Real-time Musical Interaction between
Musician and Multi-agent System

Daichi Ando, Msc.
Department of Frontier Informatics, Graduate School of Frontier Sciences,
The University of Tokyo, Japan.
e-mail: dando@iba.k.u-tokyo.ac.jp

Prof. Hitoshi Iba, PhD.
Department of Frontier Informatics, Graduate School of Frontier Sciences,
The University of Tokyo, Japan.

Abstract

The application of emergent behavior of multi-agent system to musical creation, such as controlling parameters of sound synthesizer and composition, has attracted interest recently. Although human control or programmed operation works properly, it is very complicated and seems overly monotonic. One of the features of a multi-agent system, self-organization, is suitable for controlling parameters of synthesizer and generating compositional rules. Furthermore, the system has the possibility to generate unexpected sounds and musical pieces in ways that an experienced musician would never try to generate.

In this paper, we report a research on a musical computer system, which generates synthesizer sounds and musical melodies by means of the multi-agent and interacts with human piano players in real-time. We show empirically that our interactive system plays attracting sounds. We also demonstrate that a human player feels that the interaction between the system and himself is very reliable.

1. Introduction

The problem of techniques, deterministic algorithmic composition made by only strict rules which succeeded to traditional composition is that no possibilities that can generate unexpected besides wonderful results. One of the purpose for using computers in musical creation is to generate unexpected results that cannot be generated with traditional ways. Therefore stochastic composition techniques, using random values generated by computer as parameters of composition, has been widely used. L. Hiller and L. Isaacson composed a suite named "Iliac Suite" with very early computer "ILLIAC" using the famous stochastic technique Monte-Carlo [1], and I. Xenakis generated peculiar sound called "Sound Cloud" in his pieces by stochastic technique with computer[2]. C. Roads collected famous stochastic techniques and gave a detailed explanation of them in his book[3].
Although, ordinary stochastic composition using random values simply has a problem. The problem is that the output results from computers cannot be used for composition directly without the revision of composers or programming very strict rules for random value generation. Because of that each random output value has no relationship to each other. In general, correlations between each parameter are important in composition and sound synthesis. Thus directly using output for composition generates sometime inaudible sounds, also sometime no unbearable as a music, and often not interesting sounds.

Consequently, application of an "Self-Organization" of Multi-Agent systems, such as Cellular-Automata (CA) and Artificial Life (AL, ALife), in musical creation has lately attracted considerable attention. The relationship of each parameter is important in musical creation, as mentioned before. The self-organization property bears the possibility of generating correlated parameters. In general, emergent behaviors of CA can control parameter sets for composition and sound synthesis dynamically whereas eliminating the need for the revision of computer outputted values and manual arrangement. Furthermore, recent research shows that traditional composition models of classical music can be explained with the behaviors of multi-agent systems.

Ordinary stochastic techniques used today, which involves applying random values to each musical or sound synthesis parameters, usually produces non-musical notes or inaudible sounds. Therefore composer should revise the output values from computer or put strict restriction rules when generating random values in order to obtain applicable results. In contrast, self-organization property applied to music composition and sound synthesis reduces the possibility of generating chunk data in musical terms.

Some composers and researchers have tried to apply emergent behaviors of Multi-Agent system to composition and sound synthesis. P. Bayls and D. Millen attempted to map CA's each cell state to pitch, duration and timbre of musical notes[4,5]. E. Miranda noticed self-organizing functions of CA strongly, Then he constructed the composition and sound synthesis software tools called CAMUS and Caosynth[6,7]. In the Caosynth system, he developed mapping to apply CA to controlling granular sound synthesis. P. Dahlsted has been interested in behaviors of evolution of natural creatures. He made some simulation systems consisted of many creatures which play sounds, walk, eat and interacts with each other, even die and evolve in world. Finally developed mapping for the behaviors of creatures to sounds and composition[8,9]. He also has used IEC (Interactive Evolutionary Computation) actively for his composition based on the point of views of multi-agent evolution system[10-12].

2. Interaction between live computer system with human players

As we have mentioned in section 1, application of Multi-Agent systems to musical creation has yielded promising results. The self-organization function is very useful for controlling compositional and sound synthesis parameters dynamically. Dynamic sounds alternation is required especially in sound synthesis of recent live computer music which generates melodies and sound in real-time. Besides, techniques of dynamic controlling of parameters to generate interesting melodies and sounds in interactive live computer music, such as capturing human
player's sound or performance information and computing interactive melodies and sounds, has become increasingly important in recent years. This is due to the new sound synthesis techniques, such as granular synthesis and granular sampling, which require huge sets of parameters, and also melody generation in computers. It is very difficult to apply random values to these parameters directly in real-time manually, because that many case pure random parameter sets generate unmusical sounds with complicated synthesizer and composition algorithms. An ideal type of parameter set is that each parameter alternates in time line and bears a strong correlation to each other. Therefore self-organization of each parameter works very effectively when applied to sound synthesis and melody generation in real-time.

Meanwhile, musical trials of real-time interaction between human players and computer system in musical works has also drawn considerable attention. Human's musical performance contains a huge amount of information, e.g. dynamics of articulation, tempo alternations in a melody and etc. Dynamical construction of sounds and melodies based on the captured musical information has been the typical way of creating live interactive musical works. These techniques are very exciting from the viewpoint of both performer and composer. However, programming of rules for sound and melody synthesis part is very complicated for real-time interaction. Because of the musical information captured highly depends on the condition and the mood of the performer. Also sound and melody synthesis require huge parameter sets which have strong interdependence, besides dynamic alternation of parameters is needed to make unique sounds, but system operator is not allowed to control the parameters in detail. We thought that realizing musical interactions utilizing self-organization function helps the real-time parameter control, moreover performer and composer can get more interesting and musical sounds by means of self-organization of multi-agent system.

We mentioned above, some musical research and works done in the area of multi-agent systems. In these works, processing of multi-agent system and composition or sound synthesis is non real-time, results from the fact that processing cost of multi-agent system is large, so that its not feasible to process in real-time. However, in recent years, processing power of computers has increase dynamically. For 20 years ago, we could not imagine that real-time sound processing in multi-purpose processors with very small portable computers is achievable. And also, today we get the processing power to simulate any kinds of simple multi-agent system such as small CA and ant colony. Thus, now we have many possibilities to realize interaction between computer system utilizing self-organized functions and human performers.

For these matters, mentioned above, we tried to construct a multi-agent system which interacts with human player in musical terms. Then we observed interesting musical communication that the power of self-organization of multi-agent system and human players in live piece.

3. System Construction

3.1 Overview of the system

Our system consists of two computers, first one is for running the multi-agent system, and the next is for composition and sound synthesis. Two computer are connected via Ethernet with
"OpenSound Control" protocol[13]. This enables a simple setup for the realization of a distributed processing environment. On a single computer there is the possibility of interference since the execution of multi-agent system requires high computational power which may lead to generation of noise or interruption in the sound synthesis task. OpenSound Control is widely used protocol for real-time live music creation, and in order the system to connect to other live computer systems.

For the realization of the multi-agent system, we adopted "Swarm Libraries"[14]. Swarm Libraries is a software package for multi-agent simulation of complex systems, originally developed in Santa Fe Institute. we introduced networking function with OpenSound Control to Swarm Libraries, then implemented functions to send and receive musical messages from other software.

For composition and sound synthesis, "Max" clones, "Max/MSP" and "PureData" was used. Max and Max clones are powerful software for real-time sound synthesis and control algorithmic composition. Also we have attempted to introduce other sound software to our system.

The order of processing is as follows:

1. Performance information of human player is captured with Max as sound signal or MIDI data.
2. The information is analyzed with rule sets that composer and performer programmed, then send to the next computer which runs the multi-agent system via network.
3. Multi-agent system, such as CA include the messages extracted from performance information, change states of each agent in the virtual world based on contents of the message.
4. At predetermined intervals, return information of agents states to music computer via network.
5. Music computer processes melodies or sound synthesis with informations from multi-agent system, and publish with the speaker or send to an other instrument connected via MIDI.

Fig.1. Distributed processing environment of our system
2.2 Examples of multi-agent behaviors, generating melody and sound synthesis rules

Purpose of this research is an attempt to apply multi-agent system to actual musical creation. Multiple behavioral rules for the multi-agent system are adopted in our system instead of restricting ourselves to a single rule. The idea behind this is that in practice composition of a musical piece accommodates several rules in sound and melody generation. Many algorithms are used for melody generation and sound synthesis while composing a piece in order to give a feeling of dynamic movement and represent a developing musical structure.

For generating behavioral rules for the multi-agent system in use, several rules are implemented such as 1D, 2D CA, Boids and Ant Colony. Boids is a simulation model for mimicking the movement of flocks and herds in which the behavior for each member of the group is governed by simplistic rules. On the other hand, Ant Colony simulates the behavioral models of and colonies for foraging tasks.

Fig. 2: Multi-agent system implemented. The 2D CA rule Conway’s Game of Life is running.
In order to map the behaviors emerged in the simulations to musical melody generation and sound synthesis several mappings are produced. For instance, in the case of Boids simulation, each agent's behavioral exposition and location data are connected to a set of a white noise oscillators and a band pass filter. Specifically, agent's behavior, lateral position in X line, vertical position in Y line and moving speed are mapped to the three parameters of band pass filter, center frequency, bandwidth and position between the left-right channels, respectively. Moreover, during the musical performance, the frequency of key presses of piano is adjusted based on the agents moving speed. This enables a dynamically alternating sound cluster. Another metric employed in this approach is the distance between any two agents which is used for stretching the musical note created by the agents.

4. Composition and Performance of a Piece

4.1 Composition

We conducted an experiment to compose a piece with the proposed method. The piece is named "Fellow fr MIDI piano and live interactive computer system."

The procedure of the composition is as follows: At first, we composed 6 very short pieces consisting of 16 or 32 bars and few notes using predetermined note sets with our software for composition by means of Interactive Evolutionary Computation (IEC) named "CACIE" [15]. Then a professional pianist is requested to actually play the computer generated piece with full articulations and the performance information is extracted as a MIDI file during the performance. In the next step, notes generated by the multi-agent system are sprinkled into pre-composed piece taking the MIDI file as input which is obtained in the previous step. Multi-agent system is build using "Conway’s Game of Life" based on simple CA rules with fixed mappings. These fixed mappings uses the predetermined note sets to give a feeling of unity for the audience and performers. Finally, we rearrange a final piece using all the previously generated pieces by IEC and the multi-agent system considering an artistic sense.

![MIDI sheet music image]

**Fig. 3:** A piece generated by the interaction between multi-agent system and performance of pianist.
4.2 Performance

Next, we explain the performance example with our system. In previous section, composition of the piece with our system was explained. We tried to perform the piece as a live interactive musical piece with our system.

Setting of the live interactive system for the performance is similar setting as mentioned section 3.1 and 4.1, MIDI piano is connected to musical processing computer. Input of the system is the pianist's performance information as MIDI signal, and outputs of the system are synthesized sound and played piano sequences which is generated in real-time with MIDI piano driven by MIDI signal from the system. Non-musical instruments were not used, such as a foot switch or MIDI sliders in the system. In this setting we tried to assure that interaction between the system and the performer is only in musical terms.

For the behavior of the multi-agent system, few 2D CA rules and Boids were adopted along with several mappings for melody and sound generation. These rules of multi-agent behavior and mappings for sound generation were designed to change based on the section of a piece. In addition we tried to extract some heuristics from the performance information to dynamically evolve the mapping and rules.

![Fig. 4: Pianists play the piece with multi-agent system](image)

As a result of the interaction between the pianist performer and our system, performances such as four bars exchange in jazz were observed. There were precise musical correlations between the melodies played by the pianist and the system. Despite the simple mappings used in this
experiment, no inaudible sounds were encountered during the composition and performance steps. Furthermore we received positive feedback from the professional performer.

5. Conclusion

In this paper we conducted a research about a multi-agent system for musical creation which interacts with human players in composition and performance stages. As a proof of concept, using our proposed system we composed and performed a piece interactively with a professional pianist. As a result, the piece created received positive feedback from the performer.

References


[13] OpenSound Control Homepage (http://www.cnmat.berkeley.edu/OpenSoundControl/)
[14] Swarm Homepage (http://www.swarm.org/)