ON THE LEARNING STAGES OF AN INTELLIGENT RHYTHMIC GENERATOR

Marcelo Gimenes, Eduardo Reck Miranda, Chris Johnson
Computer Music Research, School of Computing, Communications and Electronics,
University of Plymouth, UK
{marcelo.gimenes, eduardo.miranda, c.johnson}@plymouth.ac.uk

ABSTRACT

We are developing RGeme (Rhythmic Meme Generator), an artificial intelligence system for the composition of rhythmic streams inspired by Richard Dawkin's theory of memes. The system is based on intelligent agents that learn from examples and interact by generating rhythms to each other.

The system has two broad stages. In the first one, the *learning stage*, which is the main focus of this paper, Agents are trained with examples of musical pieces in order to evolve a “musical worldview” which consists of a “Style Matrix” of basic rhythmic elements (or "rhythmic memes”). In the *production stage*, Agents are able to learn from each other's "compositions" and capable of evolving new rhythmic styles by adapting to each other's rhythms.

The dynamics of this evolution is studied by analysing the behaviour of the memes logged during the learning and the interaction processes. In this paper we present the learning stage of a simulation of the system that uses rhythmic information taken from music compositions by two Brazilian composers, namely Ernesto Nazareth and Tom Jobim.

Only the learning stage is discussed in this paper. The production stage and its connexions with the learning stage will be introduced in a future paper.

1. INTRODUCTION

Computers have long being used for aiding musical composition in a number of possible ways. Some composers ([2], [6], [18], [19]) use mathematical models such as combinatorial systems, grammars, probabilities and fractals to create new pieces of music. Other systems apply standard Genetic Algorithm procedures for evolving musical materials such as melodies, rhythms, chords, and so on. One such example is Vox Populi [13] which evolves populations of chords of four notes, through the operations of crossover and mutation.

Evolutionary Computation models are also being used in many models. In one of them, CAMUS [14], the emergent behaviour of Cellular Automata (CA) is used to generate musical compositions in which case the co-ordinates of the cells are associated with the distances between the notes of a set of three musical notes.

Impett [9] uses an Agent system to generate musical compositions. Through the interaction of embodied behaviours that co-exist and interact in the same world, Agents are adaptive to the changing environment to which they belong.

A growing number of researchers are developing computer models to study cultural evolution, including musical evolution [1]. For instance, Miranda [15] investigates how musical structures can originate and evolve in artificially created environments and inhabited by virtual communities of musicians and listeners.

Some rhythmic generating systems have already been proposed [8]. Pachet [16] describes an evolutionary model where a group of agents play rhythms together in real time without prior knowledge about the music to play. Agents play in cycles to which transformation rules are applied in order to produce new variations.

The system that we present in this paper, RGeme, as described in the next Section, makes use of some of the previous experiences, in addition to concepts inspired on the theory of memes by Dawkins [4]. According to Dawkins, memes are basic units of cultural transmission in the same way that genes, in biology, are units of genetic information. The initial argument of this theory references musical aspects:

"Examples of memes are tunes, catch-phrases, clothes fashions, ways of making pots or of building arches. Just as genes propagate themselves in the gene pool by leaping from body to body via sperm and eggs, so memes propagate in the meme pool by leaping from brain to brain via a process which, in the broad sense, can be called imitation."([5], p. 206)

Cox [3] asserts that the “memetic hypothesis” is based on the concept that the understanding that someone has on sounds comes from the comparison with the sounds already produced by this person. The process of comparison involves tacit imitation, or memetic participation that is based on the previous personal experience on the production of the sound.

Gabora [7] explains that, in the same way that information patterns evolve through biological processes, mental representation, or memes, evolve through the adaptive exploration and transformation of an informational space through variation, selection and transmission. Our minds perform tasks on its replication through an aptitude landscape that reflects internal movements and a worldview that is continuously being updated through the renovation of memes.

Our aim with this research is to contribute to this trend by means of computational modelling of a memetic environment for the generation of rhythmic streams.

2. THE MODEL

RGeme is an artificial intelligence system for the composition of rhythmic passages that uses the computational framework of intelligent Agents. Maes

[12] asserts that autonomous Agents are computer systems that inhabit a dynamic and complex environment, sense and act autonomously in this environment executing a series of goals and tasks for which they were devised.

These computational entities are designed to have the ability to perceive and to act in their environment in order to achieve certain targets [17]. In our system, Agents are able to look for the existence of music compositions and to choose the ones with which they will interact; later on Agents parse and extract the rhythmic information. Conversely, Agents are also able to actuate in the system through the generation of new rhythmic streams.

At the beginning of a simulation a number of Agents are created. They are given an identity (name), a number of tasks ("Goal Matrix") and the criteria ("Evaluation Matrix") they will apply to choose the compositions with which they will interact ("Candidate Compositions").

Three types of agent tasks are envisioned (listen, practice and compose) to be accomplished during three different stages (Listener, Student, Composer) to which the Agents will belong during their lifetime. As Listeners, Agents can only execute listening tasks. In the Student stage, Agents can listen to and practice rhythms. In the last stage, as Composers, Agents can execute listening, practicing and composition tasks. Broadly speaking these stages and tasks split the model into two general concepts: the learning and the production phases. Evidently, listening and practicing tasks focus mainly on the learning phase whereas composition tasks focus mainly on the production phase.

Before the execution of listening and practicing tasks the Agents choose the Candidate Music according to the Evaluation Matrix (composer's name and/or year of composition). An Evaluation Matrix can determine the same rules for the Agent's entire lifetime or can establish different ones according to the stage in which the Agent is at a specific moment. This last possibility will be employed in the simulation described in the next Section.

The basic difference between listening and practicing tasks is in the number of times an Agent parses the Candidate Music and adapts its internal states to the contents that it finds in the music. This feature is controlled by a 'number of rehearsals' variable, which is always 1 for the listening task and can be more than 1 for the practicing task.

Once the Candidate Music is chosen, Agents parse it in order to extract rhythmic memes (Candidate Memes). In the real world, the definition of the exact length/boundaries of a musical meme is a very complex subject for a number of reasons ([10], [11]). Roughly speaking, different individuals can identify different memes in the same or in different pieces of music in accord with, among other factors, their previous personal musical background. Our model, however, was designed to produce musical material in artificially inhabited environments, although it has many features that were inspired in real life situations. Therefore, in order to keep it reasonably simple in the first steps of implementation, currently each rhythmic meme has a fixed length that corresponds to a music bar.

Agents store their musical knowledge in a Style Matrix in which every entry is related to a unique rhythmic structure (rhythmic meme) with the following information:

- the dates (represented in terms of a counter that calculates each interaction cycle) in which the memes were first and last listened to,
- the number of times the memes were listened to,
- the weight (importance) the memes hold due to the various interactions with the Candidate Memes and
- the Candidate Music the meme was listened from.

Style Matrices also hold 'Composition Maps', which correspond to the ways the Candidate Memes are interconnected in the Candidate Compositions.

RGeme represents rhythmic memes coded as vectors whose entries are 0s and 1s (Figure 1), where 1 means the trigger of sounds and 0s are used to represent rests and as time placeholders.

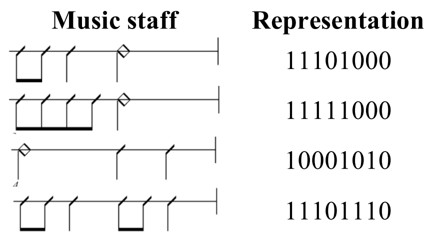

Figure 1. Musical staff and corresponding meme representation

One of the drawbacks of this representation is that information such as the position of the meme in the musical stream, the intensity of each sound and the articulation (duration of the sounds) are not taken into account. Nevertheless, this representation is still useful for the initial implementation of the system and can be extended to include the above-mentioned aspects.

Every time a composition is chosen and the Candidate Memes are parsed, a transformation algorithm is applied as follows. In the beginning of any simulation the Style Matrix is empty and receives the first parsed Candidate Meme. Its weight is set to 1 and the dates of first and last listening are set according to the general time controlled by the system. The second Candidate Meme is then compared with the first meme in the Style Matrix. If they are different the Candidate Meme is copied to the Style Matrix and its weight is set to 1. Subsequently, the other memes in the Agent's Style Matrix have their weight upgraded according to their distance to the Candidate Meme being compared.

The distance between two given memes $a = [a_1, a_2, ... a_n]$ and $b = [b_1, b_2, ... b_n]$, is defined as:

$$d(a,b) = \frac{1}{n}\sum_{i=1}^{n} | a_i - b_i | \qquad (1)$$

Equation 1: Measure of distance.

For example, the distance between the memes a = 01011101 and b = 11011101 is d(a, b) = 0.125 while the distance between the memes a and c = 11111111 is d(a, c) = 0.375. This is, roughly speaking, the so-called block distance.

Once all Candidate Memes are compared with the memes in the Style Matrix, the new memes are copied and the remaining ones have their weight upgraded, a forgetting effect is applied to the memes in the Style Matrix that don't appear in the current Candidate Memes set.

In order to make these concepts more clear, we will advance somewhat on the discussion of the simulation of the next Section and present in the following paragraphs two extracts from the Style Matrices generated by the system. In Table 1 we show the Style Matrix after an Agent listened to the first music ('Arrojado', by composer Ernesto Nazareth):

Meme	**dFL**	**dLL**	**nL**	**W**
01011101	1	1	6	1.0385
11011101	1	1	31	1.0424
10001000	1	1	1	1.0181
10010101	1	1	1	1.0171
11011010	1	1	1	1.0159
10011010	1	1	4	1.0090
10011001	1	1	4	1.0075
11111111	1	1	1	1.0040
10000000	1	1	1	1.0000

dFL: date of first listening
dLL: date of last listening
nL: number of listening
W: weight

Table 1: Extract from 1st Style Matrix

Table 2 shows the corresponding data in the Style Matrix after Agent 'A' listened to the second music ('Matuto', by same composer):

Meme	**dFL**	**dLL**	**nL**	**W**
01011101	1	2	7	1.0700
11011101	1	2	37	1.0795
10001000	1	1	1	1.0240
10010101	1	2	21	1.0588
11011010	1	2	2	1.0399
10011010	1	1	4	1.0161
10011001	1	2	10	1.0439
11111111	1	1	1	1.0114
10000000	1	2	2	1.0273
00010101	2	2	1	1.0350
10100101	2	2	1	1.0248
11011111	2	2	2	1.0223
10010111	2	2	4	1.0245
10011111	2	2	2	1.0206
11011000	2	2	1	1.0088
10000101	2	2	2	1.0090
11010101	2	2	1	1.0075

Table 2: Extract from 2nd Style Matrix

Notice, for example, that:

- After the first cycle of interaction (Style Matrix 1 or SM1), meme 11011101 (second in the list) had been listened (nL) 31 times and its weight (W) was 1.0424. After the second cycle of interactions (Style Matrix 2 or SM2) its number of listening was 37 and its weight had been raised to 1.0795.
- In SM1 meme 11111111 (8th in the list) was listened only 1 time and its weight had been set to 1.0040. In SM2, although the number of listening had been kept the same, the weight had been raised to 1.0114. The additional weighting was due to the similarities that this meme had in comparison with the other ones that were listened to after it first appeared in any Style Matrix. Also, no 'forgetting' effect was applied because it was listened in two consecutive interaction cycles.
- Meme 11010101 (last in Table 2) only appears in SM2 and its weight was set to 1.0075, which means that, after its first appearance (in which the weight had been set to 1), the weight raised due to the comparisons made with the other memes that were listened to afterwards.

Up to this point we described how the transformation algorithm alters the musical knowledge possessed by the Agents as a result of the execution of the listening and practicing activities.

Since the learning phase is the main focus of this paper, it suffices to say that in the production phase the Agents execute composition tasks mainly through the reassignment of the various Composition Maps according to the information previously stored in the learning phase. Composition tasks, beyond the production of new material, also have a transformation effect on the Style Matrix where all memes are updated according to the musical material used in the newly produced rhythms.

The model has the potential to execute intricate simulations with several Agents learning at the same time from rhythms by composers from inside and outside the system's environment.

In the next Section we present an experiment where some of the features of the system are evaluated and better explained.

3. A CASE STUDY

A group of 20 pieces by Brazilian composers Ernesto Nazareth and Tom Jobim was selected. Nazareth embodies the 'chorinho' style, characteristic of the beginning of the twentieth century in Brazil. Jobim personifies the 'bossa nova', a rhythm that emerged as a softened variation of the 'samba' in the fifties.

RGeme was configured to create only one Agent (Agent 'A') to which a series of 100 tasks was given as shown in Table 3:

	Listen	Practice	Compose
Listener	50	n/a	n/a
Student	25	25	n/a
Composer	0	0	0

Table 3. Agent's 'A' Goal Matrix

The Evaluation Matrix for the first 50 tasks established the choice of only Nazareth's works. During the following 50 tasks only Jobim works should be chosen, as shown in Table 4:

	Composer	Year begin	Year end
Listener	Nazareth	-	-
Student	Jobim	-	-
Composer	n/a	n/a	n/a

Table 4. Agent's 'A' Evaluation Matrix

"Year begin" and "Year end" can usually be employed in an Evaluation Matrix to define a date interval. In this case they were not specified which means that the algorithm returned all the compositions by Nazareth (in the Listener stage) and Jobim (in the Student stage). In each time period Agent 'A' performed a task consisting of: one Candidate piece of Music was chosen, the Candidate Memes were parsed and the Style Matrix was transformed according with the transformation algorithm. The system generated a new Style Matrix after the accomplishment of each task and all the resulting Style Matrices were logged in the system in order to observe the behaviour of each meme during the interaction processes.

After the completion of the simulation, we observed that, during the first 50 tasks, Agent 'A' learned a total of 57 memes from the music by Nazareth. In the second half of the simulation, 76 memes were learned, which indicates that 19 new memes originated from the music by Jobim. Figure 2 shows the evolution in time of the number of memes that were learned by Agent 'A'.

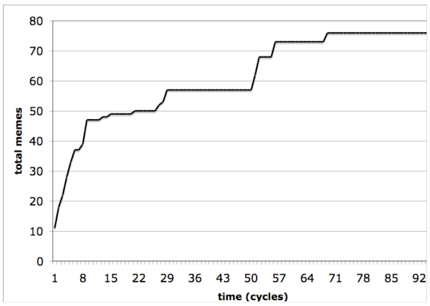

Figure 2. Number of memes learned in time

It was also possible to observe the number of times that each one of the memes were listened to by Agent 'A". The next Figure shows this quantity for the first learned 20 memes.

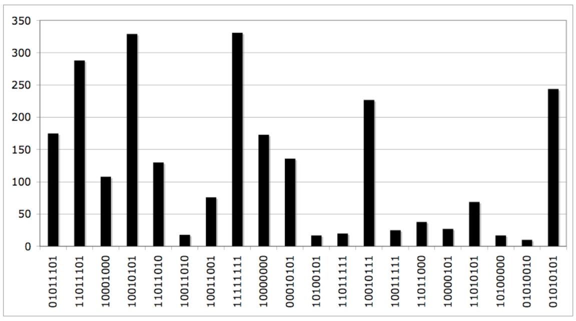

Figure 3. First 20 learned memes: number of listening.

One of the most interesting features that RGeme generates is the track of the evolution of the importance (weight) of each one of the memes during the learning phase of an Agent. The increase or decrease of the importance of the memes is the direct result of the number of times and the date they were listened to and/or practiced. The next Figure shows this analysis during Agent 'A's whole learning phase. Every time a new meme was learned a new line appeared. If a meme was not heard during a certain time, its curve started to fall (forgetting effect).

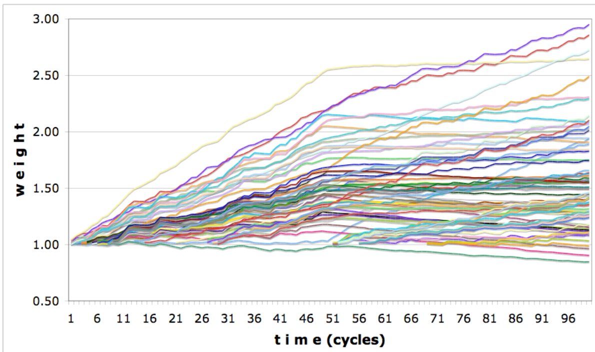

Figure 4. Memes curve of importance in time.

As it is obviously very difficult do visualize the evolution of all the 76 memes in the same graph, in Figure 5 we made a selection of a few of them. Some typical behaviour that emerged from the interactions is described in the paragraphs below.

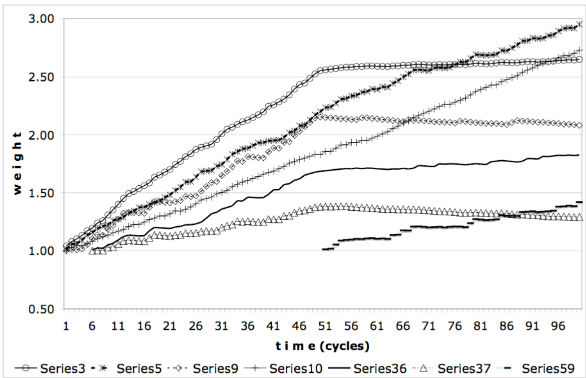

Figure 5. Memes curve of importance in time (selection)

In Figure 5, each series corresponds to the memes described in Table 5:

Series	Seq	dFL	dLL	nL	aW
3	11011101	1	50	288	2.64975
5	10010101	1	100	329	2.95175
9	11111111	1	88	331	2.08238
10	10000000	1	100	173	2.72938
36	01011111	6	93	123	1.8265
37	11110000	6	43	4	1.29287
59	00000001	51	100	12	1.41975

Table 5. Description of memes

Memes 3, 5, 9 and 10 were all first listened to in the first interaction (dFL = date of first listening) with music ‘Arrojado’, by Nazareth. Memes 36 and 37 only appeared in time 6, after Agent ‘A’ listened to music ‘Ameno Reseda’, by the same composer. Meme 59 appeared in time 51 by listening to music ‘A Felicidade’, by Tom Jobim.

Observe that Agent ‘A’ only begun to learn from the rhythms by Jobim at time 50. From this moment on the behaviour of the curve of importance clearly changed for many of the memes (see Figure 5), affecting them positively in some cases and negatively in other cases. For some memes, on the contrary, the change in the Evaluation Matrix didn’t affect too much the previous behaviour.

For instance, although meme 37 begun to be listened to in time 6, its relative importance comparing to the other memes was never very high and even begun to fall after time 50. On the other hand, meme 59, which never appeared in the music by Nazareth, and was only listened to in time 51, at the end of simulation was victorious over meme 37.

Memes 3 and 36 showed a very strong increase during the time in which Agent ‘A’ was listening only to the music by Nazareth. After it begun to listen to the music by Jobim, these memes had only a light increase, meaning that there was a balanced ratio between the number of times they were listened to and the forgetting effect.

During the first half of the simulation the importance of meme 9 had a relative important increase but after time 50 it begun to be forgotten.

At the end of the simulation, some of the memes were definitely winners over others, as Table 6 shows:

Meme	Weight
10010101	2.9517
01011101	2.8581
10000000	2.7294
11011101	2.6497
01010101	2.4907
10010111	2.3096
11010101	2.2926
00010101	2.1459
00000101	2.1012
11111111	2.0824

Table 6: Winning memes

Notice that meme 5 (see Figure 5 above), which was the winner meme above all the others, had a very strong increase from the beginning up the end of the simulation. The next Figure shows its musical representation:

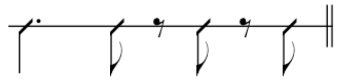

Figure 6: Musical representation of the winner meme

At last, this information will be used later on during the production phase in order to generate new rhythmic material, as mentioned above.

4. CONCLUSION

In this paper we introduced the learning stages of RGeme, an artificial intelligence system for the composition of rhythmic streams.

Besides the production stage that will be covered in a future paper, RGeme has already proved to be an efficient tool to evolve rhythmic worldviews in artificially inhabited environments. Through the description of a simulation we demonstrated how the exposure to different rhythmic material could ultimately shape the musical “knowledge” of an agent.

Experiments are being conducted with different sources of data according to musical genres and styles.

In the future, besides the rhythm information that is being currently employed, the system will deal with more complex musical structures that consider note information (pitches and vertical structures). A better parsing algorithm is being tested in order to extract memes of varied length and a new measure of distance is also being implemented.

5. REFERENCES

[1] Blackmore, S. *The Meme Machine*. Oxford: Oxford University Press. 1999.

[2] Cope, D. *Computers and Musical Style*. Oxford, UK: Oxford University Press, 1991.

[3] Cox, A. The mimetic hypothesis and embodied musical meaning. MusicæScientiæ, V(2):195–212. 2001.

[4] Dawkins, R. *The Blind Watchmaker*. Penguin Books, London, 1991.

[5] Dawkins, R. *The Selfish Gene*. Oxford University Press, 1989.

[6] Dodge, C. and Jerse, T. *Computer Music*. London, UK: Schimer Books, 1985.

[7] Gabora, L. M. The origin and evolution of culture and creativity. Journal of Memetics–Evolutionary Models of Information Transmission [http://www.cpm.mmu.ac.uk/jom-emit/1997/vol1/gabora l.html]. 1997.

[8] Horowitz, D., “Generating Rhythms with Genetic Algorithms”, *Proceedings of the 1994 International Computer Music Conference, San Francisco, ICMA*, 1994.

[9] Impett, J. “Interaction, simulation and invention: a model for interactive music,” in *Proceedings of ALMMA 2001 Workshop on Artificial Models for Musical Applications* (E. Bilotta, E. R. Miranda, P. Pantano, and P. M. Todd, eds.), (Cosenza, Italy), pp. 108– 119, Editoriale Bios, 2001.

[10] Jan, S. “Replicating sonorities: towards a memetics of music”, Journal of Memetics–Evolutionary Models of Information Transmission, 4. 2000.

[11] Jan, S. Meme Hunting with the Humdrum Toolkit: Principles, Problems and Prospects. Computer Music Journal (2004) 28(4):68-84.

[12] Maes, P. *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, MIT Press, 1991.

[13] Manzolli, J., Moroni, A., von Zuben, F. and Gudwin, R. Vox Populi: An Interactive Evolutionary System for Algorithmic Music Composition . Leonardo Music Journal, vol. 10, p. 49-54. San Francisco, USA, 2000.

[14] Miranda, E. R. “Cellular automata music: An inter-disciplinary music project,” Interface (Journal of New Music Research), vol. 22, no. 1, pp. 03–21, 1993.

[15] Miranda, Eduardo R. “The artificial life route to the origins of music”. Scientia, 10(1):5-33. 1999.

[16] Pachet, F. “Rhythms as emerging structures”, *Proceedings of 2000 International Computer Music Conference, Berlin, ICMA*, 2000.

[17] Russell, S.J. and Norvig, P. Artificial Intelligence: A Modern Approach. Prentice Hall, 2002.

[18] Worral, D. “Studies in metamusical methods for sound image and composition,” Organised Sound, vol. 1, no. 3, pp. 183–194, 2001.

[19] Xenakis, I. *Formalized Music: Thought and Mathematics in Composition*. Bloomington (IN), USA: Indiana University Press, 1971.