Emergent Rhythmic Phrases in an A-Life Environment

João M. Martins and Eduardo R. Miranda

Interdisciplinary Centre for Computer Music Research
University of Plymouth, Plymouth, Devon PL4 8AA
United Kingdom.
{joao.martins,eduardo.miranda}@plymouth.ac.uk

Abstract. The A-Life approach to Music is a promising new development. The vast majority of existing A-Life systems for musical composition employ a Genetic Algorithm (GA) to produce musical melodies, rhythms, and so on. In these systems, music parameters are represented as genotypes and GA operators are applied on these representations to produce music according to given fitness criteria. We have identified two methodological limitations of such GA-based systems: one relates to the fact that composition should not be driven by a constant set of fitness criteria and the other is to do with the fact that music is largely a cultural phenomenon driven by social pressure and this is cumbersome to model with standard GA alone. An approach improve this scenario is to build systems with A-Life algorithms designed primarily to address musical issues, rather than using algorithms that were not designed for music in the first place. The work presented in this paper contributes to this line of thought by proposing the design of algorithms that consider music as a cultural phenomenon whereby social pressure plays an important role in the development of musical conventions. We introduce three algorithms: popularity, transformation and complexity algorithms, respectively. The algorithms were implemented in the context of a system for composition of rhythms, where the user can explore their potential to generate rhythmic sequences and also monitor their behavior. Finally, we explore the composition capabilities of the system by using the rhythms developed by the agents during the simulations in a collective performance environment. This bottom-up approach automatically defines an implicit metric structure.

Keywords
Music and A-Life, emergence of rhythms, interactive autonomous agents, social pressure in cultural evolution

1 Introduction

A comprehensive overview of applications of A-Life published recently mentioned that A-Life has been applied in the field of Music and indicated a few examples of A-Life systems for musical composition [1]. Why should musicians be interested in A-Life?

From the discovery almost three thousand years ago of the direct relationship between the pitch of a note and the length of a string or pipe, to the latest computer models of human musical cognition and intelligence, musicians have always looked at science to provide new and challenging paradigms to study and compose music.

The A-Life approach to Music is a promising new development for composers and musicologists alike. For composers, A-Life provides an innovative and natural means for generating musical ideas from a specifiable set of primitive components and processes reflecting the compositional process of generating a variety of ideas by brainstorming followed by selecting the most promising ones for further iterated refinement. For musicologists, A-Life techniques might be used to model the cultural transmission and change of a population's body of musical ideas over time; e.g., to model the development and maintenance of musical styles within particular cultural contexts and their reorganization and adaptation in response to cultural exchange. In both cases, the musical evolution can be influenced by a variety of constraints and tendencies built into the system, such as realistic cognitive and environmental factors that might influence the way in which music is experienced, learned, stored, modified, and passed on between individuals.

The vast majority of existing A-Life systems for musical composition employ a standard Genetic Algorithm (GA) to produce musical melodies, rhythms, and so on. Normally, music parameters are represented in these systems as genotypes and GA operators are applied on these representations to produce music according to given fitness criteria; for a review, please refer to [2].

Because of the highly symbolic nature of Western music notation, music parameters are suitable for GA-based processing and a number of composers, including ourselves, have used such systems to compose music. However, we have identified two methodological limitations of such GA-based systems which may jeopardise further developments in this area: a) fitness criteria are not easy to define when dealing with musical composition and b) music is largely a cultural phenomenon driven by social pressure and this is cumbersome to model with standard GA alone.

The first limitation emerges from the fact that music is not an exact science. For example, it differs from engineering. Whereas the success of a piece of engineering would normally be measured by its ability to match a number of functional requirements effectively, the success of a piece of music cannot be measured so objectively. Indeed, whereas good engineers are praised for following the rules of their métier strictly, good composers (at least in the Western music tradition) are praised for clever violations of musical conventions. Moreover, in most cases, composers do not explicitly know a priory how a new piece of music will sound like until it is completed and indeed performed. Therefore, rather than tools to generate efficient solutions to problems automatically, composers need tools to explore a vast space of possible outcomes.

Biles [3] proposed an interesting approach to implement GA-based systems for the exploration of a space of musical possibilities, which takes into consideration the evaluation of the user; that is, the fitness of each generation is evaluated by the user. This is surely a very interesting idea, but this slows down the compositional process enormously. Biles is aware of this problem, which he refers to as the "fitness bottleneck" problem.

The second limitation is largely related to a problem that is endemic in the field of Computer Music, which is the tendency to design systems to generate music from algorithms that were not designed for music in the first instance. For example, in the late 1980s it became fashionable to implement systems that generated music from fractals [4], but such systems seldom produced significant pieces of music. There was a tendency at the time to overstate the adequacy of fractals for algorithmic composition. In reality, fractals are not appropriate to convey musical information, but appealing images: the eye can grasp an entire image at a fraction of the time needed to grasp even a short sound sequence.

Nowadays, we may be witnessing a similar case of overstatement on the adequacy of GA for algorithmic composition. Although we acknowledge that there have been rather successful stories (e.g. [3, 5]), we believe that additional evolutionary computation methods need to be developed in order to move the field of evolutionary computer music forward.

One way forward is to build systems with A-Life algorithms designed or suitably modified to address musical issues. A-Life methods have been previously used for music composition [6, 7] or to study the evolution of bird songs [8–10].

The work presented in this paper contributes to this line of thought by looking into the design of algorithms that consider music as a cultural phenomenon whereby social pressure plays an important role in the development of musical conventions. A plausible method to embed social dynamics in such algorithms is to design them within the framework of interacting autonomous agents.

In this paper we introduce three algorithms, referred to as popularity, transformation and complexity algorithms, respectively. These algorithms were implemented in the context of a system for composition of rhythms. In this system the user can explore the potential of these algorithms to generate rhythmic sequences and also monitor the behavior of the system. The system offers the ability to extract information about its behavior in many different ways, providing composers the means to explore the outcomes systematically.

Our research is greatly inspired by the work developed by research into gaining a better understanding of the evolution of language with computational models [11–15], particularly the work of Steels [11] on language imitation games with software agents and robots. Basically an imitation game consists of one agent picking a random sound from its repertoire and the other agent trying to imitate it. Then a feedback is given about the success of the imitation. On the basis of this feedback, the agents update their memories.

By way of related research, we cite the work by de Boer [12] on modeling the emergence of vowel systems by means of imitations games. Also, Miranda [16] has developed a variant of de Boer's games in order to model the emergence of intonation systems.

In a previous paper [17] we provided a detailed explanation on the algorithms of interaction that enable repertoires of rhythms to develop. We have also studied the development of complexity of the repertoires and similarity between agents as a result of their behaviours. In this paper we intend to go a step further placing the agents in a group performance, letting the structure of the rhythmic phrases be defined collectively.

2 The Agents

The agents are identical to each other and the number of agents in a group may vary. The agents move in a virtual 2D space and they normally interact in pairs (Figure 1). Essentially, the agents interact by playing rhythmic sequences to each other, with the objective of developing repertoires of rhythms collectively. At each round, each of the agents in a pair plays one of two different roles: the player and the listener. At each interaction, the agents may perform operations on the rhythms that they play to each other, depending on the interaction algorithm and on the status of the emerging repertoire. The agents are provided with a memory to store the emerging rhythms and other associated information.

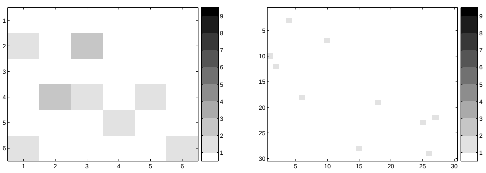

Fig. 1. 2D virtual worlds with different sizes holding 10 agents. A darker color indicates a cluster. (This will be clarified in due course.)

One interesting ability of human beings is that we are able to perceive, and more importantly, to produce an isochronous pulse [18]. Moreover, humans show a preference for rhythms composed of integer ratios of the the basic isochronous pulse [19]. Therefore, rhythms are represented here as interonset intervals in terms of small integer ratios of an isochronous pulse (Fig. 2).

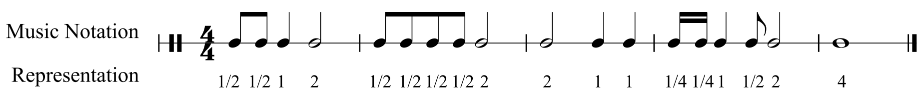

Fig. 2. Standard music notation of a rhythmic sequence and its corresponding interonset representation.

2.1 Transformations of Rhythms

At the core of the mechanism by which the agents develop rhythmic sequences is a set of basic transformation operations. These operations enable the agents to generate new rhythmic sequences and change the rhythmic sequences that they learn as the result of the interactions with other agents. The transformation operations are as follows:

– Divide a rhythmic figure by two (see Fig. 3a)
– Merge two rhythmic figures (see Fig. 3b)
– Add one element to the sequence (see Fig. 3c)
– Remove one element from the sequence (see Fig. 3d)

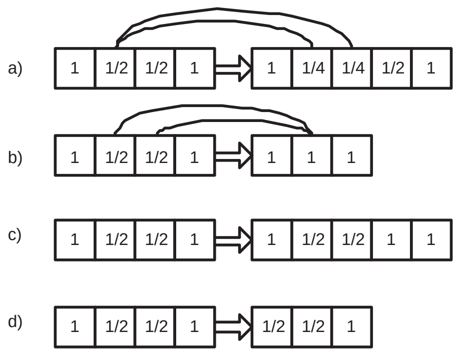

Fig. 3. Examples of rhythmic transformations.

The definition of these transformations were inspired by the dynamical systems approach to study human bimanual coordination [20] and is based on the notion that two coupled oscillators will converge to stability points at frequencies related by integer ratios [21]. Furthermore, common music notation facilitates these types of transformations. We have defined other transformations that divide a figure into three, five, and other prime numbers, but the impact of these additional transformations on the model is beyond the scope of this paper. Addition and removal transformations were introduced to increase diversity in the pool of rhythms and to produce rhythms of different lengths.

3 The Interaction Algorithms and the Experiments

The interaction algorithms and the analysis methods that we have implemented in our system are introduced below. Each algorithm is introduced in the context of illustrative experiments aimed at studying the development of repertoires of rhythmic sequences from three different perspectives:

– From the perspective of an individual agent
– From the perspective of a group of agents, referred to as the society
– From the perspective of the developed rhythms

From the perspective of an individual agent, we studied the development of the size and the complexity of the repertoire of individual agents. From the perspective of the society we averaged values of the corresponding individual measures from the agents, as well as similarity between agents and how they were clustered in terms of the rhythms that they shared. Finally, from the perspective of the developed rhythms, we measured their lifetime, the amount of rhythmic sequences that the society developed and the degree to which the agents shared similar rhythms. We traced the lifetime of a rhythmic sequence by counting the number of agents that possessed this sequence at each iteration. Fig. 4 shows graphs illustrating these various types of analyses.

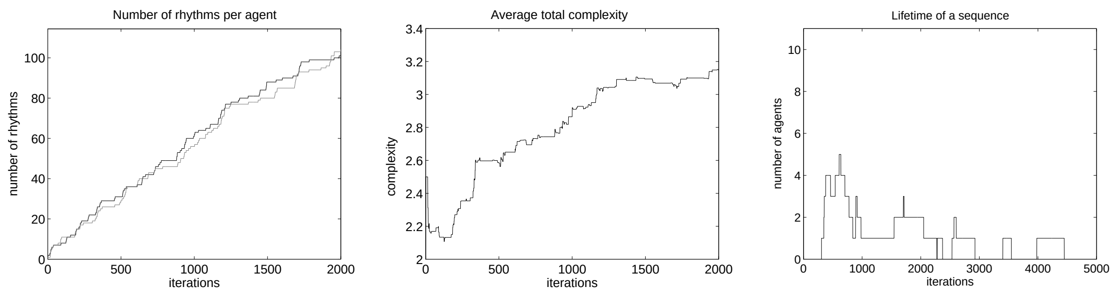

Fig. 4. a) Development of the size of the repertoire for different agents.; b) Complexity of the rhythms of the whole society; c)Number of agents sharing a particular rhythm.

The experiments were run for 5000 iterations each for a number of times, with the objective of observing their behavior under different conditions. We have run experiments with societies of 3, 10 and 50 agents. On some of the experiments we limited the lifetime of the agents to 1000 iterations; when an agent dies, another is born. Sometimes the algorithm considers the movement of the agents in the 2D space, which may or may not influence the nature of the interactions.

In this paper we focus only the results of the popularity algorithm. For a detailed exposition of the results of the other two algorithms please refer to [17].

3.1 The Popularity Algorithm

Popularity is a numerical parameter that each agent attributes to a rhythm in its repertoire. The parameter is modified both by the listener and by the player during an interaction. If the listener recognises the rhythm (that is, if it holds this rhythm in its repertoire), then it will increase the popularity index of this rhythm and will give a positive feedback to the player. A positive feedback is an acknowledgment signal, which will prompt the player to increase the popularity index of this rhythm in its repertoire as well. Conversely, if the listener does not recognize the rhythm, then it will add this rhythm to its repertoire and will give a negative feedback to the player, which will cause the player to decrease the popularity index of this rhythm. Furthermore, there is a memory loss mechanism whereby after each interaction all the rhythms have their popularity index decreased by a small value of 0.05. This accounts for a natural drop in popularity due to ageing of the rhythm. The diagram of this interaction is displayed in Fig. 6a.

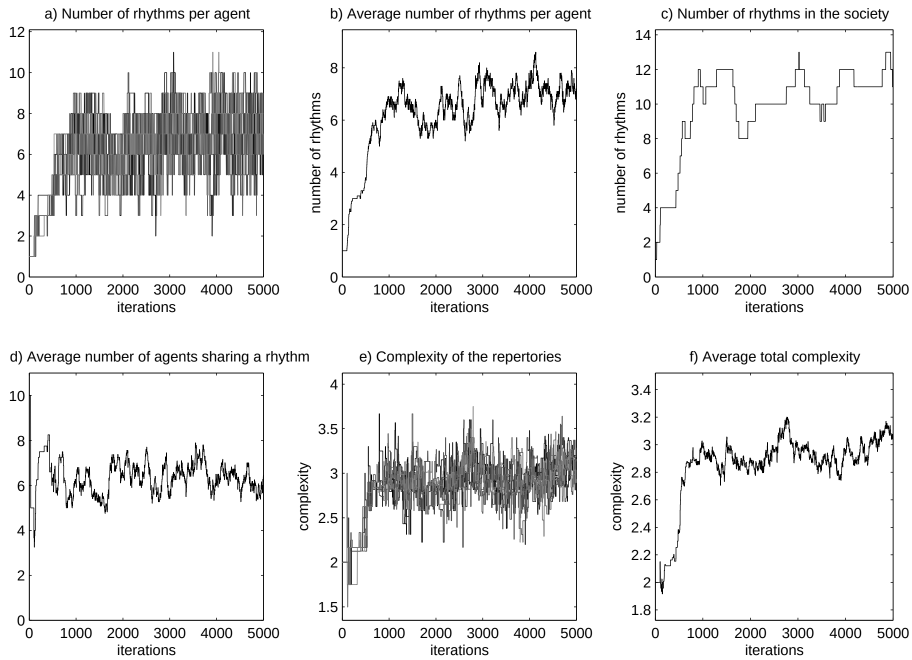

Fig. 5. Results from a typical run of the popularity algorithm with 10 agents.

Fig. 5 shows the results after 5000 iterations of the popularity algorithm without population renewal. Fig. 5a displays the development of the repertoire from the individual agents and the graph in Fig. 5b displays the corresponding average across the agents. Here the repertoire of each agent grows monotonously during 500 iterations and subsequently oscillates around a stable point.

Fig. 5c displays the development of the repertoire of the whole society being a direct consequence of the lifetime of each rhythm. The average number of agents sharing a rhythm (Fig. 5d) is calculated by summing the instant number of agents sharing a rhythm (Fig. 4c) for all rhythms, and dividing the result by the number of rhythms currently present in the society (Fig. 5c). Fig. 5d) provides the means to assess the global behavior of the society; for instance, if it develops coherently in terms of the popularity of existing rhythms.

Fig. 5e represents the development of complexity of the individual agents and Fig. 5f gives the corresponding average. Initially, the size and complexity of the repertoire of individual agents are very close to the average, but this trend is replaced quickly by repertoires of different sizes amongst the agents.

3.2 The Transformation and Complexity Algorithms

As its name suggest, the transformation algorithm (Fig. 6b) applies transformations on a rhythm whenever it is communicated between agents. The motivation behind this algorithm is to foster novelty. We conducted experiments to evaluate the degree to which transformations occurring during the interactions have an impact on the organisation of the emerging repertoire as time progresses.

The diagram of the complexity algorithm is shown in Fig. 6c. With the complexity algorithm we studied the effect of preference for particular types of rhythm; in this case, we wanted to establish whether the agents would show preference for rhythms with identical complexity; we have developed methods to measure this complexity. Here the agents include in their repertoire only those listened rhythms that fall within a window of complexity centered in the average complexity of the rhythms of the listening agent.

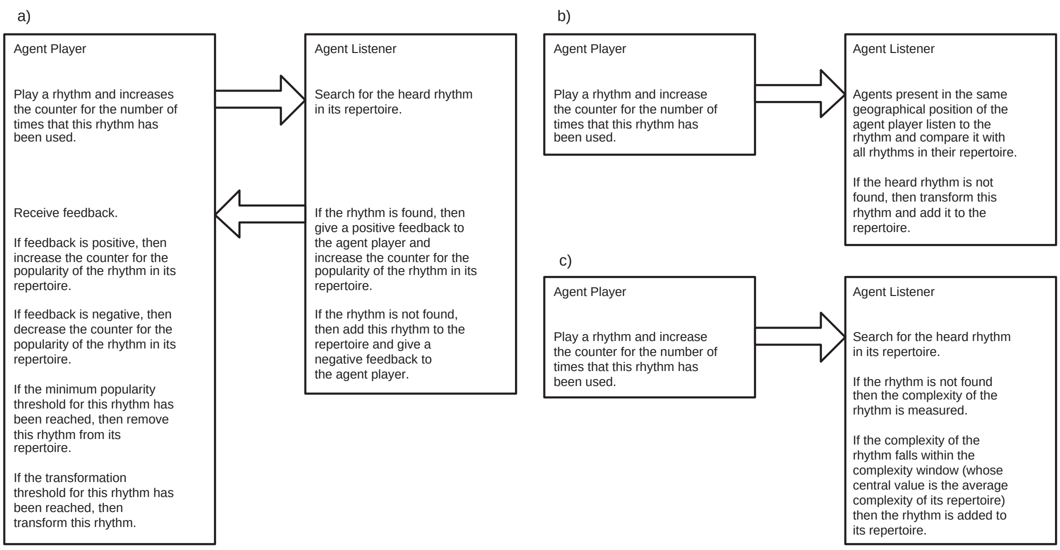

Fig. 6. a) Popularity algorithm; b) Transformation algorithm; c) Complexity algorithm.

4 Rhythmic Phrases in the Social Context

The interaction processes introduced above developed into some interesting behaviours, revealing the dynamics of the representations of the rhythmic units. From a musical point of view, however, the output of the system is not yet satisfactory. This is not surprising because our intention is that these acquired units should be considered as the basic material the agents will use when playing in a synchronised mode. In this section we will demonstrate how the agents can create a rhythmic background texture that will also establish the metric structure of a longer piece.

4.1 Emergent Phrase Length

Most often the rhythmic sections of musical pieces consist of repetitions of small rhythmic units. This fact may have the function of either reinforce or contradict a metric structure. We decided to conduct an experiment where all the agents present in a given geographic position would play their rhythms simultaneously, as opposed to the interaction algorithms presented in Sec. 3. Instead of focusing on the learning process we observed how the agents would play a collective rhythmic piece.

If each agent plays one of the rhythms from its repertoire and repeats it, there will be a strong metrical cue associated with this repetition. The rhythms that belong to the repertoires of the agents may or may not have different lengths due to the transformations (Sec.2.1). When played together there will be an instant where all the agents will hit the initial beat of their basic rhythm at the same time. The difference between two such consecutive instants defines the length of the music phrase.

When the lengths of the basic rhythms are divisible in relation to each other then the length of the longest will define the size of the phrase. In case the length values are not divisible (3:2, 4:3, 5:3,...) the repetitions will generate an interesting polyrhythmic effect.

In a polyrhythm, two or more independent rhythms are played simultaneously. Polyrhythms are particularly abundant in African music, Indian classical music, Cuban music and Jazz. For a more detailed explanation on polyrhythms please refer to Handel [22].

Algorithmically, this can be achieved by finding the least common multiple of the lengths of all the basic rhythms. As an example we let the main rhythmic phrase be composed and each agent will have an assigned rhythmic phrase to compose other rhythmic units from the repertoire. The algorithm is defined as follows:

- Select a basic rhythm from the repertoire.
- Calculate the least common multiple between the lengths of the basic rhythms of all the agents.
- Repeat the basic rhythm across the entire composition, except for its assigned phrase.

– Select from a series of rhythms contained in the repertoire to compose an individual rhythmic phrase.

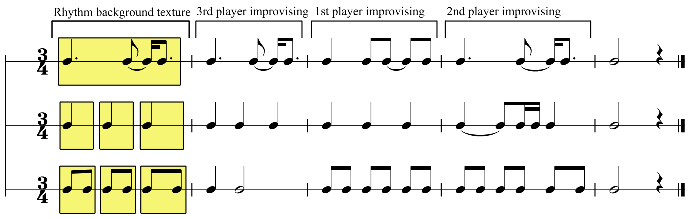

Fig. 7. Rhythmic composition resulting from the performance when three agents meet and play simultaneously.

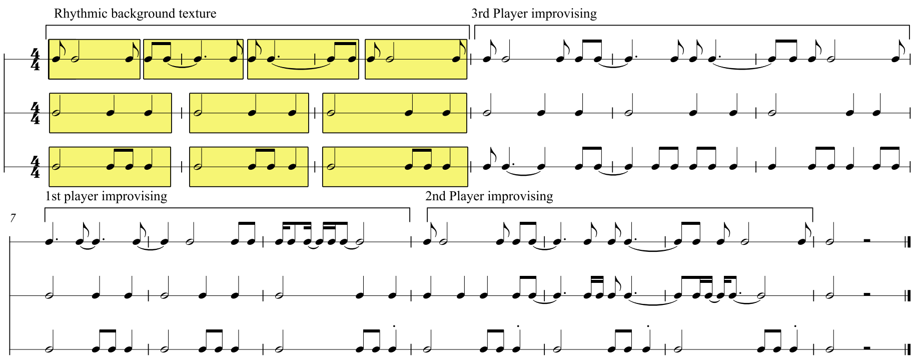

Fig. 8. Rhythmic composition where the phrase length is the result of one repetition of the rhythm from the first agent and two repetitions of the basic rhythms of the other two (polyrhythmic underlying structure).

In Figs. 7 and 8 it is possible to observe the generated score for a group of 3 agents in different stages of the experiments.

5 Conclusion

Most current approaches to musical composition with A-Life entail the application of a standard GA to produce streams of symbols representing musical parameters, such as musical notes. We suggested that one of the main limitations of this approach is that GAs are not entirely adequate for musical composition because they were not designed to address musical problems in the first place; the act of composing music seldom involves an automated selective procedure towards an ideal outcome based on a set of definite fitness criteria. As a way forward, we suggested that musical composition systems may best benefit from A-Life if the algorithms were designed to address specific musical issues. In addition to providing a more realistic music systems, such algorithms may also be useful for building models to study the evolution of music, which would follow up on the research work being conducted in the field of evolution of language [11–15].

In this paper we introduced a few algorithms, which address music as a cultural phenomenon whereby social pressure steers the development of musical conventions (in this case, repertoires of rhythmic sequences).

We also propose an algorithm that enables the agents to create longer rhythmic structures by composition of the rhythmic units that they exchange during the interactions. The algorithm suggests a bottom-up approach to rhythm structure generation. Longer phrases emerge from the usage and repetition of the rhythmic units in a collective context.

While the system is able to produce a great variety of rhythms and coherent rhythmic variations, which is what we had expected to observe in the first instance, the system also displayed a number of interesting and surprising behaviors that beg further scrutiny. We are currently studying the behaviors of these algorithms in order to ascertain whether they could be used to model the way in which rhythms emerge and develop in real societies, e.g. tribal music in Africa.

We are currently experimenting with runs involving agents with different behaviors and with agents that change their behavior during the interactions. We are also conducting experiments where the agents learn from the collective performance environment in order to observe the emergence of composition grammars and new behaviours.

Examples of the rhythms generated by the system, accompanied by a brief explanations of the behaviors that generated them are available at:

http://cmr.soc.plymouth.ac.uk/members/jmartins/research.htm

6 Acknowledgements

The authors would like to thank Simon Durrant and Steve Shaw for their valuable comments on technical aspects of the model. This work is financially supported by The Leverhulme Trust.

References

1. Kim, K.J., Cho, S.B.: A comprehensive overview of the applications of artificial life. Artificial Life **12**(1) (2006) 153–182
2. Miranda, E., Biles, A., eds.: Evolutionary Computer Music, Springer-Verlag London (2007)

3. Biles, J.A.: Genjam: A genetic algorithm for generating jazz solos. In Anderson, P., Warwick, K., eds.: Proceedings of the International Computer Music Conference, Aarhus(Denmark), International Computer Music Association (1994)
4. Mandelbrot, B.B.: The fractal geometry of nature. New York: W.H. Freeman and Co. (1982)
5. Towsey, M., Brown, A., Wright, S., Diederich, J.: Towards melodic extension using genetic algorithms. Educational Technology & Society **4**(2) (2001) 54–65
6. Brown, A.R.: Exploring rhythmic automata. In Rothlauf, F.e.a., ed.: Proceedings of the 3rd European Workshop on Evolutionary Music and Art, Lausanne(Swizerland), Springer Verlag (2005) 551–556
7. Miranda, E., Todd, P.: A-life and musical composition: A brief survey. In: Proceedings of the IX Brazilian Symposium on Computer Music, Campinas,(Brazil) (2005)
8. Todd, P.M., Werner, G.: Frankensteinian methods for evolutionary music composition. In Griffith, N., Todd, P.M., eds.: Musical networks: Parallel distributed perception and performance. MIT Press/Bradford Books, Cambridge (MA), USA (1999) 313–339
9. van den Broek, E., Todd, P.M.: Piep piep piep - ich hab' dich lieb: Rhythm as an indicator of mate quality. In: ECAL03. (2003) 425–433
10. Ritchie, G., Kirby, S.: A possible role for selective masking in the evolution of complex, learned communication systems. In Lyon, C., N.C., Cangelosi, A., eds.: Emergence of Communication and Language, Springer (2007) 387–401
11. Steels, L.: A self-organizing spatial vocabulary. Artificial Life **2**(3) (1995) 319–332
12. de Boer, B.: Self-Organisation in Vowel Systems. PhD thesis, Vrije Universiteit Brussel AI-lab (1999)
13. Kirby, S.: Natural language from artificial life. Artificial Life **8**(2) (2002) 185–215
14. Cangelosi, A., Parisi, D., eds.: Simulating the Evolution of Language. Springer Verlag, London (2002)
15. Vogt, P.: Lexicon Grounding on Mobile Robots. PhD thesis, Vrije Universiteit Brussel (2000)
16. Miranda, E.R.: Mimetic development of intonation. In: Proceedings of the 2nd International Conference on Music and Artificial Intelligence (ICMAI 2002), Springer Verlag - Lecture Notes on Artificial Intelligence (2002)
17. Martins, J.M., Miranda, E.R.: Breeding rhythms with artificial life. (Submitted to Journal) (2007)
18. Handel, S.: Listening: An Introduction to the Perception of Auditory Events. The MIT Press (1989)
19. Drake, C., Bertrand, D.: The Quest for Universals in Temporal Processing in Music. Ann NY Acad Sci **930**(1) (20) 17–27
20. Kelso, J.A.: Phase transitions and critical behavior in human bimanual coordination. Am J Physiol Regul Integr Comp Physiol **246**(6) (1984) R1000–1004
21. Beek, P.J., Peper, C.E., Daffertshofer, A..: Timekeepers versus nonlinear oscillators: how the approaches differ. In: Rhythm Perception and Production. Swets and Zeitlinger (2000) 9–34
22. Handel, S.: Using polyrhythms to study rhythm. Music Perception **1**(4) (1984) 465–484