Cellular Automata Histogram Mapping Synthesis

Histogram Mapping Synthesis (HMS) is a new sound synthesis technique using cellular automata (CA), whereby histograms of cellular automata are converted into spectrograms, which in turn are rendered into sounds. One of the mains advantages of HMS is that it affords more control over the sound design process than other sound synthesis techniques using CA. Moreover, HMS allows for simulations of some acoustic music instruments, notably percussion, in addition to sounds that would be very difficult to produce by acoustic means alone.

Cellular Automata are mathematical models introduced by mathematicians Stanislaw Ulam and John von Neumann in the 1960s. Since then, CA have been used for a variety of modeling purposes (Adamatzky et al. 2008). For instance, CA have been very useful in the field of Artificial Life as they can embody processes and phenomena occurring in nature (Langton 2004). Algorithmic techniques for the simulation of natural systems have seen a major growth over the past few decades. Such techniques have been successfully applied in computer music, contributing new insight into the search for innovative algorithms to synthesise sounds and generate music (Miranda and Biles 2007).

Cellular Automata are normally implemented as a regular grid of cells in one or more dimensions. Each cell may assume any state from a finite set of values. The automata evolve in successive generations at every time unit. For each generation, the values of all cells change simultaneously according to a set of transition rules that take into account the states of the neighboring cells. The states of the cells may represent different colors and therefore, the functioning of a two-dimensional cellular automaton may be displayed on a computer screen as a sequence of images, like an animated film.

Cellular Automata are of interest to computer musicians because of their emergent structures; patterns are created from the interaction of multiple units with relatively simple rules (Miranda 2001; Burraston and Edmonds 2005) . This dynamic process allows for the exploration of complex forms in an orderly way. In sound synthesis, CA have been used for controlling the parameters of synthesizers over time. This is attractive especially for those synthesis techniques that demand large amounts of control data over time, making it difficult to be controlled non-automatically; for instance granular synthesis (Truax 1988; Miranda, Correa and Wright 2000). CA are useful in these cases because it is possible to obtain large amounts of structured data with few parameter specifications. The key to the success of a CA-based synthesis technique is to devise a method to transfer - or map - the structured evolution of CA onto the sound synthesis domain.

Cellular Automata and Sound Design

The motivation behind our research was the limited ability to predict the outcomes of CA-based synthesizers. Although we fully acknowledge that a level of unpredictability is accepted, and often desired, in systems for generating music and sound (Chareyron 1990), our research focused on harnessing the capabilities of CA to aid sound design processes in more predictable ways. We believe that we can improve CA-based synthesizers by providing some level of control and predictability for sound design.

Rocha Iturbide (1999) offered an interesting insight referring to CA and sound design limitations, which roughly translates from the original text in French as follows: the danger of using this kind of complex automatic dynamic processes for the control of synthesis is that we tend to limit the control over them and in the end, they become more important than ourselves, who become passive beings satisfied with the observation of the results.

Although CA can allow the production of a rich variety of dynamic sounds, their practical usage requires significant amount of trials to gain an understanding of their potential, and hence relatively few musicians, usually those within a research environment, use them. Our research is aimed at systems that offer users a reasonable level of control over the synthesis and facilitate the exploration of the rich potential of CA for sound synthesis.

To the best of our knowledge, previous systems using CA to synthesize sound and music fall within the category of systems critiqued by Iturbide: they allow the user to establish the values of certain synthesis parameters, and others are controlled over time by the automaton. With this approach such systems produce a wide variety of sounds. However, the sound design possibilities are limited by the fact that the control of the CA is not addressed.

Only a handful of systems have shown some concern on the sound design problem taking into account the control of CA. Among them, the most relevant are LASy by Jacques Chareyron (1990), the CA Workstation by Andy Hunt and colleagues (Hunt, Kirk, and Orton 1991), and Chaosynth by Eduardo Miranda (1995). However, there is much room for more research along these lines, as the authors themselves suggested.

Hunt and colleagues pointed out, referring to the problem of utilizing CA algorithms:

> ‘... most of the time the composer is at the mercy of the algorithm – the music being automatically produced’ (Hunt, Kirk, and Orton 1991 p.165).

Thus, their CA Workstation ‘aims to interrupt this automatic process to let the composer interact directly with the CA at a high-level and thus to make significant artistic decisions’ (Hunt, Kirk, and Orton 1991 p.165). Their system allows the composer to start and stop the automaton and change its parameters at any point. The composer is also able to zoom in and select areas containing visually interesting patterns, which can then be used to control different music or sound parameters. In this research, the issue of gaining control over the evolution of CA is addressed, but the work did not delve into sound synthesis. Whilst it provides specific mapping procedures in the MIDI domain, it does not provide any concrete mapping method in the synthesis domain. In the absence of further published work on sound synthesis, the potential of this approach for sound design remains to be seen.

Miranda's work is concerned with the sound design possibilities of Chaosynth from a control perspective. Regarding the specific control of his automaton called ChaOs, he established that 'different rates of transition from noise to oscillatory patterns' (Miranda 1995 p.299) are obtainable by changing the values of the rule parameters. However, Miranda admitted that 'more research is needed to gain a better understanding of the role of Chaosynth's parameters' (Miranda 1995 p.299). For instance, whilst the most powerful characteristic of Chaosynth is the spectral evolutions of its sounds, these evolutions, on the other hand are highly unpredictable. Miranda subsequently attempted to address this issue, but progress has been hampered by the constraints of Chaosynth's architecture (Miranda, Correa and Wright 2000).

Finally, LASy is a synthesis technique that provides a limited level of control. Chareyron presented a case study from where we seemed to have been able to predict three types of sound evolution, namely:

> - 'sounds whose amplitude and spectral complexity grow in time' and 'the opposite behavior' (Chareyron 1990 p.32).
> - 'transition rules whose actions may be seen as a linear filter' (Chareyron 1990 p.32).

- 'a variation of the Karplus-Strong algorithm' (Chareyron 1990 p.32).

From this brief survey of previous work one can learn that in order to develop a sound synthesis technique capable of allowing a sound design process, more research is needed regarding the control of CA. Furthermore, apart from the control of CA, another capability that our synthesis technique should demonstrate is the possibility of producing a variety of sounds. In this respect, the following criteria are established in order to estimate the potential of a system for producing a variety of sounds:

- Variety of CA used: Does the technique use only one automaton or can many types be used? We would expect that the more CA rules available the more variability there can be in the resulting sounds.

- Variety of mappings used: Does the technique work with only one mapping process or can it use many? We would expect that the more mappings that can be used, the more variability there will be in the resulting sounds.

- Variety of synthesis engines used: Does it use only one synthesis engine or can it use many? Again, we can expect that the more engines that can be used, the more variability in the sounds produced.

- Type of synthesis: Is it pure synthesis or/and audio processing? By incorporating both these types of synthesis, the results should provide more variability.

On the basis of these criteria, we observe that none of the three systems above addresses the use of a variety of synthesis engines. The following table summarises the critical analysis undertaken in this section by presenting the reviewed systems in relation to the adopted criteria.

[SUGGESTED PLACE FOR TABLE 1]

Note that “control” and “prediction” are two concepts that go hand in hand here. In general terms, the more unpredictable a generative process is, the more difficult it will be to control it. Our research seeks to strike a good balance between the predictable and the unpredictable.

Problem Analysis and the DSP Approach

The problem of developing a synthesis technique based on CA capable of allowing a sound design is twofold. Firstly, in order to develop a sound synthesis technique based on CA, it is necessary to devise a mapping procedure. Secondly, in order to allow a sound design process it is necessary to provide control mechanisms for the synthesis technique.

Regarding the design of control mechanisms we have seen that the control of CA is complex but crucial to address our problem. It is difficult to make generalizations because different CA offer different control possibilities. Also, for a same automaton, the devised control mechanisms may differ depending on the mapping used. Nevertheless, we identify two general problems to be faced, common to virtually all the automata, which stem from the nature of CA:

- The autonomous evolution of CA: an automaton is defined as a device operating under its own hidden power. The term is derived from the Greek "automatos" which means acting of one’s own will or independently, self-acting, self-moved.

- The CA unpredictability: the value that a cell will hold after a number of generations is not possible to be ascertained in advance. The only way to find out the value is to compute all the previous CA generations. There is no shortcut in this process, and in this sense, the automaton is unpredictable. Under unpredictability conditions the design of control mechanisms is hindered.

The approach adopted to deal with these issues will be explained below. Let us now examine the mapping problem. Designing a mapping procedure is a trial-and-error endeavour open to the most arbitrary solutions. However, mappings producing interesting musical results are not straightforward to obtain.

As a first step towards a mapping design, noteworthy is the practice of some authors who manage to identify analogies between CA behaviors and sonic phenomena or musical processes. Then, they try to design a suitable mapping process which captures or reflects these analogies. Miranda's work is an example of this methodology. For instance, in Chaosynth he managed to capture the behavior of ChaOs which according to Miranda:

> 'resembles the way in which most of the natural sounds produced by acoustic instruments evolve: sounds tend to converge from a wide distribution of their partials (for example, noise) to oscillatory patterns (for example, a sustained tone)' (Miranda 1995 p.297).

He also adopted this strategy, in the music domain (symbolic note level), for his system CAMUS. Miranda explains:

> 'We have devised a number of experiments in order to study whether cellular automata that exhibit pattern propagation behavior could be used or adapted to model the propagation of musical patterns' (Miranda 2007 p.191).

In order to identify these analogies between CA behaviors and sonic phenomena an important question arises: what type of information is available to us in order to characterise the CA behaviors? Miranda analysed them visually, which is a very limited type of immediate analysis. It can also be unreliable or misleading; note that the visual display of CA is the result of an elementary mapping: cell values are normally assigned to grey levels or different colors and, for instance, with different palettes of colors the same output can be displayed in many different ways, especially in the case of multi-state CA.

A similar situation to the abovementioned limitation is found in the analysis of sounds. By displaying the waveform we can perceive some basic information such as its degree of timbral complexity, but it is through Fourier analysis that we obtain exhaustive information about its frequency content. Analogously, a DSP analysis of CA evolutions could reveal information beyond what can be perceived from their visual display.

Therefore, we propose to consider the output of CA as digital signals and to use DSP procedures to analyse them. This approach opens a great variety of possibilities to better understand the self-organization process of CA, with a view on identifying:

A) Analogies from the sound synthesis domain: if certain analogies are found we would be more likely to design natural mappings, in contrast with more artificial or arbitrary mappings designed when there are not clear correspondences between CA behaviors and sonic phenomena. Even though no clear analogies are found, the output of such a DSP analysis may provide useful information about the CA evolution that could be the inspiration for other musical applications.

B) Possibilities of CA control for sound design with respect to the two main problems stated above:

- With a DSP analysis it may be possible to monitor the CA evolution, being therefore in a better position to make decisions on when (and how, as stated in the following paragraph) to alter the autonomous evolution.

- With a DSP analysis it may also be possible to better characterise not only the CA behaviors, but also the behavioral effects caused to the CA by the control mechanisms that may be devised. That would help to establish cause-effect relationships which could bring a degree of predictability regarding the type of sounds that we could obtain.

To the best of our knowledge this approach has never been pursued before in order to develop methods for sound synthesis based on cellular automata. Histogram Mapping Synthesis is a result of such an approach.

Histogram Mapping Synthesis (HMS)

The core of Histogram Mapping Synthesis is a statistical analysis of CA evolution. The functioning of a two-dimensional multi-state automaton is considered as a sequence of digital images, each of which is analysed by means of histogram measurements. Such a DSP analysis gives a histogram sequence that can be displayed in a 3D plot.

The histogram of a grey-level digital image is a graphical representation of the number of occurrences of each grey level in the image. However note that in this paper we often refer to colors instead of grey levels because CA are usually displayed using a palette of different colors. By dividing the number of occurrences by the total number of pixels of the image, the histogram is normalised and expressed in probabilistic terms giving an estimate of the probability of occurrence of each grey level in the image.

Initially, this kind of histogram analysis was chosen for two main reasons. In the histogram sequences the values adopted over time by the cells of multi-state CA are registered and displayed in a meaningful way. This is especially important when a large number of cell states are brought into play, whose values escape the visual perception. This kind of analysis opens up a new dimension of the self-organization process of CA. Delving into this new dimension provides unique insights that complement the spatial information gathered from the CA visual output.

Moreover, one can think of a certain analogy between histogram sequences and spectrograms. Firstly, their z-axes both correspond to the time domain. Secondly, the x-axis of the histogram sequence, which represents the automaton’s cell values, parallels the frequency domain represented by the x-axis of the spectrogram (in other words, there is a parallelism between the bins of the histogram sequence and the spectrogram’s bins). Finally, the probability scale on the y-axis of the histogram sequence parallels the spectral magnitude represented by the spectrogram’s y-axis. Note that this analogy is formulated here strictly between two types of graphical representations. Any analogy between the concepts behind these graphs would be more awkward to state.

Continuing the theme of the abovementioned analogy, we established that from an appropriate automaton, in the histogram sequences, which can be seen as temporal structures, it is possible to identify time-varying structural elements resembling spectral components of a sound such as sinusoidal components, noise components, transients, spectral envelopes, formants, etc.; please refer to **Figure 1**. With such structural elements we can design the time-varying frequency content of sounds. That is, we can build spectrograms. Such spectrograms can be rendered into sounds using different synthesis techniques.

[SUGGESTED PLACE FOR FIGURE 1]

Seen from another perspective, these structural elements can be used as control data to drive different synthesis techniques. Depending on the resemblance of these structural elements, different mappings onto appropriate synthesis parameters can be established. For instance, structural elements resembling partials can control additive synthesis through controlling the time-varying amplitudes and frequencies of sinusoidal oscillators. This process can be seen as one that creates spectrograms out of histograms. In other cases, histogram sequences resembling sequences of spectral envelopes can control subtractive synthesis by controlling the time-varying magnitude response, or filter shape, of a time-varying filter. If the sound source is, for instance, a white noise generator, this process can still be seen as building spectrograms out of histograms. However this is not the case if the sound source is, for instance, a sawtooth wave oscillator, because the harmonics introduced by such an oscillator are not initially represented in the histogram sequence.

Such possible mappings based on such resemblances make HMS distinctive; in most other techniques there is not an intuitive correspondence between the components of an automaton and the components of a sound.

Control

This section presents an important feature of HMS, namely, a control possibility for sound design, which can be generally applied regardless of the automaton used. Nevertheless, it should be noted that most of the control possibilities that HMS can offer over the synthesis depend to a significant extent on the automaton used. Due to space limitations, below we will briefly present some control mechanisms that belong to only one CA model, the *Hodge Podge Machine*. In this section it will also be helpful to re-examine, from the HMS perspective, the two main CA control problems highlighted above: the autonomous evolution and the unpredictability of CA.

Regarding the unpredictability of CA, unlike most other techniques, HMS does not map directly from the CA cells. These cells are the elements that introduce the unpredictability. Instead, the mapping processes are created from histogram sequences (**Figure 2**). Therefore, with those CA for which the histogram sequences are a means of characterising their behaviors and, whose behaviors are predictable by their parameter values, then by designing mappings after a histogram analysis it is largely avoided the unpredictability of cell values in favour of the predictability of behaviors.

[SUGGESTED PLACE FOR FIGURE 2]

Regarding the autonomous evolution of CA, it is only when a specific automaton is under study when potential mechanisms for controlling its evolution can be identified. Nevertheless, closely related to the autonomous evolution of CA is the automatic character that CA imprint on most synthesis techniques, something that reduces the possibilities of an offline sound design process.

In this respect, a key control possibility of HMS is that we can make decisions before rendering the sounds; we can develop an offline sound design process from the structural elements of the histogram sequences. Virtually infinite sounds can be designed from a single histogram sequence, by manipulating its structural elements and controlling the way they are mapped onto the sound synthesis domain. The fact of being operating in the frequency domain facilitates the control over all the sound attributes: intensity, duration, pitch and timbre. This is accomplished by means of spectral and mapping processes applied to the structural elements of the histogram sequences; for example the assignment of frequencies, amplitude modifications, pitch shifting, time stretching, and so on.

The possibility of designing a timbre and being able to reproduce it with different pitches, intensities and durations, constitutes a special case of sound design, namely, instrument design. Instrument design can be boosted with specific CA in which it is possible to predict the location and characteristics of the structural elements within histogram sequences resulting from different CA runs. With this level of prediction, the automatic characteristic introduced by CA is no longer a problem, but an advantage. Once a type of sound has been designed offline, it will be possible to automatically reproduce multiple instances of the original sound which will be similar but not identical; very much like the sounds produced by acoustic instruments. Of course, for each of these sound instances it will be possible to change some of the sound attributes, such as the pitch, duration and intensity, whilst largely preserving the originally designed timbre.

Modes of Use

Having introduced various ways of synthesising sound with HMS, let us study in detail the different modes of use of HMS.

<u>Manual vs. Automatic Modes</u>: these refer to the mapping process between the CA domain and the synthesis domain. In Manual Mode there is user involvement at the point between the CA run and the synthesis. After the CA run, an offline sound design process can be developed by manipulating the histogram sequence, establishing which synthesis parameters will be controlled by which structural elements of the histogram sequence, and setting the values of the rest of the synthesis parameters. In Automatic Mode there is not user involvement at the point between the CA run and the sound synthesis. Usability depends on the level of predictability. With predictable histogram sequences, in the best case scenario, any process previously developed in Manual Mode can be automatically reproduced. Hence, HMS can automatically produce different sound instances of a type of sound previously designed, behaving like an instrument. Some synthesis parameters can be established beforehand for each instance (for example, to control the pitch). On the other hand, with a low level of predictability, even in the worse case scenario it is always possible to render sounds with random procedures involved in the mapping process.

Real-Time vs. Non-Real-Time Modes: these refer to whether the sound is rendered while the automaton is evolving or after the run is completed. In Real-Time Mode the sound is synthesised as the automaton is evolving. It has not been implemented yet, although it is potentially possible. In Non-Real-Time Mode the sound is heard after the CA run is completed.

Interactive vs. Non-Interactive Modes: these refer to whether there is user control over parameters (of the automaton or the synthesis engine) on-the-fly, i.e. while the sound is being rendered.

The Search for Automata

The search for CA, which are suitable for Histogram Mapping Synthesis, is largely a matter of experimentation. Traditionally, an automaton was considered potentially interesting for sound synthesis after a visual analysis of its behavior. Within the context of HMS, an automaton can be considered interesting depending on the results obtained from a histogram analysis of its evolutions.

The results of a histogram analysis can be more or less expectable depending on the automaton in question. For instance, for some CA models, the main characteristics of its histogram sequences are, in general terms, as one might expect provided that one understands the model's behavior. On the other hand, the histogram sequence structural elements of some behaviors of the Hodge Podge Machine (see below) are by no means possible to expect in advance from its visual perception. It is the DSP analysis which reveals such useful information. It is worth mentioning that, although the first experiments with HMS resulted in essentially flat histogram sequences, the first successful results were these kinds of unexpected structures from the Hodge Podge Machine. It was a discovery in its own right.

There is a requirement that every automaton must satisfy: it must be multi-state so that its corresponding histogram’s x-axis is large enough so as to accommodate enough structural elements for the synthesis of complex sounds.

Our work so far has focused on two-dimensional CA, however HMS can be produced using CA grids of different dimensions, and also with other computational models. Generally speaking, it is possible to obtain histogram sequences from virtually any stream of numerical information. However, it is our belief that self-organised systems are the most fruitful for producing musically interesting histogram sequences. Self-organised systems can be modelled using means other than CA, for instance, differential equations and Monte Carlo simulations.

In the following section we will present some HMS results using a specific automaton: the Hodge Podge Machine. We have achieved relevant results with other CA such as the *Multitype Voter Model* and the *Plurality Vote Rule*, but due to space limitations, these cannot be documented here. (Anonymous XXX, XXX – removed for review).

HMS with the Hodge Podge Machine

The Hodge Podge Machine (HPM) is a 2D cellular automaton that models certain chemical reactions (Gerhardt and Schuster 1989). One important characteristic of this automaton is its capability of exhibiting different behaviors depending on the values of its rule parameters and other specifications such as the type of neighbourhoods, and so forth. Starting the CA run from a uniform random distribution of colors, each of these behaviors is characterised by the emergence of a particular type of spatial patterns. For instance, the Spiral behavior is characterised by the emergence of spiral waves.

For sound synthesis with HMS we have identified and studied three different behaviors. We refer to them as Spiral, Quasi-Synchronic and Fast-Infection behaviors. In the context of our histogram analysis it is relevant the fact that the resulting histogram sequences are a means of characterising each of these behaviors. For sound synthesis purposes, in such histogram sequences we can identify a variety of structural elements resembling spectral components of sounds. Some examples of these resemblances are illustrated in **Figures 3** and detailed as follows. **Figures 3**a, 3b and 3c are zooms of histogram sequences of the Quasi-Synchronic behavior. In **Figures 3**a, we can see a number of narrow bands resembling sinusoidal components (Sound Example 1). In **Figures 3**b, apart from two sinusoidal components, we can see a number of peaks at the beginning of the histogram sequence that could be considered as transients. In **Figures 3**c we see a couple of wide bands that could be considered as steady-state noise components, as formants, or in general terms, as a sequence of spectral envelopes. **Figures 3**d and 3e are zooms of histogram sequences of the Fast-Infection behavior showing non-sustained structures. In **Figures 3**d we can see a number of narrow bands that resemble sinusoidal components (Sound Example 2). In **Figures 3**e we can see a more compact structure that can be considered as a sequence of spectral envelopes (Sound Example 3). Finally, **Figures 3**f is a zoom of a histogram sequence of the Spiral behavior showing a number of narrow bands resembling sinusoidal components (Sound Example 4, which is a sound texture in contrast with the previous examples that are note-type of sounds). Although for space limitations this paper focuses on the study of the HPM automaton only, we wish to include some sound examples rendered with other automatons. Sound Example 5 is an example of another texture using the Multitype Voter Model.

[SUGGESTED PLACE FOR FIGURES 3a, 3b, 3c, 3d, 3e AND 3f]

An important general aspect of controllability of the HPM is that the different behaviors that it can exhibit are determined by its parameter settings. Once a certain behavior is achieved by a set of parameters, then, different runs with the same settings, but starting from different uniform random distributions of colors, will produce the same behavior; i.e., the spatial patterns that the automaton develops are of the same type every run, although different from each other. However, the level of prediction goes beyond this. We have found an invariance property since different runs produce histogram sequences with the same type of structural elements which moreover are located in the same positions. That is to say, in all runs the automaton self-organises through the same set of predominant colors. The time-varying amplitude evolutions of each structural element is slightly different every run.

The level of prediction available through this invariance property is useful for instrument design in Automatic Mode. Once a timbre has been designed in Manual Mode, all the processes developed in this design stage can be replicated in order to automatically obtain multiple instances of the original type of sound (with different pitch, intensities and durations, as desired). All instances will share the same spectral structure, but with differences in the time-varying amplitudes of the spectral components. In conclusion, we can design instruments able to produce similar but not identical sounds. Sound Example 6 serves to illustrate this similarity feature which has been also achieved with other automatons.

To conclude this highly summarised section dedicated to the HPM, it is worth mentioning a control mechanism we developed based on coupled CA. Two CA are used, a HPM and a growth model (GM). The latter serves as control of the former. This method explores the fact that different simultaneous behaviors can be evolved within the same HPM if we bring into play different sets of parameter values.

However, we restrict the number of parameter sets to two. Therefore, the GM will have only two states and will delimit two dynamic zones in the HPM each of which governed by a different set of parameter values. The predictable evolution of the two GM zones will produce a controlled dynamic sound spectrum. Among all the possibilities that this process affords we highlight its application to the attack portion of a sound, making it dynamically more complex than the rest of the sound (see Figure 4 and Sound Example 7).

[SUGGESTED PLACE FOR FIGURE 4]

Frequency Trajectories

We have seen how to produce amplitude trajectories with HMS. However, sounds produced by acoustic means also involve frequency trajectories. Frequency trajectories in HMS can be generated by the automaton and/or by the user as will now be shown.

Regarding the latter possibility, leaving the generation of amplitude trajectories to the automaton, any kind of frequency trajectories can be specified or generated by the user. The possibilities could range from the design of quasi-random frequency fluctuations, which are often found in sounds produced by acoustic instruments, to the production of glissandi that the user can specify at will.

Frequency trajectories can also be generated by the automaton itself. Different automatons provide different possibilities and the discovery and practical use of these is a matter of experimentation. One approach is modulating the angle of the sinusoidal-like components (using for example, frequency modulation) with their same amplitude trajectories. The results range from interesting frequency fluctuations, which can be obtained from using amplitude trajectories similar to those found in **Figure 4** (Sound Example 7), to glissando gestures (Sound Example 8).

Finally, another possibility is to identify "frequency" trajectories in the histogram sequences. Interesting results have been obtained using modified versions of the HPM (Figures 5a and 5b).

[SUGGESTED PLACE FOR FIGURE 5a AND 5b]

Conclusion

This paper introduced Histogram Mapping Synthesis (HMS), a new sound synthesis technique using Cellular Automata.

Controllability was the main criterion established in our research to determine the sound design possibilities of a CA-based synthesizer. Controllability is the main characteristic of HMS: a considerable number of HMS-based control mechanisms have been identified and developed. HMS also fulfils all four criteria introduced in this paper for producing a variety of timbres using CA, namely, allowing the use of a variety of: CA, mappings, synthesis engines and types of synthesis. We highlighted the importance of being able to use a variety of synthesis engines because none of the reviewed CA-based synthesizers provides this capability. In our research, sinusoidal additive synthesis and subtractive synthesis by means of FFT Convolution filtering have been the main approaches. As for variety of CA, in this paper we have presented only the Hodge Podge Machine. However, we can report that have achieved excellent results with other CA such as the Multi-type Voter Model and the Plurality Vote Rule, but these have not been documented here due to space limitations.

The sounds produced with HMS range from those that are novel to those that are imitations of sounds produced by acoustic means. All the sounds obtained present dynamic features and many of them, including some of those that are novel, retain important characteristics of sounds produced by acoustic means. As an example of this, a control mechanism has been presented that leads to the production of sounds with more complex spectra in the attack portion than in the rest of the sound. As a final point, most of the control mechanisms allow the production of sounds which are similar but not exactly identical, behaving in this sense like an acoustic instrument.

HMS has significant potential for sound synthesis because CA research is very active and new CA development is ongoing. Therefore, the investigation of new HMS possibilities using other multi-state CA is a vast venue to explore. The search for CA suitable for Histogram Mapping Synthesis is largely a matter of experimentation. It is only after histogram analysis that one can determine the suitability of the automaton in question. This is because on many occasions the results of a histogram analysis are by no means possible to expect in advance from the CA visual perception. A case in point is the Hodge Podge Machine, briefly studied in this paper, where unexpected time-varying structures with "sound-like" features were discovered in the histogram sequences. Expectations for discoveries like this will always be present.

References

Adamatzky, A., et al. (Eds.), 2008. *Automata-2008: Theory and Applications of Cellular Automata*. Frome, UK: Luniver Press.

Burraston, D. and Edmonds, E. 2005. "Cellular Automata in Generative Electronic Music and Sonic Art : Historical and Technical Review". *Digital Creativity* 16 (3):165-185.

Chareyron, J. 1990. "Digital Synthesis of Self-Modifying Waveforms by Means of Linear Automata". *Computer Music Journal* 14 (4):25-41.

Gerhardt, M., and Schuster, H. 1989. "A cellular automaton describing the formation of spatially ordered structures in chemical systems". *Physica D* 36 (3):209–221.

Hunt, A., Kirk, R., and Orton, R. 1991. "Musical Applications of a Cellular Automata Workstation". Proceedings of the 1991 International Computer Music Conference, pp. 165-168.

Langton, C. G. 2004. "Studying artificial life with cellular automata". *Physica D: Nonlinear Phenomena* 22(1-3):120-149.

Miranda, E. R. 1995. "Granular Synthesis of Sounds by Means of a Cellular Automaton". *Leonardo* 28 (4):297-300.

Miranda, E. R., Correa, J. S., Wright, J. (2000). "Categorising Complex Dynamic Sounds".*Organised Sound*, 5(2):95-102.

Miranda, E. R. 2001. *Composing Music with Computers*. Focal Press, UK: 2001.

Miranda, E. R. 2007. "Cellular Automata Music: From Sound Synthesis to Musical Form". In E. R. Miranda and J. A. Biles (Eds.) *Evolutionary Computer Music*. London: Springer, pp.170-193.

Miranda, E. R., and Biles, J. A. 2007. *Evolutionary Computer Music*. London: Springer.

Rocha-Iturbide, M. 1999. "Les techniques granulaires dans la synthèse sonore". Doctoral thesis, University of Paris VIII, U.F.R. Arts, Philosophie et Esthetique.

Truax, B. 1988. "Real-Time Granular Synthesis with a Digital Signal Processor". *Computer Music Journal* 12 (2):14-26.

Appendix: Sound Examples

Note that these examples represent only a few of the possibilities of HMS, since virtually infinite sounds can be rendered from a single histogram sequence by developing different offline sound design processes.

Also, different sound engines can endow the final sounds with a different character. In the below examples, the structural elements of the histogram sequences have been used to control two types of synthesis engine: sinusoidal additive synthesis, with or without angle modulations, and subtractive synthesis by means of FFT convolution filtering.

"Harmonic" and "non-harmonic" are terms used here to distinguish two ways of producing sound with sinusoidal additive synthesis. The former refers to the assignment of frequencies to the partials following the harmonic series, and the latter refers to the assignment of random frequencies to the partials. These random values are bounded between the lowest and the highest frequency values selected by the user.

Sound Example 1: A harmonic sound rendered with a number of narrow bands from a Quasi-Synchronic behavior.

Sound Example 2: This example consists of two sounds from a Fast-Infection behaviour, being the first one non-harmonic and the second one harmonic.

Sound Example 3: This sound has been synthesized using subtractive synthesis of white noise by means of FFT Convolution filtering.

Sound Example 4: A non-harmonic sound rendered from a Spiral behavior histogram sequence. Only the most prominent narrow bands have been included in the synthesis, which have been considered as sinusoidal components.

Sound Example 5: A non-harmonic texture rendered from a Multitype Voter Model histogram sequence. A long fade-in (up to the midpoint of the sound) and a long fade-out (from the midpoint to the end) were applied externally.

Sound Example 6: This example illustrates the possibility of creating similar but not identical sounds. It has been rendered using the Multitype Voter Model. The sounds have interesting spectro-temporal differences between their attack portions. To facilitate the perception of this nuance, the melody begins with a sequence of groups of three notes with the same pitch. The purpose of this example is to illustrate this similarity feature, though it is admitted that, perhaps, the timbre of the designed instrument is not particularly interesting.

Sound Example 7: Two sounds rendered from the structure in Figure 4. The first is harmonic and the second is non-harmonic. These sounds present greater spectral complexity in the attack portion than in the rest of the sound. Before rendering the sounds, the following offline sound design process was carried out. The bins of the histogram were sorted according to the energy that each partial would contribute to the sound. Then, the third bin of the newly sorted structure was doubled in amplitude. The frequencies were assigned in ascending order, so that the higher partials are the short-lived ones. This is an example where a small angle modulation has been applied to the sinusoidal components. The modulation is driven by the amplitude trajectories so that the amplitude fluctuations are translated into frequency fluctuations.

Sound Example 8: A non-harmonic sound rendered from a Multitype Voter Model histogram sequence. An angle modulation has been applied to the sinusoidal components using FM. The modulation is driven by the amplitude trajectories of the histogram bins, so that these are translated into frequency trajectories in the form of glissandi.

FIGURES & TABLE CAPTIONS:

Figure 1. One 2D CA image and its corresponding histogram, which is similar to a sound spectral analysis with those peaks resembling partials. The time dimension, not included in this figure, can be appreciated in Figure 3.

Figure 2. Mapping process from histograms instead of directly from CA.

Figures 3a, 3b, 3c, 3d, 3e, and 3f. Different structures in the histogram sequences resembling spectral components of sounds. Note that some of these figures do not have the probability axis because an amplitude envelope has been estimated. Once the envelope of a histogram sequence has been computed, the y-axis of its 3D plot no longer informs on a probability measure, and hence, it is not displayed in the figures.

Figure 4. Zoom of a HPM histogram sequence after envelope estimation. The HPM exhibits two simultaneous Quasi-Synchronic behaviors and is controlled over time by a GM. The bins of the histogram sequence can represent the time varying amplitudes of sound partials, some of which are short-lived in the attack portion. We can also observe amplitude compensations between the disappearing components and the permanent ones.

Figure 5. "Frequency" trajectories found in the histogram sequences. Convergent glissandi (a). Glissandi in the "attack" portion of two narrow bands (b).

Table 1. Previous systems in relation to defined criteria.

FIGURE 1:

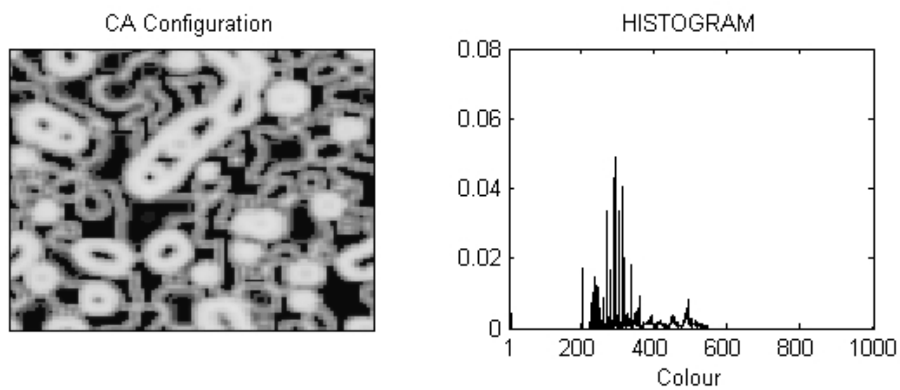

FIGURE 2:

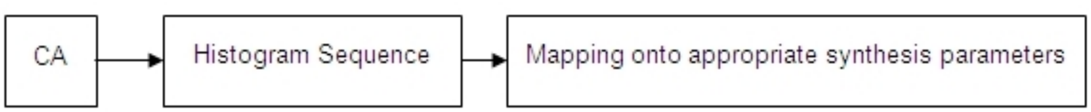

FIGURE 3a:

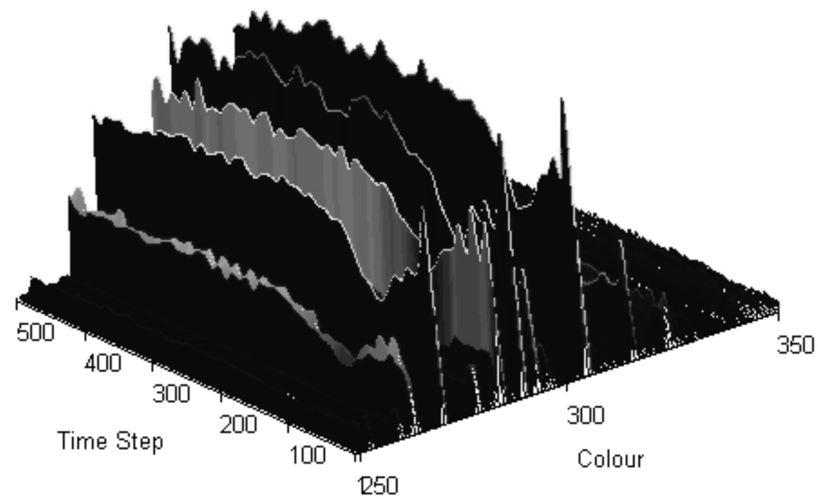

FIGURE 3b:

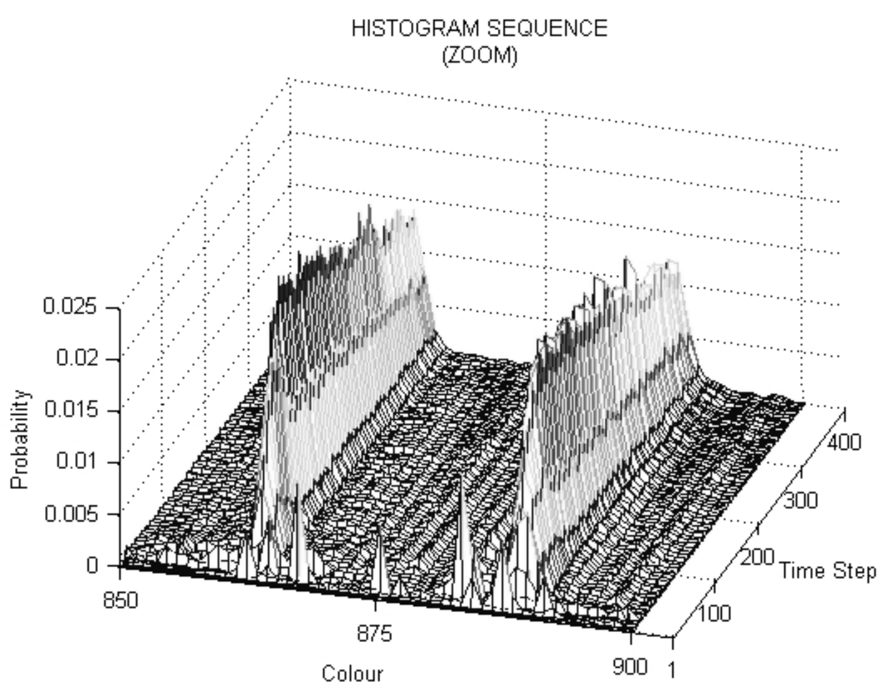

FIGURE 3c:

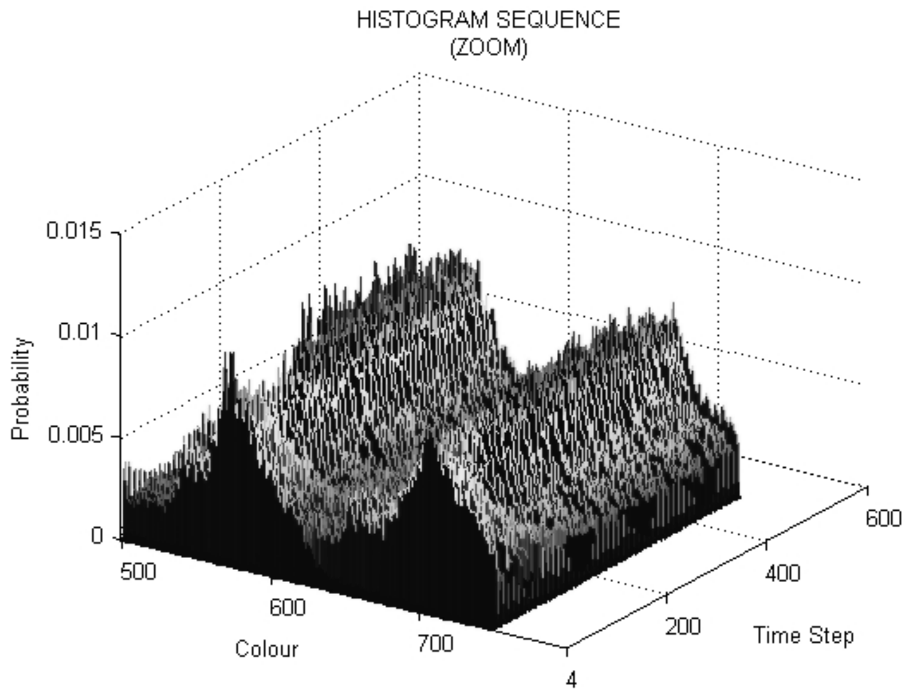

FIGURE 3d:

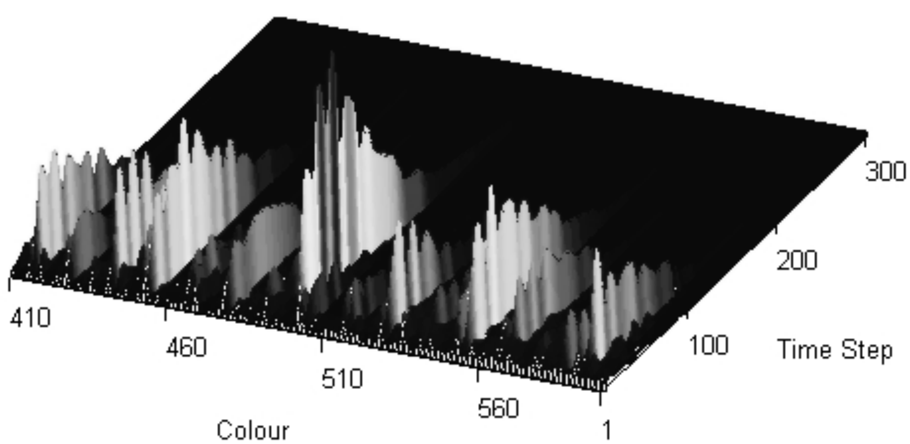

FIGURE 3e:

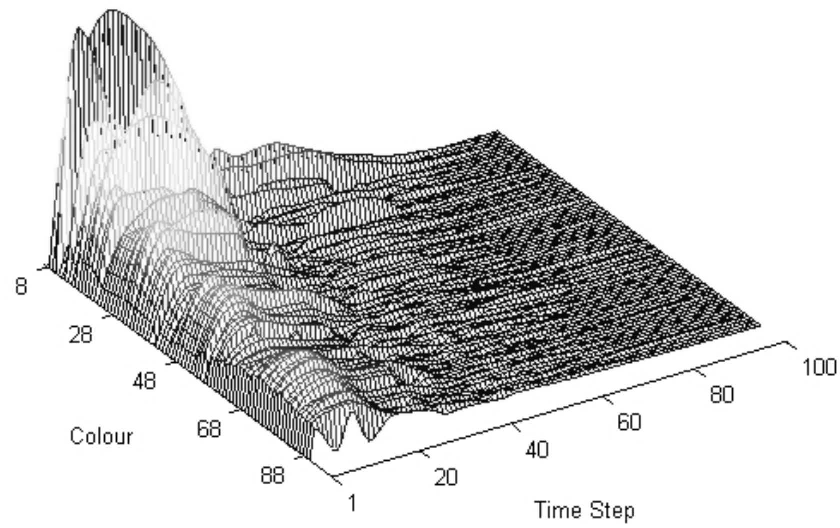

FIGURE 3f:

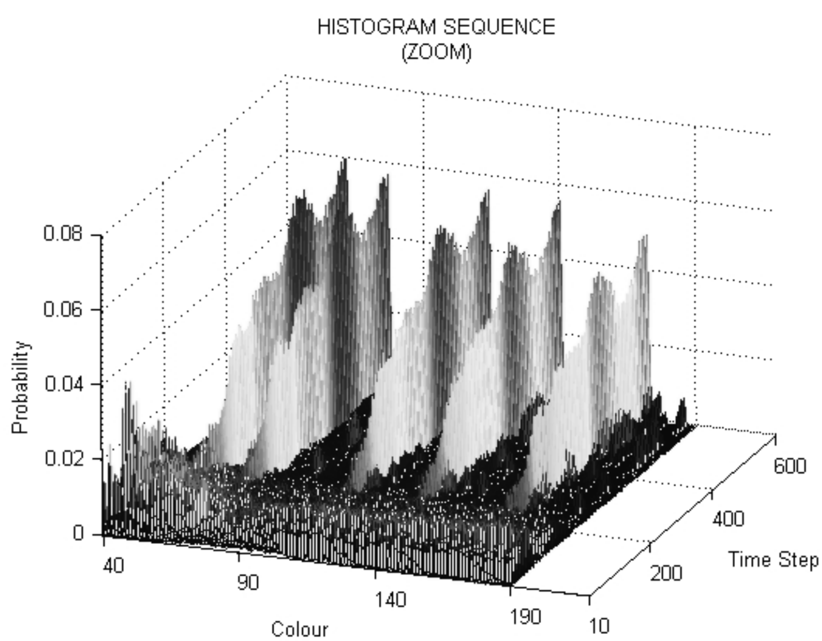

FIGURE 4:

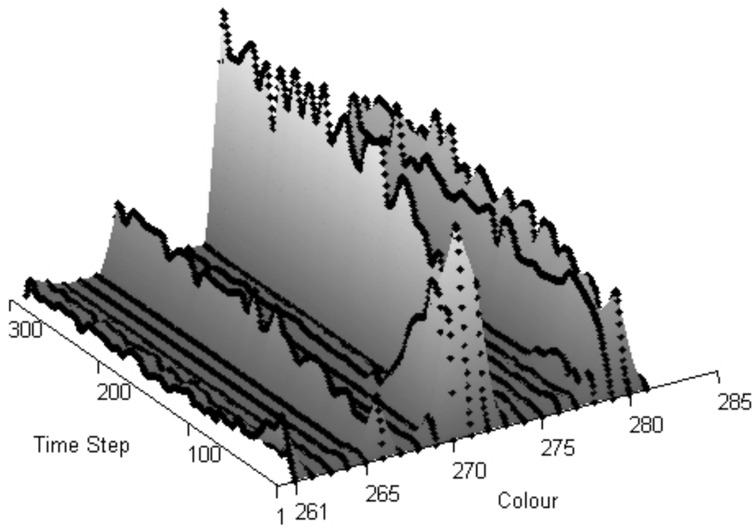

FIGURE 5a:

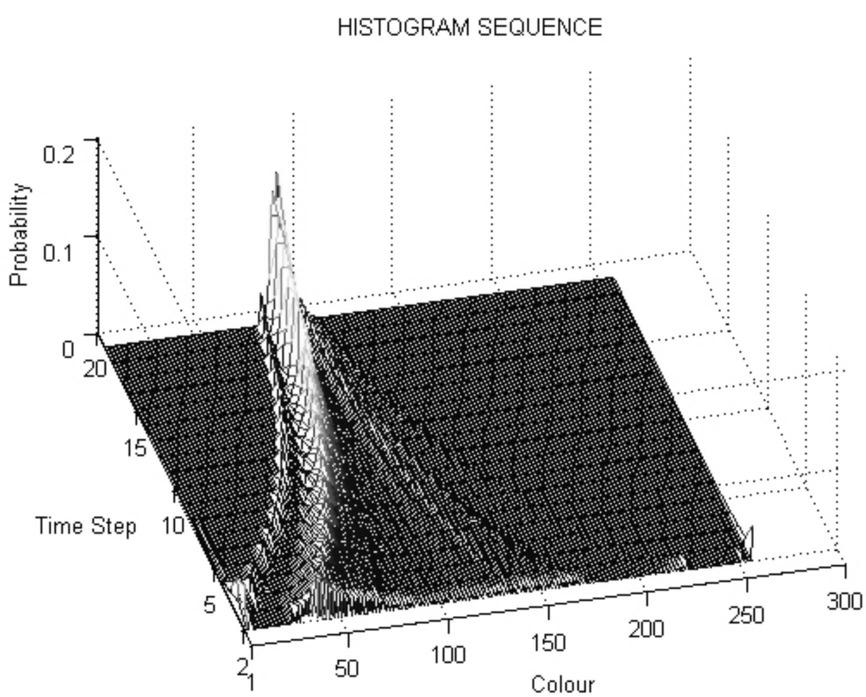

FIGURE 5b:

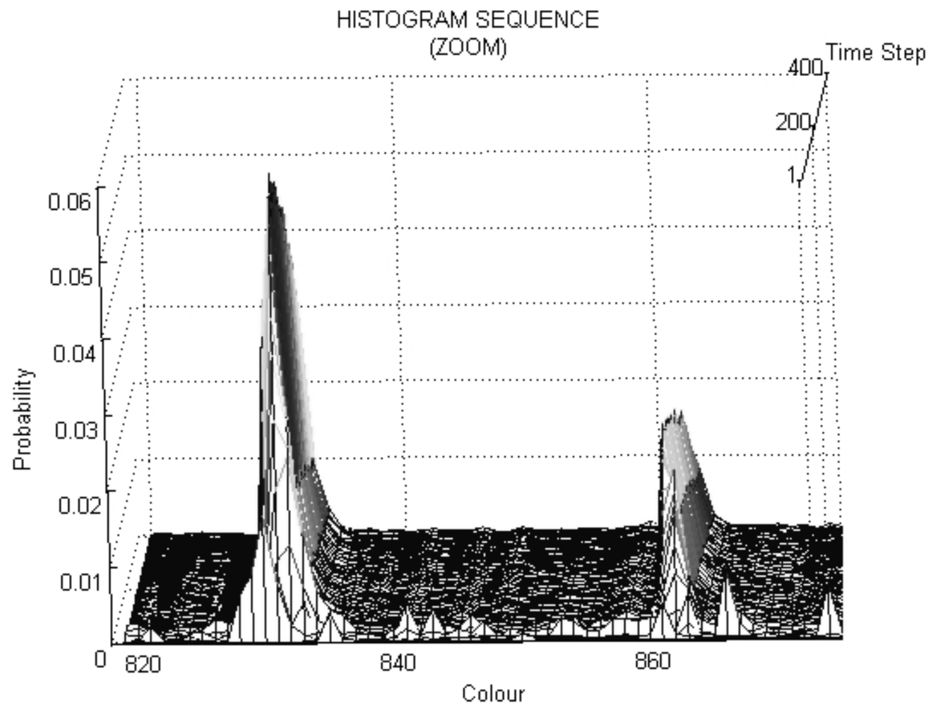

TABLE 1:

	CA Workstation	Chaosynth	LASy
Variety in CA	Yes	No**	Yes
Variety in mappings	Yes	No	No
Variety in engines	No	No	No
Type of synthesis	Pure synthesis *	Pure synthesis	Audio processing
Control / prediction	In MIDI domain	Low	Low

* Only suggested.

** Variety achieved through different behaviors.