GRANULAR SYNTHESIS OF SOUNDS THROUGH MARKOV CHAINS WITH FUZZY CONTROL

Eduardo Reck Miranda

Computer Music Research
Faculty of Technology
University of Plymouth
UK

Adolfo Maia Junior

NICS and IMECC
Universidade Estadual de Campinas
Brazil

ABSTRACT

In this paper we introduce a new model for granular synthesis using Markov Chains and Fuzzy Sets. Whereas Markov Chains are used to control the evolution of the sound in time, Fuzzy Sets are employed to define the internal structure of the sound grains. We provide the mathematical foundations of the model and briefly discuss the implementation of a demonstration.

1. INTRODUCTION

Granular synthesis [1, 2] is commonly known as a technique that works by generating a rapid succession of tiny sounds, metaphorically referred to as sound grains [3]. Granular synthesis is widely used by musicians to compose electronic or computer music because it can produce a wide range of different sounds, but it also has been used in speech synthesis [4, 5]. Clearly a discussion about musical aesthetics may arise from these developments. Although such discussion would be a very interesting topic on its own right, we will not deal with these matters in this paper. A good discussion on the aesthetics of microsound can be found in [6].

Granular synthesis is largely based upon D. Gabor idea of representing a sound using hundreds or thousands of elementary sound particles [7].

In this work we take C. Roads' definition of sound grain as a point of departure to develop a formal but flexible granular synthesis model. Our model uses stochastic processes, namely Markov Chains with Transition Probability Matrix modulated by Membership Functions of the grains with values in the interval $[0, 1]$, which give fuzzy characteristics to the grains. Thus, we propose a new method for controlling the grains by intertwining Stochastic Processes and Fuzzy Set Theory, where the content of the grains (or internal variables) can change their transition probabilities between states. For the sake of clarity, we have chosen a very simple *State Space* to introduce the model, where each grain is itself a state of a Grain Vector $\mathbf{G}$. Therefore, the membership functions in this case modulate the transition probabilities between states (i.e., grains), changing their ordering position in the time domain. In this paper we present just one of the several possible modes of interaction between internal and external control variables.

2. FUZZY GRAIN AND ITS MATRIX REPRESENTATION

Let us denote Ω the space of all possible oscillators, that is the *frequency* $\times$ *amplitude* space of the ordered pair (ω, a), where the variables ω and a vary in some suitable real intervals. Ω is referred to as a Parameters Space. Formally, we define a grain as a finite collection of points $\{(\omega_i(t), a_i(t)), i = 1, 2, \ldots, r\}$ in Ω, which is taken here as a state of a Markov Chain. A grain can be described by its Fourier Partials inside a real interval I. Its spectral content can be written, without loss of generality, as

$$G(t) = \sum_{n=1}^{r} a_n sin[2\pi\omega_n t + \delta_n], \tag{1}$$

where a_n, ω_n, δ_n reads for amplitude, frequency and a possible phase, respectively.

In granular synthesis a sound can be viewed as a fast stream of grains that, from a geometrical point of view, describes a trajectory in the Ω space.

A *fuzzy grain* can be represented as a 3 column matrix

$$G^i = \left[\begin{array}{cc|c} \omega_1^i & a_1^i & \alpha_1^i \\ \omega_2^i & a_2^i & \alpha_2^i \\ \vdots & \vdots & \vdots \\ \omega_r^i & a_r^i & \alpha_r^i \end{array}\right] \tag{2}$$

where the elements of the first column are frequencies, those of second column are the associated amplitudes and those of the third column (which for the sake of clarity are separated from the two other ones by a vertical bar) are the corresponding membership values of each Fourier partial of the grain G^i. This third colunm, also referred to as *Membership Vector*, defines the fuzzy character of the grain, or in a musical jargon, its *weighted* harmonic content. Note that for $\alpha_1^i = \alpha_2^i = \ldots = \alpha_r^i = 1$ we get an ordinary (non fuzzy) grain. The *Grain Vector* is defined as $\mathbf{G} = (G_1, G_2, \ldots, G_N)$. In our model it is also the *State Vector* of the Markov Process.

3. MARKOV PROCESSES FOR FUZZY GRAINS

Fuzzy sets, first proposed by Zadeh [8], are able to handle uncertainty, imprecisions or vagueness. Below we show how the membership functions of fuzzy grains can modify the Markov Transition Matrix in order to control the Markov Chain in a fuzzy way. Please refer to [9] for a good text on Fuzzy Sets. Let us consider a grain described by its Fourier-like equation (1). Each subset of points in Ω represents a grain with particular Fourier partials, that is, it is a sum of basic sinusoidal frequencies.

With the above defined matrices G^i, it is possible to define an unambiguous time evolution of grains through Markov Chains. This can be accomplished using a Fuzzy Transition Matrix, constructed as follows: a transition matrix for ordinary grains, that is, with no membership vector yet, can be written as

$$\mathbf{p} = \begin{bmatrix} p^{11} & p^{12} & \dots & p^{1N} \\ p^{21} & p^{22} & \dots & p^{2N} \\ \vdots & \vdots & \vdots & \vdots \\ p^{1N} & p^{2N} & \dots & p^{NN} \end{bmatrix} \tag{3}$$

where each entry of the above matrix is the conditional (or transition) probability $p^{ij} = p\left(g^i|g^j\right)$ between the i-th to j-th states. Now, we define a Fuzzy Extended Probability Transition Matrix (or simply Fuzzy Transition Matrix)

$$\mathbf{Q} = \mathbf{\Phi} * \mathbf{p} \tag{4}$$

where the symbol $*$ means a matrix operation (a scalar product, a matrix product or any other well defined operation). The elements ϕ^{ij} of the matrix Φ are generated as a finite number of applications of the following basic operations of fuzzy sets: for $i, j = 1, 2, \dots, N$, we define

1. $$\phi^{ij} = \max_{1 \le k \le r} \left\{ \alpha_k^i, \alpha_k^j \right\}, \tag{5}$$

   where α^i and α^j are the membership vectors of the grains G^i and G^j respectively.

2. $$\psi^{ij} = \min_{1 \le k \le r} \left\{ \alpha_k^i, \alpha_k^j \right\}, \tag{6}$$

   where α^i and α^j are the membership vector of the grains G^i and G^j respectively.

3. $$\alpha_c^i = 1 - \alpha^i. \tag{7}$$

   and α_c^i can now be used in any of two operations above.

If we perform these operations in a sequence we obtains, typically, a product of ϕs by ψs, like

$$\Phi^{ij} = \phi_1^{ij} \psi_1^{ij} \dots \phi_n^{ij} \psi_m^{ij}. \tag{8}$$

Note that since the membership function modulates the probability values p^{ij}, the condition for the probability sum $\sum_{j=1}^{N} Q^{ij} = 1$ can be violated. In order to solve this problem we renormalize the matrix Q^{ij} as follows: denoting $q_i = \sum_{k=1}^{N} Q_{ik}$ we define the elements of matrix $\mathbf{P}$ as

$$P_{ij} = Q^{ij}/q^i \qquad i, j = 1, 2, \dots, N \tag{9}$$

Now the probability property $\sum_{j=1}^{N} P_{ij} = 1$ is clearly satisfied. The above definition shows that the internal fuzzy content of the grains have a weight (through the function Φ^{ij}) for their transition to a next state of the Markov Chain. In this simple model, a transition from one state to another corresponds to a jump from a particular grain to another in the grain vector $\mathbf{G}$. In addition the fuzzy content of a grain, that is, its membership vector, can have a significant weight on the probability transition. Since the process is finite, a criterium to halt the process is needed. This will be discussed in the next section.

The above model is suitable for different types of matrix operations on internal as well external variables controlling the behaviour of the grains in time. There is plenty of room for the definition of a great number of different methods to generate and control the time evolution of the grains. We present one of such methods below.

4. CONTROL OF GRAIN STREAMS

A metric is an intuitive and very efficient mathematical tool to control and halt a time process. Therefore, a metric is a very convenient way to formalize notions of approximation and comparison of sound grains, to control their time evolution and sound streams.

Here we introduce one that uses a metric known as Hausdorff Metric [9]. This metric is suitable to measure the distance between sets and, in our case, grains, which are just finite and discrete subsets of Ω. The fuzzy character of grains acts upon the time evolution in our model according to equation (4). In this way, Fourier partials with low membership coefficients contribute little for the time evolution of the Markov process.

Below we indicate the three stop criteria that we devised to halt our Markov Chain, and methods for Grains Updating.

A) Halting Criteria

1. *Convergent Type*: If the distance between the last generated grain and a fixed grain (target) is smaller than a prefixed arbitrary number ϵ then the process halts.

2. *Cauchy Type*: If the distance between two states is smaller than ϵ then the process halts.

3. *Maximal Number of Steps Type (MNS)*: Fix the maximum number of steps for the process to halt.

Maximal Number of Steps Type is the simplest one, since no metric is required. In our demonstration program we implemented fully the MNS and partially, the Cauchy type, at the Hausdorff Metric level, but not at the Fuzzy Metric level. We implemented the Hausdorff Metric as an inequality, so that system runs in loops until it is satisfied. We obtained good results for both controls of the grains streams working together.

In addition we can also specify a number of different formal settings to update the internal content of the grains at each step of the Markov Chain. This provides the means to control the evolution of the sound in the Ω Space. Two updating methods are introduced below.

B) Grains Updating

1. *No Updating: no change in the internal content*

   In this case, each grain G_i corresponds to a state of the grain vector $\mathbf{G}$ and no operation is applied to the internal structure of the grains. Nevertheless this procedure takes into account the fuzzy nature of grains as the role of the membership vectors is to produce new arrangements in time (that is, permutations) for the prefixed grains.

2. *Core Merged Grains*

   In this case we update l-th step grain as a subset merging l previous grains of the Markov Chain; e.g., $G^0, G^1, \ldots, G^{l-1}$. For the sake of clarity, we ignored all other subindexes. Define the *l-Mean Frequency* as

$$\overline{\omega}^{(l)} = \sum_{k=1}^{r} \frac{\omega_k^0 + \omega_k^1 + \ldots + \omega_k^{l-1}}{l} \tag{10}$$

   and take the r closest frequencies from the set $U_l = \bigcup_{k=0}^{l-1} G^k$ to the mean frequency $\overline{\omega}^{(l)}$ and update G^l (with the same letter) as

$$G^l = \left[(\omega_{k_1}, a_{k_1}), (\omega_{k_2}, a_{k_2}), \ldots, (\omega_{k_r}, a_{k_r})\right].$$

This procedure leads to a concentration of frequencies within a narrow bandwidth, but with a large bandwidth for the amplitudes. The *halting criterion* here can be taken as the *Cauchy type*. Given an arbitrary (but small) number ϵ, the process stops if $d_H(G_i, G_{i+1}) \leq \epsilon$, where the distance between two points used for defining the above Hausdorff Distance is given by, for example:

$$d((\omega_i, a_i), (\omega_j, a_j)) = \max_{1 \leq k \leq r} |\omega_i - \omega_j|. \tag{11}$$

If we fix a particular grain in the Ω space, such as $\overline{G}$, then we can consider the *Convergent halt criterion*, that is the process stops if $d_H\left(G_i, \overline{G}\right) \leq \epsilon$.

We can also take the *mean frequency* of the last m grains only and so it reads as

$$\overline{\omega}^{(l)} = \sum_{k=1}^{r} \frac{\omega_k^{l-m} + \omega_k^{l-m+1} + \ldots + \omega_k^{l-1}}{m} \tag{12}$$

and then take the r closest frequencies to $\overline{\omega}^{(l)}$ from the set $U_l = \bigcup_{k=l-m}^{l-1} G^k$. Clearly, for $m = l$ we obtain the previous model.

5. IMPLEMENTATION

We have implemented a demonstration of our model where Membership Matrices modulate a Transition Probability Matrix of a Markov Chain, but the internal content of the grains are not changed during the Process. Thus, the demonstration can be thought of as a *Coarse Grain Fuzzy Synthesis*. We have used the MNS and the Cauchy criteria, by applying the Hausdorff Metric on the Grain (State) Space, in order to halt the process. Weakly convergent process have lead to rich varieties of timbre along sound streams. This is because the system has time enough to explore the possibilities during the Markov Process. Amplitude modulation by means of time windows is used to avoid glitches between the grains. In addition, we included a few special sound effects. For example, we included crescendo and decrescendo effects on a macro-level. In fact we implemented this more generally by using a modulation function with an arbitrary number of peaks and regions of increasing or decreasing rates.

Input information and control parameters are as follows:

1. **Control Parameters for the Markov Process**

   markov_type = $\{$1= random, 2= non random$\}$

   N = number of states (or number of grains)

   n = number of steps of the Markov Process

   init_vect_type = $\{$ 1=defined by user, 2=random uniform, 3=sparse normal random $\}$

2. **Grain Parameters**

   fs = sampling rate

   dur = grain duration in seconds

   r = number of points in a grain, where each point is a Fourier Partial

   grain_type = $\{$1 = deterministic, 2 = random uniform, 3 = sparse normal random $\}$

3. **Fuzzy Control Parameters**

   The type of Membership Matrices is given by:

   The type of vector (α) to construct Membership Matrices is given by:

   alpha_type = $\{$1= random uniform, 2= sparse normal random, 3= non-fuzzy $\}$

   memb_type = $\{$1 = internal product: $< \alpha^i, \alpha^j >$, 2 = $sum(max(\alpha^i, \alpha^j))$, 3 = $max(max(\alpha^i, \alpha^j))$, 4 = non-fuzzy $\}$

To begin with, the algorithm generates a Transition matrix for the Markov Process (Eq. (3)). After a manipulation with membership matrices (Eq. (4)) we obtain a fuzzyfied Transition Matrix as in Eq.(9). The grain parameters control the internal content of the grains, and their output is a sum of Fourier partials (Eq.(1)). The last group of parameters controls the fuzzy characteristic of the grains as described by Eqs. (5)-(8).

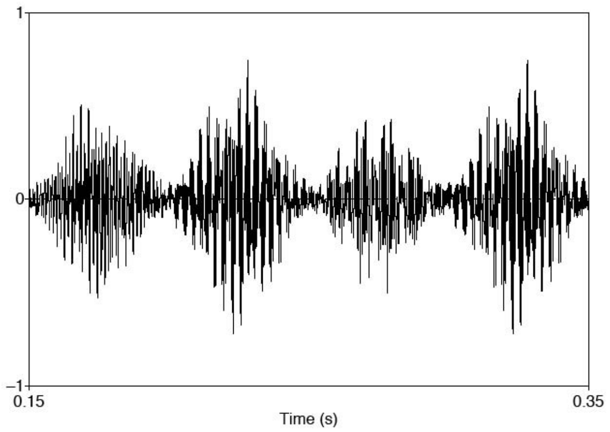

Figure 1. A typical granular sound generated by Fuzzkov 1.0 with Gaussian Envelope Effectc

Sound grains are generated randomly by uniform or Gaussian 3D matrices A with dimensions $2 \times r \times N$, which include r normalized frequencies and amplitudes for N grains (Fourier Partials). From this we obtain a Matrix $B(2, 1, N)$ with the sum of Fourier Partials for the N grains. A Markov transition Matrix $p(N, N)$ is generated and modified by a Membership Matrix $Memb(N, N)$. A number of diferent operations are available for this modification. In this way a fuzzyfied Markov Matrix $Q(N, N)$ is generated, which operates on an array of probability vectors $u(n + 1, N)$. Next, we included a filter, which selects the index of the maximal value of each probability vector $I(1, n + 1)$. Finally, the system reorders the Grain matrix $B(2, 1, N)$ along the index vector $I(1, n+1)$ and produces the sound output.

Figure 1 shows the first four grains of a sound produced with the following settings: *N* = 10, *n* = 15, *init_vect_type* = 1, *dur* = 0.05, *r* = 5, *grain_type* = 2, *memb_type* = 1 and *alpha_type* = 1. In this example, the Hausdorf Metric was used to control the length of the resulting sound. The process runs until the Hausdorf inequality is fulfilled.

6. CONCLUSION

In this paper we presented a model for granular synthesis using a Markov Chain in which each grain is a possible state of a Grain Vector **G**. A major feature of our model is that the spectral components of the grains are coupled with the state transition probability through grain's membership vectors. This allows the user more flexibility and more variability to control the sequence of grains of the Markov Chain. We implemented a demonstration system, where the membership functions modulate the Transition Probability Matrix. In this demonstration, the internal contents of the grains are not changed by the Matrix. In this way, the demonstration synthesiser can be considered as a *Coarse Grain Fuzzy Synthesis*. Currently, we are working on a more complex implementation of the model using Fuzzy functions to emphasize specific components of the spectrum of the grains. We plan to drive the sound flow by updating the grains at each time-step by merging and selecting the most representative frequencies and amplitudes for the next grain of the stream. As Halt Criteria we used the Maximal Number of Steps, as well the Cauchy type with the Hausdorff Metric. A model where the states of the Markov Chain are related to grains's subsets (*Fine Grain*), as well an effective use of Fuzzy Metrics, is currently being tested in our laboratory.

7. ACKNOWLEDGMENTS

This work was supported by FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo) and CAPES (Coordenadoria de Aperfeiçoamento de Pessoal de Nível Superior), Brazil. AMJ is grateful to the Computer Music Research team at the University of Plymouth for their hospitality during his sabbatical.

8. REFERENCES

[1] C. Roads, *Introduction to Granular Synthesis*, Comp. Mus. Jour. **12**(2), 11-13 (1988.)

[2] C. Roads, *Microsound*, MIT Press, Cambridge,MA, (2001).

[3] C. Roads, *Computer Music Tutorial*, MIT Press, Cambridge, MA (1996).

[4] E. R. Miranda, *Generating Source Streams for Extralinguistic Utterances*, Journal of the Audio Engineering Society, 50(3):165-172, (2002).

[5] E. R. Miranda, *Computer Sound Design: Synthesis techniques and Programming*, Oxford: Focal Press (2002).

[6] P. Thomson, *Atoms and errors: towards a history and aesthetics of microsound*, Organized Sound **9**(2), 207-218, (2004).

[7] D. Gabor, *Acoustical Quanta and the Theory of Hearing*, Nature**159** (4044), 591-594,(1947).

[8] L.A. Zadeh, *Fuzzy Sets*, Informat. Control, **8**, 338-353, (1965).

[9] P. Diamond and P. Kloeden, *Metrics of Fuzzy Sets: Theory and Applications*, World Scientific, (1994).