TIME DOMAIN ATTACK AND RELEASE MODELING
Applied to Spectral Domain Sound Synthesis

Cornelia Kreutzer, Jacqueline Walker
Department of Electronic and Computer Engineering, University of Limerick, Limerick, Ireland
cornelia.kreutzer@ul.ie, jacqueline.walker@ul.ie

Michael O'Neill
School of Computer Science and Informatics, University College Dublin, Dublin, Ireland
m.oneill@ucd.ie

Keywords: Audio Signal Processing, Spectral Music Synthesis, Modeling Real Instrument Sounds.

Abstract: We introduce a time-domain model for the synthesis of attack and release parts of musical sounds. This approach is an extension of a spectral synthesis model we developed: the Reduced Parameter Synthesis Model (RPSM). The attack and release model is independent from a preceding spectral analysis as it is based on the time domain sustain part of the sound. The model has been tested with linear and polynomial shaping functions and produces good results for three different instruments. The time-domain approach overcomes the problem of synthesis artifacts that often occur when using spectral analysis/synthesis methods for sounds with transient events. Moreover, the model can be combined with any synthesis model of the sustain part and offers the possibility to determine the duration of the attack and release parts of the sound.

1 INTRODUCTION

In the standard sinusoidal model used for speech (McAuley and Quatieri, 1986) and musical sounds (Serra, 1989; Serra and Smith, 1990), the harmonic part of a given signal is modeled as a sum of sinusoidal components with time-varying amplitude, frequency and phase. The remaining sound components are then usually added to the model by using some type of noise model. However, these methods are not sufficient to model transient parts of the signal. Transients mainly occur during the onset of a sound and have long been known to be important for our perception of timbre (Grey, 1977; McAdams and Cunibile, 1992). A number of methods have been introduced to provide a sinusoidal sound model that is also capable of modeling transients more accurately. Jensen (Jensen, 1999) proposed an amplitude model in the frequency domain where the amplitude envelope of each harmonic partial is fitted with appropriate functions. Verma and Meng (Verma and Meng, 2000) proposed an extension of the Spectral Modeling Synthesis (SMS) framework to model transients by performing sinusoidal modeling in the frequency domain. This is based on the observation that transient components of a signal show the same behavior in the frequency domain as sinusoidal components in the time domain. Methods using exponentially damped sinusoids to model transient events more accurately have been proposed by (Nieuwenhuijse et al., 1998; Boyer and Essid, 2002; Hermus et al., 2005). Meillier and Chaigne (Meillier and Chaigne, 1991) applied an autoregressive model which improved the spectral analysis of percussive sounds compared to the standard FFT approach. In (Masri and Bateman, 1996) the spectral analysis is improved by synchronizing the analysis window to transient events. This overcomes the problem that transient events, which occur at a certain time, become diffused during the synthesis process when using the standard sinusoidal model.

All these approaches focus on improving the sinusoidal sound model in the spectral domain. Thus, the transient of the analyzed sound is captured more accurately and artifacts during the synthesis process are reduced. However, these interventions are inherently limited in efficiency by the time-frequency uncertainty principle.

In contrast to that, we propose a time domain model for attack and release parts of musical sounds. The model is combined with a spectral synthesis model: the Reduced Parameter Synthesis Model (RPSM) (Kreutzer et al., 2008). We model the sound attack and release independently from a preceding spectral analysis of these parts of the signal. Therefore, we exclude artifacts that might occur when using a transient analysis-synthesis model. These artifacts are due to interpolations of the sound partials between signal frames when it comes to the synthesis process. Our time domain approach in combination with RPSM also leads to a reduction in computational requirements, because it does not require us to model the amplitude envelope of each partial individually in detail.

2 REDUCED PARAMETER SYNTHESIS MODEL

2.1 Frequency estimation

To determine the frequency values within the synthesis model we use a flexible model that is not based directly on a preceding spectral analysis but on the basic knowledge about the sound. The fundamental frequency, or pitch, as well as the number of harmonic partials are user defined values. This is particularly important if the synthesized sound lies outside the range of the instrument the model is supposed to mimic. Also, within the range of an instrument there is no restriction of the pitch value or the number of harmonics that can be chosen, since both values are entirely user defined. Consequently we can model whole tones, semitones or quarter tones of an instrument as well as other notes whose pitch value is anywhere in between or outside these tones.

We apply a random walk to several frequency partials in order to reconstruct the naturalness of the sound. Figure 1 (top) shows a representative result of the SMS partial tracking algorithm (Amatriain et al., 2002): in this particular case the result for a flute note (A4, played forte, non Vibrato (RWC, 2001, Instrument Nr.33, Flute:Sankyo)). As illustrated, some of the partials, especially the upper ones, show a certain amount of variation or *noisiness*. Due to this noisiness a reconstruction of the sinusoidal parts of the sound does keep the sound characteristics of the original recording, although the residual part of the signal is neglected for the reconstruction. Based on this observation we incorporate this noisiness into the sinusoidal partials of our synthesis model rather than defining a separate noise model. This is achieved by the use of a one-dimensional random walk (Feller, 1968). A one-dimensional random walk can be described as a path starting from a certain point, and then taking successive steps on a one-dimensional grid. The step size is constant and the direction of each step is chosen randomly with all directions being equally likely.

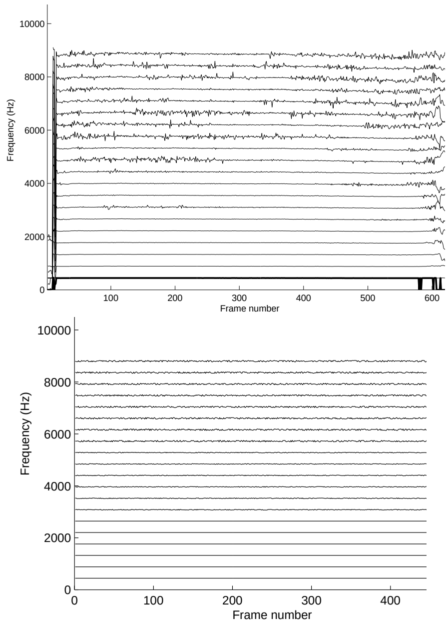

Figure 1: SMS frequency analysis result of a flute note recording (A4 forte, non Vibrato (top) and estimated frequency tracks for a note with 20 harmonics with the same fundamental frequency (bottom)

For the purpose of our synthesis model random walks are applied to certain harmonic partials in the following way. First, the harmonic partials are divided into three groups, where each group represents a third of the overall number of harmonics. This follows from the results of the SMS analysis which shows different levels of variations for the lower, the middle and the upper harmonics. Concerning the lowest third of the harmonic partials - starting from the fundamental frequency - no random walk is applied as the analysis of these lower partials shows very little variation. For the middle and the upper harmonic partials a random walk is applied, where the starting point of the random walk is determined by the basic frequency of the harmonic partial. Basic frequency in that case means the integer multiple of the fundamental frequency. Again, from the analysis result it can be seen that the upper harmonics show more variation than the middle ones. Due to that, and after testing several levels of noisiness, the step size of the random walk was set to 30 Hz for the upper harmonics and to 15 Hz for the middle ones. Figure 1 (bottom) shows the estimated frequency tracks for the synthesis model compared to an SMS analysis result with the same conditions (top).

2.2 Amplitude estimation

In contrast to the frequency estimation which is not directly taken from the sound analysis results, we use SMS analysis results as a basis for estimating the amplitude values of the harmonic partials.

However, we reduce the number of parameters to provide a flexible synthesis model that is mostly independent from the preceding sound analysis process. This also reduces the computational complexity of the synthesis process. Additionally, our main concern is to keep the quality and naturalness of the musical sound after the synthesis process in order to mimic real instruments. Therefore, three different methods have been applied to the analysis amplitude data. In particular we have carried out amplitude estimation by means of local optimization, lowpass filter estimation and polynomial fitting.

We start by applying a standard SMS analysis (Amatriain et al., 2002) to obtain the amplitude parameters. To increase the number of spectral samples per Hz and improve the accuracy of the peak detection process, we apply zero-padding in the time domain - using a zero-padding factor of 2. The STFT was performed with a sampling rate of 44.1 kHz and a Blackman-Harris window with a window size of 1024 points and a hop size of 256 points. From the resulting frequency spectrum, 100 spectral peaks were detected and subsequently used to track the harmonic partials of the sound. The number of partials to be tracked was set to 20. This analysis has been applied to sound samples taken from the RWC database (RWC, 2001), in particular to all notes over the range of a flute, a violin and a piano. Given the amplitude tracking results only one representative note for each instrument has been chosen to provide the basis for the amplitude values of the RPSM. However, this could be changed in the future into using more than one amplitude template, e.g., using different templates for the low notes and the high notes within the range of an instrument.

2.2.1 Local optimization.

The SMS analysis provides one amplitude value for each harmonic partial and for each frame of a given sound signal. We reduce that parameter size by determining the local maxima of each amplitude track. This reduces the number of parameters to about a third of the SMS analysis result. For example, for the flute note (A4, played forte, non Vibrato) the SMS analysis consists of 12680 amplitude values. This is reduced to 3015 values which represent all the local maxima of the 20 harmonic partials.

We determine the local maxima of each amplitude envelope by using the first derivative of the amplitude envelope function f_e. Suppose we want to determine if f_e has a maximum at point x. If x is a maximum of f_e, then f_e is increasing to the left of x and decreasing to the right of x. The same principle applies for local minima of f_e. If x is a minimum of f_e, then f_e is decreasing to the left of x and increasing to the right of x. In contrast, if f_e is increasing or decreasing on both sides of x, then x is not a maximum or a minimum. In terms of the first derivative of f_e this means, that f_e is increasing when the derivative is positive, and that f_e is decreasing when the derivative is negative.

To compute the shape of each amplitude track, necessary for the synthesis process, we then perform a one-dimensional linear interpolation between the local maxima of the track. Figure 2 (top right) illustrates an example of estimated amplitude tracks using this approach as well as the SMS analysis results (top left) for a violin note. As can be seen the shape of the tracks are close to the SMS analysis result. However, this is not the case for the attack and the release part of the sound.

2.2.2 Lowpass filter estimation.

The second curve fitting method applied uses a lowpass filter to estimate the overall amplitude envelope of each partial. We apply a 3^{rd} order Butterworth lowpass filter to the analysis data. We perform zero-phase digital filtering by processing the input data in both the forward and reverse directions. After filtering in the forward direction, the filtered sequence is reversed and runs back through the filter. The resulting sequence has precisely zero-phase distortion and double the filter order. As shown in Figure 2 (bottom left) the envelope shapes of the estimated amplitude tracks are similar to the local optimization estimation. However, the estimation takes significantly longer to be performed. Similar to the local optimization method, no sufficient estimate for the synthesis of the attack and the release of the sound signal is obtained.

2.2.3 Polynomial interpolation.

Additionally we performed polynomial fitting to obtain an estimate for the several amplitude tracks. For each amplitude envelope the coefficients of a polynomial of degree 6 are computed that fit the data - in our case the analysis result - in a least squares sense. This computation is performed using a Vandermonde matrix (Meyer, 2000)

$$V = \begin{bmatrix} 1 & \alpha_1 & \alpha_1^2 & \dots & \alpha_1^{n-1} \\ 1 & \alpha_2 & \alpha_2^2 & \dots & \alpha_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha_m & \alpha_m^2 & \dots & \alpha_m^{n-1} \end{bmatrix} \qquad (1)$$

since solving the system of linear equations $Vu = y$ for u with V being an $n \times n$ Vandermonde matrix is equivalent to finding the coefficients u_j of the polynomial

$$P(x) = \sum_{j=0}^{n-1} u_j x_j \tag{2}$$

of degree $\leq n-1$ with the values y_i at α_i (Meyer, 2000).

An example for the estimation result is shown in Figure 2 (bottom right). Unlike the two other methods being used, the results are very smooth amplitude envelopes. That is, all the small variations that can be seen in the SMS analysis result are missing. Nevertheless, the synthesized sounds preserve the timbre of the particular instrument and the sound quality of the original recordings. Regarding the flute and the violin, the polynomial estimation also gives a sufficient estimate for the attack and the release part of the sound.

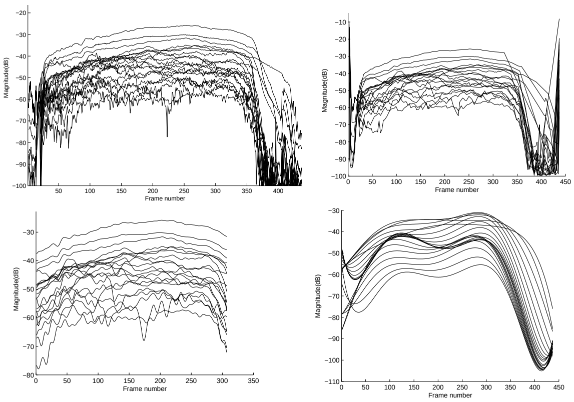

Figure 2: Violin note, A#3, forte, non vibrato: SMS amplitude analysis result, estimated amplitude tracks using local optimization, LP filter estimation, and polynomial fitting (from top left to bottom right)

2.3 Spectral synthesis

With the calculated frequency and amplitude parameters we synthesize a new sound using an additive synthesis method, which is based on spectral envelopes and the inverse Fast Fourier Transform (Rodet and Depalle, 1992). Compared to the traditional use of oscillator banks for additive synthesis, this is a more efficient and faster approach.

3 TIME DOMAIN ATTACK AND RELEASE MODELING

To improve the RPSM model in terms of sound attack and release, we extend the synthesis model with a time domain attack and release model. The approach we are using corresponds to multiplying the sustain portion of the sound by a time domain window. Thus, we accomplish the desired transformation in the frequency domain. This reduces the complexity of the model significantly, as the alternative is to map each amplitude partial in detail through the attack and release stages in the frequency domain.

3.1 Linear Modeling

To synthesize the attack and release portions of the sound we require the sustain part of the RPSM synthesized signal in the time domain and the durations of the attack and the release parts we want to model. Both duration times may be user defined and thus can be changed according to the signal length and the instrument.

The attack is computed as follows. From the beginning of the sustain signal we take a part with the same length as the attack duration. This is the part of the signal that is shaped to gain the attack portion. Then, we carry out a point wise multiplication of this part with a linear shaping function $y_{att}(n) = k * x(n)$, for $n = \{1, 2, \ldots, N\}$. This can be compared to the application of a time domain window. The parameters of y_{att} are set according to the given signal, with $y_{att}(1) = 0$ to ensure that the sound starts at 0. The length N of the shaping function is equal to the duration of the attack. The slope k of the function is determined by $k = y_{att}(N) - y_{att}(1)/(N-1)$. Thus, $y_{att}(N) = 1$. This allows for a smooth transition between the attack and the sustain portion of the sound when they are joined.

For the release part of the sound the procedure is similar to the attack, but here we perform the shaping at the end of the sustain signal. From the end of the sustain signal a part with the same length as the release duration is taken. To compute the sound release we carry out a point wise multiplication of this signal part with a linear shaping function $y_{rel}(m) = -k * x(m)$, for $m = \{1, 2, \ldots, M\}$. The function length M is equal to the duration of the release and $y_{rel}(M) = 0$ to ensure that the sound terminates to 0. The negative function slope k is determined by $k = -y_{att}(N) - y_{att}(1)/(N-1)$. To ensure a smooth transition between the sustain and the release part of the sound the function parameters are set so that $y_{rel}(1) = 0.5$. Although setting $y_{rel}(1) = 0.5$

works well for the three different instruments we have tested so far, it must be noted that this value is largely dependent on the shape of the given sustain signal.

After the computing attack and the release portions, both are connected to the original sustain part of the RPSM synthesized sustain signal. To do so, the three separate waveforms are concatenated in the order attack - sustain - release.

3.2 Polynomial Modeling

To obtain a more smooth and realistic attack and release, we also used a second order polynomial as a shaping function. Setting the function parameters and computing the particular attack and release signals has been performed similarly to the linear shape.

To compute the attack a part of length N - equal to the attack duration - is taken from the beginning of the sustain signal. This waveform is then point wise multiplied with the polynomial function $y_{att}(n) = k * x(n)^2$, for $n = \{1, 2, \ldots, N\}$. As with the linear shaping function, the function parameters are set to ensure $y_{att}(1) = 0$ and $y_{att}(N) = 1$ Therefore, the sound starts at 0 and we gain a smooth transition between the attack and the sustain portion of the synthesized waveform.

For the sound release we also use a second order polynomial, but this time with a negative slope. From the end of the sustain signal a part of length M - equal to the release duration - is taken. Subsequently, this waveform is point wise multiplied with the polynomial function $y_{rel}(m) = -k * x(m)^2$, for $m = \{1, 2, \ldots, M\}$. The function parameters are set so that $y_{rel}(1) = 0.5$ and $y_{rel}(M) = 0$. This provides for a smooth transition between the sustain signal and the release portion and ensures that the sound terminates to 0. Again, note that the setting of $y_{rel}(1)$ depends on the shape of the given sustain signal. For the instruments we have tested so far, 0.5 has shown to be the most suitable setting.

After computing the sound attack and the release, both are connected to the original sustain part of the RPSM synthesized sound to form the overall synthesized time domain signal. The presented shaping functions have produced good synthesis results for the tested instruments. However, the method to determine the shaping function for the attack and release model could be further improved to overcome any possible dependencies on the actual sound signal. Another way to determine the parameters of the shaping function would be by modeling the shape of the release component on the actual shape of the time domain envelope.

4 EMPIRICAL EVALUATION

Figure 3 shows comparisons of the original sound sample, the SMS synthesis result and the RPSM synthesis result with the time-domain attack and release for the three different instruments being used. In all three cases the RPSM amplitude values have been estimated using the local optimization method described in Section 2.2.1. For the SMS results we applied a standard SMS analysis/synthesis (Amatriain et al., 2002).

The presented RPSM model has been tested for notes covering the whole range of a flute (37 notes), a violin (64 notes) and a piano (88 notes). An SMS analysis has been carried out for all these notes using recorded samples from the RWC database (RWC, 2001). The analysis was done to find a representative note for the presented amplitude model and to compare the synthesis results obtained by our model with the standard SMS results. The frequency estimation works well and allows a large flexibility when choosing the fundamental frequency. Due to the random walk that is applied to higher frequency partials the synthesized sound keeps the natural noisiness of the real instrument recording without the need for a separate noise model. Concerning the three different amplitude estimation methods, all of them perform well when estimating the sustain part of the signal. Although, only the polynomial fit gives a satisfactory estimate for the attack and the release parts of the signal at the same time. The combination of the basic RPSM model with the time domain attack and release model overcomes these difficulties and provides an efficient method to model the beginning and the end of the sound. Moreover, the attack/release model is independent from a preceding spectral analysis and from the computation of the sustain portion of the sound. Using this approach we avoid artifacts that result from smoothing transient events, a problem connected with spectral transient analysis/synthesis methods. Together with the user defined duration, the new approach presented here allows for a flexible synthesis model.

5 CONCLUSIONS

We introduced a time domain attack and release model as a an extension of a Parametric Synthesis Model for musical sounds. To obtain the shape of the note onset and release we use linear and polynomial shaping functions. The RPSM model has been tested for notes covering the whole range of three different instruments; a flute, a violin and a piano.

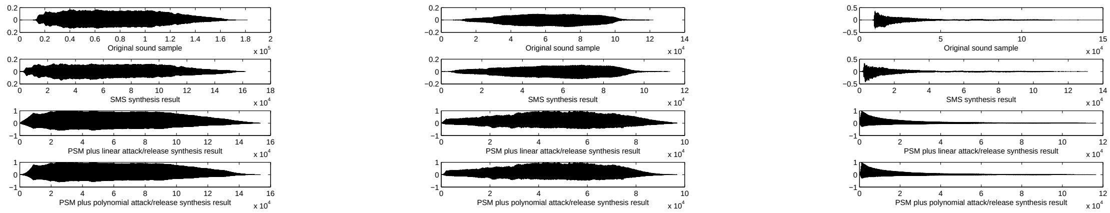

Figure 3: Time domain plots of original sound, SMS result and RPSM result with attack/release model(from left to right: flute, violin, piano)

Future work will be focused on analyzing the effects of the time domain model on the spectral representation of the signal and using the actual sound envelope for shaping the sound attack and decay. Moreover, we are going to perform listening tests to gain detailed results for a comparison between the original recorded sound samples, SMS synthesis results and the presented RPSM model.

6 ACKNOWLEDGMENTS

This work was supported by the Science Foundation Ireland (SFI) under the National Development Plan (NDP) and Strategy for Science Technology & Innovation (SSTI) 2006-2013.

REFERENCES

Amatriain, X., Bonada, J., Loscos, A., and Serra, X. (2002). *Spectral Processing* in *DAFx – Digital Audio Effects*, chapter 10, pages 373–439. edited by Udo Zoelzer. John Wiley & Sons.

Boyer, R. and Essid, S. (2002). *Transient Modeling with a Frequency–Transform Subspace Algorithm and "Transient+Sinusoidal" Scheme*. pages 865–868, vol.2. IEEE International Conference on Digital Signal Processing (DSP). Thera, Greece.

Feller, W. (1968). *Introduction to Probability Theory and its Applications*. Wiley series in probability and mathematical statistics. John Wiley & Sons, 3rd edition.

Grey, J. M. (1977). *Multidimensional Perceptual Scaling of Musical Timbre*. Journal of the Acoustical Society of America, 61(5):1270–1277.

Hermus, K., Verhelst, W., Lemmerling, P., Wambacq, P., and van Huffel, S. (2005). *Perceptual Audio Modeling with Exponentially Damped Sinusoids*. Signal Processing, 85(1):163–176. Elsevier North-Holland, Inc., Amsterdam, The Neatherlands.

Jensen, K. (1999). *Timbre Models of Musical Sounds*. PhD thesis, University of Copenhagen, Copenhagen, Denmark.

Kreutzer, C., Walker, J., and O'Neill, M. (2008). *A Parametric Model for Spectral Sound Synthesis of Musical Sounds*. International Conference on Audio, Language and Image Processing (ICALIP). Shanghai, China.

Masri, P. and Bateman, A. (1996). *Improved Modelling of Attack Transients in Music Analysis–Resynthesis*. pages 100–103. International Computer Music Conference (ICMC), IEEE. Hong Kong, China.

McAdams, S. and Cunibile, J.-C. (1992). *Perception of Timbral Analogies*. Physical Transactions of the Royal Society, 336. Series B.

McAuley, R. and Quatieri, T. (1986). *Speech Analysis/Synthesis Based on a Sinusoidal Representation*. 34:744–754. IEEE Transactions on Acoustics, Speech and Signal Processing.

Meillier, J.-L. and Chaigne, A. (1991). *AR Modeling of Musical Transients*. pages 3649–3652. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Toronto, Canada.

Meyer, C. (2000). *Matrix Analysis and Applied Linear Algebra*, chapter 4. SIAM, Philadelphia, PA.

Nieuwenhuijse, J., Heusdens, R., and Deprettere, E. (1998). *Robust Exponential Modeling of Audio Signals*. pages 3581–3584. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Seattle, Washington, USA.

Rodet, X. and Depalle, P. (1992). *Spectral Envelopes and Inverse FFT Synthesis*. 93rd AES Convention, San Francisco, AES Preprint No. 3393 (H-3).

RWC (2001). *Real World Computing (RWC) Music Database – Musical Instrument Sound*. RWC–MDB–I–2001 No. 01–50, Tokyo, Japan.

Serra, X. (1989). *A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition*. PhD thesis, Stanford University.

Serra, X. and Smith, J. (1990). *Spectral Modeling Synthesis:A Sound Analysis/Synthesis Based on a Deterministic plus Stochastic Decomposition*. Computer Music Journal, 14(4):12–24.

Verma, T. S. and Meng, T. H. Y. (2000). *Extending Spectral Modeling Synthesis with Transient Modeling Synthesis*. Computer Music Journal, 24(2):47–59.