# ENHANCED RESAMPLING FOR SINUSOIDAL MODELING PARAMETERS

*Martin Raspaud and Sylvain Marchand*

LaBRI – CNRS, University of Bordeaux 1
351 cours de la Libération, F-33405 Talence cedex, France
firstname.lastname@labri.fr

## ABSTRACT

The sinusoidal modeling parameters can be regarded as control signals, and resampling can be used for synthesis or time-scaling purposes. However, these signals are not zero-centered, and consist of a slow time varying envelope together with modulations (vibrato, tremolo). Using directly the classic resampling method could have disastrous effects.

We present an addition to classic resampling aimed at achieving better results on such non zero-centered signals. Applied to the control signals of sinusoidal modeling, the method locally removes a polynomial envelope to the signal to perform better resampling on the residual modulations.

## 1. INTRODUCTION

The sinusoidal model [1] represents sounds as sums of sinusoids. Each sinusoid is controlled in time by phase, frequency, and amplitude parameters, often measured at a lower rate than the original sampling rate of the sound. These sinusoidal modeling parameters then have to be interpolated for resynthesis. This can be done by polynomial interpolation (see [1, 2]). Another possibility is to use signal processing tools, such as resampling [3].

Indeed, we propose in [4] to use resampling for synthesis and even time-scaling of the sound. However, whereas the classic resampling method works well for zero-centered signals, poor attention has been paid to the more general case of non zero-centered signals. In sinusoidal modeling, the amplitude, frequency, and phase of each partial are examples of such complicated signals.

Considering the parameters of the partials as sums of polynomials and sinusoids (see [2]), we thus designed a way to resample the control signals without the artifacts usually present if classic resampling is used for non-zero centered signals.

This paper is organized as follows. After an overview of the sinusoidal modeling context in Section 2, Section 3 describes the classic resampling technique. Then, in Section 4, we show the drawback of the classic technique and we present an enhanced method, taking the envelope into account. Finally, in Section 5, we present results that clearly show the improvements.

## 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 sinusoidal modeling representation used by McAulay and Quatieri [1] for speech signals and by Serra and Smith [5] for music signals. The audio 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 $\phi_p$ are the instantaneous frequency, amplitude, and phase of the $p$-th partial, respectively:

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

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

Sinusoidal modeling is used in many analysis / synthesis programs such as SMS [5] or InSpect [6] for examples.

### 2.1. Synthesis Using Resampling

The frequency, amplitude, and phase parameters are slow varying functions of time. Hence, at the analysis stage, they are often measured only each $H$ – the hop size – samples of the original audio signal. As a consequence, since the parameters of the partials were measured at a ($H$ times) lower sampling rate than the one needed for the output sound, at the synthesis stage it is necessary to find the missing amplitude, frequency, and phase values between the ones which were measured, in order to be able to apply Equation (1), and also Equation (2) to reconstruct the phase if only frequency and amplitude parameters are used.

Since the very beginning of sinusoidal modeling, polynomial interpolation techniques were proposed [1]. Many of them have been tested in [2], but they only work for small values of $H$ and do not take the global evolutions of the parameters into account. Moreover, they hardly handle modulations like vibrato / tremolo often present in musical signals. Indeed, a polynomial of finite degree cannot model sinusoidal modulations of the frequency (vibrato) or amplitude (tremolo) parameters.

On the other hand, we show in [4] that the parameters of the partials can also be regarded as (control) signals. Then, we show that finding the missing values can be regarded as uniform reconstruction – upsampling by a constant factor $H$. One advantage of considering the parameters of the partials as time signals is that $H$ can be set below the half of the period of the modulation (vibrato or tremolo), so that the Nyquist condition is respected and the original modulations can be accurately reconstructed.

### 2.2. Time-Scaling Using Resampling

Using the same principles as for the synthesis we just presented, we show in [4] a simple way to perform time-scaling using the sinusoidal model. Indeed, we can reconstruct the sinusoidal parameters at any time, and not necessarily at the original sampling periods of the input sound. More precisely, the technique consists in first scaling the time axis, then reconstructing the parameters according to this new scale, which is roughly equivalent to resampling. The time evolutions of the sinusoidal parameters are then scaled, but the values of these parameters are preserved. This way, we perform time-scaling while preserving the pitch, intensity, and

978-1-4244-1619-6/07/$25.00 ©2007 IEEE

timbre of the original sound. Note that for the phase parameter – considered unwrapped throughout this paper – the scaled version also has to be multiplied by the scaling ratio in order to be consistent, because of the relation between frequency and phase given by Equation (2).

The problem is yet to be able to resample the sinusoidal parameters, which are not zero-mean signals – neither the amplitude nor the frequency. The case of the – unwrapped – phase is even worse, since it exhibits a quasi-linear behavior if the frequency is quasi-stationary.

## 3. CLASSIC RESAMPLING

Let $s(t)$ be a continuous signal, and let us denote by $s[n] = s(nT_s)$ the sampled (discrete-time) version of this signal, with $T_s$ being the sampling period, that is the inverse of the sampling frequency $F_s$. Moreover, $s(t)$ is supposed to be bandlimited to $F_s/2$. According to the Shannon-Nyquist theorem, $s(t)$ can be reconstructed from its sampled form $s[n]$ by convolving the discrete signal by a reconstructor: a (windowed) sinc function [7]. In practice we use an algorithm similar to the one proposed by Smith [8, 3], except that we chose to use the Hann window instead of the family of Kaiser windows to build the reconstructor.

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. According to the Shannon-Nyquist theorem, the continuous version of the signal $s$ is reconstructed simply by convolving this impulse train by the ideal reconstructor $r_i$, based on the *cardinal sine* (sinc) function:

$$r_i = \text{sinc}(F_s t) \tag{3}$$

the classic reconstruction of $s(t)$ being:

$$s(t) = \sum_{n=-\infty}^{+\infty} s[n] \cdot r_i(t - nT_s) \tag{4}$$

that is, for the whole signal, $s = s * r_i$ ($*$ denoting the convolution).

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:

$$w_N(n) = \frac{1}{2}\left(1 - \cos\left(2\pi n/(N-1)\right)\right) \tag{5}$$

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

$$r_p(t) = w_{2k+1}(k + F_s t) \cdot \text{sinc}(F_s t). \tag{6}$$

Using Equation (4), it is easy to obtain the continuous signal $s(t)$ from its sampled version. Once we have the continuous 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 fulfill the Nyquist condition. We then have to use $F_m = \min(F_s, F_s')$, where $F_s'$ is the destination sampling rate, instead of $F_s$ in Equations (3) and (6) to define the appropriate reconstructor. The classic reconstruction procedure is summarized in Algorithm 1.

**Algorithm 1** classic_reconstruct($s$,$t$)

1: Center $r_p$ on time $t$
2: Compute an approximation of $s(t)$ by using $r_p$ instead of $r_i$ in Equation (4)

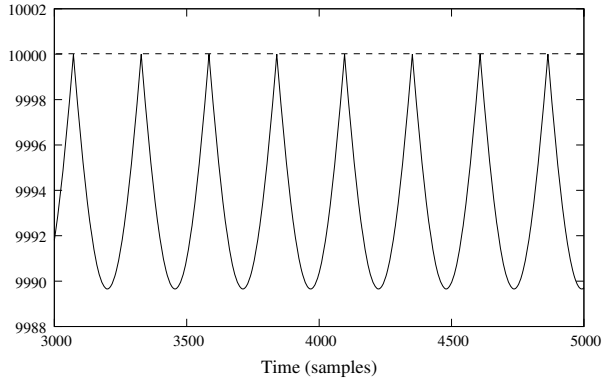


Figure 1: *The classic resampling technique is not adapted for the resampling of non zero-centered signals. Here is an example of the resampling of a constant signal at 10000 (represented by a dashed line). The resampling ratio is 256. The result of this resampling is given as a solid line, far from the constant we would have expected.*

## 4. ENHANCED RESAMPLING

### 4.1. Drawback of the Finite Reconstructor

In theory, resampling works on every kind of signal. However, in practice, only careful resampling can give good results. Practical resampling, as explained earlier, is based on a windowed sinc function. With this finite version of the cardinal sine, artifacts may appear if the signal to resample is not zero-centered. Indeed, an infinite number of samples of the signal are left out of the computation of the resampled signal by the finite version of the cardinal sine.

In fact, while the ideal (infinite) cardinal sine reconstructor considers the values of the entire signal to compute new samples, the practical (finite) reconstructor uses only the values closest to a given new sample. Hence, when using the practical reconstructor, the influence of the signal samples that are not included for the reconstruction is neglected. Of course, the shape of the cardinal sine function makes the center sample far more important, and the further from the center the samples, the less significance and influence they have for the newly computed sample. When the signal to resample is zero-centered, the left out samples can indeed be neglected. However, for a discrete-time $s[n] = c$ ($c$ is a constant), we would expect a reconstructed (continuous) $s(n) = c$. As shown on Figure 1, for signals with a large offset (here $c = 10000$, which is a typical value for the frequency parameter – in Hz – of partials in sinusoidal modeling), it is not the case. In the latter case, if we had added a sinusoidal modulation with an amplitude of 5 on the constant signal, the resampling would have shown a disastrous effect, completely loosing the modulation in the artifacts of the resampling.

The sinusoidal modeling parameters are control signals that fall into this category of signals that are not zero-centered, and are often modulated (for example vibrato for the frequency control signal). Hence, the classic resampling technique is not adapted to our synthesis / time-scaling purposes.

### 4.2. Enhancing the Resampling Technique

The sinusoidal modeling parameters are slow time varying. More precisely, we have shown in [4, 9] that each parameter consists of modulations below 20 Hz (*e.g.* tremolo or vibrato) together with an envelope, whose time evolution is slow – with a frequency content below 3 Hz.

Whereas the modulations are zero-centered, this is not the case for the envelope. In order to enhance the resampling, we propose to remove the envelope of the signal by modeling it as a piecewise polynomial. The idea is to perform the centering on parts of the signal that are about to be used for the reconstruction of a given sample. This means that we apply the centering on the samples that fall under the span of the practical reconstructor. The enhanced reconstruction procedure we use is summarized in Algorithm 2.

**Algorithm 2** enhanced_reconstruct($s$,$t$)

1: Center $r_p$ on time $t$
2: Compute $\Pi_d$ on the span of $r_p$
3: $m \leftarrow$ classic_reconstruct($s - \Pi_d$,$t$)
4: Return $m + \Pi_d(t)$

Here, $\Pi_d$ represents a polynomial of degree $d$, computed using the well-known least-square method (see for example [9] for details). Hence, removing polynomials of different degrees will have different effects.

Removing a null polynomial before performing the reconstruction is the classic resampling technique, which works very well for (zero-mean) PCM audio signals, but has the drawback we explained earlier. Removing the local mean seems to significantly enhance the result of the resampling, as shown on Figure 2. This is equivalent to removing a constant polynomial ($d = 0$), the constant being the mean, well suited for the frequency parameters when the sound is stationary. Removing a linear polynomial ($d = 1$) works well with the (unwrapped) phase parameters of stationary sounds. In the case where the frequency is based on a linear envelope, a quadratic polynomial ($d = 2$) is necessary to resample the corresponding unwrapped phase; and for quadratic frequency envelopes, we need cubic polynomials ($d = 3$) for the unwrapped phases.

By performing these subtractions to the signal, the envelope is removed and only the modulations are kept. These modulations are zero-centered and thus well resampled using the classic resampling techniques.

## 5. RESULTS

The effect of the removal of a constant is shown on Figure 2. We can see that the overall quality of the resampling in significantly improved. However, the steep parts of the signal are problematic.

In our experiments, the polynomials we have removed are constant, linear (first degree), quadratic (second degree) and cubic (third degree). The comparison of the performances of all these resampling techniques on a sinusoidal signal is shown on Figures 3 and 4. The results of resampling on a sine signal with a quadratic envelope are shown on Figure 5.

It has to be noted that the error of the resampling technique using a higher degree polynomial is lower only for low frequency signals, while being almost identical for higher frequency signals. This is explained by the fact that from a given frequency, the sinusoidal signal is no longer an envelope to be removed but a modulation, which is thus no longer captured by the polynomial.

This enhancement is thus significant for our purpose of resampling the control signals since they contain slowly evolving envelopes that are well taken into account by the polynomials.

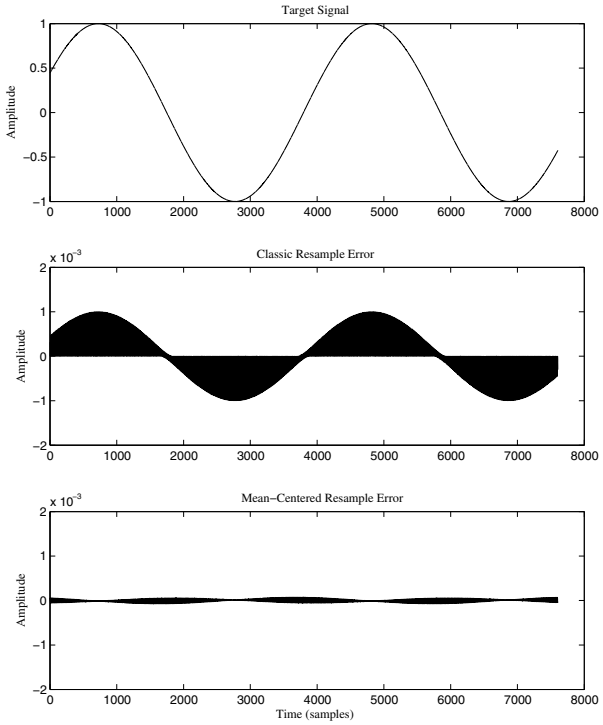


Figure 2: *Resampling a simple low-frequency sine function using classic resampling and the local mean-centering resampling method. The target signal was downsampled, then resampled back to the original rate. The resampling ratio does not influence the maximal error signal. The error depends on the amplitude of the signal to resample for the first method, while it depends only on the derivative of the signal for the second method. The resampling errors at the boundaries have not been shown on this figure. The reconstructor has a size of $N = 21$ samples.*

Moreover, during informal listening tests, it appeared obvious that the classic resampling of control signals produced unwanted artifacts (adding another tone to the sound), whereas using our new method the artifacts were removed.

One last point is about the windowing of the reconstructor function. As said earlier, the sinc reconstructor has to be windowed in order to be applicable in the finite case. Thus, we have compared the Kaiser-windowed sinc that is used in Matlab against the Hann-windowed sinc we use in our version of the resampler. The results are shown in the Figure 6. For our purpose (resampling low frequency signals), the Hann-window reconstructor performs best. It has to be noted that the maximal error values seem to be describing the Kaiser and the Hann windows respectively, and that the width of the bumps seems to be related to the window size.

## 6. CONCLUSION

We have presented a new way to resample signals when dealing with non zero-centered signals. The method is quite simple and yet efficient. The results are significantly improved, from the points of view of error measures and informal listening tests.

This new technique has been presented here in the context of additive synthesis. The results also confirm the validity of the sinusoidal plus polynomial model for the parameters of sinusoidal modeling. However, it could be applied to other types of signals.

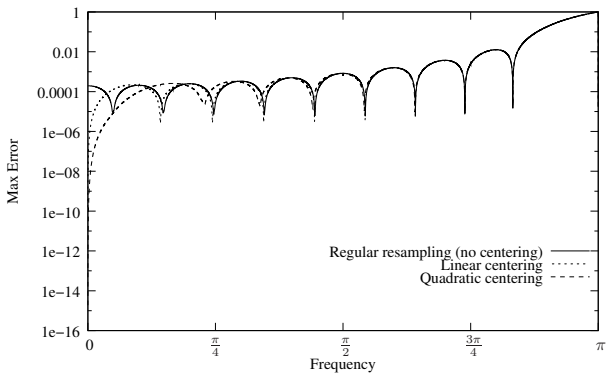


Figure 3: *Maximal resampling error using various resampling techniques plotted against the frequency of the sinusoid they have been applied on. Here, the signal is already zero-centered.*

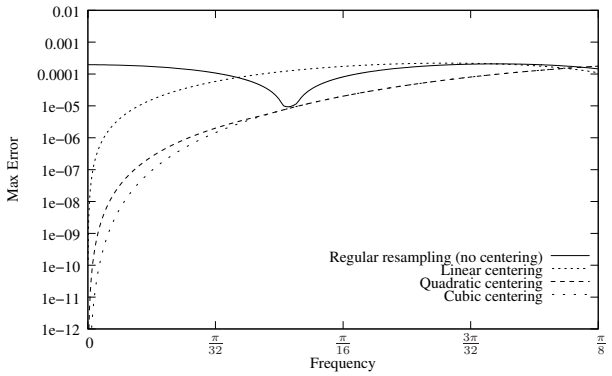


Figure 4: *Maximal resampling error using various resampling techniques plotted against the frequency of the sinusoid they have been applied on. We have zoomed on the lower frequency to show that higher order polynomials help enhancing the resampling. Mean (constant) centering has not been plotted since it is very similar to linear centering.*

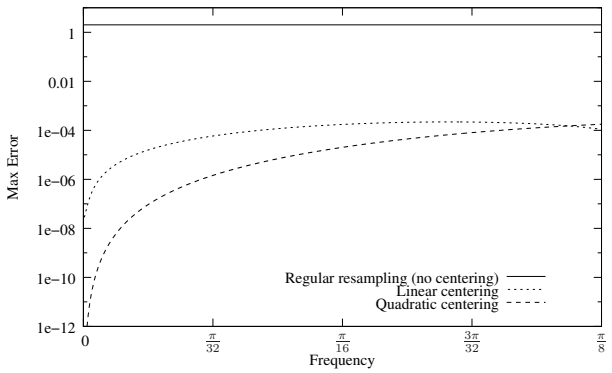


Figure 5: *Maximal resampling error using various resampling techniques plotted against the frequency of the sinusoid they have been applied on. The test signal is a sinusoid of given frequency added to a second degree polynomial (here $x^2$). Here, the linear-centering method is not as good as on the previous figure, while the quadratic method still performs very well (for low frequencies).*

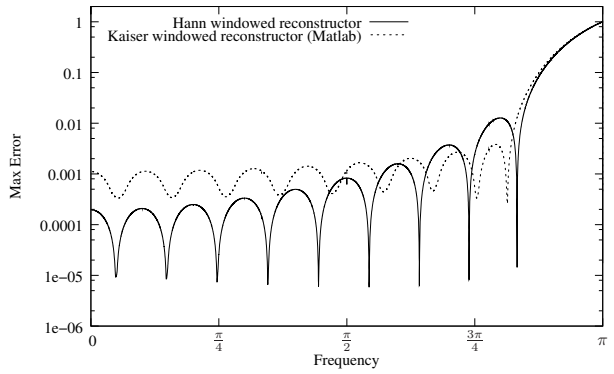


Figure 6: *Maximal resampling error using various resampling techniques. Here, Matlab's resampling technique using a Kaiser-windowed ($\beta = 5$) cardinal sine as reconstructor is compared to the technique using a Hann-windowed reconstructor over single sinusoids of various frequencies. The reconstructor contains $k = 10$ wings (lobes) on each side.*

## 7. ACKNOWLEDGMENTS

This research was carried out in the context of the SCRIME *(Studio de Création et de Recherche en Informatique et Musique Électro-acoustique)*, and was supported by the French GIP ANR *(Agence Nationale de la Recherche)*, DESAM project (ANR-06-JCJC-0027-01).

## 8. REFERENCES

[1] R. J. McAulay and T. 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.

[2] L. Girin, S. Marchand, J. di Martino, A. Röbel, and G. Peeters, "Comparing the Order of a Polynomial Phase Model for the Synthesis of Quasi-Harmonic Audio Signals," in *Proc. IEEE WASPAA*, New Paltz, New York, USA, October 2003.

[3] J. O. Smith, "Digital Audio Resampling," World Wide Web, URL: http://ccrma.stanford.edu/~jos/resample/.

[4] S. Marchand and M. Raspaud, "Enhanced Time-Stretching Using Order-2 Sinusoidal Modeling," in *Proc. Digital Audio Effects (DAFx) Conference*, Naples, Italy, October 2004, pp. 76–82.

[5] X. Serra and J. O. Smith, "Spectral Modeling Synthesis: A Sound Analysis/Synthesis System Based on a Deterministic plus Stochastic Decomposition," *Computer Music Journal*, vol. 14, no. 4, pp. 12–24, 1990.

[6] S. Marchand and R. Strandh, "InSpect and ReSpect: Spectral Modeling, Analysis and Real-Time Synthesis Software Tools for Researchers and Composers," in *Proc. Int. Computer Music Conference*, Beijing, China, October 1999, pp. 341–344.

[7] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Prentice Hall, 1999, second edition.

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

[9] M. Raspaud, S. Marchand, and L. Girin, "A Generalized Polynomial and Sinusoidal Model for Partial Tracking and Time Stretching," in *Proc. Digital Audio Effects (DAFx) Conference*, Madrid, Spain, October 2005, pp. 24–29.