Providing Rhythm Patterns in Sound Synthesis

Lars Graugaard

Department of Software and Media Technology
Aalborg University Esbjerg
lag@cs.aaue.dk

Abstract

A parameterized periodic synthesis update model for applying organized rhythm and periodicity to sound synthesis is described. An event pattern is combined with a distance pattern to create a dynamic pattern in a normalized space, which is used for parameter update according to distance to the subsequent parameter value. The model has been tested with linear and non-linear synthesis techniques, and it has been used in performance of interactive music.

1 Introduction

Rhythm is generally not part of timbre synthesis. Issues of rhythm are not inherent to most synthesis methods, though some techniques such as granular synthesis use a time continuum from random to fixed periodicity, or other time structuring methods.

Rhythm patterns are mostly generated as global amplitude envelopes applied after the synthesis step, in some cases affecting aspects of the sound such as the fundamental frequency. The envelope may be derived from acoustic instruments, often as an attack-decay-sustain-release sequence (ADSR). Values for the ADSR envelope are often derived from the regular categories of acoustic instruments, such as percussion instruments, wind instruments, and strings. Do-it-yourself drum machines abound in laptop music, and procedurally applying ADSR envelopes to a signal fed into the machine in real-time is a trivial yet highly effective technique.

Another technique is to affect the spectral envelope with local envelopes in parallel, independently affecting separate aspects of the sound.

Very few reports on generalized integration of independent rhythm structures into sound synthesis are available. In (Lyon 1997) an approach is described that incorporates rhythm matrices into an audio processing schemata for separate processing of each sonic event. A more elaborate work on pattern generation is described in (Morris 2004), but sound synthesis is soundfile playback. Both reports imply a stylistical binding which the work reported in this paper has avoided.

Quite a few works on rhythm are available to us but they treat rhythm as a separate expressive entity with self-referencing descriptions of its properties (Stockhausen 1957, Schillinger 1976, Cooper et al. 1960, Lerdahl et al. 1983, Dobrian 1995). Rhythm in music is constituted by several interconnected elements such as accent, metre, and tempo and theories requiring periodicity as the fundamental aspect of rhythm are opposed by theories that accept non-recurrent configurations.

The present paper outlines a method for rhythm as a periodic means to affect timbre inside the synthesis technique's own parameter space. It does not propose any specific way of treating rhythm in electronic music and it does not propose or imply any sort of "theory" for its use. It proposes the application of rhythm at an earlier stage of the sound synthesis flow, namely at the moment of deciding timbrically significant parameter values. It also proposes a consistent way to handle these parameter values whereby timbrical changes can be differentiated in ways similar to those of rhythm patterns in the accent/duration combination.

One of the consequences of this approach is that rhythm and timbre become integrated parts since the sound synthesis methods will determine the timbrical distance of the rhythm patterns, and hereby determine their perceived strength and degree of discernibility in the overall sound.

2 Application

In our work rhythm is applied to standard synthesis techniques through a sequence of timed synthesis parameter updates (Figure 1). The model is divided into *Event Pattern*, *Distance Pattern*, and *Parameter Space* mapped onto the *Synthesis Step*. The *Shape Settings* step define properties of the Sound Synthesis step's transition between successive parameter updates. A *feedforward* connection between Sound Synthesis step and *Output* is inserted to handle unwanted artefacts of our method.

The degree of parameter disparity is expressed by the parameter distance value (PDV) in the range 0.-1. (Figure 2). The PDVs are mapped to the parameter range and the parameter range is defined independently, as a property of the chosen sound synthesis method. No amplitude envelope is applied, and the synthesis parameters affected are defined by the synthesis method and determined by musical or compositional decisions. The result is a dynamic, rhythmizised pattern of changes to the synthesis timbre.

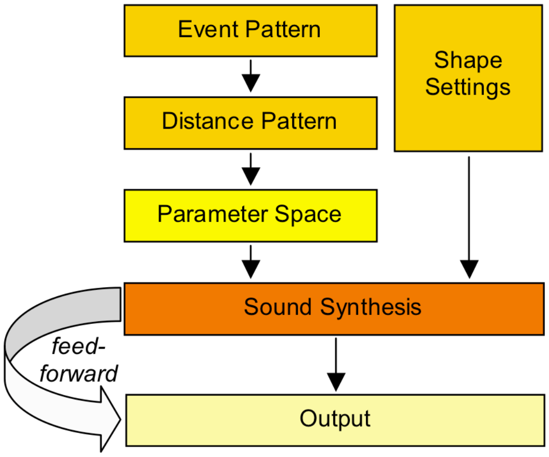

Figure 1. System layout.

2.1 Parameter Storage

The primary parameters are divided into a *Distance Pattern* for PDV storage and an *Event Pattern* for metrical and probabalistic control. A sequence of event delta times and event propabilities make up the Event Pattern, and the Distance Pattern contains a sequence of PDVs. The Event and Distance Patterns can contain different number of entities and the model is exposed to real time affecting. A feedforward link to the synthesis step has been inserted as discussed in section 2.4, to compensate for artefacts mainly caused by lack of amplitude control. Secondary parameters are a number of *Shape Settings*, discussed in section 2.3.1.

2.2 Event Pattern

Events are algorithmically generated, securing constant statistical characteristics inside a constantly varying rhythm. Event Patterns can have a variable length in the range 1-32 main events. Duration and weight of each event can be dynamically updated, and events at pulse subdivision locations as well as a window of randomization for each event can be applied. A stability value for measure group, single measures, and at inter-pulse level provide a focus/nebulous continuum for dynamic event variation.

2.3 Distance Patterns

A Distance Pattern is a sequence of PDVs in the normalized range. It is used in place of a traditional accent pattern, but can better be compared to a sound pattern. A PDV determines the distance in parameter space from the preceeding value, and the PDV sequence expresses the varying diversification of the parameters as a sequence of implied timbre differences of a synthesis model.

This suggests a timbre affection layer, but the model has no knowledge of the timbre itself. Consequently, some degree of entropy is introduced, but the patterns are clearly discernable according to the Distance Pattern, because the sequence of PDVs provide enough difference to perceptualy articulate the pattern. PDV is hereby equivalent to amplitude because larger distances provide clearer event onset. The synthesis method is fundamental in determining the *sound* output, but the parameter range has a large effect on the *intensity* of the output. Parameter ranges are chosen and affected for the desired degree of contrast among rhythm events, where larger contrast provides higher intensity.

Different procedures for fanning out the PDVs have been tested. The PDVs are used for calculating the next output in a normalized space according to the familiar equation

$$n = (n-1) \pm d + \left(R - 0{,}5 * r\right), \qquad \textbf{(1)}$$

where d is distance and r is the window size centered at d relative to n-1 for n to fall within, as a random number R in the range 0-r. An algorithm takes care of keeping n inside the range 0.-1. at the expected distance.

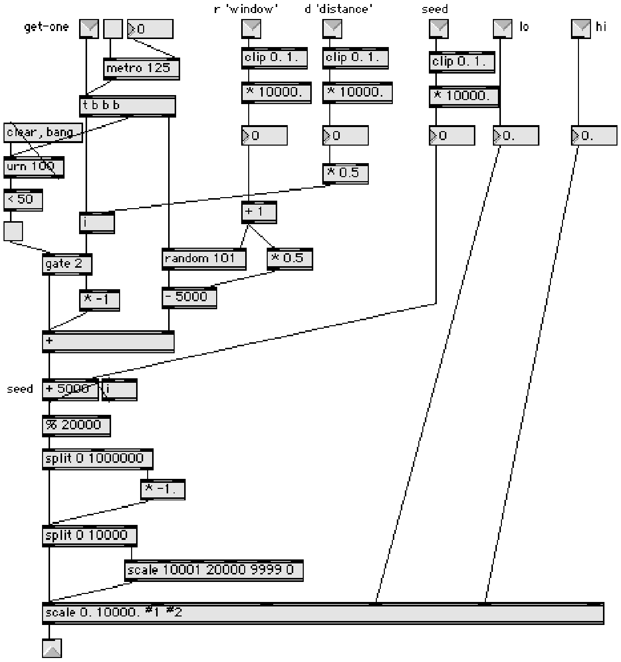

Figure 2. PDV instantiation – code example.

The window size r provides irregularity to the pattern and to the pattern's overall direction. The changing sign will vary the pattern while the PDV sum will direct the pattern's spectral centroid upwards if $\sum d$ is positive and downwards if negative. The changing of the sign of d will be the determining factor of this. The sign changes statistically 50/50 but the execution of each pattern will normally be either positive or negative, because even in cases where the number of positive/negative signs is at 50/50, then the values will not balance out if the sum of positive *d* does not equal the sum of negative *d*.

2.3.1 Shape Settings

Transition time between parameter values are used to affect a pattern's articulation. It can be compared to the attack portion of an ADSR envelope, and can be instantaneous or gradual up to the total duration of the event. This will change the degree of 'roughness' of the rhythmization of the sound synthesis.

An equivalent to the release stage of the ADSR envelope is introduced, in that the parameter is ramped to a fixed value in the last portion of the event's duration.

2.4 Real-time Transformation

An interesting case of real-time transformation is through analysis of the synthesis output. This is a required step because we introduce a degree of entropy, and we want to make sure that we stay within the limits of what is perceptually acceptable (and technically legal). We therefore want to feed perception related data into the output step of the model. Ironically, amplitude data is the most relevant for this. Not addressing the amplitude in our model does not mean that amplitude does not affect the synthesis in significant ways. On the contrary, output amplitude may be very much affected but it is a property particular to the chosen synthesis method, the exposed parameters, and their range. Our primary concern is therefore to compensate for unwanted changes in output amplitude, and the feedforward step will handle this. Feedforward into the output step must be handled case-by-case, and is discussed further in sections 3.1 and 3.2.

3 Output Models

A couple of cases are presented that present vastly different results according to the two categories of linear and non-linear synthesis.

3.1 Linear Synhesis: additive synthesis

We update parameters continuously and only output when the Event Pattern requests a new rhythm event.

Sound analysis/resynthesis by frequency components is not readily applicable to the PDV when no modification of the spectrum is taking place. The continuity of the resynthesis is broken, since its update does not happen in accordance with the timed receipt of the analysis data. The resynthesis becomes instead a 'snapshot' of the analyzed sound, defined by the delta time of the Event Pattern. This means that sound is not continuous any longer, and the relationship between the analysis and the resynthesis becomes vague, if not entirely obscure.

A more conventional mapping space comes into play when a PDV pattern is inserted between input signal analysis and spectrum modifications of its continuous resynthesis. The PDV determines the degree of modification in accordance with properties of the mapping space and the signal being analyzed.

An analysis-feedforward step in the time-scale <3 sec. between signal input and the model's output compensates for amplitude spikes outside legal values.

3.2 Non-linear Synthesis: signal modulation

Simple frequency modulation is used where carrier frequency, modulator frequency, and modulator index is subjected to the output of our model. Since the synthesis is algorithmically generated, the Distance Pattern has significant influence on the long-term timbre evolution. The range of timbre change is determined by the range of the exposed synthesis parameters but the Distance Pattern clearly creates discernible timbre events inside this timbre space. The shape parameters ramp-on and ramp-off influence the smoothness and distinctness of the events and the overall direction of the sequence becomes also a significant period identifier.

The direction taken by a sequence depends on the sum of PDVs so that if the sum is negative the timbre of the sequence moves in one direction, while a positive sum moves in the opposite direction. Direction is understood as directional tendency of the spectral centroid and a sequence will only move if the sum of the PDVs is non-zero. The sign is important in determining the overall direction of the sequence. A sequence consists of a sum of distance values, and if the sequence shouldn't move then the sum of the distance values should be zero.

The analysis-feedforward connection is inserted between the synthesis and output. This catches sudden energy increases due to some unforeseen combination of the synthesis parameters affected by our model.

4 Performance Use

The Event Patterns are flexible in performance because they can display recognizable pulse, irregular pulse, or no pulse. The random window function is capable of randomizing the Distance Patterns to the extend that the patterns are entirely hidden and only the random values are applied. Event Patterns can be aligned with other actions in the system, period or trigger based, making for easy integration. The model does not require much computing power and does not display any latency. The model generates autonomous patterns, but the patterns can be interferred with in performance without upsetting their immediate direction of development in an unnatural way.

5 Conclusions

The model holds certain promise because it is able to produce sonically coherent results that are straight-forward and intuitive to affect procedurally, manually, and interactively. The model produces very different results with different synthesis methods, yet retains its control variables. The pattern output can be restricted to take place in a desired sonic *ambitus*. The model can function on its own or in connection with a subsequent rhythm processing module based on the ADSR model. The computing power required is modest, the synthesis method and the quantity and resolution of its dimensions being the factors to take into account in the PDV nexus.

Some problems are inherent to the model's simultaneous affection of several synthesis parameters. As a result, unwanted changes in amplitude may come about. The feedforward step inserted attempts at solving this, but a more elegant way to handle abrupt changes should be found. This might point to a fine-tuning of the model for different synthesis methods, but this is against the original idea of universality of the model.

In the future the model will be applied to other synthesis methods in order to find its best design delimitations. It is already clear that the model will not be usefull to all synthesis methods, but it is also seems likely that such methods can be developed and transformed to suit the model's properties.

References

Cooper, G., Meyer, L. B., 1960. *The Rhythm Structure of Music*, ISBN 0-226-11522-4, Miami, USA.

Dobrian, C., 1995. Algorithmic Generation of Temporal Forms: Hierarchical Organization of Stasis and Transition. *Proceedings of the International Computer Music Conference 1995*. San Francisco, USA.

Lerdahl, F., Jackendoff, R. 1983. *A Generative Theory of Tonal Music*. The MIT Press, Cambridge, MA, USA.

Lyon, E., 1997. Rhythmic Rendering. *Proceedings of the International Computer Music Conference 1997*, pp. 485-486, Thessaloniki, Greece.

Morris, J., 2004. A Dynamic Model of Metric Rhythm in Electroacoustic Music. *Proceedings of the International Computer Music Conference 2004*, pp. 480-483, Miami, USA.

Schillinger, J., 1976. *Encyclopedia of Rhythms.* Da Capo Press, New York, USA.

Stockhausen, K-H., 1957. …wie die Zeit vergeht… *Der Reihe*, No. 3.