phase* spectrum, which is often completely neglected, to specify the original position of the time-frequency component within the analysis window rather than locating them at the geometrical center of the analysis window.

This is done by assuming that the region with the slowest phase variation within the window (the so-called "center of gravity") contributes most to the analysis result. The location of this point can be determined by computing

$$\hat{t} = \tau - \frac{\partial\phi(\tau,\omega)}{\partial\omega} \quad \text{and} \tag{28}$$

$$\hat{\omega} = \frac{\partial\phi(\tau,\omega)}{\partial\tau} \ , \tag{29}$$

which are nothing else than the group delay and the instantaneous frequency, respectively.

4.3.3 Computation

The correction terms for time and frequency are effectively obtained by computing the ratios of two "adjusted" STFTs and the "standard" Short-Time Fourier Transform (the complex conjugate is applied to avoid complex division):

$$\hat{t}_{k,n} = n - \Re\left\{\frac{X_{t,n}(k)X_n^*(k)}{|X_n(k)|^2}\right\} \tag{30}$$

$$\hat{\omega}_{k,n} = k + \Im\left\{\frac{X_{t,n}(k)X_n^*(k)}{|X_n(k)|^2}\right\}, \tag{31}$$

where $X_{t,n}(k)$ and $X_{f,n}(k)$ represent a time- and frequency-corrected STFT, respectively. These corrected STFT matrices are computed using special analysis windows: for the time-corrected STFT, the windowing function is scaled by a time ramp from $-\frac{N-1}{2}$ to $\frac{N-1}{2}$ (for an odd N); while for the frequency-corrected STFT, the ramp is muliplied in the frequency domain. The latter can also be achieved by differentiating the windowing function with respect to (discrete) time:

$$h_t(n) = nh(n) \qquad \text{and} \qquad h_f(n) = \frac{dh(n)}{dn}\,. \tag{32}$$

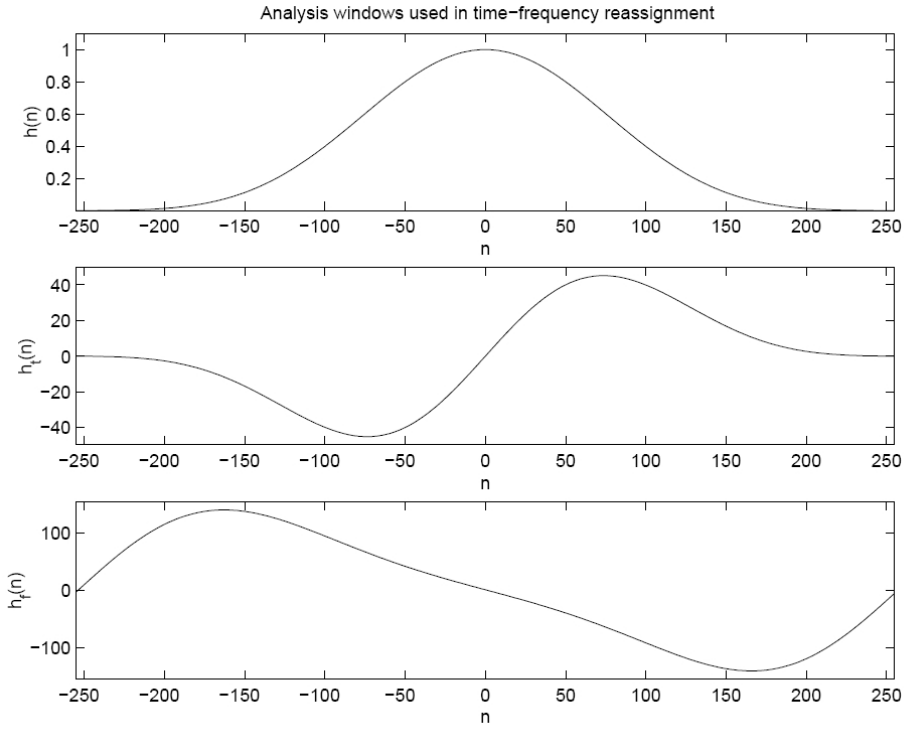

Figure 14: Comparison of the original (top) and the modified windowing functions (middle: time-scaled, bottom: frequency-scaled), for a 501-point Kaiser window with shaping parameter 12. (from [3])

4.3.4 Results

To visualize the benefit of the discussed approach, we compare this method to the "classic" sinusiodal model presented in section 3. It can clearly be seen that the reassigned data is distributed continuously in time and frequency, whereas the basic model is confined to the discrete time-frequency grid, which itself depends on the respective resolution due to the analysis window.

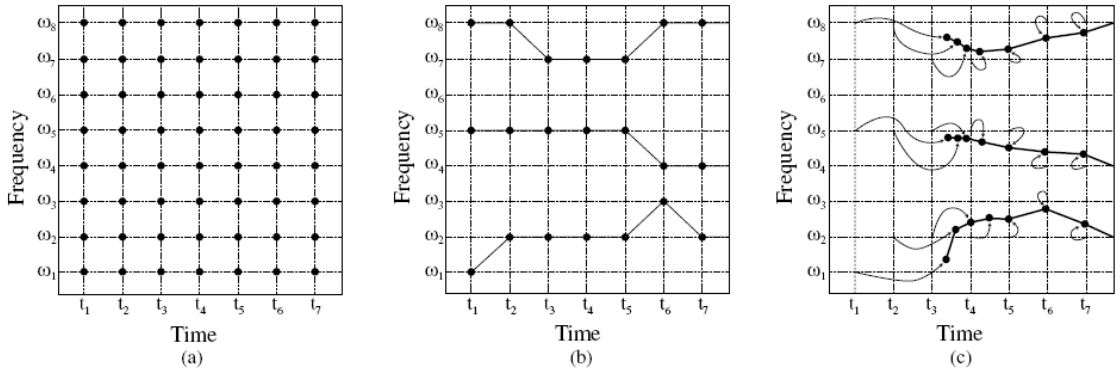

Figure 15: Partial tracks in the STFT representation (left), analyzed using the McAulay-Quatieri method (center) and the time-frequency reassigned version (right). (from [3])

4.3.5 Cropping

Since off-center components (i.e., those with a center of gravity far from the window center) can be easily identified due to the large time correction value, it makes sense to remove this unreliable data if it is likely to be better represented in the following (overlapping) frame.

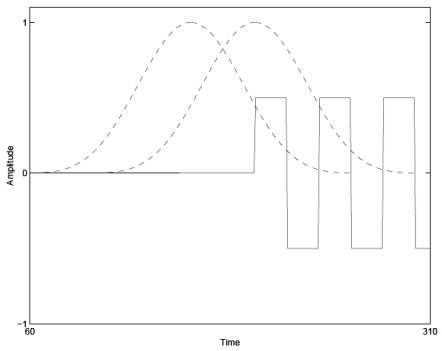

Figure 16: The abrupt turn-on of the square wave signal will not be represented well in the first depicted analysis window, but much better in the second one. (from [3])

4.4 Linear Prediction

4.4.1 Motivation

The basic approach presented in section 3 implies the assumption that the frequency and amplitude trajectories of the single partial tracks are constant from frame to frame; i.e., the center of the matching interval is positioned exactly at the frequency ω_n^k, whose evolution over time is currently evaluated.

However, for the frequency, this stability is rarely the case, since for many musical instruments (as well as singing voice), vibrato or portamento are commonplace; the same instationarity is true for the amplitude.

To yield a more precise tracking even for polyphonic sounds, an approach introduced by [4] predicts the parameters of the partials not just by considering the past value, but by taking into account a *linear combination* of past values.

4.4.2 LPC Method

To predict the temporal evolution of the partial tracks, the *Burg Method* is chosen. According to the authors, it combines the advantages of the auto- and the cross-correlation method: as the autocorrelation method, the Burg method is minimal phase ($\forall i, a_i < 1$). And as the covariance method, the Burg method estimates the coefficients a_i on a finite support.

4.4.3 Results

The effect of adopting Linear Prediction techniques for better partial evolution estimation is shown in figure 17. Here, the evolutions of different predictors for a saxophone vibrato are compared to each other, namely *constant* or *hold* (which equals the McAulay-Quatieri algorithm), *linear extrapolation* and *LPC* using the described method.

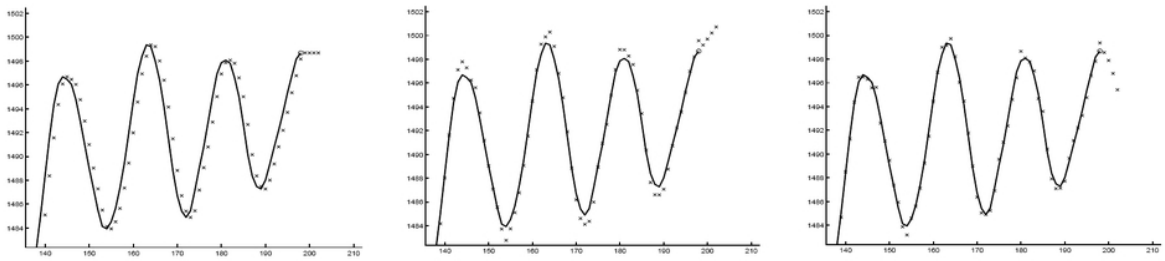

Figure 17: Evolutions of different predictors: constant (left), linear (center), and LPC (right) for a saxophone vibrato. (from [4])

Having a more precise estimate for the evolution of the partial track enables us to reduce the extent of the matching interval and thus to improve the uality of the estimate.

5 Conclusion and Outlook

The sinusoidal model, a framework for modelling speech and music signals, has been presented. Various modifications concerning improved frequency estimation and track continuation have

been explained yielding better estimates for the analysed signal.

Sounds synthesized with the original implementation of McAulay and Quatieri [1] lack of sharpness during transient parts of speech or musical signals. This is due to the incapability of the model to capture noisy sounds well. Though it is very inefficient to model noise like sounds as a set of sinusoids it is possible in principal, given the density of sinusoids meets the requirements imposed by the Karhunen-Loeve expansion.

Another strategy would be decomposing the signal into deterministic and stochastic parts and using different models for the different portions of a sound as proposed by [5]. These models capable of producing sounds of very good audio quality fail modelling percussive sounds well. An explicit transient model would be needed for capturing this last unmodelled portion left in the data.

The method of time-frequency reassignment has opened new possibilities in terms of analysis accuracy. In combination with the so called bandwidth enhancement a technique introduced by [3], very good sound quality could be attained. There partials are not strictly sinusoidal but a combination of sinusoidal energy and noise energy, a single partial having time-varying amplitude, frequency, and bandwidth parameters yielding a homogenous model able to represent very different musical sounds and preserve very good audio quality in synthesis.

References

[1] R. McAulay, Th. Quatieri: "Speech Analysis/Synthesis Based on a Sinusoidal Representation", in *IEEE Transactions on Acoustics, Speech, and Signal Processing, August 1986*

[2] J. Smith III, X. Serra: "PARSHL: An Analysis/Synthesis Program for Non-Harmonic Sounds Based on a Sinusoidal Representation"

[3] K. Fitz, L. Haken: "On the Use of Time-Frequency Reassignment in Additive Sound Modelling"

[4] M. Lagrange, S. Marchand, M. Raspaud, J.-B. Rault: "Enhanced Partial Tracking using Linear Prediction", in *Proc. of the 6th Int. Conference on Digital Audio Effects (DAFx-03), September 2003*

[5] X. Serra: "A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition", *Thesis, Stanford University, 1989*