A GENERALIZED POLYNOMIAL AND SINUSOIDAL MODEL FOR PARTIAL TRACKING AND TIME STRETCHING

Martin Raspaud†, Sylvain Marchand†, and Laurent Girin‡

†SCRIME – LaBRI, Université Bordeaux 1
351 cours de la Libération
F-33405 Talence cedex, France
firstname.name@labri.fr

‡Institut de la Communication Parlée – INPG
46 avenue Félix Viallet
F-38031 Grenoble cedex 1, France
girin@icp.inpg.fr

ABSTRACT

In this article, we introduce a new generalized model based on polynomials and sinusoids for partial tracking and time stretching. Nowadays, most partial tracking algorithms are based on the McAulay-Quatieri approach and use polynomials for phase, frequency, and amplitude tracks. Some sinusoidal approaches have also been proved to work in certain conditions. We will present here an unified model using both approaches, which will allow more flexible partial tracking and time stretching.

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 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].

At the same time, most parameters are modeled using polynomials, such as in the well-known and widely-used partial-tracking algorithm proposed by McAulay and Quatieri [3].

In this article, we introduce a new sound model of great interest for digital audio effects. Indeed, it mixes both approaches in a single model made of polynomials and sinusoids.

Moreover, we follow the multi-level sinusoidal modeling approach we introduced in [4]. Indeed, the parameters of the partials of the basic sinusoidal model can be also regarded as (control) signals. This way, we can re-analyze these signals to obtain "partials of partials", also called order-2 partials. This multi-level modeling is well-suited for high-level musical transformations. In the remainder of this paper, the original time-domain signal is the order-0 signal, the partials are in fact order-1 signals, and we also deal with those new order-2 partials.

One advantage of this multi-level polynomial and sinusoidal model is the fact that the polynomial part will represent the slow time-varying envelope of the signal (at any order), while the sinusoidal part will model order-1 partials and will handle the musical modulations they may contain, such as the vibrato and the tremolo. The vibrato and tremolo represent a slight sinusoidal variation of the sound frequencies and amplitudes, respectively.

We also demonstrate two applications of this new model. The first one is a classic enhancement to partial tracking, and more precisely is peak prediction from past peaks to follow more accurately the partials, allowing the algorithm to choose more precisely the next peak of a tracked partial. By using our new model, we leave aside linear prediction – used till then as shown in [5] – and thus we obtain a more consistent algorithm.

The second application is a challenging digital audio effect: time stretching. We aim at achieving this effect without audible artifacts, but most of all without any modification of timbre or vibrato and tremolo rates. This is possible thanks to the second-order analysis we perform with our model. The sounds we focused on for our study are without noise or transients (because of the limitations of the sinusoidal model).

After a brief introduction in Section 2 to the basics of our new Poly-Sin model, we introduce in Section 3 the analysis method for our model, then we present the modification to the peak prediction for the partial-tracking algorithm in Section 4. Finally, we will explain the synthesis procedure of this model in Section 5, and in Section 6 the 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. POLYNOMIAL AND SINUSOIDAL (POLY-SIN) MODEL

We present here the components of our Poly-Sin model, a generalized polynomial plus sinusoids model. For the sake of clarity, we first present the basics of the model, that we will extend to multi-level modeling at the end of this section.

2.1. Polynomial Modeling

To ensure the accurate reconstuction of a partial – especially for the phase – it is very important to estimate the coefficients of the polynomial within the analysis window we are currently analyzing (locality property of the polynomial).

Such a local polynomial interpolation is used by the McAulay-Quatieri partial-tracking algorithm, where the phase is a third-degree polynomial interpolation of the measured phase values, thus a second-degree polynomial interpolation for the frequency, and where the amplitude is interpolated using a first-degree polynomial (linear interpolation).

However, these finite-degree polynomial approximations will not be able to approximate correctly sinusoidal modulations. Moreover, those modulations are better analyzed using sinusoidal modeling, thus using for the control parameters a model which is particularly well-suited for the signal itself.

2.2. Sinusoidal Modeling

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 quasi-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 signal s 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 [6], SMS [7], or InSpect [8].

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

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

2.3. Poly-Sin Model

From the preceding models, we build our new model. We can express it from Equation 1 as

$$s(t) = \Pi(t) + \sum_{p=1}^{P} a_p(t) \cos(\phi_p(t)) \qquad (3)$$

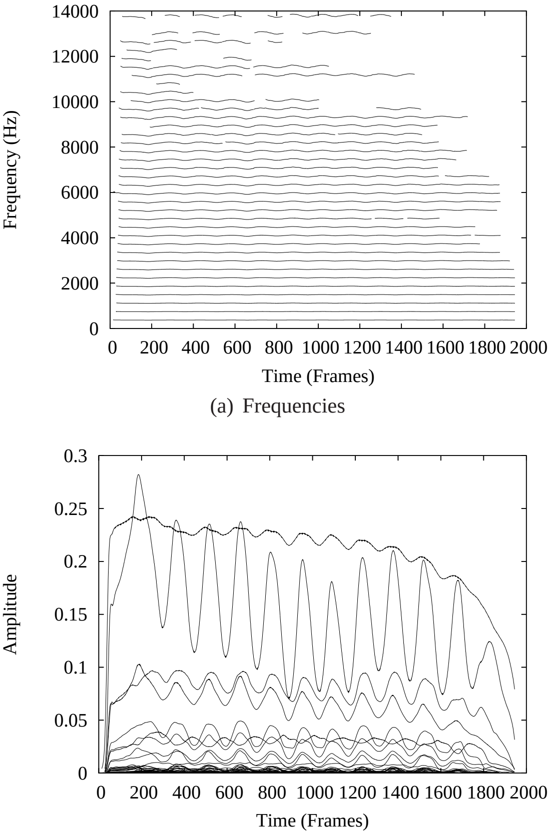
where $\phi_p(t)$ is given in Equation 2 and $\Pi(t)$ is a polynomial.

This model is thus more general than the two others. Indeed, as presented in [9], polynomial models have shown their limitations regarding vibrato and tremolo. In fact, it is not possible to approximate correctly a sinusoidal modulation with a finite-degree polynomial. Considering that the vibrato and the tremolo created by an instrumentalist are almost sinusoidal, or at least pseudo-periodic, we can then suppose that our model will perform perfectly for those kinds of sounds, and thus open more perspectives for applications on digital audio effects, while still being well-suited for sounds correctly handled by any of the preceding models.

Throughout the remainder of this document, the polynomial part of our model will be called envelope. Indeed, the polynomial will gather the very slow modifications of the signal, in other words the very low frequencies, while the modulations – higher frequencies – will be gathered by the sinusoidal analysis. Since the set of polynomials and the set of sinusoids both constitute a base of the signal space, the combination of the two is over-complete. Despite this over-completeness, we think that with a correct tuning of the separation between envelope (low frequency) and modulations (high frequency), a simple decomposition might be easily found.

2.4. Multi-Level Model

We then follow the multi-level sinusoidal modeling approach we introduced in [4]. The original time-domain signal is the order 0 of the hierarchy, and Equations 3 and 2 usually deal with partials we will call now order-1 partials. These equations can in turn be re-used to deal with order-2 partials – obtained from the analysis of the evolutions of the (order-1) partials – useful for handling musical modulations. We use Equations 3 and 2 at each level of our hierarchy. However, in the case of zero-mean signals, the polynomial part of Equation 3 disappears and thus Equation 3 turns into Equation 1. This is the case for the first level of our hierarchy.



(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). The frames are estimated every 64 samples, using 1024-sample windows, on a CD-quality signal (44100-Hz sampling frequency).*

3. ANALYSIS

The next step is to estimate the parameters of our model. We will use a short-term windowed analysis method, as opposed to previously proposed models which perform polynomial or sinusoidal analyses on long-term signal ranges [10].

3.1. Sinusoidal Analysis

To faithfully imitate or transform existing sounds, this model requires an analysis method 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 audible artifacts due to analysis errors.

The analysis method we use is made of two steps: spectral peaks are first extracted from the sound using a short-time spectral analysis (*i.e.* using a short-term sliding analysis window), then these peaks are tracked from frame to frame to reconstruct the partials. This is explained in further details in [4].

Another part of the analysis procedure is the extraction of the envelope (polynomial part) of the signal. This envelope is considered constant and equal to zero for the first-order analysis, because the analyzed signal is supposed to be zero-mean.

3.2. Polynomial Analysis

The other part of the analysis for our new model is the polynomial analysis. In the scope of our study we have used the least squares method to estimate the polynomial. Other methods exist though. We used the weighted least squares method at first, but it performed badly in the cases where the signal contained slow oscillations[1]. By minimizing the squared error equally on the whole analysis window, the least squares method allowed us to obtain a "smoother" polynomial for slowly-oscillating signals.

3.2.1. The Least Squares Method

The aim of the least squares method is to minimize the squared error of the polynomial approximation. So let N points located at positions $(s_k, g(s_k))$ with $k = 0 \ldots N-1$ being the sampling of the function g. We wish to find a globally-defined function $\widetilde{g}(s_k)$ that approximates the given values $g(s_k)$ at points s_k in a least-square sense, that is

$$\overline{g} = \min_{\widetilde{g}\in\Pi_m} \sum_{k=0}^{N-1} (\widetilde{g}(s_k) - g(s_k))^2 \qquad (4)$$

where Π_m is the set of polynomials of total degree m and $\widetilde{g}$ can be written as

$$\widetilde{g}(s_k) = \pi(s)^T A \qquad (5)$$

where $\pi(s) = \begin{bmatrix} 1 \\ s_k \\ s_k^2 \\ \vdots \\ s_k^p \end{bmatrix}$ and $A = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_p \end{bmatrix}$ is the vector containing the coefficients of the polynomial we are looking for, and p is the degree of the approximating polynomial.
In other terms,

$$\widetilde{g}(s_k) = a_0 + a_1 s_k + a_2 s_k^2 + \ldots + a_p s_k^p \qquad (6)$$

so, the function to minimize is

$$\overline{g}(a_0, a_1, a_2, \ldots, a_p) = \sum_{k=0}^{N-1} \left(\pi(s_k)^T A - g(s_k)\right)^2 \qquad (7)$$

A necessary – but not sufficient – condition to identify the minimum in a N-dimension space is

$$\nabla \overline{g} = 0 \qquad (8)$$

In other words

$$\frac{\partial \overline{g}}{\partial a_0} = \frac{\partial \overline{g}}{\partial a_1} = \frac{\partial \overline{g}}{\partial a_2} = \ldots = \frac{\partial \overline{g}}{\partial a_p} = 0 \qquad (9)$$

Then, the equation to solve is

$$MA = B \qquad (10)$$

with $M = \sum_k \pi(s_k)\pi(s_k)^T$ and $B = \sum_k \pi(s_k) g(s_k)$.

For further details, see for example [11].

3.2.2. Estimating the Polynomial

As we said earlier, the polynomial is used to approximate the global envelope of the signal we analyze. This requires some constraints. First we have to adjust the degree of the polynomial so that only slow variations of the signal are taken into account. High-degree polynomial could vary very quickly, so we decided to take a maximum degree of 3. This decision was also motivated by the fact that natural sounds rarely have more than third-degree polynomial shaped phase tracks (see [9]). Second, the signal has to be long enough to show more than two periods of oscillations, so that the polynomial will not approximate those oscillations.

Even though a signal can be long and contain more than two periods of a signal, the signal is analyzed locally using a short-term sliding analysis window. Thus the window size has to be large enough to contain the required number of periods of the modulation.

3.3. Poly-Sin Model Estimation

The two estimation methods proposed above have been found to be the best ones among those we have been acquainted with so far. However, other estimation methods, especially high-resolution methods, might be worth trying as they seem quite powerful, and maybe suitable for our requirements [12].

Now that we have at our disposal two estimation methods, we can estimate the parameters of our new model. The corresponding analysis will be windowed. For each window the analysis will be twofold. First we find the best-fitting third-degree polynomial using the least-square regression discussed earlier. The solving of the matrix system (Equation 10) gives us the coefficients of our polynomial. We then subtract this polynomial from our signal and proceed to the second step of our analysis, which consists of the sinusoidal analysis of the residual. The sinusoidal analysis consists in retrieving the spectral peaks of the signal in terms of amplitude, frequency, and phase using the method described in [13].

3.4. Two-Level Analysis

As presented in [4], the approach we use is a multi-level approach. That is, we are first analyzing the sound with a classic sinusoidal analysis and then we are analyzing the parameters of each partial using the same technique. The major drawback of this method is that it is not possible to correctly analyze the phases of the partials and thus we have to perform the second level analysis on the frequency (together with the amplitude) to have our second-order partials.

The idea behind our new phase model is that we need second-order partials to be based on phase rather than frequency tracks, so that we can explore even higher levels. Thus, the use of a polynomial estimation would allow us to correctly analyze the phase tracks of our partials.

[1] Around two periods in the analysis window.

Indeed, we assume that the phase tracks (and also the amplitude tracks) of our partials are in fact composed of sums of sinusoids and polynomials. Though one could argue that polynomials could be approximated by sinusoids and *vice-versa*, in our analysis procedure, the polynomial is gathering the global envelope of the signal while the sinusoids are gathering the oscillations of the signal, thus separating our signal in the two required components: sinusoids and envelope.

An illustration of this decomposition, obtained after the reanalysis of an order-1 partial, is shown on Figure 2. The sinusoidal part is the result of the re-synthesis of order-2 partials.

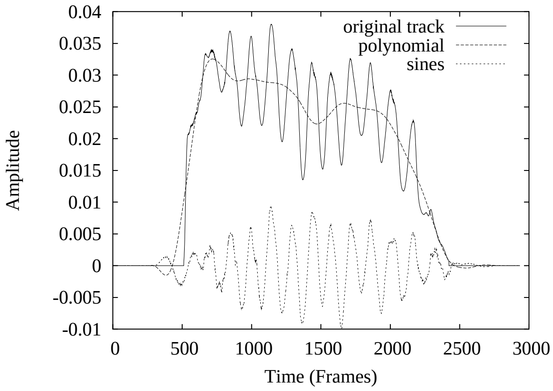


Figure 2: *Decomposition of the amplitude of a partial (control signal) showing the two steps of an analysis using a polynomial and sinusoids. The (order-2) frames are estimated at each sample of the evolution of the (order-1) partial, using 256-sample windows.*

Figures 3 and 4 show an amplitude track together with its order-2 partials and envelope. We can see that the order-2 partials are mainly present when the modulation is mostly active.

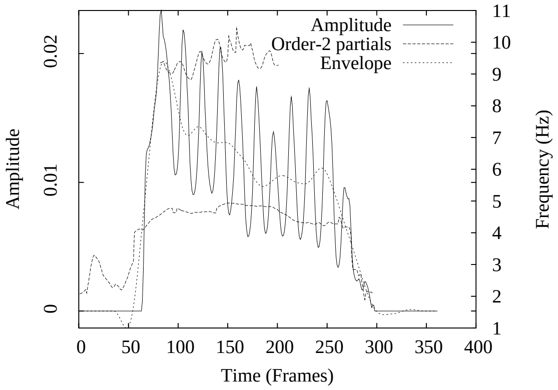


Figure 3: *Order-2 frequency partials and envelope of an amplitude track. The solid line represents the original amplitude track, the dashed lines represent the order-2 frequency partials, and the dotted line represents the polynomial envelope of the amplitude track.*

These order-2 partials are obtained by the re-analysis of the partial, again using an analysis window. A suitable window size is when the analysis window contains at least two periods of the oscillations, if any. In our context, these oscillations represent musical parameters of the sound such as vibrato and tremolo. We have considered in the scope of our study that the minimal vibrato and tremolo rates are around 5 Hz. This estimation leads to an easy computation of the minimal window size (to have at least two periods of the vibrato in the analysis window), which is two times the sampling frequency divided by the minimal vibrato and tremolo rates.

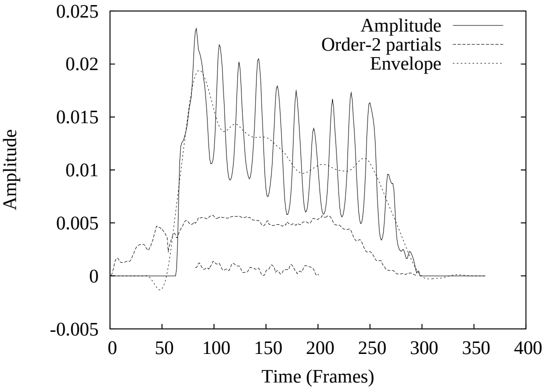


Figure 4: *Order-2 amplitude partials and envelope of an amplitude track. The solid line represents the original amplitude track, the dashed lines represent the order-2 amplitude partials, and the dotted line represents the polynomial envelope of the amplitude track.*

The proposed multi-level Poly-Sin model can then be used for several purposes, as explained in the following sections.

4. ENHANCED PARTIAL TRACKING

During the process of partial tracking with the McAulay-Quatieri algorithm [3], the peak selection algorithm is very important to follow the right tracks. Indeed, choosing the wrong peak during a partial tracking can be quite disastrous. To enhance the peak selection, various methods have been investigated.

Indeed, the algorithm we use is based on prediction of the following peak from the past peaks of the partial. To choose the best peak candidate in the ones available in the next frame, the past peaks of the currently-considered partial are used to compute a virtual – predicted – peak from which we take the closest peak candidate among the measured peaks.

The constant and linear methods are working quite well for really stationary sounds, but as most natural sounds – including singing voice – may contain vibrato and tremolo, those methods have their limitations. In [5], we showed that linear-prediction methods might work better for natural sounds, including correlation, covariance, and Burg algorithms. The Burg method proved to work best for partial tracking.

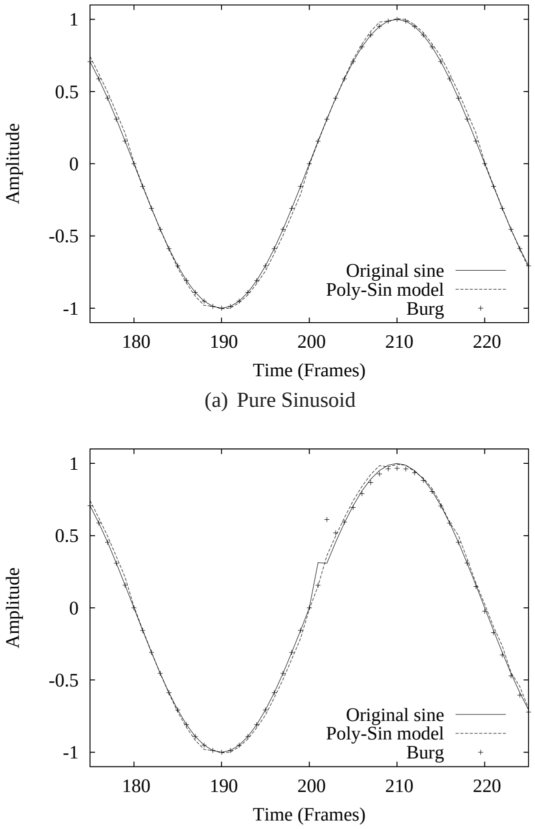
However, the Burg method tends to minimize prediction errors at the expense of spectra which are not well-suited for sinusoidal analysis (see [14] for more details).

In this paper, we choose the opposite approach: we take advantage of the spectral analysis done on the order-1 partials to propose a prediction method based on spectral extrapolation.

Indeed, we obtain consistent spectra we can extract sinusoids from, and the order-2 partials resulting from the analysis of the past evolutions of a given (order-1) partial are synthesized a bit further to obtain the predicted peak for this partial.

As for the spectral extrapolation described above, the polynomial part is in turn extrapolated in order to ensure the global envelope.

Thus, the way we perform the prediction is quite simple. The first step is to find the parameters of our model on the past samples used for the prediction. Once those parameters are found, we just consider that the parameters of the sinusoids will be constant over the predicted samples, and we compute the next values of the polynomial using the coefficients we found. Adding the two parts of the computation gives us the predicted samples.



(a) Pure Sinusoid

(b) Altered Sinusoid

Figure 5: *1-sample predictions of a pure sinusoid (a) and an altered sinusoid (b). The window size for Poly-Sin prediction is 128 samples, the order for Burg prediction is 8.*

Figure 5 shows some results of 1-sample predictions on a simple sinusoid. Figure 5(a) shows the predictions on a pure sinusoid. Burg performs better, but our method shows promising results. The small deviations are due to the polynomial estimation method which is not perfect. Figure 5(b) shows the predictions on a pure sinusoid where we displaced a sample to simulate a local error. Our method performs better in that matter since it is not disturbed by the noise, whereas the Burg method deviated a bit from the sinusoidal trajectory.

On longer predictions however (extrapolating more than one sample), our method might not be as good. Indeed, the polynomial is not guaranteed to be stable outside the analysis window. Thus, diverging may occur in certain conditions, among which a very small number of periods of the sinusoidal part of the signal. A solution would be to lower the degree of the polynomial (the lower the degree, the more stable the polynomial), but that would mean possibly lowering the quality of our estimation. This is a trade-off between approximation quality and long-term prediction stability.

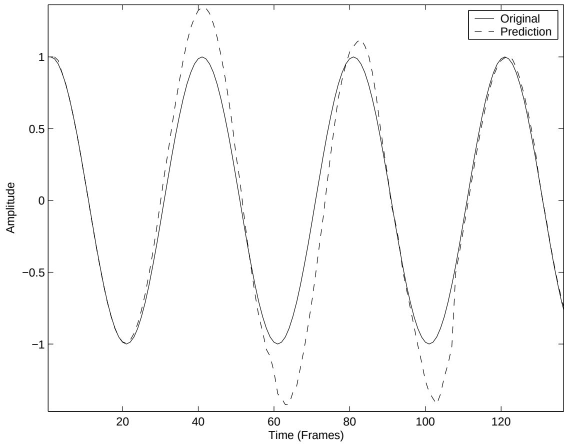


Figure 6: *1-sample predictions of a pure sinusoid computed from all the samples preceding the currently predicted sample.*

As for short signals, our method does not perform very well until more than two periods of the modulation are present in the analysis frame. This is illustrated by Figure 6, where we see that the predictions are diverging at the extrema of the signal. This can be explained by the fact that the third-degree polynomial is still not “flattened” until the modulation is really present in the signal, meaning that the polynomial is trying to approximate the sinusoid.

However, for very short signals, the Poly-Sin method is equivalent to polynomial extrapolation which works quite well on very slowly evolving signals. This is illustrated by the 20 first samples in Figure 6.

Moreover, the method is self-adapted to the number of samples available from the past, and even one sample is enough for (constant) extrapolation.

5. SYNTHESIS

Once all the parameters have been found, we then have to synthesize the signal to have our analysis-synthesis loop complete.

Our analysis having been windowed, we have a set of parameters for each window. A first solution is to consider everything as being constant on the short-term range of the analysis / synthesis windows. Thus, we create the polynomial from its coefficients, taking as many values as necessary (as many samples as the final sample-rate requires), and create the modulation from the spectral peaks we found. We consider that the modulation is constant over the window.

Then we simply overlap and add the sum of the envelope and the modulation with the next window, using a overlap factor of 50%. This will give us synthesized order-1 partials we can then use for the second synthesis. This last synthesis is then performed simply by combining the phases and amplitudes applying the formula of Equation 1.

Another solution is to resample the parameters of the order-2 partials to have them at the same sample rate as the desired sound output, as in [4]. Doing so would allow us to apply Equation 1 to obtain both the amplitudes and the phases of order 1 (adding the envelope each time) and then re-apply the same formula to have the final sound. In this case, we also generate the envelope by using the overlap-add technique explained before.

The results on both synthetic and natural sounds from those synthesis methods of informal listening tests are quite satisfactory, since it is not possible to hear any difference between original and re-synthesized sounds. However, the SNR measurements are not as good. Indeed, since we analyze and synthesize the phase partials with approximative estimators, the resulting phase is slightly different from the original. Thus, the shape of the original and synthesized sounds are not similar. SNR measurements being based on wave shapes, our results in that domain are not good.

6. CONSERVATIVE TIME STRETCHING

Basing our current study on our previous work on *enhanced time stretching* [4], we applied the Poly-Sin modeling on the analysis of partial parameters to perform conservative stretching. We call here conservative stretching, stretching where timbre, vibrato, and tremolo are conserved. In the previous work, we exposed that it is possible to compute order-2 partials to stretch more accurately the sound. Indeed, the order-2 partials are gathering the main parameters of the vibrato and tremolo of the sound. Thus, stretching simply consists in stretching the envelope and resampling the order-2 partials.

With the proposed Poly-Sin model, we can obtain the same order-2 partials with possibly an even better envelope. Indeed, at each analysis frame, instead of gathering the first bin of the Fourier transform (thus leading to a constant offset), we compute a polynomial estimate of higher degree, thus less stationary.

The stretching is then performed in the same way as before for the order-2 partials, and just consists in taking more values of the polynomial of the envelope.

The results we obtain from this method are indeed conservative as they preserve vibrato and tremolo. However, they are not perfect. In fact, some small artifacts can be heard. This is due to the very high precision and very fine tuning needed by the two-level analysis, because every error in an order-2 partial might have disgraceful consequences on the corresponding order-1 partial and thus on the re-synthesized sound. The resample methods not being perfect, and most of all the estimators not being very precise, the artifacts are then easily introduced.

7. CONCLUSION AND FUTURE WORK

In this article, we have presented a new Poly-Sin model – composed of a polynomial and sinusoids – with explanations for spectral analysis and synthesis with this model. We have also presented two major applications of this model: partial tracking and conservative time-stretching using order-2 partials.

Of course, the work presented here is still preliminary. As for any new model, a lot has to be done to obtain viable results, especially when we want a model general enough to fulfill most requirements of an analysis-transformation-synthesis loop. The major flaws of our results coming from erroneous estimation, our main goal in the future will be to find better estimators, especially for polynomial regression, thus maybe with high-resolution sinusoidal estimators. This would allow us to compete with other works regarding SNR measures for example.

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] Sylvain Marchand and Martin Raspaud, "Enhanced Time-Stretching Using Order-2 Sinusoidal Modeling," in *Proc. DAFx*, Naples, Italy, October 2004, Federico II University of Naples, pp. 76–82.

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

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

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] Andrew Nealen, "An as-short-as-possible introduction to the least squares, weighted least squares and moving least squares methods for scattered data approximation and interpolation," URL: `http://www.nealen.com/projects/`, May 2004.

[12] K. W. Chan and H. C. So, "Accurate Frequency Estimation for Real Harmonic Sinusoids," *IEEE Signal Processing Letters*, vol. 11, no. 7, pp. 609–612, July 2004.

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

[14] Florian Keiler, Can Karadogan, Udo Zölzer, and Albrecht Schneider, "Analysis of Transient Musical Sounds by Auto-Regressive Modeling," in *Proc. DAFx*, Queen Mary, University of London, United Kingdom, September 2003, pp. 301–304.