

### Conclusão

O *Chaosynth* é um sintetizador cujos parâmetros de síntese são controlados por um autômato celular (AC) chamado ChaOs. A técnica de síntese do *Chaosynth* se assemelha muito com a técnica de síntese granular, pois ela produz uma grande quantidade de partículas sônicas simples, que no total formam eventos sonoros mais complexos. Pelo menos duas características diferem o *Chaosynth* dos sistemas tradicionais de síntese granular: (a) o fato de que eu utilizo o ChaOs para controlar a produção de partículas sônicas e (b) cada partícula é o produto de uma síntese aditiva de várias ondas senóides, enquanto que nos sistemas tradicionais cada partícula é composta por apenas uma senóide.

A versão atual do sistema utiliza apenas um método para mapear o comportamento do ChaOs com os parâmetros de síntese. Em futuras versões eu pretendo deixar esse mapeamento em aberto, de tal forma que o musicista possa inventar e experimentar outros métodos.

No momento eu estou investigando a possibilidade de paralelizar o módulo de síntese sonora, com o objetivo de minimizar ao máximo o tempo necessário para computar as amostragens digitais (ou *samples*) para produção do som. Assim o *Chaosynth* também poderá funcionar como uma espécie de instrumento musical, tocado por intermédio de controladores apropriados; como por exemplo a *luva interativa* que está sendo desenvolvida na Unicamp, pela equipe do Jônatas Manzolli, cujos resultados de pesquisa também estão sendo apresentados no II SBC&M. A longo prazo, eu pretendo unir o *Chaosynth* ao sistema CAMUS (também de minha autoria). O sistema CAMUS utiliza os autômatos celulares para controlar níveis mais altos de composição; por exemplo, o nível formal da composição de fragmentos musicais (Miranda, 1993; 1994).

O *Chaosynth* foi implementado em um supercomputador CM-200, no Centro de Computação Paralela de Edimburgo (EPCC), na Escócia, com o auxílio técnico de Martin Westhead and Robert Fletcher.

### Referências Bibliográficas

- Carpenter, R.H.S. (1990), *Neurophysiology*, Edward Arnold, GB.
- Cood, E.F. (1968), *Cellular Automata*, Academic Press, GB.
- Brown, S. (1994), *Parallellising Grid-Based Applications with PUL-RD*, relatório técnico, Edinburgh Parallel Computing Centre, GB.
- Dewdney, A.K. (1989), "A cellular universe of debris, droplets, defects and demons", *Scientific American*, August:88-91.
- Dodge, C. and Jerse, T. (1985), *Computer Music - Synthesis, Composition and Performance*, Schirmer Books, UK.
- Eccles, J.C. (1958), *The Physiology of Imagination*, WH Freeman & Co (reimpressão de artigo publicado na revista *Scientific American*), EUA.
- Garbor, D. (1947), "Acoustical Quanta and the Theory of Hearing", *Nature*, 159(1044):591-594.
- Hamman, M. (1991), "Mapping Complex Systems using Granular Synthesis", *Montreal ICMC 1991 Proceedings*, 475-478.
- Millen, D. (1990), "Cellular Automata Music", *Glasgow ICMC Proceedings*, 314-316.
- Miranda, E.R., Nelson, P. and Smaill, A. (1992), "ChaOs: a Model for Granular Synthesis using Cellular Automata", *Annual Report & Project Directory 1991-92*, Edinburgh Parallel Computing Centre, 153-156, GB.
- Miranda, E.R. (1993), "Cellular Automata Music: An Interdisciplinary Project", *Interface*, 22(1):3-21.
- Miranda, E.R. (1994), "Cellular Automata Music Composition", *Languages of Design*, 1994(2):105-107.
- Orton, R., Hunt, A., and Kirk, R. (1991), "Graphical Control of Granular Synthesis using a Cellular Automata and the Freehand Program", *Montreal ICMC 1991 Proceedings*, 416-418.
- Roads, C. (1991), "Asynchronous Granular Synthesis", *Representation of Music Signals*, De Poly, G. et al (editores), MIT Press.
- Truax, B. (1988), "Real Time Granular Synthesis with a DSP Computer", *Computer Music Journal*, 12(2):14-26.
- Vercoe, B. (1991), *Csound Manual*, disponível via Internet ftp no diretório *music* da máquina *media-lab.mit.edu*.
- Westhead, M.D. *Chaosynth: a System for Granular Synthesis using a Cellular Automaton*, EPCC Summer Course Project Report, Edinburgh University, GB.
- Wiener, N. (1964), "Spatial-Temporal Continuity, Quantum Theory and Music", *The Concepts of Space and Time*, Capek, M. (editor), Reidel.
- Wilson, G. (1988), "The life and times of cellular automata", *News Scientist*, October(8):44-47.
- Wolfman, S. (1983), "Statistical mechanics of cellular automata", *Reviews of Modern Physics*, 55(3):601-623.

### One Tool, Two Programs and Several Ideas for Composition with Spectral-Modelling Synthesis

CELSO AGUIAR

Center for Computer Research in Music and Acoustics (CCRMA)  
Department of Music, Stanford University  
Stanford, California 94305-8180 USA  
aguiar@ccrma.stanford.edu

#### Abstract

Sound can be approached from complementary perspectives. This paper describes aspects of one of my works, 'Piece of Mind' for tape, where time domain techniques derived from 'musique concrète' are blended to powerful spectral modelling (SMS) in the processing of natural sounds. Programs written in collaboration with Xavier Serra have made SMS available in CLM's synthesis environment. Deterministic components are synthesized by a highly efficient IFFT algorithm that relieves the delay of pleasure composers have historically accumulated with additive synthesis. Bell sounds and singing voice are hybridized or cross-faded to provoke ambiguity of identity. On the other hand, the way SMS handles its stochastic part exposes the fact that dsp guys and composers may not always speak the same language. Being aware of what synthesis techniques are geared to and the ability to present proper solutions are first requirements. A graphical interface to display SMS information on a single view is introduced.

#### SOME PREMISES

A prediction was once made by researcher Julius O. Smith III where he affirmed that time domain techniques for sound synthesis and transformation would migrate, or be absorbed in the future, into frequency domain ones (Smith, 1991). By time domain techniques he meant all synthesis techniques derived from 'musique concrète' like sampling and granular synthesis. Under frequency domain he listed several techniques including spectral modelling synthesis (SMS) and inverse Fourier synthesis.

His prediction was made about four years ago and considered the fact that sampling and granular techniques can add very interesting new colors to the sonic palette, but they are very difficult to control. To obtain control over these techniques, more general sound transformations would be required. As these transformations are to be understood in terms of what we hear, and the best way to understand what we hear is through the short-time spectrum it seems logical to assume that time domain techniques point towards spectral modelling.

The word 'control' and its meaning in his text may stir polemic if directly transposed to a compositional context. Composers, including myself, may want to deal with the uncontrollable, and that stands very far from avoiding control over materials. However, I still think his idea is interesting enough to deserve some discussion and consideration. In one side we have time domain techniques that are straight-forward to implement but which impose barriers to effectively manipulate sounds. On the other hand we have very powerful analysis-synthesis techniques which allow us to modify sound in very effective ways but whose implementation is not as immediate as in the previous case and which involve much more computation than the mere reading of files or of samples of recorded samples.

For the composer the amount of delayed pleasure associated with additive synthesis, for example, is such

that few of us will risk to write a piece depending entirely upon it. It's a very well known fact that the computational cost of sum-of-sinuosides directly increases with the number of partials in a sound. This limitation in part explains what has been going on recently with composers who rely in some way on digital synthesis to produce their musical works. Tired of the predictability of sounds obtained through abstract methods of synthesis they feel quite comfortable and relieved when turning on to time domain methods in the hope that the sonic aspect of their piece will retain the complexity of the original sound sources. These methods have derived mostly from the renewed computer practice of concrete procedures but the point is: who can blame them?

On the other hand physical modelling still seems to be quite in its infancy and hasn't yet been able to clearly fulfill the promise of providing controllable instruments with a small number of simple and intuitive parameters. Besides that fact, physical modelling seems also a lot geared towards performance, an aspect of computer music which shouldn't be discarded but whose full potential is only obtained through the use of controller devices interfacing a particular algorithm to human performers. Just like playing an old natural instrument again, Science seems to be trying to reproduce nature in a box.

<cl> (EQUALP FANCIER\_TOOLS BETTER\_MUSIC)  
NIL

My basic drive when composing 'Piece of Mind' was to write a piece where a strong sense of movement and direction would be present. One who is not used to composing tape music on non real-time systems may still easily perceive how hard it is to keep up that urge.

Besides the technical difficulties and constraints that computer music in general normally imposes on the composer, tape music in particular is a medium which also inflicts its own hardships upon the act of composition. The necessity of maintaining attention focused on the large scale aspects of a composition is severely damaged by the repeated hearing of sounds and by the rate at which these sounds are being created by a certain algorithm. On the one hand they naturally have to be examined in detail to verify if our initial setting of parameters corresponds to what we actually hear, but on the other, this repeated hearing quickly wears out the impact this sound would have to suggest new associations on our imagination. The most precious gift in tape music, being able to previously hear while planning what a piece will actually sound like during the concert, can easily become our most dangerous weakness.

This process of give and take, if not economically carried out, will make it very easy to loose track of our initial intentions and will make us zoom into details which don't really matter for the overall picture and will easily be missed by the audience (Smalley, 1986). A minimally clear plan has to be previously set up. The nature of this plan may differ from composer to composer in the idiosyncrasies their creative process may have inherited or acquired in the use of the medium but it still has to be there if a minimum of success is desired from the result.

From another perspective, complexity can exert a great deal of appeal over all of us. It's quite easy to associate fancier tools with better results in our minds. Composers in particular tend to think that the more advanced tools will automatically generate better music. I've been finding this piece of thought to be one of the most insidious lies propagated nowadays in computer music composition. Despite the fact that we have to be open to the new, it's easy to find this overabundance of systems and tools to be hiding a deficiency in creativity if not a resistance to the true demands for the new in art, exceptions made.

The idea of approaching sound from two different and complementary angles should allow for a balance in the demands to keep the focus of attention and a minimum degree of complexity in the sonic result. Particularly for this piece I wanted to be able to either avoid or fulfill expectations by the simple interplay of textures juxtaposed in time. Movement of sound in space was not really as important as being able to transmit an impression of imminent instability, an impression that things are about to happen. Higher portions of the available frequencies should be explored as well, and they are easy to forget if one is not conscious of the real difference they can make in the overall result of a piece.

The use of voice and natural sound sources would provide for a better communication with audience but above all I wanted to explore the consequences of making clearly audible transformations in or between them. What we may be avide to find is a musical discourse where it would be possible to incorporate complexity and yet not be complicated.

## AN ALGORITHM

Xavier Serra was telling me of all these new spectral modelling researches that had been appearing after his famous SMS thesis. Of how spectral modelling (based on sound as it arrives our ears) will in the near future look much like the physical modelling approach (based on how the sound is actually produced at its very source) in trying to recreate sounds from natural instruments and in general. One of this improvements was an IFFT algorithm for additive synthesis developed by Xavier Rodet. A student of Serra in Spain had been working on it without positive results yet. I decided that I should set out to reconstruct it even if from very scarce bibliography (Rodet, 1992).

The main idea behind the IFFT algorithm is that, instead of performing the old time-consuming method of sum-of-sinuosides, we use the Fast Fourier Transform for the conversion of domains involved in additive synthesis. In order to do that some assumptions have to be previously made. We start by constructing a frame of our spectra in the frequency domain. For this frame of spectra we have to assume that our partials have been imposed some kind of windowing in order to reduce the spectral splatter along the bins of our FFT. The Blackman window seems as a natural candidate to achieve this. It concentrates most of the energy of a partial in a total of nine bins if we consider a FFT size of 256 points. In the Blackman window all other bins remain at least at a comfortable 200 Db below the peak amplitude of our partial (Fig 1).

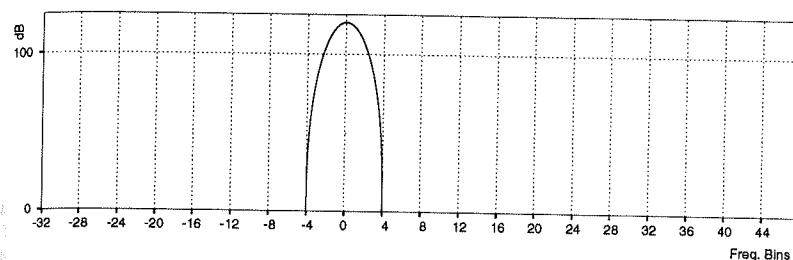


Fig. 1 - The 9 most significant bins in a frequency domain Blackman window (256 FFT size).

In our implementation of the program we decided that a resolution of 288 points for in-between bins frequencies would be enough. Higher resolutions can be easily implemented. The function displayed in Fig2 is used as a table for sampling the partials when constructing our frequency domain frame of spectra.

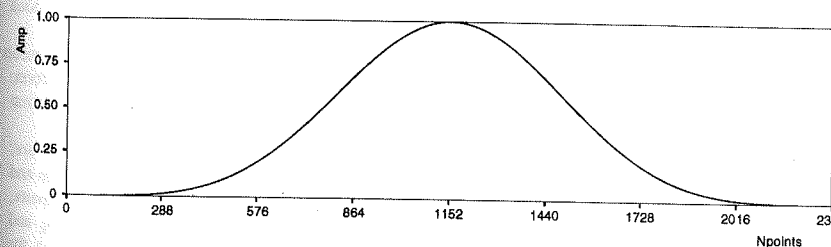


Fig. 2 - Frequency domain Blackman (undB'd version of above) and resolution in our IFFT CLM program.

The resulting frequency of a partial obtained after we perform the inverse FFT depends on the position we start sampling the 9 bins within the table. It's also easy to notice that in reality we have only 8 bins and not 9 as

one of them will always falls off the limits of our table look-up process. For smaller partial amplitudes this number can be further reduced due to the loss of significance in the least central bins.

When deciding the size of the FFT we have to consider the fact that it has to be small enough to avoid perceptual problems. A 256 points FFT at 44.1 kHz seems like a good choice. It is long enough to permit a fine frequency resolution and is short enough to avoid our perception of the overlap-add frame rate. We also have to keep track of the phase information for each of our partials in the spectra when performing the overlap-add portion of the algorithm since they're tied to the frequency information.

We then perform the inverse Fourier transform. Once in the time domain, we now can divide our resulting waveform by a time domain Blackman window in order to compensate for our initial assumption that our spectra was windowed. To perform a 50% overlap-add in the short-time spectra we also multiply our waveform by a triangular window precisely splicing the waveforms together. The two steps described above can be collapsed into an one multiplication operation. The necessary window is displayed in Fig. 3.

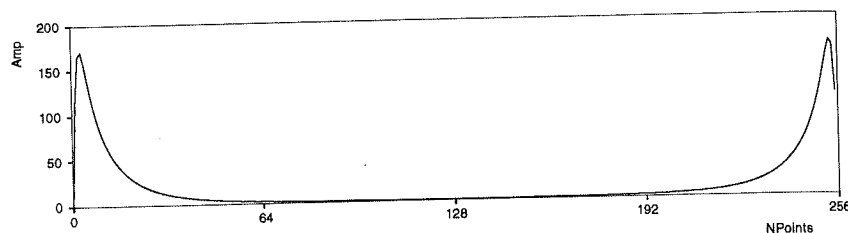


Fig. 3 - Time domain Triangular/Blackman windowing.

One problem that we had to face when writing the algorithm regards to what happens when we have partials sounding below 344.53 Hz, or the 4th bin in our 256 point FFT. The lower bins will soon fall off the limit of our DC bin component and the balance in our Blackman windowed spectra will be lost. At this point we just assumed that the negative ones would reflect into the positive domain with a change of sign, or suffer an inversion of phase, just like in the case of FM synthesis (Chowning, 1973). The same can be demonstrated for the Nyquist frequency case.

Another interesting point was that Xavier Serra believed that we would be able to add the SMS stochastic data into the spectra of partials (deterministic data) before performing the inverse FFT in order to resynthesize them altogether. We found this not to be possible. As we are assuming a Blackman windowed spectra in the input to the IFFT it's easy to notice what will happen when we divide by a Blackman window once in the time domain. SMS' stochastic data are made out of piece-wise linear segments that approximate noise bands of the spectra in the frequency domain, lending themselves difficult to window compared to what could easily be done in the case of partials.

The operation of the CLM instrument built from this algorithm to resynthesize deterministic components requires the memory storage of the following look-up tables: **wave**, for our time domain output waveform; **window**, the function in Fig. 3; **blackman**, the function in Fig. 2; **sine**, **cosine**, for phase interpolation calculations.

The parameters which can be controlled in this basic deterministic resynthesis instrument (IFFTdet) are: **beg**, **dur**, **amp** and **freq**, common parameters for time, duration, amplitude and pitch of a note; **file-amp** and **file-freq**, two SMS format files containing the information from the partials of analyzed sounds; **ampenv**, **freqenv** and **sclev**, envelopes for amplitude, frequency and scaling of the spectra of a sound; **recFrq** and **recAmp**, envelopes for record positioning inside an SMS file; **spectEnv**, for imposing an envelope on the magnitudes of a spectra; **partialAmps** and **partialFrqs**, lists with partial numbers for recombining their structure; **coord** and **rev-amount**, for reverberation and spatial localization.

Our implementation of the IFFT algorithm has achieved a very good level of performance when resynthesizing SMS deterministic data. In CLM, by nature a non real-time environment, our specs for 22 kHz, 30 partials sound using a complex IFFT have been 3:1 on a NeXT slab and .7:1 on a Pentium 60 MHz.

## EXPLOSION OF HYBRIDIZED SOUNDS

Hybridization and cross-fade of sounds are only two of the new possibilities opened by approaching sounds in their inner spectral content (Serra, 1994). They provide us with objective ways to play with the identity of natural sounds. The power they have in suggesting new ambiguous associations is at the root of electroacoustic music thought.

Hybridization and cross-fade of sounds have already been tried with other synthesis methods and systems. I believe our implementation for sound hybridization and cross-fade in CLM to be able to render very clear and precise results, especially regarding intelligibility of articulated voice sounds (Harvey, 1986).

Fig. 4 below displays the spectrogram of a cross-fade from a complex type of bell sound to a segment of singing voice. It's possible to notice how the many partials in the complex bell sound are actually merging together and transforming into singing voice with its more transparent nature.

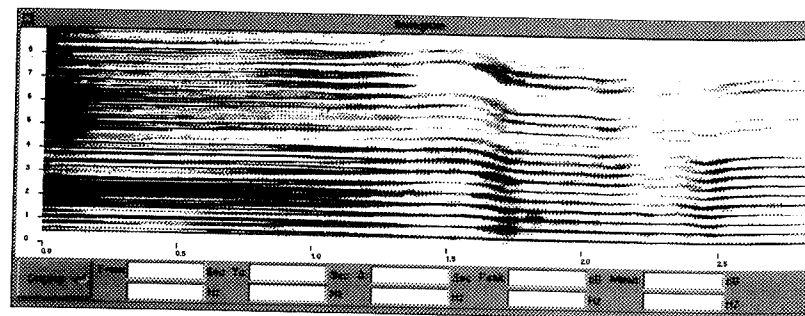


Fig. 4 - Cross-faded sound (bell -> voice) showing interpolation of partials.

The following example (Fig. 5) explores the opposite case where departing from a more simple and transparent bell sound we arrive at the same singing voice segment in the previous example. In this case the partials have to bifurcate in order to merge the partials of the subsequent sound.

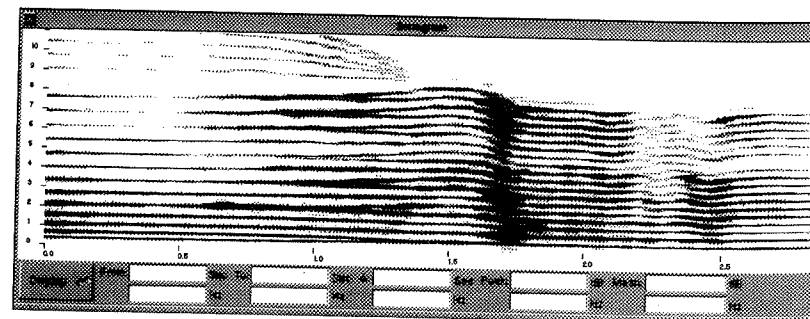


Fig. 5 - Cross-faded sound (another bell -> voice) showing contrary merging and interpolation of partials.

A program was written to perform a match in the partials of the two sound identities to be cross-faded or hybridized. These sound identities have to be previously analyzed in their short-time spectral content with the tools in SMS. Their resulting files are then used as input to that routine which by its turn performs the match or comparison in the proximity of the two groups of partials depending on the points between which we want them

to be cross-faded. Output lists inform which partials in the departure sound should then be interpolated to what other partials in the target sound. The process above should be able to work for any natural sound whose spectral content can be translated into deterministic components.

Besides the common standard parameters for duration, amplitude and frequency, we have this additional set of parameters to control in our IFFTFade instrument: **file-amp1**, **file-frq1**, **file-amp2**, **file-frq2**, four SMS format files containing frequency and magnitude of the sounds to be cross-faded; **ampenv**, **freqenv1** and **freqenv2**, envelopes for amplitude and fundamental frequencies in each sound; **recFrq1** and **recFrq2**, envelopes for frequency record positioning inside an SMS file; **recAmp1** and **recAmp2**, envelopes for amplitude record positioning inside an SMS file; **partials1**, **amps1**, **partials2**, **amps2**, output lists from the matching program to control partial interpolation; **spectEnv1**, **spectEnv2**, for imposing envelopes on the magnitudes of each spectra; **scl1** and **scl2** scaling parameters to be imposed on the spectra of the sounds; **iFrqenv**, **iFrq**, **iFrbase**, **iAmpenv**, **iAmp** and **iAmbase**, parameters to control the sections to be cross-faded in the sounds; **coord** and **rev-amount**, for reverberation and spatial localization.

Improvements can be made and new possibilities can be open within SMS yet. Some of them point in the direction of being able to analyze and transform larger portions of sound files. There's a clear necessity for more intelligence from the part of the analysis programs provided by SMS. The time consumed in the choice of the right parameters to perform the analysis is still big compared with the time spent in the analysis itself. This things can and should be made automatic.

The idea of sounds decomposed in deterministic and stochastic parts is very interesting but the rendering of the stochastic part seems still unsatisfactory, at least to me. In the process of composing the piece realized that Xavier had meant it to be geared towards data compression. A new residual sound file in the size of the original sound is generated during the decomposition process of SMS. I could achieve more precise results just by phase-vocoding this residual sounds and then mixing them to the deterministic part instead of applying the stochastic model.

One idea that can be implemented is to apply graphically based processes to SMS files considering its data to be a huge matrix that can be transformed in different ways. This idea has been suggested while building a new interface for displaying SMS data with the *RenderMan* package of graphical rendering routines. The nice thing about this interface is that all SMS data can be displayed on a single view. The picture can then be moved around via mouse and be viewed from different angles during the process of analyzing and transforming sound. This same picture could then be viewed as a surface object that could suffer whatever modifications graphic surfaces can undergo.

## REFERENCES

- Chowning, J. (1973). "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation." *J. Audio Eng. Soc.*, vol. 21, no. 7, pp. 526-34.
- Harvey, J. (1986). "The Mirror of Ambiguity." In *The Language of Electroacoustic Music*, edited by Simon Emmerson. Macmillan: London, pp. 175-190.
- Smalley, D. (1986). "Spectro-Morphology and Structuring Processes." In *The Language of Electroacoustic Music*, edited by Simon Emmerson. Macmillan: London, pp. 61-93.
- Smith, J. O. (1991). "ViewPoints on the History of Digital Synthesis." Keynote Paper, *Proceedings of the International Computer Music Conference*.
- Rodet, X. and P. Depalle. (1992). "Spectral Envelopes and Inverse FFT Synthesis." 93rd Convention of the Audio Engineering Society, San Francisco.
- Serra, X. (1994). "Sound Hybridization Based on a Deterministic plus Stochastic Decomposition Model." *Proceedings of the International Computer Music Conference*.

## ACKNOWLEDGMENTS

My thanks go first of all to composer David Soley from Stanford University who allowed me to have his ears as my third and fourth ears, in my opinion a very healthy practice when composing music for tape. I also thank Xavier Serra for his detached collaboration and teaching during CCRMA's DSP Summer Workshop of 94. The paper was written while under a DMA fellowship from Brazilian research support agency CAPES.