

人間の演奏者と音楽インタラクションを行う マルチエージェントシステムの構築

安藤 大地† 伊庭 斉志†

† 東京大学大学院新領域創成科学研究科基盤情報学専攻

自己組織化などの特徴を活かした、マルチエージェントシステムの作曲や音響合成などの音楽創作への応用が、近年盛んに試みられている。また、計算機の処理能力向上にとともに、リアルタイムでマルチエージェントシステムと音響合成、旋律生成などの処理を行うことが可能になってきた。そこで、マルチエージェントシステムを応用し、人間の演奏者とリアルタイムに音楽的なインタラクションを行うライブインタラクティブコンピュータシステムを構築の構築と、実際にシステムを利用したコンサート作品の創作を行い、その有効性を確認した。

Constructing a multi-agent system which interacts with a human player in real-time manner

Daichi Ando† Hitoshi Iba†

† Dept. of Frontier Informatics, Graduate School of Frontier Sciences,
The University of Tokyo.

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 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 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 はじめに

1.1 過去の作品例

古典的な作曲手法を引き継いだ決定論的作曲手法を用いたアルゴリズム作曲は、厳密なルールを用いた作曲が可能であるという利点があるが、アルゴリズム作曲ならではの「人間が創造できない音楽」をつくり出すことは不可能である。そこでコンピュータによる

アルゴリズム作曲の利点を活かした、乱数を用いた確率論的作曲手法が使われるようになった [6]。Xenakis は「音の雲」と呼ばれる音響効果を独自の確率論的音楽 (Stochastic Music) 手法を用いて作り出している [10, 11]。

しかしながら、従来試みられてきた単純な乱数を用いた確率論的作曲手法では、コンピュータが出力した結果はそのままでは曲として成り立たないことが多く、作曲家が出力を任意に修正するか、もしくは、どのよ

うな乱数列でも音楽、音響として成り立つような厳密なルールをプログラミングする必要があった。

そのような中で、Artificial Life(人工生命、Alife, AL)やCellular-Automata(CA)を始めとするマルチエージェントのシステムを音楽創作に応用する試みが、脚光を浴びてきている。例えばCellular-Automataの特徴のひとつとして自己組織化がある。E. Mirandaらは、CAの自己組織化をマッピングすることにより、従来では困難だった音楽の旋律や音響を生成するための多数のパラメータを容易にコントロール可能であると主張している[3, 8]。

従来の個々の乱数を音楽もしくは音響の各パラメータに割り当てる方式では、実際には音楽や可聴な音響になることがまれであり、乱数の生成のルールを厳しくするか、コンピュータの出力した結果に作曲家の手を加える必要があった。この状態では確率論的手法の最大のメリットである、人間が創造できない音楽が生まれるという特徴が薄くなってしまふ。そこでCAのような複数のArtificial Lifeをマルチエージェントシステムとし、システムの振るまいを作曲や音響合成にマッピングする方法を用いることが提案された。

CAを用いた自動作曲システムでは、各セルの状態を音の高さ、音の長さ、音色の選択に応用したP. Beyls[1]とD. Millen[7]の例がある。この時作曲家は、各セルの状態からピッチ、音の長さ、音色にマッピングするルールを用意することになる。

音響合成にCAを用いた例としては、グラニュレーションセンシス用のパラメータを生成したP. Bowcott[2]の例、Self-modifying 波形を生成したJ. Chareyron[4]の例などがある。また前述のBeylsは、加算合成のための振幅エンベロープを発生させ、それを作曲に応用している。

このような音楽創作にCAを応用した諸研究はE. Mirandaらによってまとめられている[3, 8]。さらにMirandaはさまざまな数学的な手法を盛り込み、マルチエージェントシステムにより作曲や楽音合成を行った作品を多く創作している[9]。CAMUSシステムは2状態と4状態の二つのCAを重ねあわせたものから、ごく短い旋律を生成する。生成された旋律を1ステップごとに並べていくことで、変化をともなった旋律の繰り返しという典型的な音楽の展開が得られたとMirandaは報告している。またCaosynthシステムは、CAをグラニュレーションセンシスに用い、一つのGrainを構成する16のサイン波のオシレータを統合的に制御している。Mirandaの作品において、このマッピング方法は非常に有効に働き、単純な乱数を使った並べ替えによるグラニュレーションセンシスでは得られなかった音響が得られたと報告されている。

またALによるマルチエージェントシステムを音楽創作に用いた例としては、P. DahlstedのLiving Melodies[5]がある。同作品はALのアルゴリズムの音楽創作への応用のポテンシャルの発見を試みたアー

トプロジェクトである。それぞれが音響を生成する複数のALモデルを世界に存在させ、リアルタイムに音楽的なコミュニケーションを取りながら進化させる試みを行い、興味深い音響を生成することに成功している。

1.2 ライブ作品でのリアルタイムなインタラクション

このような背景をもとに、著者らはマルチエージェントシステムの自己組織化を用い、人間の演奏者とマルチエージェントシステムが、リアルタイムに音楽的なインタラクションを行うシステムを構築し、コンサート作品を創作できないかと考えた。

近年のコンピュータの処理能力の向上に伴って登場してきた、リアルタイムに音響合成や旋律の生成を行い演奏する作品では、音響や旋律のダイナミックな変化が重要視されている。しかしながら、前述の通り、音響合成や作曲にしようするパラメータセットは非常に巨大で、リアルタイムに演奏する作品において、オペレータが制御することは非常に難しい。

リアルタイムに演奏する作品の中でも、人間の演奏者とコンピュータのインタラクションにより音響を生成するライブインタラクティブ作品では、人間の演奏者の演奏情報を解析し、音響合成や作曲のパラメータセットにアサインし、音響のダイナミックな変化を得ることができているが、解析された演奏情報は実際には乱数に近く、かなり厳密なフィルタを通さなければ使用することができない。アサインするべきパラメータの数が多ければ多い程この傾向は強くなり、演奏者の演奏が強く反映されたインタラクションであるとは言いがたい状況になる。

そこで、マルチエージェントシステムの自己組織化機能を用いた演奏情報のフィルタリングを行うことで、演奏者の演奏を巨大なパラメータセットの生成に利用すれば、インタラクション性が高まると考えた。

マルチエージェントシステムの実行には、非常に高いコンピュータの処理能力が必要であり、従来ではマルチエージェントシステムをリアルタイムに実行しながら音合成を行うのは難しかった。しかしながら、リアルタイム音響合成と同じく近年はその問題は解決されてきており、実装方法や音響や旋律への変換マッピングを工夫することにより、マルチエージェントシステムと音響、旋律の合成の双方を連動させながらリアルタイムに実行することが可能となった。

今回構築したシステムは、リアルタイムに、演奏者の情報を入力に取り音響信号の生成や旋律の生成を行うものである。

2 実装

2.1 概要

構築したシステムは、二台のコンピュータから構成されている。一台はマルチエージェントシステムを実行するコンピュータで、もう一台は旋律合成と音響合成や演奏情報の解析などを担当するコンピュータである。このような分散型のシステムを構築した理由は、特に音響合成とマルチエージェントシステムのタスクは、共に計算負荷が非常に高いことにある。一台のコンピュータで双方を実行した場合に、リアルタイムで生成され演奏される音響が途切れてしまう可能性があり、ライブ作品としては無視できない障害になると考えられる。そこで、二台のコンピュータを準備し、双方を Ethernet を用いて接続し相互通信を行うことで、この問題を解決した。

相互通信のためには、前述の通り Ethernet と “OpenSound Control”¹ というプロトコルを用いた。一般的に音楽用の通信としては、コンピュータの環境に依存しない MIDI が使用される事が多いが、MIDI は設定が複雑でありコンサート時のセッティングでトラブルが多発しやすいこと、近年 OpenSound Control を実装した音楽用アプリケーションが多く搭載していることを踏まえ、今回は OpenSound Control を採用した。これにより、システムのマルチエージェントパートは、様々な音楽用アプリケーションと連動して動作することが可能になった。

マルチエージェントシステムの実装には、“Swarm” ライブラリ²を採用した。Swarm は Santa Fe 研究所で開発されたマルチエージェントのシミュレーションや複雑系システムなどを容易に構築することができるライブラリパッケージである。著者らはこの Swarm ライブラリに前述の OpenSound Control を用いた通信機能を追加し、他のソフトウェアからの音楽のためのメッセージの送受信を可能にした。

旋律生成や音響合成、演奏情報の分析用としては、Max/MSP や PureData などの Max クローンを用いた。

図 1 に典型的な接続の例を示す。またミキサと音響信号の代りに MIDI を用いることも可能である。処理の流れは以下のようなになる。

1. 図 1 の左に示されるミキサから入力された、人間の演奏者の演奏の音響信号、または MIDI による演奏情報の信号が、図中中央に示される音楽担当のコンピュータに取り込まれる。
2. 音楽担当のコンピュータで演奏情報の抽出や解析が行われる。この情報は任意のフォーマットで、

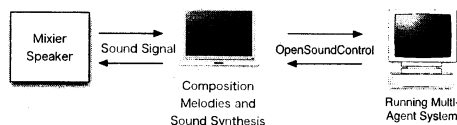


図 1: Distributed processing environment of our system.

OpenSound Control を用い図中右に示されるマルチエージェントシステムに送られる。

3. マルチエージェントシステムは演奏情報を基に処理を行い、決められたインターバルの後の世界の状態の結果を音楽用コンピュータに返信する。
4. 音楽担当コンピュータは予め決められたルールを用いて、音響信号や旋律を生成し、ミキサ、もしくは MIDI 信号として楽器に送信する。

2.2 マルチエージェントシステムのルールとマッピング

実際の音楽生成にマルチエージェントシステムを用いる場合、システムの振るまいを規定するルールが一種類の場合、曲中で使用される旋律や音響に統一感が生まれるという利点があるが、その反面、曲中のセクションの変化やダイナミクスの変化、楽曲の構造の展開をつけることが難しいという欠点がある。そのような理由から、本システムではマルチエージェントシステムの振るまいを規定するルールを複数実装した。また同様の理由で、マルチエージェントシステムの状態を旋律、音響に変換するマッピングも、各振るまいのルールに対して複数用意している。

また、音へのマッピングは、なるべく位置やピッチがダイナミックに変化するものを選んで実装している。

現時点で実装しているマルチエージェントシステムの振るまいのルールは、いくつかの 1D と 2D の CA ルール、Boid、Ant Colony である。Boid は鳥の群れなどの動きのシミュレーションであり、一方 Ant Colony は蟻の群れが餌を集めるシミュレーションである。

音への変換のマッピングは、Boid の場合、各個体にホワイトノイズオシレータとバンドパスフィルタ接続し、各固体の位置と移動スピードをバンドパスフィルタのセンター周波数とバンド幅、パンニングにアサインしている。これにより、音の固まりが音場を移動しつつダイナミックに音を変化するような音響を得られた。解析された演奏情報は、各個体の移動スピード

¹<http://www.cnmat.berkeley.edu/OpenSoundControl/>

²<http://www.swarm.org/>

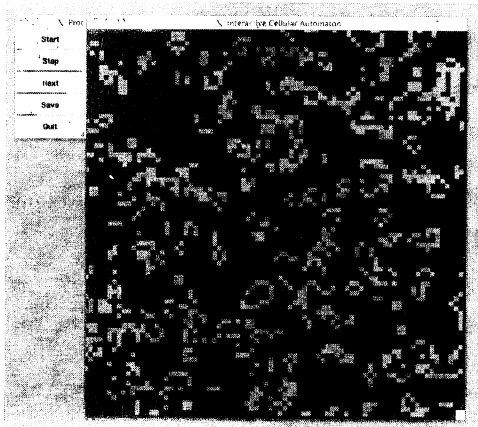


図 2: Multi-agent system implemented. The 2D CA rule Conway's Game of Life is running.

を強制的に変化させるなどの方法で入力されている。また旋律の生成のマッピングとして、任意の二つの個体間の距離を音符のピッチにアサインする方法を実装している。

一方 Ant Colony は入力された音のピッチを餌の出現位置に、各個体には Boid と同様にホワイトノイズのオシレータとバンドパスフィルタのを接続しパンとピッチに、それぞれアサインすることにより、例えばピアノの一鍵盤を弾いた時に、その周辺にノイズが段々集まってくるというような効果を得ることが出来る。

CA の場合のマッピングは、主に旋律の生成に使用し、前述の Miranda のマッピングを使用したり、単純なリズムパターンへのマッピングなどを行っている。

図 2 は 2D の CA を本システムで実行した表示画面である。ルールは “Conway ライフゲーム” を使用している。

3 作曲と演奏

著者らは、構築した本システムの有効性を確かめるため、“Fellow for MIDI piano and live interactive computer system” というピアノの曲を作曲し、演奏した。セッティングは前述した図 1 を拡張し、音楽用コンピュータに MIDI インタフェースを取り付け、そこに MIDI 信号を送受信できるピアノを接続したものである。

本システムは、作曲の過程と演奏時の双方で使用している。

3.1 作曲

まず、本システムを用いた作曲について説明する。処理の流れは、以下の通りである。

1. 少ない音数から構成される 16 小節から 32 小節の短い曲を 6 つ生成した。生成には著者らによる、対話型進化論的計算を用いた作曲支援システム CACTE[12] を用いた。
2. 生成した各曲をピアニストにアーティキュレーションをつけて演奏してもらい、それぞれ演奏情報を採取した。演奏には MIDI 出力ができるピアノを用い、演奏情報は MIDI ファイルとして演奏をそのまま用いた。
3. 採取した MIDI ファイルから予め決められたルールを用いてフィルタリングを行い、結果をマルチエージェントシステムに取り込んだ。
4. マルチエージェントシステムを一定のインターバルの間の処理を行い、結果の状態を出力する。出力結果には入力した曲に合わせてクオンタイズをかけた。
5. 得られた結果の 6 つの旋律と 1. で生成した 6 つの短い曲を用いて著者らによる再作曲を行い、最終的な一つの曲を得た。

図 3 に、ピアニストの演奏を入力に取った時のマルチエージェントシステムの出力結果を示す。結果は原曲の変奏のような形になっている。従来のアルゴリズム作曲のような厳密なルールを設定しなくても、入力の変奏が得られることで、マルチエージェントシステムの自己組織化機能が有効に働いていることを確認した。

なお、使用した演奏情報の抽出ルールや旋律への変換のマッピングは入力した 6 つの曲ごとに違うものを使用している。

3.2 演奏

次に作曲した曲の演奏について説明する。システムの構成は、前述のように図 1 のシステムに MIDI を送受信できるピアノを接続したものである。フットスイッチやスライダーなどの楽器以外のスイッチは一切使用しておらず、また演奏時のシステムのオペレータも、曲のセクションの切替えなどの必要最低限の操作のみを行い、スライダーなどでリアルタイムにパラメータを変更するなどの操作は行っていない。そのため、ピ



図 3: A piece generated by the interaction between multi-agent system and performance of pianist.



図 4: Pianist play the piece the with multi-agent system.

アニストと本システムは純粋に演奏情報のみでインタラクションを行うことになる。

マルチエージェントシステムの結果から生成された旋律は、MIDI 信号でピアノに送信され自動演奏される。演奏情報の採取と解析、旋律と生成と演奏はリアルタイムに行われる。またピアニストの演奏情報の入力に対して、変奏の関係にある旋律を出力するルールを用いたため、結果として、ジャズにおける四小節交換のようなインタラクションが、ピアニストと本システムの間に成立し、興味深い演奏が得られた。また今回演奏を依頼した職業ピアニストの評価として、非常にポジティブな感想を得た。図 4 は演奏中のピアニストを示したものである。

用いたマルチエージェントシステムのルールは、いくつかの 2D の CA と Boid である。また、それぞれのルールに対して、複数の旋律生成と音響合成のマッピングを用意し、曲中でこれを切替えて使用している。

4 おわりに

本稿では、人間と音楽的なインタラクションを行うマルチエージェントシステム構築し、実際に作曲と演奏に応用した事例の報告を行った。厳密なルールを用いずに巨大なパラメータセットを制御することが可能なマルチエージェントシステムの自己組織化機能は、非常に有効に働いており、様々な作品への応用が期待できる。

今後の展開として、マルチエージェントシステムの各個体のコミュニケーションを強くしたり、演奏中にリアルタイムに進化し演奏を生成するなどの、人工生命の要素をより多く盛り込んだシステムと作品創作を行っていきたいと考えている。

参考文献

- [1] P. Beyls. The musical universe of cellular automata. In *Proceedings of the 1989 International Computer Music Conference*, pp. 34–41, San Francisco, September 1989.
- [2] P. Bowcott. Cellular automata as a means of high level compositional control of granular synthesis. In *Proceedings of the 1989 International Computer Music Conference*, pp. 55–57, San Francisco, September 1989.
- [3] D. Burraston, E. Edmonds, D. Livingstone, and E. R. Miranda. Cellular automata in midi based computer music. In *Proceedings of the 2004 International Computer Music Conference*, pp. 71–78, Miami, September 2004.
- [4] J. Chareyron. Digital synthesis of self-modifying waveforms by means of linear automata. *Computer Music Journal*, Vol. 14, No. 4, pp. 25–41, 1990.
- [5] P. Dahlstedt and M. G. Nordahl. Living melodies: Coevolution of sonic communication. *Leonardo*, Vol. 34, pp. 243–248, 2001.
- [6] L. Hiller and L. Isaacson. *Experimental Music*. MacGraw-Hill, New York, 1959.
- [7] D. Millen. Cellular automata music. In *Proceedings of the 1990 International Computer Music Conference*, pp. 314–316, San Francisco, September 1990.
- [8] E. R. Miranda. *Composing Music with Computers*. Focal Press, 2001.

- [9] E. R. Miranda. At the crossroads of evolutionary computation and music: Self-programming synthesizers, swarm orchestras and the origins of melody. *Evolutionary Computation*, Vol. 12, No. 2, pp. 137–158, 2004.
- [10] I. Xenakis. Elements of stochastic music. *Gravesaner*, Vol. 18, pp. 84–105, 1960.
- [11] I. Xenakis. *Formalized Music*. Indiana University Press, Bloomington, 1971.
- [12] 安藤大地, P. Dahlstedt, M. G. Nordahl, 伊庭斉志. 対話型 gp を用いたクラシック音楽のための作曲支援システム. 芸術科学会論文誌, Vol. 4, No. 2, pp. 77–86, 2005.