M.: Ants can play music

Conference Paper *in* Lecture Notes in Computer Science · September 2004

DOI: 10.1007/978-3-540-28646-2_29 · Source: DBLP

Christophe Guéret
Accenture Labs

Nicolas Monmarché
University of Tours

Ants can play music

C. Guéret, N. Monmarché, and M. Slimane

Laboratoire d'Informatique de l'Université de Tours
École Polytechnique de l'Université de Tours - Département Informatique
64, Avenue Jean Portalis - 37200 Tours - France
{nicolas.monmarche,mohamed.slimane}@univ-tours.fr

Abstract. In this paper, we describe how we can generate music by simulating moves of artificial ants on a graph where vertices represent notes and edges represent possible transitions between notes. As ants can deposit pheromones on edges, they collectively build a melody which is a sequence of Midi events. Different parameter settings are tested to produce different styles of generated music with several instruments. We also introduce a mechanism that takes into account music files to initialize the pheromone matrix.

1 Introduction

Science and music have a long common history made of mutual interactions [19]. Many examples can be given to show how music can lead to new perceptions or theories of scientific realities or how scientific researchers can improve music performance or composition.

This paper investigates more precise relations between music and science with computer generated music and Artificial Intelligence (AI) from the swarm intelligence point of view. We will demonstrate how the collective behavior of ants can lead to a compositional system. As far as we know, artificial ants have never been used to generate music and this paper is the first attempt to address this type of method. Real ants are known to produce sounds like crickets do but we will be interested in this paper by their collective behavior such as their capacity to build paths between their nest and a food site in their environment.

After a little introduction to ant algorithms and AI methods that have been used to generate music, we will present the AntMusic project and give some details about the algorithm and its parameters. Finally, we will present results as far as we can do it on paper before investigating future directions.

1.1 Little things about artificial ants

Ants have recently been used as a new kind of inspiration for computer science researchers and many successful works deal with combinatorial optimization (see a review in [4]). For most of the cases, a global memory is used to guide the search agents toward promising solutions as real ants spread volatile substances, known as pheromones, on their path leading to food sources. In this case, artificial pheromones are real values that are used with a positive feedback mechanism which reinforce promising solutions. We can also notice that as not all of the ant species use pheromones, for instance, we have shown that their artificial translation can exploit other capabilities to solve optimization [16] or clustering problems [15]. All these collective problem solving methods belong to the emerging "swarm intelligence" research field.

We have already used artificial ants to produce another kind of computer generated art stuff: artificial paintings [1]. In this application, ants are used to spread colors on a painting and compete against each other to make their color dominant. A interactive genetic algorithm have been used to find the best parameters (i.e. colors and ants properties) from the user point of view.

1.2 Generating music with AI

Artificial Intelligence (AI) has been used to conceive computer music systems since the beginning of computer science (see a review in [13]). Several models have been proposed to capture musical knowledge such as finite and infinite automata [11], neural networks [5, 8], analogical reasoning [13], constraint programming [10], multi agent systems [22] or Evolutionary Computation (EC) (see for instance [18] and [21, 2] for a larger review).

These systems can be classified in three fields: compositional, improvisation and performance systems. In the following, we will focus on the two first types since the last one deal with producing artificial realistic performances. In the former category, we can cite the pioneering work of Hiller and Isaacson's (in 1858) where notes are pseudo randomly generated by Markov chains and selected accordingly to classical harmony and counterpoint rules. In the EMI project [6], the style of various composers was emulated by searching patterns in several compositions of a given author. In the later category, neural networks are used for instance to learn how to improvise jazz solos [9] or with an interactive genetic algorithm [3]. Interactions with the user can also be integrated for instance as an agent by the way of a Midi instrument [22], by the evaluation of melodies in an interactive genetic algorithm [18] or by the body gestures [17].

In this paper, we will be interested by using the collective behavior of ants to produce music. Todd and Miranda [20] has proposed that there are three ways to generate music with an artificial life system: (1) music can be an expression of the movement of agents which are not aware of what they produce. In this case, music can be considered as a representation of the artificial world. (2) Each individual produces music and its survival depends on it. This approach belongs to evolutionary algorithms technics. (3) Agents produce music that has an influence on other agents' behavior. According to this classification, we will demonstrate that our system belongs to the third category: music is the result of multiple social interactions of ant like agents.

2 The AntMusic project

2.1 Artificial ants on the graph of notes

In the AntMusic project, we use artificial ants to build a melody according to transition probabilities and taking advantage of the collective behavior of marking paths with pheromones. Artificial ants are agents that are located on vertices in a graph and can move and deposit pheromones on edges. The more ants choose an edge, the more pheromones will be present and the more other ants will choose this edge. As mentioned in section 1.1, this general principle has been exploited for combinatorial optimization problems like the well known Traveling Salesman Problem (TSP). One of the first algorithm designed to deal with this problem was called Ant System (AS) [7] and we have adopted in this project the same principles. In our case, the vertices correspond to Midi events (note, duration,...) and a melody corresponds to a path between several vertices. Each edge (i, j), between vertices i and j, is weighted by a positive pheromone value τ_{ij} (see figure1).

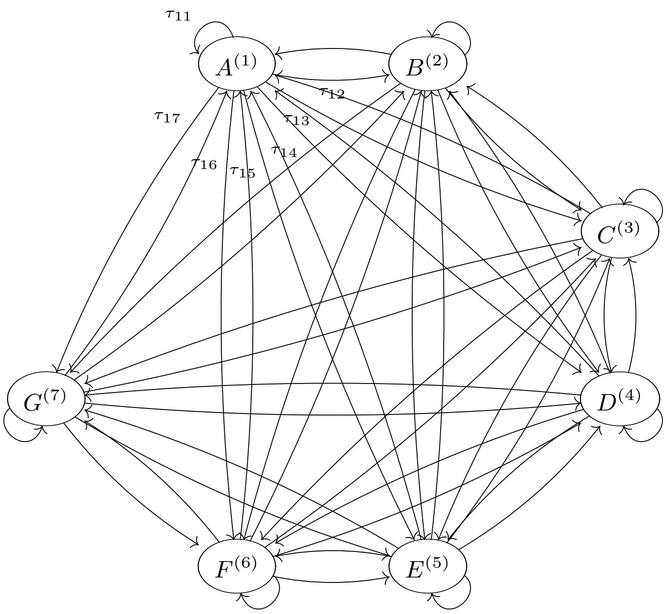

Fig. 1. Example of graph with 7 notes: $A^{(1)}, \ldots, G^{(7)}$ (numbers are indices) on which ants can move (pheromones have only been represented when moving from note A).

A transition rule is used to decide which of the vertices an ant will choose when located in vertex i: the probability of choosing vertex j is given by:

$$p_{ij} = \frac{\tau_{ij} \times \eta_{ij}^{\beta}}{\sum_{k \in N_i} \tau_{ik} \times \eta_{ik}^{\beta}} \tag{1}$$

where η_{ij} is a numerical value representing the desirability and which is set to $1/(d(i,j)+1)$ with $d(i,j)$ representing a distance (in our case it corresponds to the number of half tones between the two notes i and j but we could use another distance). This desirability encourages ants to choose closest notes from their current position to form a melody. The parameter β controls the influence of the style: high values increase the importance of the desirability and low values give more importance to the pheromone values for the transition choices between notes. Finally, N_i stands for the set of possible vertices that can be reached from i.

Each time an ant moves from a vertex i to a vertex j, it deposits a pheromone quantity τ_0:

$$\tau_{ij} \leftarrow \tau_{ij} + \tau_0 \tag{2}$$

Finally, as it occurs in natural systems, pheromones slowly evaporates :

$$\tau_{ij} \leftarrow (1-\rho) \times \tau_{ij} \tag{3}$$

where ρ is a parameter called evaporation ($0 \leq \rho \leq 1$).

For one ant, the algorithm that simulates its movements on the graph and generates a melody of $T_{\max}$ notes is the following:

1. Initialization:
   (a) randomly choose a vertex for the initial position of each ant
   (b) initialize pheromone values: $\tau_{ij} \leftarrow \tau_0 \quad \forall i,j$
   (c) $t \leftarrow 0$
2. while $t < T_{\max}$ do
   (a) choose the next Midi event j according to the transition rule (formula 1)
   (b) deposit pheromones on chosen edge (formula 2)
   (c) move to vertex j and play the corresponding Midi event
   (d) $t \leftarrow t+1$

The same algorithm can be used simultaneously for several ants in order to compose several voices for the selected instrument. We can notice that it is not necessary that all ants play the Midi event they encounter. This can be useful for instance to build quickly a path in the graph: a few ants play the instrument while a large number collectively build the melody and remain silent. Finally, the evaporation step is performed regularly and independently from the ants movements. Pheromones evaporation is useful in optimization methods because it allows the search not to be kept in local minima. For the compositional system we have conceived, it allows the generated melody to evolve during long runs.

The main underling principle that we emphasize in this system is that a musical melody is often identified by the human mind as a repetition of notes and rhythms in the same way which is presented in [14]. So, the reinforcement performed when an ant lays down pheromones can be considered as an encouragement to repeat cycles in the graph.

2.2 Additional features

Several instruments. As many levels of complexity can be added to the system, we have introduced several instruments that can be played at the same time. In this case, we only have to build one graph for each desired instrument and different ants can move independently on these graphs in the same way that has been described before.

Limiting the graph size. The graph that contains Midi events can be very large. In order to decrease its size, we can use any Midi file to initialize the graph: the file is scanned to build a graph that only contains the Midi events that are present in the file. With this method, it is possible to obtain a melody which sounds like the music contained in the Midi file since only these events can be played. Moreover, pheromones can be initialized according to the Midi event transitions that can be found in the file. In this case, ants will not only use the same events found in the file, they will also have a tendency to build a melody that sounds like the melody contained in the Midi file.

2.3 Additional parameters

In order to let the user define his/her will, for each instrument, the following parameters can be independently set (default values are indicated within brackets):

- instrument: one of the 128 Midi instruments (1: piano)
- minimum and maximum values of notes (0–127)
- volume (0–127)
- length of the melody: $T_{\max}$ (25)
- number of ants: silent (0) and playing ones (1)
- style: parameter β (1.0)
- possible durations: 2, 1, 1/2, 1/4 (1)
- quantity of pheromones laid down by ants: τ_0 (0.1)
- evaporation parameter: ρ (0.01)

3 Results

As the generated music is saved in Midi files we present in this section the scores that correspond to the obtained files[1].

The score of the figure 2 has been generated with the following parameters: one instrument (piano), 3 playing and 10 silent ants, 50 and 70 as minimum and maximum notes, $\beta = 1.5$, $T_{\max} = 15$, possible durations are 1-1/2-1/4, $\tau_0 = 0.1$ and $\rho = 0.01$. We can observe that 3 voices can be found (3 notes can be played at the same time) because we have used 3 playing ants.

[1] The corresponding Midi files can be listen on this url: `http://www.hant.li.univ-tours.fr/webhant/index.php?pageid=39`

Fig. 2. Example of score obtained with 3 playing ants (see the text for other parameters).

The score of the figure 3 has been generated with the following parameters: two instruments (two pianos), for the first one: only 1 playing ant, 60 and 80 as minimum and maximum notes, $\beta = 2.0$, $T_{\max} = 15$, possible durations are 1-1/2-1/4, $\tau_0 = 0.1$ and $\rho = 0.01$. For the second piano: 2 playing and 5 silent ants, 40 and 65 as minimum and maximum notes, $\beta = 1.5$, $T_{\max} = 15$, possible durations are 2-1-1/2, $\tau_0 = 0.3$ and $\rho = 0.01$. This example shows how it is

Fig. 3. Example of score obtained with 2 pianos (see the text for other parameters).

possible to use two instruments to obtain two independent voices for the same instrument: the first piano plays in a high range of notes (60-80) whereas the second one plays in a low range (40-65).

The last example in figure 4 shows a score obtained with three instruments: 2 pianos and 1 strings.

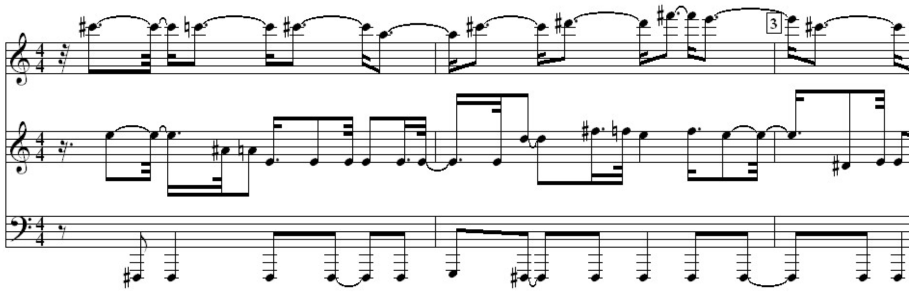

Fig. 4. Example of score obtained with 3 instruments (piano and strings).

We have noticed that the parameters β (weight for desirability) and ρ (evaporation) have an evident impact on the generated music: the former allows us to build melodies without any wide gap between notes (the style of the music is smoother) and the latter allows us to control how often the sequences of notes are repeated. The influence of this last parameter also depends on the number of ants on the graph, either playing or silent ones.

4 Conclusion

The compositional system presented in this paper is inspired of the natural behavior of ants that build paths using pheromones. Their natural ability to build paths in their environment has been exploited to make paths between notes, that is building melodies.

The results can only be evaluated more regarding to the strategy presented in this paper than from earlier composing and elaboration technics.

Many directions remain to be explored. For instance, in its current version, the system can produce melodies for different instruments simultaneously but without any harmonization between them. This could be addressed by introducing harmonization rules that would reduce the sets N_i at the transition step. Moreover, the style variation can only be influenced by the β parameter and its influence only concerns the distance between notes. Several ways can be proposed: we can add a Midi entry which could be used by a a midi instrument musician or by a given midi file to influence the pheromone values either as an initialization step or as an interactive and real time system.

We have also worked on a system that generates paintings according to the movements of ants (in this case, pheromones correspond to colors) [1]. We plan to merge the two systems to build a compositional systems with a graphical aesthetic view of the process as this type of experiments can be found in [12].

Finally, one drawback of this compositional system is its large number of parameters and consequently the huge number of possible melodies. To improve this point, we need to help the user to find easily the parameters that will produce a music he/she appreciates. One possible way is to use an interactive genetic algorithm to perform an intuitive and very subjective exploration of this huge search space.

Acknowledgment

The authors would like to thanks the computer science department students R. Conchon and V. Bordeau for their contribution to this work.

References

[1] S. Aupetit, V. Bordeau, N. Monmarché, M. Slimane, and G. Venturini. Interactive Evolution of Ant Paintings. In *IEEE Congress on Evolutionary Computation*, volume 2, pages 1376–1383, Canberra, 8-12 december 2003. IEEE Press.

[2] P. Bentley and D. Corne, editors. *Creative Evolutionary Systems*. Morgan Kaufmann, 2001.

[3] J. Biles. GEMJAM: a genetic algorithm for generating jazz solos. In *Proceedings of the International Computer Music Conference*, pages 131–137, San Francisco, 1994. International Computer Music Association.

[4] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, 1999.

[5] C. Chen and R. Miikkulainen. Creating melodies with evolving recurrent neural networks. In *Proceedings of the 2001 International Joint Conference on Neural Networks (IJCNN-2001)*, pages 2241–2246, 2001.
[6] D. Cope. Pattern matching as an engine for the computer simulation of musical style. In I. C. M. Association, editor, *Proceedings of the International Computer Music Conference*, pages 288–291, San Francisco, 1990.
[7] M. Dorigo, V. Maniezzo, and A. Colorni. The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 26(1):29–41, 1996.
[8] D. Eck and J. Schmidhuber. Finding temporal structure in music: Blues improvisation with LSTM recurrent networks. In H. Bourlard, editor, *Proc. of IEEE Workshop on Neural Networks for Signal Processing XII*, pages 747–756, 2002.
[9] J. Franklin. Multi-phase learning for jazz improvisation and interaction. In *Proceedings of the eighth Biennal Symposium on Art and Technology*, New London, Connecticut, 1-3 March 2001.
[10] M. Henz, S. Lauer, and D. Zimmermann. COMPOzE — intention-based music composition through constraint programming. In *Proceedings of the 8th IEEE International Conference on Tools with Artificial Intelligence*, pages 118–121, Toulouse, France, Nov.16–19 1996. IEEE Computer Society Press.
[11] T. Johnson. *Self-Similar Melodies*. Edition 75, 1996.
[12] V. Lesbros. From images to sounds, a dual representation. *Computer Music Journal*, 20(3):59–69, 1996.
[13] R. Lopez de Mantaras and J. Arcos. AI and music, from composition to expressive performance. *AI magazine*, 23(3):43–57, 2002.
[14] M. Minsky. Music, Mind and Meaning. In S. Schwanauer and D. Levitt, editors, *Machine Models of Music*, pages 327–354. MIT Press, 1993.
[15] N. Monmarché, M. Slimane, and G. Venturini. On improving clustering in numerical databases with artificial ants. In D. Floreano, J. Nicoud, and F. Mondala, editors, *5th European Conference on Artificial Life (ECAL'99)*, volume 1674 of *Lecture Notes in Artificial Intelligence*, pages 626–635, Swiss Federal Institute of Technology, Lausanne, Switzerland, 13-17 September 1999. Springer-Verlag.
[16] N. Monmarché, G. Venturini, and M. Slimane. On how *Pachycondyla apicalis* ants suggest a new search algorithm. *Future Generation Computer Systems*, 16(8):937–946, 2000.
[17] R. Morales-Manzanares, E. Morales, R. Danenberg, and J. Berger. SICIB: An interactive music composition system using body movements. *New Music Research*, 25(2):25–36, 2001.
[18] A. Moroni, J. Manzolli, F. Von Zuben, and R. Gudwin. Vox populi: An interactive evolutionary system for algorithmic music composition. *Leonardo Music Journal*, 10:49–54, 2000.
[19] R. Root-Bernstein. Music, creativity and scientific thinking. *Leonardo*, 34(1):63–68, 2001.
[20] P. Todd and E. Miranda. Putting some (artificial) life into models of musical creativity. In I. Deliege and G. Wiggins, editors, *Musical creativity: Current research in theory and practise*. Psychology Press, 2003.
[21] P. Todd and G. Werner. Frankensteinian Methods for Evolutionary Music Composition. In N. Griffith and P. Todd, editors, *Musical Networks: Parallel distributed perception and performance*. MIT Press/Bradford Books, 1998.
[22] R. D. Wulfhorst, L. V. Flores, L. Nakayama, C. D. Flores, L. O. C. Alvares, and R. M. Viccari. An open architecture for a musical multi-agent system. In *Proceedings of Brazilian symposium on computer music*, 2001.