ENHANCED TIME-STRETCHING USING ORDER-2 SINUSOIDAL MODELING

Sylvain Marchand and Martin Raspaud

SCRIME – LaBRI, Université Bordeaux 1
351 cours de la Libération, F-33405 Talence cedex, France
{sm|mraspaud}@labri.fr

ABSTRACT

In this article, we introduce a 2-level sinusoidal model and demonstrate its aptitude for a challenging digital audio effect: time-stretching without audible artifacts. More precisely, sinusoidal modeling is used at the two levels of the new sound model. We consider the frequency and amplitude parameters of the partials of the classic sinusoidal model as (control) signals, that we propose to model again using a sinusoidal model. This way, higher-level musical structures such as the vibrato and tremolo in the original sound are captured in the “partials of partials” of this order-2 sinusoidal model. We propose then a new time-stretching method, based on this new hierarchical model, which preserves not only the pitch of the original sound, but also its natural vibrato and tremolo.

1. INTRODUCTION

Spectral models provide general representations of sound in which many audio effects can be performed in a very natural and musically expressive way. Based on additive synthesis, they contain a deterministic part consisting of a – often huge – number of partials, which are pseudo-sinusoidal tracks for which frequencies and amplitudes evolve slowly with time. The spectral modeling parameters of this deterministic part consist of the evolutions in time of the controls of the partials, thus leading to a large amount of data.

We have already shown in [1] that the redundancy in the evolutions of these parameters can be used to reduce these data [1] and that the re-analysis of spectral parameters can help us in extracting higher-level musical parameters such as the pitch [2].

In this article, we introduce a new 2-level sound model of great interest for digital audio effects. The first level of this new model is the well-known sinusoidal model, leading to partials whose parameters – frequencies and amplitudes – continuously evolve slowly with time. For the second level, we consider these parameters as (control) signals, that we propose to model again using a sinusoidal model. This way, higher-level musical structures such as the vibrato and tremolo in the original sound are captured in the “partials of partials” of this order-2 sinusoidal model.

We then demonstrate a straightforward application of this new model to digital audio effects: time-stretching. We chose to focus on this effect, although many others can be performed in this model. More precisely, we show how the new order-2 hierarchical model allows us to enhance the quality of this challenging audio effect, by preserving the natural vibrato and tremolo together with the pitch of the sounds while stretching them. We are specially interested in transforming the deterministic part – no noise or transients for now – of pseudo-harmonic instrumental sounds as well as the human voice.

After a brief introduction in Section 2 to the basic sinusoidal model and a survey of the existing time-stretching methods based on this model in Section 3, we introduce in Section 4 the new hierarchical model and we present in Section 5 a new method for time-stretching while preserving not only the pitch of the original sound, but also its natural microscopic variations such as its vibrato and tremolo.

2. SINUSOIDAL MODELING

2.1. Model and Parameters

Additive synthesis is the original spectrum modeling technique. It is rooted in Fourier’s theorem, which states that any periodic function can be modeled as a sum of sinusoids at various amplitudes and harmonic frequencies. For stationary pseudo-periodic sounds, these amplitudes and frequencies continuously evolve slowly with time, controlling a set of pseudo-sinusoidal oscillators commonly called *partials*. This is the well-known McAulay-Quatieri representation [3]. The audio signal a can be calculated from the additive parameters using Equations (1) and (2), where P is the number of partials and the functions f_p, a_p, and ϕ_p are the instantaneous frequency, amplitude, and phase of the p-th partial, respectively. The P pairs (f_p, a_p) are the parameters of the additive model and represent points in the frequency-amplitude plane at time t. This representation is used in many analysis / synthesis programs such as Lemur [4], SMS [5], or InSpect [6].

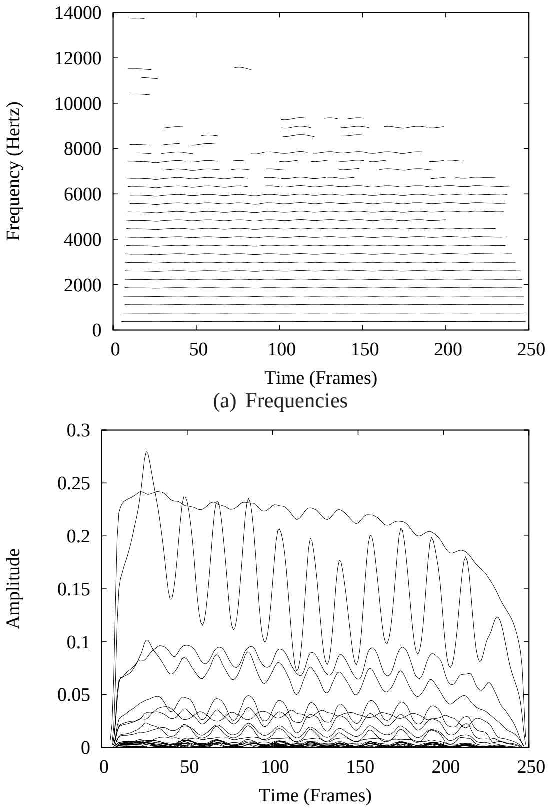
$$a(t) = \sum_{p=1}^{P} a_p(t)\ \cos(\phi_p(t)) \quad (1)$$

$$\phi_p(t) = \phi_p(0) + 2\pi \int_0^t f_p(u)\, du \quad (2)$$

2.2. Analysis Procedure

In order to faithfully imitate or transform existing sounds, this model requires an analysis method in order to extract the parameters of the partials from sounds which were usually recorded in the temporal model, that is audio signal amplitude as a function of time. The accuracy of the analysis method is extremely important since the perceived quality of the resulting spectral sounds depends mainly on it. Moreover, the main interest of an accurate analysis method, providing precise parameters for the model, is to allow ever deeper musical transformations on sound by minimizing deformations due to analysis artifacts.

The analysis method we use is made of two steps: spectral peaks are first extracted from the sound using a short-time spectral



(a) Frequencies

(b) Amplitudes

Figure 1: *Frequencies (a) and amplitudes (b) of the partials of an alto saxophone as functions of time (during approximately 2.9 s).*

analysis, then these peaks are tracked from frame to frame in order to reconstruct the partials.

It is extremely important to note that, during the sinusoidal analysis, we also keep the first coefficient of the Fourier transform as an amplitude track (of frequency 0 Hz, that is the DC component) in order to have the global envelope of the analyzed signal. This is of little interest for zero-mean sounds – this envelope being always very close to 0 (zero). But that will be very important for Sections 4 and 5.

2.2.1. Extraction of Spectral Peaks

First, a short-time Fourier analysis produces a series of short-term spectra taken on successive temporal windows on the original signal. Information about the local maxima in magnitude (so-called peaks) is then extracted from these short-term spectra using the derivative algorithm we proposed in [7], in order to provide the model with accurate spectral parameters (frequency, amplitude, and phase).

As for the practical side of this analysis, we used an analysis window of 2048 samples, moving by steps of $H = 512$ samples. These settings were used to preserve a good computation speed and memory usage in our software implementation. However, in the near future, this should be improved to gain better resolution. The test sound used for the figures of this article was a 16-bit, 44100-Hz mono recording of an alto saxophone playing at a fundamental frequency around 370 Hz with vibrato and tremolo.

2.2.2. Tracking of Partials

Since the Fourier analysis above delivers a short-time spectral representation of the analyzed sound, we consider local maxima in the magnitude spectrum (so-called peaks, see above) to be the instantaneous representation of partials. We have then to link peaks of successive frames to recover the continuous evolution of the partials. For this purpose, we use the enhanced partial-tracking algorithm we proposed in [8, 9]. This algorithm improves the classic McAulay-Quatieri algorithm [3] by using linear prediction in order to forecast, from their past, the future evolutions of the trajectories of the partials.

As for the practical side of this analysis, the maximal frequency difference between two successive frames for each partial was set to $\Delta = 10$ Hz. Partials whose amplitude was always below 0.001 or length was smaller than 0.1 s were considered as noise, since we are interested only in reliable – long and strong – partials.

2.3. Resampling the Parameters

In the remainder, we consider the frequency f, amplitude a, and phase ϕ parameters of the model as continuous signals. These parameters are measured at the center of each analysis frame. As a consequence, the corresponding signals get sampled at the analysis stage with a sampling period equals to the hop size of the analysis window (512 samples of the original sound – at 44100 Hz – here, see above). Since we need to know their values at each sound sample at the synthesis stage, we must be able to upsample these parameters (by a factor 512 in the example above).

More precisely, let us consider some signal s. We can reconstruct its continuous-time function $s(t)$ from its sampled (discrete-time) version $s[i]$, where $\forall i, s[i] = s(iT_s)$, T_s being the sampling period – that is the inverse of the sampling frequency F_s. For that purpose, we convolve the discrete signal by a reconstructor – a windowed sinc function – using an algorithm similar to the one proposed by Smith in [10, 11], except that we chose to use the Hann window instead of the family of Kaiser windows.

In theory, we consider the impulse train made of the samples of the discrete signal where they are known – at times multiple of the sampling period – and 0 (zero) elsewhere. The continuous version of the signal s is reconstructed simply by convolving this impulse train by the ideal reconstructor, the $\text{sinc}(tF_s)$ function, using the sinus cardinal function defined by:

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}, \text{ for } x \neq 0 \quad \text{and} \quad \text{sinc}(0) = 1 \qquad (3)$$

In practice, the ideal reconstructor cannot be used because of its infinite time support, and we need instead a reconstructor of finite support. In the remainder of this Section, let us denote by N this finite size expressed in samples. This size allows us to tune the trade-off of reconstruction quality versus computation time in the resampling process. We obtain this practical reconstructor by multiplying the ideal reconstructor by some window of finite support. We chose a symmetric Hann window of odd size $N = 2k + 1$ (k being some positive integer), defined by:

$$w_N(n) = \frac{1}{2}\left(1 - \cos\left(2\pi n/(N-1)\right)\right) \qquad (4)$$

for n in the $[0; N-1]$ range, and 0 (zero) elsewhere. The practical reconstructor is then given by:

$$r(t) = w_{2k+1}(k + tF_s) \cdot \text{sinc}(tF_s) \qquad (5)$$

Another problem arises at the boundaries of the discrete signal s. Indeed, this signal is of finite support and, during the convolution, we need some of its values before its beginning and after its end. The signal has then to be extrapolated. One solution is to use the classic reflection method (*i.e.* for samples before the beginning, that is for any sample index i which is a negative integer, we define $s[i] = 2s[0] - s[-i]$). This ensures the continuity of both the signal and its first derivative. However, this is not the best way for extrapolating the signal. For the amplitudes (of the audio signal a, of the partials a_p, etc.), extrapolating the signal with 0 (zero) values seems to be a natural choice, since those signals fade in and out from / to zero. However this will smooth the attack and this zero-padding technique cannot be used for other kinds of signals such as the frequencies f_p or the phases ϕ_p of the partials. For this reason, we use the extrapolation by the Burg method as proposed in [12, 13] for sound signals and generalized in [8, 9] for partials.

Once we have the continuous $s(t)$ function, upsampling by a factor u ($u \geq 1$) the signal s is straightforward since we can compute this function at any time, all the more at multiples of the new sampling period T_s/u. Upsampling is like considering the $s(t/u)$ function. Downsampling s by a factor d ($d \geq 1$) is slightly more complicated, since high frequencies have to be filtered out in order to respect the Nyquist condition. This is done by replacing F_s by F_s/d in the sinc function of the reconstructor.

In the remainder, we will often upsample parameters by integral upsampling factors. More precisely, the analysis is done frame by frame, with more than one sample between to frames. In order to get the values of the parameters at each sound sample, the evolutions of these parameters have to be upsampled by a factor corresponding to the hop size H used for the frames at the analysis stage.

3. SYNTHESIS AND TIME-STRETCHING

Once we have a sinusoidal model and an accurate analysis method for this model, we need a synthesis algorithm. Most synthesis methods can also be used to perform time-stretching. Let us now denote by T_s the sampling period of the sound to be synthesized.

3.1. Resampling the Frequency

The easiest way is to incrementally recompute the phase of each partial p at each sound sample by a discrete approximation of Equation (2), for k being any (positive) sample index:

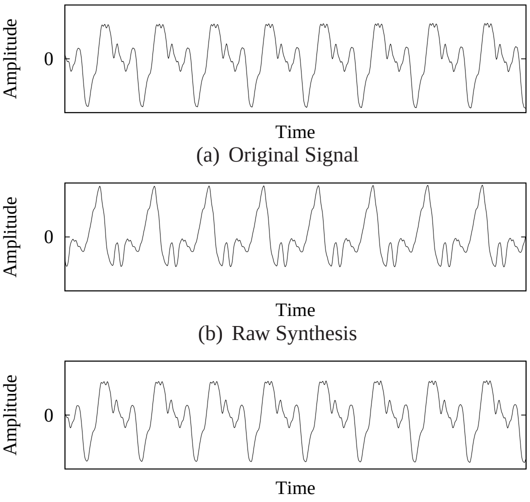
$$\phi_p((k+1)T_s) = \phi_p(kT_s) + 2\pi f_p(kT_s)T_s \tag{6}$$

so that the relations between phases and frequencies are maintained, and then compute the complete sound by using Equation (1).

Since we need, for each partial p, the values of its frequency f_p and amplitude a_p at each sound sample, we have to upsample these parameters using the technique described in Section 2.

Then, in order to perform time-stretching by a factor k ($k > 0$) during the synthesis, the frequency f_p and amplitude a_p of each partial can simply be resampled according to this k factor prior to the synthesis algorithm itself, to match the targeted length.

Since this technique does not take the measured values of the phase into account, except $\phi_p(0)$ at the time origin, the resulting sound has not the same shape as the original sound (see Figure 2(b)), although this makes no audible difference for most sounds.



(a) Original Signal

(b) Raw Synthesis

(c) Shape-Invariant Synthesis

Figure 2: *Raw (b) versus shape-invariant (c) synthesis methods. The latter is very close to the original signal (a).*

3.2. The McAulay-Quatieri Method

To be consistent with other articles on this topic (see for example [14]), let us introduce the following notations for the phase and frequency of partial p measured at the center of frame number k:

$$\theta^k = \phi_p(kHT_s) \tag{7}$$

$$\omega^k = 2\pi f_p(kHT_s) \tag{8}$$

where H is the number of samples between two consecutive frames. For simplicity sake, we will consider only one partial and omit the partial subscript.

The McAulay-Quatieri model [3] for phase reconstruction of each signal partial between the k-th and $(k+1)$-th synthesis frames consists of an order-3 polynomial, given by:

$$\theta(n) = \theta^k + \omega^k n + \alpha n^2 + \beta n^3 \tag{9}$$

where θ^k and ω^k respectively denote the phase and frequency of the partial measured at the junction of synthesis frames k and $k+1$ (which is chosen as the local origin $n = 0$). Assuming

1. continuity of the phases and frequencies – which are the derivatives of the phases – at frame junctions,
2. unwrapping of the phase with a “maximally smooth” constraint on the phase model

leads to the model parameters α and β, given by:

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} 3/N^2 & -1/N \\ -2/N^3 & 1/N^2 \end{bmatrix} \cdot \begin{bmatrix} \theta^{k+1} - \theta^k - \omega^k N + 2\pi M \\ \omega^{k+1} - \omega^k \end{bmatrix} \tag{10}$$

where N is the size of the synthesis frame (equals to the analysis hop size, so here $N = H$), and M is the “phase unwrapping” integral factor given by:

$$M = e\left[\frac{1}{2\pi}\left((\theta^k - \theta^{k+1}) + (\omega^k + \omega^{k+1})\frac{N}{2}\right)\right] \tag{11}$$

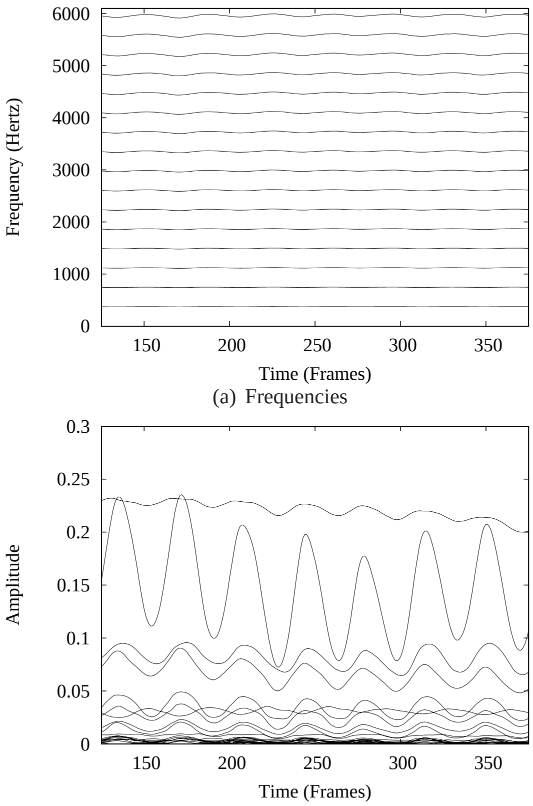
where $e[x]$ denotes the nearest integer from x. Since the phase, measured at the analysis stage, is known modulo 2π, the unwrapping factor M is used to find the real value of this phase, incrementally: if θ^k has been unwrapped (at the previous frame), then the unwrapped version of θ^{k+1} is $\theta^{k+1} + 2\pi M$.

This phase model is a piecewise-polynomial model of order 3. Polynomial phase models of different orders have also been proposed (see [14]).

This model can also be used for time-stretching during the synthesis (see for example [15, 16]). For a stretching factor k ($k > 0$), the unwrapped phase must be multiplied by k (then re-wrapped), together with the size of the synthesis frame which becomes kN.

3.3. Resampling the Unwrapped Phase

We show here that it is possible to consider the (unwrapped) phase of each partial as a continuous function. More precisely, we first unwrap the phase of each partial, from frame to frame, by considering the unwrapping factor M (see above). Then, we can upsample the unwrapped phase by a factor H in order to get the value of the phase at each sound sample, using the technique presented in Section 2. Then the amplitude of the partial is upsampled in the same way, and the complete sound can be computed using Equation (1), as it was for the technique presented in Section 3.1.



(a) Frequencies

(b) Amplitudes

Figure 3: *Normal time-stretching (order 1) of the saxophone sound, by a factor 2. The frequencies (a) and amplitudes (b) of the partials are shown at the middle of the resulting sound. The rates of the vibrato and tremolo have changed (see Figure 1).*

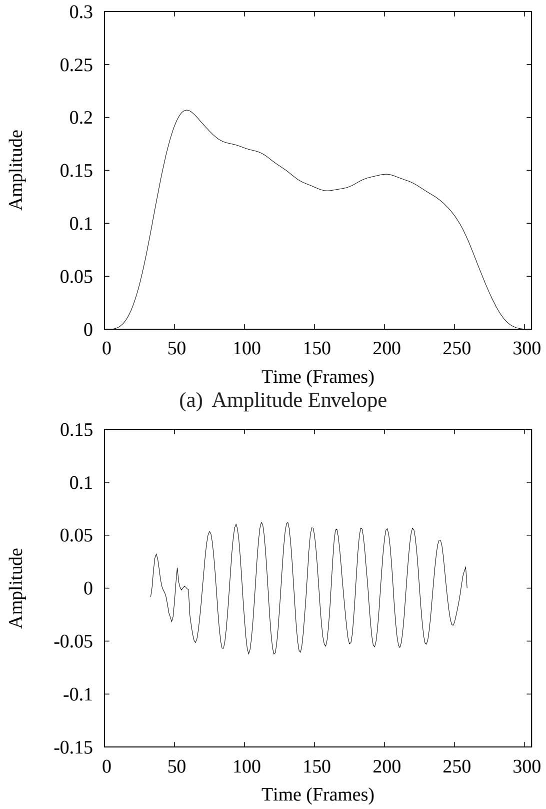
The phase-resampling technique proposed here, and more precisely the underlying continuous phase model, has to be tested in the near future with both synthetic and natural sound examples. Phase modeling is a very interesting research topic. As with the McAulay-Quatieri method described above, phase models are often polynomial (see [14]). However, in this article we will model the amplitude and frequency parameters of the sinusoidal model using the same sinusoidal model (see Section 4 below). Thus, the amplitude and frequency parameters will be sums of sinusoids (more precisely cosine functions, see model Equation (1)), the first one being of frequency zero thus leading to a constant term. Since the frequency is the derivative of the phase (see model Equation (2)), the phase will be a linear term plus a sum of sinusoids. A similar model for the phase is presented in [17] for speech signals.

Series of tests for several polynomial phase models can be found in [14], the McAulay-Quatieri method described above being the order-3 polynomial of this study. Among these tests, the third synthetic example shows sinusoidal evolutions for the frequencies (vibrato). The vibrato (sinusoidal variation of the fundamental frequency) is tested with and without tremolo (sinusoidal variation of the amplitude). It is clear that these sinusoidal evolutions cannot be perfectly approximated by polynomials of finite degrees like the order-3 polynomial phase model of the McAulay-Quatieri method. But provided that the frequency of a sinusoidal evolution remains below the Nyquist frequency, this evolution can be – in theory – perfectly reconstructed using the sinus cardinal reconstructor (see Equation (3)). That is the reason why we could expect an improvement in quality when using the resampling technique.

In [14], the signal-to-noise ratio (SNR) measures the energy ratio between the original signal and the residual part (noise) obtained by subtracting the resynthesis from the original. The greater is the SNR, the more accurate is the phase model for the considered example. For the third – vibrato + tremolo – example, the SNR for the McAulay-Quatieri method is 76.21. For the phase-resampling method using Equation (5) with $k = 32$, we compute a SNR of only 21.04, but with $k = 256$ the SNR is then 76.61 and thus outperforms all the polynomial phase models listed in [14]. The SNR is a growing function of k, and quality is at the expense of computation time. However, large values of k are needed only for the phase. Indeed, the resampling of the amplitude parameter is not problematic: by resampling only the amplitude with $k = 32$ and using the ideal phase, the SNR would have been 110.32. The problem is with the imprecision on the reconstructed phase, because of its linear term. Although the error remains very small in percentage, the large values of the unwrapped phase at the end of the partials lead to error in the magnitude of π. The synthesized signal is then from time to time put out of phase from the original one, thus the SNR is poor, but the perceived quality is still high. The SNR is indeed an imperfect perceptual metric.

3.4. Preserving the Vibrato

The previously presented time-stretching methods work well on really stationary sounds, but slight variations such as vibrato or tremolo are not properly conserved (see Figure 3). More precisely, their rate is changing with the stretching factor, leading to unnatural sound artifacts. To solve this problem, Arfib and Delprat suggested in [18, 19] the use of a hybrid method. They start by doing an analysis of the sound in order to obtain the frequency of each partial and then find its mean frequency curve – the frequency envelope – by using a low-pass FIR filter. Subtracting this frequency envelope from the frequency of the partial, they extract the vibrato (the modulation). Those two parts (envelope and modulation, illustrated for the amplitude instead of the frequency of the partial on Figures 4(a) and 4(b)) will then be stretched independently. From there, they find the mean frequency for the vibrato and then synthesize a new vibrato of the same frequency and of appropriate length. Adding this newly created vibrato to the resampled frequency envelope, they obtain the desired result: a stretching with vibrato conservation.



(a) Amplitude Envelope

(b) Tremolo

Figure 4: *Envelope (a) plus modulation (b) decomposition of the amplitude of the second partial of the saxophone. The envelope is the low-pass filtered version of the amplitude of the partial. The modulation is obtained by subtracting this envelope from the original amplitude (plain curve of the Figures 6 and 7).*

However good, this method did not seem satisfactory to us. Indeed, the resulting vibrato was synthetic and not very well defined. We wanted a more accurate vibrato, one that would be very close to the original, both in shape, rate, and depth, implying variations in frequency and amplitude, so that it would sound more natural.

4. MODELING THE PARAMETERS

On Figure 1, we see that partials issued from a sinusoidal analysis can contain sinusoidal-like components. In musical terms, these oscillations in both frequency and amplitude of a sound are respectively the vibrato and tremolo. Thus, our idea was to re-analyze those partials to extract the sinusoids from the sinusoidal parameters (this idea of re-analyzing the evolutions of the sinusoidal parameters appears for example in [1]). For that purpose, the same technique as presented in Section 2 was used to perform the analysis, simply considering the frequencies and amplitudes of the partials of a sound as regular signals.

In this article, we will denote by "order 0" the level corresponding to the sound signal in the temporal model, that is its amplitude as a function of time – $a(t)$ in Equation (1). Then, the classic sinusoidal modeling is the order 1: we obtain partials from the order-0 signal. Re-analyzing those partials, and modeling them again with sinusoidal modeling, leads us to "partials of partials", as illustrated in Figures 5, 6, and 7. This sinusoidal modeling of the parameters of the order-1 sinusoidal model constitutes another level in our hierarchical sinusoidal model: the order 2. In this article, we will stop at the order 2 since we are interested in simple microscopic sound structures such as the vibrato and tremolo, although further levels could help for macroscopic or higher-level musical sound structures.

On Figure 5 we can see the frequency of the second partial of the saxophone sound (plain) as well as the frequencies of the associated order-2 partials (dashed). We can clearly see the vibrato around 5 Hz and the corresponding order-2 partial. Since this vibrato is not perfectly sinusoidal, other order-2 partials (harmonics) are also present. Figure 6 shows the frequency of the order-2 partials of the amplitude of the second partial of the saxophone. This time, we can clearly see the tremolo, with the same frequency as the one of the vibrato. Figure 7 shows the amplitude of the order-2 partials of the amplitude of the second partial of the saxophone. The order-2 partial with the greatest amplitude is the DC component (corresponding to a frequency of 0 Hz, see Section 2), and is very close to the envelope as defined by Arfib and Delprat (see Section 3.4 and Figure 4(a)). However, our version of this envelope is not as good as it should be, because of the main drawback of sinusoidal modeling: the fast variations, such as the attack and decay, are smoothed. The depth of the tremolo can be read on the amplitude of the next order-2 partial. The partial-tracking algorithm is not perfect, and failed around the frame 25 (the order-2 partial was wrongly split into two parts). The amplitudes of the other order-2 partials are very low, because the shape of the tremolo is nearly a perfect sinusoid.

However the settings for the order-2 sinusoidal analysis had to be changed. Indeed, for the order-1 analysis, we considered the "interesting" frequencies to be audible, between 20 and 20000 Hz. Here, the frequencies we are looking for are around 5 or 10 Hz only, and not audible. Hence, the analysis window having to be large enough to contain at least 2 periods of the sinusoids we are looking for, we chose a window size of 64 samples at the sampling frequency of the (order-1) partials, that is $44100/512 \approx 86.13$ Hz. This allows us to take up sinusoids down to a frequency of 2.69 Hz, thus sufficient to take a regular vibrato or tremolo into account. The analysis window was moved by steps of $H' = 1$ sample. The maximal frequency difference between two successive frames for each order-2 partial in the partial-tracking algorithm was set to $\Delta' = 0.2$ Hz. Partials whose amplitude was always below 0.0001 or length was smaller than 0.2 s were considered as noise.

Moreover, to obtain a correct analysis, we usually need extra samples before and after the signal itself. For example, if we center our first analysis window on the first sample, then the first half of this window should be filled with some extrapolated samples. The same problem as the one we faced at the end of Section 2 occurs, hence we used the same solution: extrapolation using the Burg method.

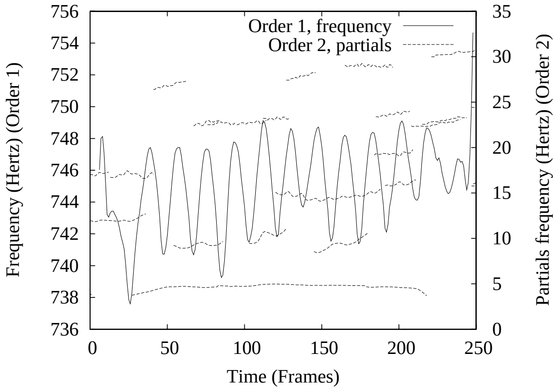


Figure 5: *Frequencies of the order-2 partials (dashed) for the frequency of the second partial (plain) of the saxophone.*

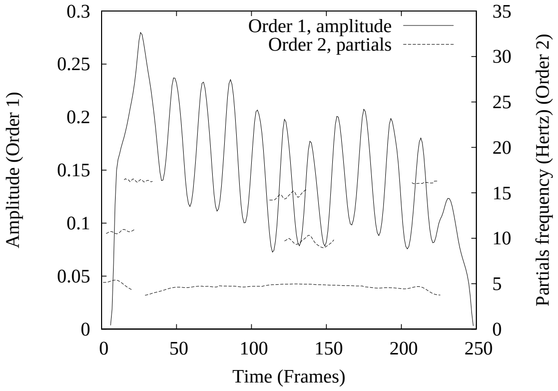


Figure 6: *Frequencies of the order-2 partials (dashed) for the amplitude of the second partial (plain) of the saxophone.*

5. ENHANCED TIME-STRETCHING

The synthesis within the order-2 model consists of two levels.

First, the frequency and amplitude parameters of the (order-1) partials are reconstructed – resynthesized – from the order-2 parameters (phase and amplitude only) using the synthesis technique presented in Section 3.3. This method is shape-invariant, and thus the vibrato and tremolo are kept as close to the original as possible. We obtain partial trajectories almost identical to the original, except for the transients though.

Second, the (order-0) audio signal is synthesized from these synthetic order-1 partials (using their frequency and amplitude) by the technique described in Section 3.1. Note that this technique is not shape-invariant, but for now we can only use the frequencies and amplitudes, not the phases, of the synthetic partials to generate the result. However, as mentioned before, this makes almost no audible difference for most sounds.

Using the classic (order-1) sinusoidal parameters, the time-stretching operation would consist in resampling these parameters to the desired length. This was explained in Section 3, and we concluded that vibrato and tremolo were not conserved. We use the same technique for our enhanced time-stretching, however applied to order-2 partials (*i.e.* partials of partials) instead of order-1 partials. By resampling order-2 partials to the desired length, we obtain after the synthesis a stretched sound with the original vibrato and tremolo rate. Moreover, since we use a shape-invariant method to synthesize the extended vibrato and tremolo, they are very similar to the original in shape too.

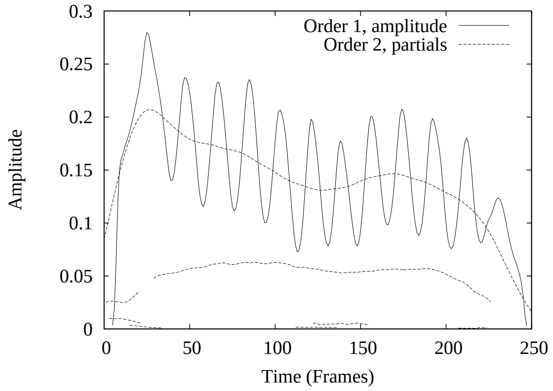


Figure 7: *Amplitudes of the order-2 partials (dashed) for the amplitude of the second partial (plain) of the saxophone.*

Figure 8 is an example of this enhanced time-stretching on the saxophone sound shown on Figure 1(b). We can see that the tremolo shape and rate on the first figure are very close to the ones of the second, but that the evolution of this tremolo was slowed down, as the global envelope was stretched.

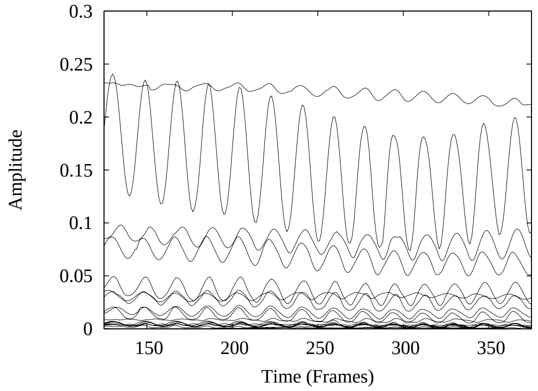


Figure 8: *Enhanced time-stretching (order 2) of the saxophone sound, by a factor 2. The amplitudes of the partials are shown at the middle of the resulting sound. The shape and rate of the tremolo are preserved (see Figure 1(b) for a comparison).*

However, the drawback of this method is the drawback of sinusoidal modeling in general, that is the smoothing of sudden changes such as the attack.

6. CONCLUSIONS AND FUTURE WORK

In this article, in the context of sinusoidal modeling, we have considered the frequency, amplitude, and phase parameters of the partials as continuous signals. By considering these signals and not their piecewise-polynomial approximation as it is generally done, we were able for example to resample them and re-analyze them with the same kind of methods and models that those used for audio signals.

This unified approach lead us to a 2-level sinusoidal model of great interest for digital audio effects, such as time-stretching while preserving not only the pitch but also higher-order sound structures such as the vibrato and tremolo of musical sounds.

Of course, the results presented here are still preliminary. They were computed using a software we developed in Common Lisp. This is a high-level language allowing a level of abstraction necessary to handle easily complex data structures. This software is still in early stages of development, thus no public release is available yet. However, some sound examples – including the saxophone used for the figures of this article – are available online[1].

In the near future, we will study the possibility of preserving the shape of the signal (order 0) together with the shape of the vibrato and tremolo (order 1). We will investigate hierarchical models of orders greater than 2, allowing us to deal with higher-level musical sound structures. We also intend to take into account transients and noise in the basic sinusoidal model.

7. ACKNOWLEDGMENTS

This research was carried out in the context of the SCRIME project which is funded by the DMDTS of the French Culture Ministry, the Aquitaine Regional Council, the General Council of the Gironde Department and IDDAC of the Gironde Department. The SCRIME project is the result of a cooperation convention between the *Conservatoire National de Région* of Bordeaux, ENSEIRB (school of electronic and computer scientist engineers) and the University of Bordeaux 1. It is made of composers of electro-acoustic music and scientific researchers. It is managed by the LaBRI (laboratory of research in computer science of Bordeaux). Its main goals are research and creation, diffusion and pedagogy.

8. REFERENCES

[1] Sylvain Marchand, "Compression of Sinusoidal Modeling Parameters," in *Proc. DAFx*, Verona, Italy, December 2000, Università degli Studi di Verona and COST, pp. 273–276.

[2] Sylvain Marchand, "An Efficient Pitch-Tracking Algorithm Using a Combination of Fourier Transforms," in *Proc. DAFx*, Limerick, Ireland, December 2001, University of Limerick and COST, pp. 170–174.

[3] Robert J. McAulay and Thomas F. Quatieri, "Speech Analysis/Synthesis Based on a Sinusoidal Representation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 4, pp. 744–754, 1986.

[4] Kelly Fitz and Lippold Haken, "Sinusoidal Modeling and Manipulation Using Lemur," *Computer Music Journal*, vol. 20, no. 4, pp. 44–59, Winter 1996.

[5] Xavier Serra, *Musical Signal Processing*, chapter Musical Sound Modeling with Sinusoids plus Noise, pp. 91–122, Studies on New Music Research. Swets & Zeitlinger, Lisse, the Netherlands, 1997.

[6] Sylvain Marchand and Robert Strandh, "InSpect and ReSpect: Spectral Modeling, Analysis and Real-Time Synthesis Software Tools for Researchers and Composers," in *Proc. ICMC*, Beijing, China, October 1999, ICMA, pp. 341–344.

[7] Myriam Desainte-Catherine and Sylvain Marchand, "High Precision Fourier Analysis of Sounds Using Signal Derivatives," *Journal of the AES*, vol. 48, no. 7/8, pp. 654–667, July/August 2000.

[8] Mathieu Lagrange, Sylvain Marchand, Martin Raspaud, and Jean-Bernard Rault, "Enhanced Partial Tracking Using Linear Prediction," in *Proc. DAFx*, London, United Kingdom, September 2003, Queen Mary, University of London, pp. 141–146.

[9] Mathieu Lagrange, Sylvain Marchand, and Jean-Bernard Rault, "Using Linear Prediction to Enhance the Tracking of Partials," in *Proc. ICASSP*, Montreal, Canada, May 2004, IEEE.

[10] Julius O. Smith and Phil Gossett, "A Flexible Sampling-Rate Conversion Method," in *Proc. ICASSP*, San Diego, 1984, IEEE, vol. 2, pp. 19.4.1–19.4.2.

[11] Julius O. Smith, "Digital Audio Resampling Home Page," Tech. Rep., CCRMA, Online. URL: `http://www-ccrma.stanford.edu/~jos/resample/resample.html`, 2000.

[12] Ismo Kauppinen, Jyrki Kauppinen, and Pekka Saarinen, "A Method for Long Extrapolation of Audio Signals," *Journal of the AES*, vol. 49, no. 12, pp. 1167–1180, December 2001.

[13] Ismo Kauppinen and Kari Roth, "Audio Signal Extrapolation – Theory and Applications," in *Proc. DAFx*, Hamburg, Germany, September 2002, University of the Federal Armed Forces, Hamburg, pp. 105–110.

[14] Laurent Girin, Sylvain Marchand, Joseph di Martino, Axel Röbel, and Geoffroy Peeters, "Comparing the Order of a Polynomial Phase Model for the Synthesis of Quasi-Harmonic Audio Signals," in *Proc. WASPAA*, New Paltz, New York, USA, October 2003, IEEE.

[15] Thomas F. Quatieri and Robert J. McAulay, "Shape Invariant Time-Scale and Pitch Modification of Speech," *IEEE Transactions on Signal Processing*, vol. 40, no. 3, pp. 497–510, 1992.

[16] Riccardo Di Federico, "Waveform Preserving Time Stretching and Pitch Shifting for Sinusoidal Models of Sound," in *Proc. DAFx*, Barcelona, Spain, November 1998, Audiovisual Institute, Pompeu Fabra University and COST.

[17] Laurent Girin, Mohammad Firouzmand, and Sylvain Marchand, "Long Term Modeling of Phase Trajectories within the Speech Sinusoidal Model Framework," in *Proceedings of the INTERSPEECH – 8th International Conference on Spoken Language Processing (ICSLP'04)*, Jeju Island, Korea, October 2004.

[18] Daniel Arfib and Nathalie Delprat, "Alteration of the Vibrato of a Recorded Voice," in *Proc. ICMC*, Beijing, China, October 1999, ICMA, pp. 186–189.

[19] Daniel Arfib and Nathalie Delprat, "Selective Transformations of Sounds Using Time-Frequency Representations: An Application to the Vibrato Modification," in *104th Convention of the AES*, Amsterdam, the Netherlands, May 1998, AES, Preprint 4652 (P5-2).

[1] `http://dept-info.labri.fr/~sm/DAFx04/index.html`