Music composition through Spectral Modeling Synthesis and Pure Data

Edgar Barroso
PHONOS Foundation
P. Circunval.lació 8 (UPF-Estacío França)
Barcelona, Spain, 08003
ebarroso@iua.upf.edu

Alfonso Pérez
MTG – Pompeu Fabra University
P. Circunval.lació 8 (UPF-Estacío França)
Barcelona, Spain, 08003
aperez@iua.upf.edu

Abstract

A major problem of the composition process involving music technology is the selection of the tools within the broader context of the ever-changing language of multimedia production.

The focus of this contribution will be on creative applications of the Spectral Modeling Synthesis (SMS) and Pure Data (PD) in two specific works: Searching your Synesthesia (2005) for flute, clarinet, cello, piano and live electronics and ODD (2006) an electro-acoustic multi channel 5.1 Surround piece.

This paper will provide a brief view on how this open source software offers the user the possibility to experiment with a vast number of compositional techniques in very flexible programming environments.

This view is based on the analysis of the research that has been carried out within the compositional course of action concerning several pieces using these systems. It is intended to share a particular framework for enabling interactions between specific tools for the creative compositional process.

Keywords

Composition, Pure Data, Audio tools, Interaction, Spectral Synthesis.

1 Introduction

During the last recent years there has been a huge proliferation of music software designed to build interactive music systems and audio signal processing. The rapid development in personal computers and the increasing number of audio application users, have situated composers into endless aesthetics, and ethical crossroads, that could result overwhelming and perhaps discouraging. This is particularly true for those initiating composers in computer-generated music composition. Even though experimentation is a fundamental part of almost every composition process, time spent familiarizing with endless different softwares could be enormous, and will take away precious time to deal with the core of the creative process.

In this contribution we present two compositions that deal with interactive systems and spectral modeling synthesis and make use of open source software, in order to show specific practical examples of the usage of those technologies. The aim is to facilitate and quicken the user decision on whether or not this tools have meaningful applications for their own compositional necessities. In the first composition "Searching your Synesthesia" the composer makes use of the graphical programming language Pure Data (PD) developed by Miller Puckette in the 1990s for the creation of interactive computer music and multimedia works. PD is an open source project and has a large developer base working on new extensions to the program. In the second composition, ODD, is inspired in SMS developed by Xavier Serra [1] which is a set of techniques and software implementations for the analysis, transformation and synthesis of musical sounds. These techniques can be used for synthesis, processing and coding applications, while some of the intermediate results might also be applied to other music related problems, such as sound source separation, musical acoustics, music perception, or performance analysis. Two open source implementations of SMS are cited, for Octave and CLAM frameworks.

2 Searching your Synesthesia – Musical Control and Live Performance

This is a piece for flute, clarinet, cello, piano and live electronics, first performed in the Paine Hall of the Harvard University Department of Music on March of 2006. Composer selected this piece to explain a succinct overview of the software he programmed in Pure Data, to show aspects of its flexibility and some of the features

and possibilities that can be achieved with this programming environment. The only intention of this paper is to share a specific thinking framework within a complete performed work, any aesthetic or technical statement must be interpreted only as a subjective opinions of the authors.

2.1 The Patch

All the programming for the performing of the piece was made in Pure Data version 0.40-2 downloaded from the Miller Puckette [2] web page. It uses only two external objects, the first named *counter* and the second one is the *freeverb* object written by Olaf Matthes [3].

The patch contains the following features:

- Recording Machine: allows to record and store in tables and/or hard disk up to 128 audio files. There is a possibility to do it manually or automatically. It also has the capability to load and send this files to any of the sample triggers modules located in the interface.
- Multi Triggering Sampler: Permits to trigger up to eight different simultaneous audio files, each one of them has an independent pitch shift, amplitude, volume, panning, and duration parameters.
- Loop Sampler Triggering: triggers in loop up to 16 voices of the same file, each one of them will have individual controls for the panning, pitch shifting, volume, duration of the loop, starting reading point, and number of voices.
- Microphones / Mixer: Controls all the microphones input and output gains.
- Audio Effects: Resonator-Multi tap Delays and three different reverbs.
- Score Follower: Order the sequence of the events in time during the piece, it uses a MIDI pedal switch to advance to the next musical event, all the rest of the parameters are automated.
- Scratcher: Records in real time and store the recordings in tables, that can be later retrieved in different positions, repetitions and speed .
- Pitch Follower Device: It works with filtered oscillators that can follow or make aleatory counterpoint and harmonization of the incoming signal.
- Amplitude contour device: It uses an amplitude threshold to decide a series of conditions and constrains.
- MIDI: A module that sends midi notes to a midi synthesizer or to a controller. You can activate or de-activate a continuous stream of midi notes by detecting the input gain of the microphone.
- Quadraphonic & Stereo diffusion system: can diffuse every module and sampler trigger individually up to four individual speakers.

2.1.1 The Interface

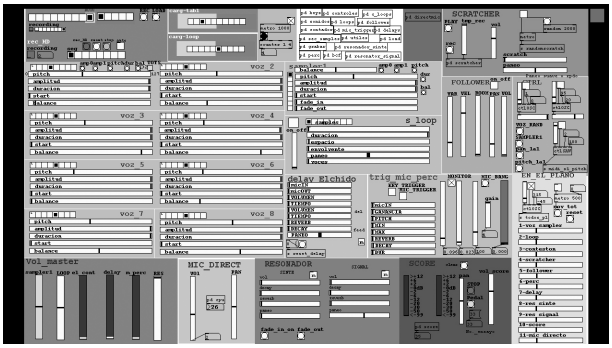

Figure 1: Searching your Synesthesia Pure Data Interface

In this particular work, the attainment of the interface was much more oriented to be an aesthetic platform defining musical structures and a practical controller of the sound engine, in opposition to the idea of conceiving the interface as an instrument itself as in [4]. It was built with a series of modules that work independently and provide the performer the necessary data to control the complete piece only with the use of a MIDI switch pedal (which is controlled by the conductor) which sends commands to the score follower (see figure 1).

One of the goals in the designing process of this interface was the riddance of any direct human control, (except for the MIDI pedal) therefore, the complete piece is built with automated parameters. All the rest of the real time processes are carried out by the computer itself in all senses. One particular challenge of the research was to avoid the overly mechanical sounding, typical of computers that produce every time the exact same parametric control quantities. In the presented patch the computer controllers were implemented trying to simulate (at some level) the human error with the use of random parameters, and inconstant sliders movements to achieve gestural

expressiveness and versatility in the outcome audio results.

Another important aspect that the composer tried to achieve in this piece, was the experimentation with dynamically changing sound processing, were one or more morphological characteristics of the input sound are analyzed and immediately used to control one or more aspects of treatment [5]. The fact that sound itself controls its own treatment was implemented in the programming, although only at specific rather short parts of the sore.

2.1.2 The Score Follower

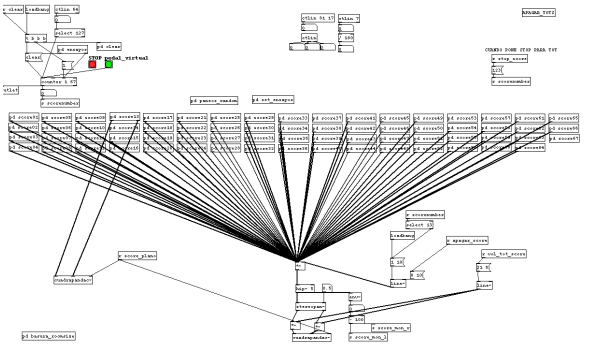

Figure 2: Searching your Synesthesia Pure Data Score Follower

The core of the program is the score follower. It is a work-in-progress device which allows you to divide the score with several cues, and create a single patch for each one of them. The advantage of this system is the economization of processing resources of the computer. The composer created sub-patches (see figure 2), each one having a totally independent self-contained patch that communicates with the interface. These events are always following the triggers coming form the MIDI pedal, using a counter to define the position in time during the performance. Inside this boxes all audio output parameters are controlled. This method was develop since it was much more easy to follow short events rather than long periods of time in which the liability of the analysis could get inaccurate, specially due to the relationship among the measured loudness and other external conditions. Even though the system is divided in this boxes, it allows you to make transitions between particular events in different ways, they can be done in a sudden movement, smoothly or even mixing two or three events simultaneously. For the score following the composer used 4 features that can be recognized: 1) MIDI switch pedal controlled by the conductor or the computer performer. 2) Tone events (notes of the score), 3) Phrase events, 4) Amplitude contour.

2.1.3 Synchronizing vs Chaotic Interaction

In the early stages of the creative process of this piece, the aim was to take advantage of the flexibility of the software and try to combine these tools in order to include the following ideas.

- Elaborate a system in which the interaction could be as clear as possible, however preventing from being too obvious (action - reaction).

- Elaborate a system that generates some indeterminacy and is able to control computer composition that would help performers to add tension and coherence to the musical ideas, yet only for specific parts of the piece.

Some of the strategies that the composer followed to achieve these two goals was to simulate the interaction of the classical period chamber music, where the interaction between the musicians is constant and very clear, however, the idea of variation is constantly present in the music. The composer particularly studied the string quartets from L. V. Beethoven Op. 59 and 132 and tried to apply the same way of thinking about interactivity as he did with the electronics and the instrumentalists.

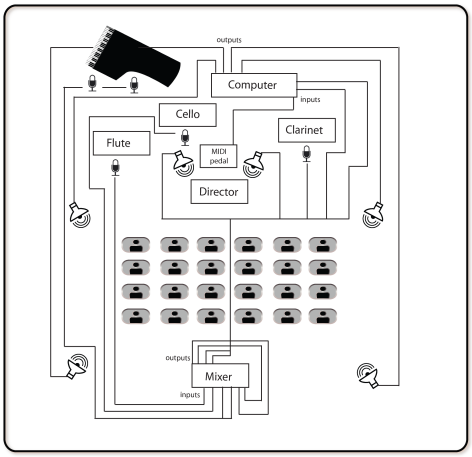

Figure 3: Searching your Synesthesia performing technique diagram.

Another issue concerning the composition of the piece and the use of the synchrony–chaos idea was the diffusion of the total outcoming audio of the ensemble. The first performance took place at the Paine Hall of the Harvard University, and was diffused with the loudspeaker orchestra *Hydra* [6] which has 24 speakers. For this performance,

special arrangements were made to the patch in order to get advantage of the number of speakers. Withal, it was necessary that an additional performer controlled the diffusion system in real time. In normal conditions, the piece uses a four channel configuration (see figure 3). As previously mentioned, every individual sample trigger and module of the interface can have two panning options: left - right and front - back. The aim of using this system is that it allows to make transitions very rapidly and effectively, from totally chaotic individual panning to very subtle panning remarks. Again the same principal: chaos and synchronization in the same piece.

2.1.4 Indeterminacy and Pure Data flexibility.

The possibilities to create an open work with the use of pure data are endless. The composer also includes in the design of the software a few indeterminacy elements. He lefts some composition and structure decisions to the computer using the parameters of sound itself, and some probability results from the analysis of pitch and amplitude that the program re-uses, for instance, to decide when to start or finish producing sound or determine the pitch and location of a prerecorded samples among others. Even though these considerations, there are in fact (few) specific moments of this indeterminacy, the composer considered the compositional structures to be linear, and with sections of music ordered sequentially as in a traditional score. From the composer point of view the piece leaves no room for improvisation coming from the musicians, it was a better idea to leave all indeterminate actions only to the computer. As the composer is not a programmer, the capacity to adapt the system to a new environment and resilience in recovering from a crash or from a human error (MIDI switch pedal) was also a very important and not at all an easy task. Nevertheless Pure Data proved to be a robust, liable environment in which the flexibility can be determined by your imagination.

3 ODD – Composition with Spectral Modelling Synthesis

ODD was conceived based on the SMS technology. SMS is a set of techniques and software implementations for the analysis, transformation and synthesis of musical sounds. The aim of this work is to get general and musically meaningful sound representations (specifically for harmonic sounds) based on analysis, from which musical parameters might be manipulated while maintaining high quality sound. There are two main open source implementations of SMS, one is based on Octave (code can be found at [8]), and the other one on CLAM [7].
As is shown in ODD, this technique can also be used as a "**creative"** tool. (Although there is no single, authoritative perspective or definition of creativity. Most experts agree that is a mental process involving the generation of new ideas or concepts, or new **associations** between existing ideas or concepts.) The composer makes use of the SMS technology dealing with sound morphing, transformations, separation of residual-noise components of sound objects and also is inspired in the graphical representation of analysis data that provokes and stimulates musical concept associations as if it was a score.

3.1 SMS Overview

During analysis, sound is divided into temporal frames that are spectrally analyzed. At each frame the signal is separated into harmonic content and residual or noise. The harmonics of a frame are matched with the ones in the next frame, giving rise to tracks (usually called sinusoidal tracks) that indicate the temporal evolution of each harmonic. This way we can represent any sound as a parametric harmonic part plus a noise residual, and we can get back to the original waveform representation by the resynthesis process.

The parametric representation of the harmonic part of the sound is very useful in order to make sound transformations. If any parameter is modified, the re synthesized sound will be different. SMS lets the composer to experiment with the analysis parameters can produce very interesting and unexpected results.

3.2 SMS Transformations

Transformations are another important process that allows you to have a way of making "variations" of the same sound, from very subtle changes to totally unrecognizable sound source.

Typical transformations include pitch shift, time stretch, timber manipulations, etc. and a special type of transformations called morphing that consists of a cross-synthesis between two sounds, resulting in a new one that has hybrid characteristics. In this process it is quite interesting to provoke ambiguity of identity of the sound source, by approaching sounds in their inner spectral content. Among the endless possibilities of transformation we can emphasize morphing

between instruments and voice, or between "natural" sounds and synthetic ones, the so called "hybrid sounds".

In am more semantic or conceptual context, transformation are defined as the genetic alteration of a cell sound resulting from the introduction of foreign DNA. In music, transformation refers to any operation or process that a composer or performer may apply to a musical variable. The transformation concept of the SMS tools reminded me the so called "Variations" which is a formal musical technique where material is altered during repetition; reiteration with changes. The internal structure of "ODD " is constructed thinking in variations of all sounds, to keep the piece in constant movement during time.
Also the concept of Morphing lead us to the concept of curiosity, which by definition is the unknown result of combining two more more different elements or in this case sounds. It is a very useful tool to generate complex unique sounds. Many experts agree that curiosity is any natural inquisitive behavior. and is the emotional aspect of living beings that engenders exploration, investigation and learning.

3.2 The residual component

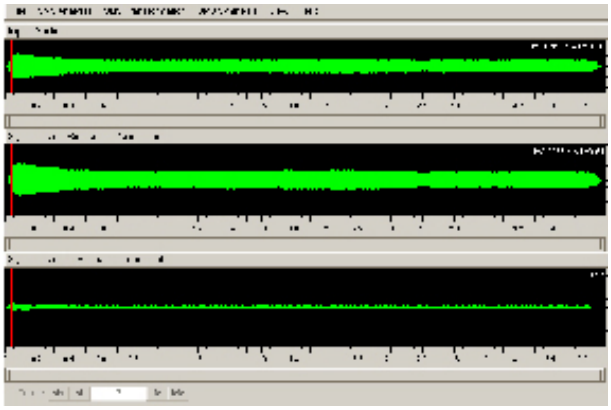

Figure 4. Residual Graphic Representation

Apart from the transformations above mentioned, the possibility of isolating the residual component of the sound is fascinating for creative aims. As an example, we can extract the noise part of an instrumental sound or the non-pitched element of a vocal sample. In figure 4 we can see at the top the original sound , in the middle the resynthesized sound and the bottom the residual component. The composer used the concept of residual , understanding the residual as the quantity left over at the end of a process; a remainder. The composer found that recycling this sounds, in other words to put or pass through a cycle again, as for further treatment would be a interesting idea. This thought became the core of the composition, the subjective analogy of extracting useful materials from otherwise unwanted frequencies, and make a complete composition structure with it..

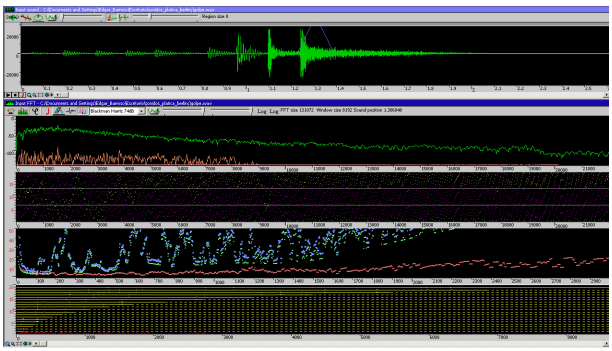

Figure 5. Graphical Representation of Analysis parameters.

3.3 Graphical Representation of Analysis parameters as a score generator

3.0.1 Evolution in time of SMS analysis data can be easily visualized. Among this data we can mention the FFT, sinusoidal tracks, residual part, pitch, sine amplitudes, etc. (see Figure 5)

Non-traditional music notation has always looked for inspirational ideas on graphic structures. The graphical representation of the analysis parameters can be seen as such.

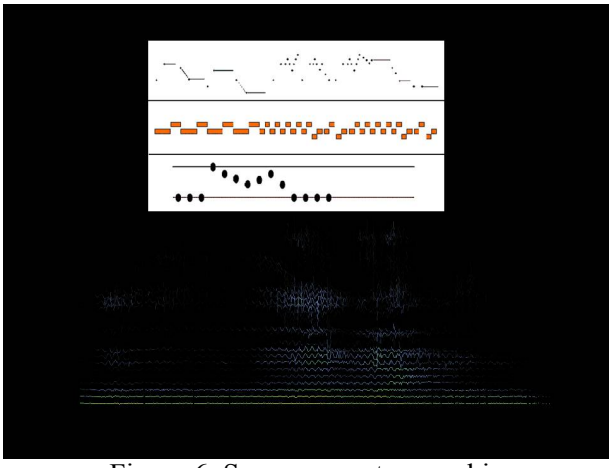

Figure 6. Score generator graphic

Graphical representation of some parameters resulting from the analysis of sounds created for the piece, is used as feed-back by the composer to determine the upcoming musical ideas (see figure 6). The word "fractal" in colloquial usage, denotes a shape that is recursively constructed or self-similar, that is, a shape that appears similar at all

scales of magnification and is therefore often referred to as infinitely complex.

4 Conclusion

We had introduced the creative applications of the Spectral Modeling Synthesis (SMS) and Pure Data (PD) in and its use for music composition. We have shown that the open source software SMS and PD could be very well suited to the design and development of music compositions. Future work will involve increasing the flexibility of the interactive system and extending the experimentation with SMS to obtain new processed sounds.

5 Acknowledgements

The authors would like to thank to Hans Tutschku, Matteo Sistisette, Ricard Marxer, Berio Molina and Günter Geiger for the advise in the programming and musical aspects of the pieces. We also thank Folkmar Hein, Stefan Kersten, Miguel Álvarez-Fernández and the Inventionen Festival, the MTG and the PHONOS Foundation for providing the infrastructure for developing these two pieces.

References

[1] Xavier Serra. "Musical Sound Modelling With Sinusoids Plus Noise". Published in C. Roads, S. Pope, A Picialli, G. De Poli, editors. 1997. "Musical Signal processing", Swets&Zeitlnger Pubishers.

[2] Miller Puckette. http://crca.ucsd.edu/~msp/software.html 2006.

[3] Schroeder/Moorer reverb model implementation, version 0.2c written by Olaf Matthes, based on Freeverb by Jezar. <olaf.matthes@gmx.de>

[4] Thor Magnusson. "ixi software: The Interface as Instrument". Proceedings of the 2005 International Conference on New Interfaces for Musical Expression (NIME05), Vancouver, BC, Canada.

[5] Hans Tutschku. "Portfolio of Compositions" Department of Music School of Humanities. The University of Birmingham March 2003

[6] Harvard University Studio for Electroacoustic Composition. http://huseac.fas.harvard.edu/pages/05hydra/1hydra.html 2007

[7] CLAM Framework. http://clam.iua.upf.edu/

[8] Octave code for SMS Framework. http://www.iua.upf.es/mtg/sms/