

# インタラクティブな関係性における音楽表現のための システム構築についての考察 —人間とコンピュータの多様な関係性において—

大野 雅夫

## Abstract

In the previous paper, I studied how computers can participate with humans more actively, and created a program that runs under the following two main relationships between humans and computer. Based on this relationship, several works were produced and analyzed. Finally, I mentioned the future possibility of the interactive relationship between a computer performer and a human performer. In this paper, I constructed a system for musical expression in the interactive relationship between a human and a computer, focusing on three roles of computer:

- ① A computer as a musical instrument
- ② A computer as a tool for executing a program (thoughts by a computer)
- ③ A computer as a performer

In order to use a computer as a musical instrument, I constructed a system that uses Max to output the MIDI note number of the harmonic overtone by the bang message. Furthermore, to recognize the motion of the human hand and send a bang message wirelessly to a computer, I used the R-IoT module developed by collaboration between IRCAM and PluX. In order to use the computer as a tool for executing the program, several systems were constructed. When displaying the score on Max, I used Max's external library "bach". In order to make the computer play a role as a performer, I built a system, using Max's external library "Antescofo". Through this research, I mention the possibility of musical expression by computer calculation, which is difficult for human beings.

キーワード……コンピュータ インタラクティブ Antescofo Max

## 1 はじめに

### 1-1 本論文の位置付け

筆者はこれまで、人と Antescofo を用いたコンピュータ(Max)との関係性におけるコンピュータの果たす役割として、「伴奏者」よりも積極的に人間との音楽演奏に関わる「共演者」としての役割を果たさせるための方策について、作曲上の音楽的表現における人の思考と身体性のコ

コンピュータを用いた模倣について研究を行い、実際の作品の創作を通して考察を行った。また、考察を通して、コンピュータ同士によるアンサンブルの音楽の表現手段としての可能性、並びに、コンピュータには無い人間の演奏のズレに着目した MIDI における身体性の再現が、人間の演奏者と演奏者としてのコンピュータとのインタラクティブな関係性における音楽作品制作に繋がる可能性について示した(大野 2017)(大野 2018)。

本研究においては、大野(2018)において課題として示された、人間と共演者としてのコンピュータとの、音楽におけるインタラクティブな関係性においてシステムを構築するという創作上のアイデアを基に、コンピュータに3つの役割を担わせることにより、Max を用いたシステムとして“Response”を構築することに成功した。人間と共演者としてのコンピュータとの、音楽におけるインタラクティブな関係性の成立について、実際の“Response”のシステムの製作を通して、コンピュータが担う3つの役割に主眼を置き、システムの分析、研究を行う。本稿は大野(2017)からの継続研究である。

本研究においてコンピュータが担う3つの役割は、以下の3つである。

- ① 楽器としてのコンピュータ
- ② 計算(思考)するプログラムを実行するツールとしてのコンピュータ
- ③ 演奏者としてのコンピュータ

#### 1-2 コンピュータによる人間の演奏の模倣と人間の思考との関係性について

演奏者としてのコンピュータによる演奏は、人間の演奏の場合において考えられる身体的・技術的制約を受けず、共演者としてのコンピュータの存在を考えたとき演奏の不自然さを招くだけでなく、人間の演奏者とのコミュニケーションを困難にする(大野 2017)。堀内、藤井、田中(1995)も言及しているように、人間では有り得ないような正確すぎる機械的な楽譜の再生は、再現芸術における再現性の演奏者の裁量の範囲を考慮しても不自然であり、人間とコンピュータとがインタラクティブな関係性を構築する上で、コンピュータの演奏が人間らしい自然な演奏になるよう考慮するという事は、人間の演奏者とのコミュニケーションの成立のために必要な要素であると考ええる。

しかし、本稿におけるコンピュータによる人間の演奏の模倣は、あくまでも MIDI のノートナンバーが `makenote` オブジェクトに入力され、`noteout` オブジェクトから MIDI のデータが実際に出力される(演奏される)段階においてであり、それ以前の計算を行う(思考する)段階においては、人間の思考について模倣するといったことは一切行わない。なぜなら、本稿においては、コンピュータに、人間には瞬時に計算し得ない計算(思考)をあえてさせることにより、コンピュータにしか果たし得ない計算機としてのコンピュータの役割を担わせることが目的であるからである。

次に、本稿においてコンピュータが果たす役割について詳述する。

## 2 コンピュータが担う 3 つの役割

### 2-1 ①楽器としてのコンピュータ

コンピュータは楽器としての役割も果たす(清水、大野 2015)。今回はこの楽器としてのコンピュータを、演奏者としての人間と演奏者としてのコンピュータが共用する。

この楽器は bang の入力に反応して MIDI のメッセージを出力する。演奏者としてのコンピュータは同じ Max 上で機能しているので、bang の入力にももちろん問題はないが、人間が楽器としてのコンピュータに bang メッセージを送るためには何らかのインターフェイスが必要となるので、今回は人間の腕の動きを bang として送信するのに適したセンサーモジュールとして、写真 1 に示されている R-IoT を用いた。R-IoT の開発者の一人である IRCAM の Emmanuel Flety は、R-IoT について、デジタル変換を行うワイヤレスのセンサーで、モーションセンシング、ジェスチャー認識、ライブパフォーマンスアートを行う際の重要なツールであると説明している(Emmanuel Flety 2017)。また R-IoT は、IRCAM と PluX によって共同開発されている。

今回の筆者のパッチにおいては、R-IoT のジェスチャー認識の機能を用い、以下の順序で bang メッセージを楽器としてのコンピュータに送信している。

1. 演奏者としての人間が R-IoT を持った手を指揮者のように振り下ろす。
2. OSC のテクノロジーを用い、無線 LAN を介してセンサーのデータをコンピュータに送信する。
3. Max がそのエクスターナルライブラリである MuBu を用いてデータを解析する。
4. 楽器としてのコンピュータのパッチに bang メッセージを送信する。



写真 1. R-IoT モジュール

(出所) “A comprehensive guide to using, programming & flashing the BITtalino R-IoT WiFi sensor module” (2017) より

## 2-2 ②計算(思考)するプログラムを実行するツールとしてのコンピュータ

今回コンピュータによる思考を司る主なプログラムとして、以下のパッチが挙げられる。

- ・マイクを通して音を分析しピッチを検出するパッチ
- ・コンピュータの演奏を Antescofo を用いてスコアフォローイングを行うパッチ
- ・人間の演奏の際の bang の間隔を順番に記録するパッチ
- ・記録した人間の演奏の bang の間隔の順番を入れ替えるパッチ
- ・人間の演奏を MIDI としてリアルタイムに記録するパッチ
- ・記録された MIDI のデータを楽譜のデータに変換し、アルゴリズムによって処理を行うパッチ

これらの中で楽譜としてのデータを扱うパッチがあるが、Max には本来この処理を行う、楽譜のデータを扱う機能がない(簡易的にピッチのみ楽譜として表示するオブジェクトは存在する<sup>1)</sup>)。そのため、楽譜を用いた Max 上での処理のため、エクスターナルライブラリである bach を用いた。

### 2-3 bach とは

Andrea Agostini らの論文において言及されているように、bach はリアルタイムに処理を行う Max 上において、高度な楽譜表記のための作曲家のヘルパーとして機能する(Agostini, Daniele Ghisi 2012)。

今回の筆者のパッチにおいては、以下の処理を通して、記録した人間の演奏者による演奏に、bach を用いたアルゴリズムによる処理を行っている。

1. MIDI として記録した人間による演奏のデータを bach.roll オブジェクトを用いて、プロポーショナル・ノーテーション<sup>2)</sup>の表記の楽譜に変換する。
2. bach のライブラリを用いたアルゴリズムによる処理を行う。
3. bach.quantize オブジェクトを用いてクオンタイズを行い、bach.score オブジェクトを用いて、クラシカル・ノーテーションの表記の楽譜に変換する。
4. bach.playkeys オブジェクトを用いて、MIDI として bach.score の楽譜を再生する。

### 2-4 ③演奏者としてのコンピュータ

1-2 において述べたように、演奏者としてのコンピュータが人間の演奏に対する返答として演奏を行う際、人間の演奏を模倣し、人間らしい自然な演奏になるよう考慮する。そのための方策として以下の2つのパッチを構築した。

- ・演奏者としてのコンピュータによる、Antescofo を用いた楽器としてのコンピュータを演奏するパッチ
- ・bach.playkeys オブジェクトを用いて、MIDI として bach.score の楽譜を再生する際に、演奏にズレを生じさせるパッチ

この 2 つのパッチは、人間による演奏を受け、コンピュータが計算(思考)した後コンピュータ側が演奏する際に機能する。このパッチによる演奏の後、これを受けた人間による演奏が開始するが、この対話のシステムについて次に詳述する。

### 3 3つの役割を果たすコンピュータと人間とが対話するシステムの概要と分析

#### 3-1 “Response”のシステムの概要

筆者は、人間と共演者としてのコンピュータとの音楽におけるインタラクティブな関係性の構築という創作におけるアイデアに基づき、Max を用いたシステムとして“Response”を制作した。“Response”は、人間による演奏とコンピュータによる演奏とが交互に繰り返され、その行程は図 1 に示すとおり閉じた円環になっており、その順序を幾度となく繰り返す仕組みとなっている。

このシステムは、人間が手に持ったセンサーによる信号の入力により、楽器としてのコンピュータの MIDI による演奏が始まり、それに呼応するコンピュータが、人間の演奏の拍の数によって場合分けの計算を行い、その結果によって 2 つの思考回路から 1 つの思考回路を選び、人間の演奏に反応するための計算を行う。その後の演奏者としてのコンピュータによる MIDI の演奏をもって、円が一巡する。

以下の項において、図 1 の順序に則り、その細部の構成について分析を行う。なお今回 MIDI の出力先としては、Ableton の Live10 を使用しており、それが考慮されたパッチの構造となっている。

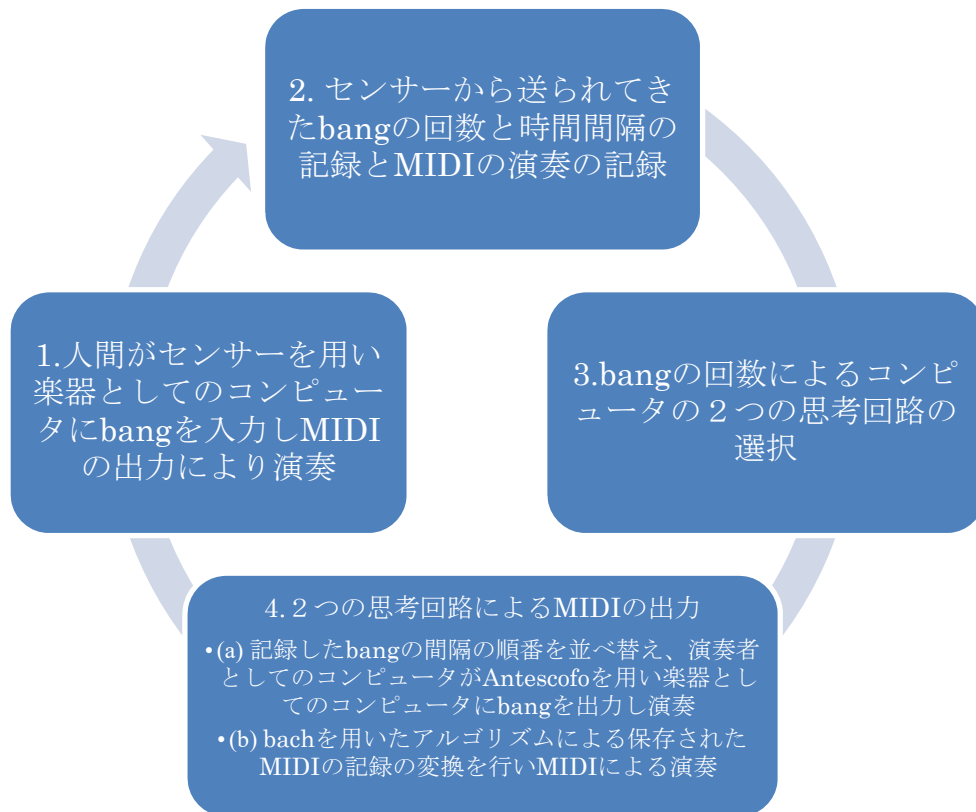


図 1. 人間とコンピュータによる呼応の過程が循環する“Response”のシステムの全体図  
(筆者による作成)

### 3-2 センサーから送られてくる bang に対応して楽器としての Max が演奏するシステム

図 2 に示されているように、この楽器によって演奏される MIDI のノートナンバーは倍音列の計算によって求められている。その基音となる周波数はマイクから入力される波形データの計算によって求められ、実際に検出されたピッチの 2 オクターブ下のノートナンバーを基に、第 4・6・8・10・14・16・18・20 倍音の 8 つの周波数が計算によって導き出され、楽器としてのコンピュータによって演奏されるピッチとして使用される。この処理はリアルタイムで常に行われており、倍音列による構造には影響を及ぼさないが、その基音は常に変化している。しかし、一度ピッチが検出されると 2 秒間は新たに更新はしない。

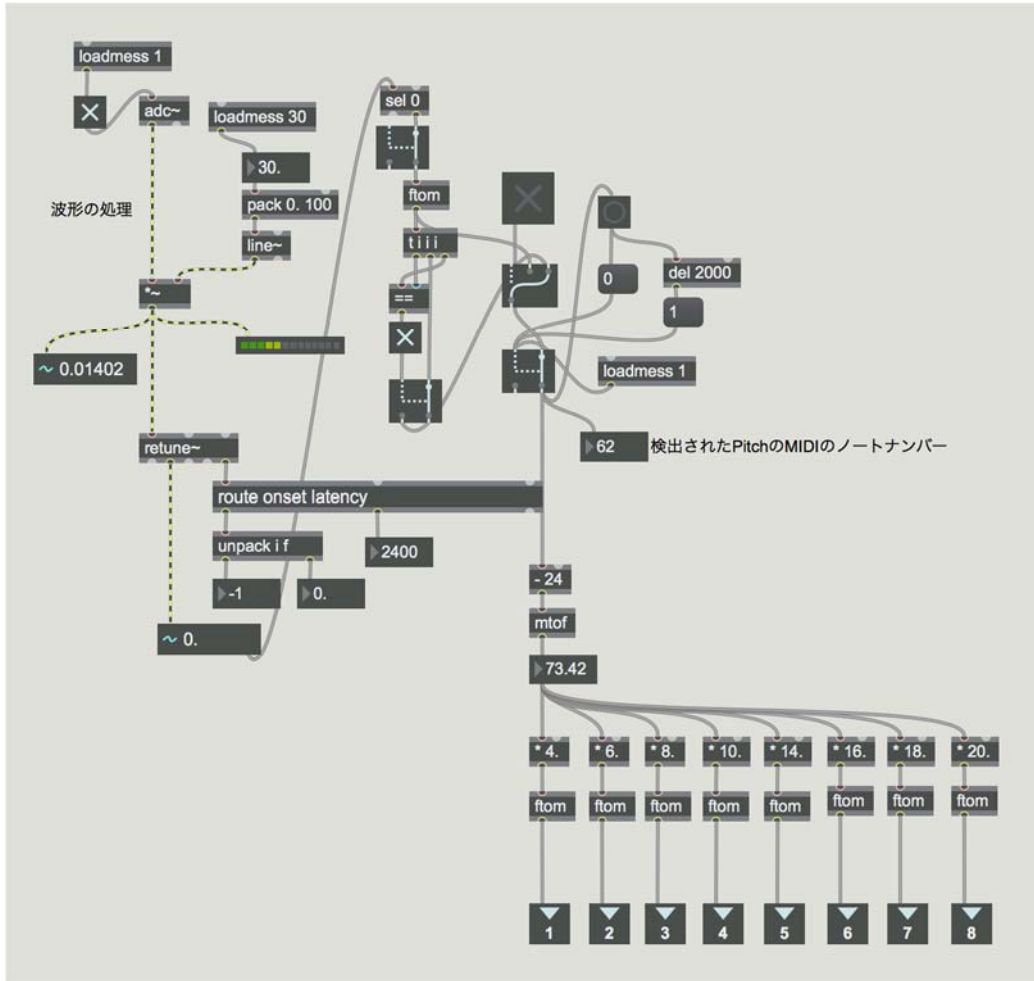


図 2. 楽器としてのコンピュータによって演奏されるピッチを計算するパッチ  
(筆者による作成)

人間が手の動きによって演奏を開始しようとする際、センサーから送られてくるデータは MuBu のオブジェクトを用いたパッチにより分析され **bang** が出力される。このシステムでは、その **bang** メッセージは“Response”とは別のパッチ上で出力され、**send** オブジェクトにより “Response”のパッチに送信されるようになっている。

図 3 にあるように **Receive** オブジェクトによって受け取った **bang** によって楽器としてのコンピュータが演奏されることとなるが、**trigger** オブジェクトによって以下の 5 つの順序で機能が作動することにより、演奏が行われる。

1. 1度の bang の入力に対して鳴る 8 つの音の時間間隔をそれぞれ deley オブジェクトによって 0~200ms の範囲で指定する。
2. bang の時間間隔を timer オブジェクトと coll オブジェクトとの組み合わせにより計測し図 4 の形式で記録する。
3. counter オブジェクトを用い、入力される bang の回数を記録する。
4. 図 2 のパッチの処理によりあらかじめ用意されたピッチを、loadbang オブジェクトによる bang をきっかけにランダム化された順序で、counter オブジェクトに bang を入力することより MIDI として再生する。その際の Velocity の値は 2.で計測された bang の間隔が短ければ高く、長ければ低く設定される。
5. 4.の bang によって MIDI が再生された後、1.で間隔が指定された deley オブジェクトにより bang が遅れて counter オブジェクトに入力されることにより、いくつかの音が遅れて再生される。その数は 2.で計測された bang の間隔が短ければ多く、長ければ少ない。

3.における行程によって記録された bang の回数によって、2つの思考回路(図 1-4.)の内どちらが選択されるかが場合分けの計算によって決定される。このシステムにおいては 1 小節 4 拍として計算され、4 小節以下の場合(bang の回数が 19 回以下) (a)の思考回路、5 小節以上の場合(bang の回数が 20 回以上) (b)の思考回路となる。

人間による楽器としてのコンピュータの演奏は、(a)の思考回路の場合でも、(b)の思考回路の場合でも MIDI として記録されるが、演奏者としてのコンピュータによる楽器としてのコンピュータの演奏の際には、演奏は保存されない。



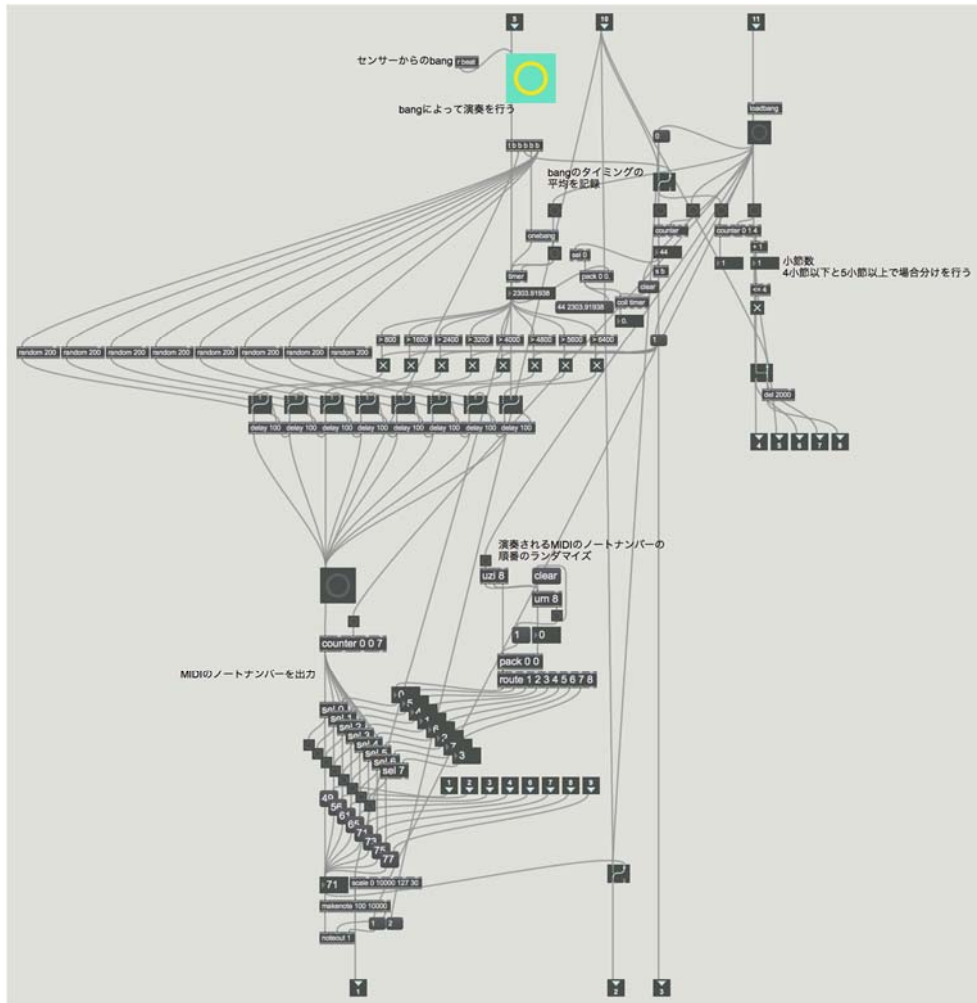


図 3. 楽器としてのコンピュータのパッチ (筆者による作成)

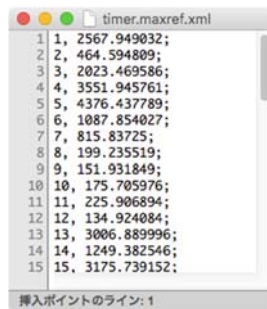


図 4. Coll オブジェクトによって記録された人間の演奏の bang の回数と時間間隔(ms) (筆者による作成)

### 3-3 人間の演奏を MIDI のデータとしてリアルタイムに記録するシステム

図 5 のパッチにより、人間による演奏は seq オブジェクトによって常に保存される。センサーから送られてくる最初の bang をきっかけに記録を開始し、最後の bang が送られてから 10 秒経過すると、自動的に記録が停止する。その後、seq オブジェクトにメッセージとして “write Record” が送られることにより “Record.mid” として、パッチが置かれているフォルダと同じフォルダに MIDI のデータとして保存される。この時点におけるセンサーから送られてきた bang の総数によって場合分けが行われ、思考回路(a)、もしくは思考回路(b)が動作を開始する。

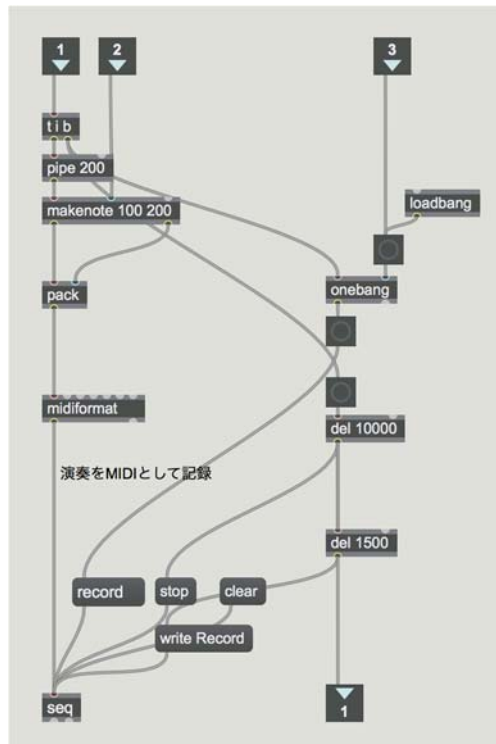


図 5. 人間の演奏を MIDI のデータとして保存するパッチ（出所）筆者による作成

### 3-4 記録した人間の演奏の bang の間隔の順番を入れ替え Antescofo を用い再生する

#### システム：思考回路(a)

小節数が 4 以下だった場合この思考回路が動作する。図 6 のパッチでは、記録された人間の演奏の際の bang の時間間隔を urn オブジェクトによってランダムに並べ替え、その間隔に則り bang を出力する。urn オブジェクトによってランダムに並べ替えられた bang の間隔は、平均されれば人間による演奏の bang と同じだが、1-2 で述べたように、不自然な時間間隔で bang が出力されることとなる。そこでこの bang の拍をきっかけとして Antescofo を用いてスコアフォロ

ーイングを行い、ランダムに並べ替えられた bang の間隔を Antescofo が人間による bang の入力として認識することにより、人間の演奏として不自然な拍は除外され、Antescofo からは、人間の拍の間隔を元にしたコンピュータによる bang を、人間の演奏者による bang として認識した Antescofo により bang が出力され、独特な間隔で拍が出力されることとなる。演奏者としてのコンピュータによる bang としてこれを出力し、楽器としてのコンピュータの演奏を行う。また、スコアフォローイングのためのテキストデータは、1 小節の場合から 4 小節の場合まであらかじめ用意されており、①の思考回路が選択された際、自動的に Antescofo に読み込まれる。

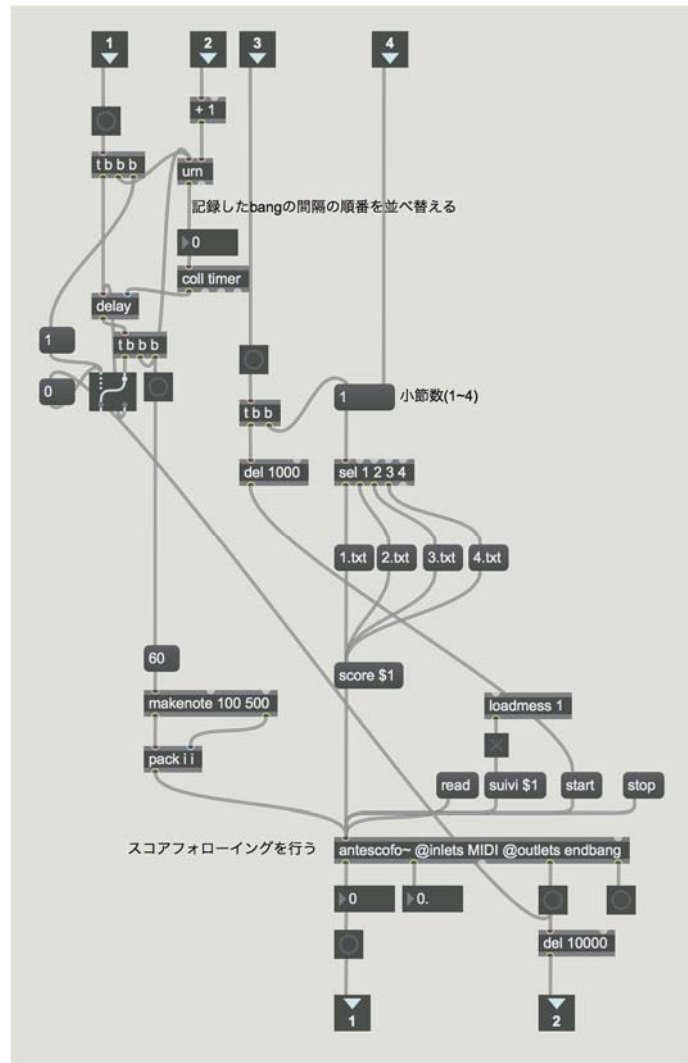


図 6. 記録した人間の演奏の bang の間隔の順番を入れ替え Antescofo を用い再生するパッチ (筆者による作成)

### 3-5 Antescofo の役割

このシステムにおいて Antescofo が果たす役割として以下の 3 つが挙げられる。

- ・人間による演奏の小節数による、演奏者としてのコンピュータが出力する bang の回数の指定
- ・コンピュータによって不自然な間隔で出力される bang のタイミングの修正
- ・演奏者としてのコンピュータによる演奏の終了の合図

演奏者としてのコンピュータが出力する bang の数は antescofo から出力される bang の数であり、coll オブジェクトから出力される bang の間隔には拠らない。なぜなら、coll オブジェクトから出力される時間間隔によって delay オブジェクトから出力される bang は、不自然な間隔で出力された場合スコアフォローイングの際に除外される可能性があるからである。スコアフォローイングが最後まで終了するとアトリビュートの指定により第 4 アウトレットから bang が出力されるので、これをきっかけに楽器としてのコンピュータがリセットされ、人間による演奏の待機状態となり、(a)の思考回路の場合の円(図 1)が一巡する。

### 3-6 保存された MIDI のデータを bach を用いて楽譜のデータに変換するシステム

：思考回路 (b)

小節数が 5 以上だった場合この図 7 の思考回路が動作する。図 5 のパッチにより保存された MIDI のデータを bach.roll オブジェクトで読み込むことにより、プロポーショナル・ノーテーションの表記の楽譜に変換を行う。演奏が波形データとして記録されていた場合はこのような変換はできない。MIDI のデータを楽譜として bach.roll オブジェクトに読み込むことにより、MIDI として記録された演奏の音符単位での編集が可能になるが、これも演奏が波形データとして記録されていた場合には不可能である。

### 3-7 楽譜に変換されたデータを bach のオブジェクトを用いたアルゴリズムを通して変換し演奏するシステム

このアルゴリズムは人間による楽器としてのコンピュータの演奏と連動している。人間の演奏の際の bang の総数を scale オブジェクトを用いて 0.8~1.5 の範囲の値に変換し、MIDI からプロポーショナル・ノーテーションの表記の楽譜へと変換を行った音符の MIDI セント全てに掛け合わせている。この計算により、人間の演奏の際の bang の回数が少なければ、全体のピッチは低く音程間隔が狭くなり、bang の回数が多ければ、全体のピッチは高く音程間隔が広がる。

また、MIDI からプロポーショナル・ノーテーションの表記の楽譜へと変換を行った楽譜に、bach によるアルゴリズムにより、逆行・反行・反逆行の 3 声部を追加し、順行の声部は 1 オクターブ高く、反逆行の声部は 1 オクターブ低くなる計算を行った。そしてこの段階においては、

演奏のタイミングは人間の演奏が元になっているので、クオンタイズを行いクラシカル・ノーテーションの表記に楽譜を変換することにより、人間の演奏らしい自然なアゴギクを一旦排除する。

`bach.score` オブジェクトにクラシカル・ノーテーションの表記で楽譜を表記させ、`bach.playkeys` オブジェクトを用いて楽譜を再生する際には `pipe` オブジェクトと `random` オブジェクトを用い、MIDI のノートナンバーが `makenote` オブジェクトに入力されるタイミングにズレを生じさせることにより、1-2 で言及したような不自然な演奏になることを回避している。楽譜が最後まで再生されると `bach.score` オブジェクトから“end”が出力されるので、これをきっかけとした `bang` により、10 秒が経過した後、楽器としてのコンピュータがリセットされ、人間による演奏の待機状態となり、(b)の思考回路の場合の円(図 1)が一巡する。

### 3-8 “Response”のシステムの総括

人間の手の動きから始まるこのシステムの行程は、図 1 にあるように円になっており、演奏を始めた人間がコンピュータの演奏に対して呼応をやめない限り、その行程は幾度となく繰り返される。ただし、この繰り返しはただのループではなく、一度として同じ循環が行われることはない。コンピュータは人間による演奏の度に新たに計算(思考)を行い、人間もそれに呼応し演奏を行う。

また、このシステムで行われているコミュニケーションは全て `bang` の回数の値とその時間間隔の値を元に成り立っている。数の計算を行うコンピュータと拍を認識する人間との共通言語としてこれらの値を用いることの有効性を“Response”のシステムの構築を通して確認した。

## 4 本稿の総括及び今後の展望

本稿では、大野(2018)において課題として示された、人間と共演者としてのコンピュータとの、音楽におけるインタラクティブな関係性においてシステムを構築するという創作上のアイデアを基に、実際の“Response”のシステムの製作を通して、コンピュータが担う 3 つの役割に主眼を置き、システムの分析、研究を行った。

“Response”のシステムにおいてコンピュータは以下の 3 つの役割を果たした。

#### ① 楽器としてのコンピュータ

この楽器によって演奏される MIDI のノートナンバーは倍音列の計算によって求められた。その基音となる周波数はマイクから入力される波形データの計算によって求められ、実際に検出されたピッチの 2 オクターブ下のノートナンバーを基に、第 4・6・8・10・14・16・18・20 倍音の 8 つの周波数が計算によって導き出され、楽器としてのコンピュータによって演奏されるピッチとして使用された。人間が手の動きによって演奏を開始しようとする際、センサーか

ら送られてくるデータは MuBu のオブジェクトを用いたパッチに分析にされ、動きが認識されると bang が出力される。その bang をきっかけとして5つの順序で機能が作動することにより、演奏が行われることを確認した。

### ②計算(思考)するプログラムを実行するツールとしてのコンピュータ

人間による演奏は seq オブジェクトによって常に保存される。センサーから送られてくる最初の bang をきっかけに記録を開始し、最後の bang が送られてから 10 秒経過すると、自動的に記録が停止し、その後 seq オブジェクトにメッセージとして“write Record”が送られることにより“Record.mid”として、パッチが置かれている場所と同じ場所に、MIDI のデータとして保存されることを確認した。

人間の演奏の際の bang の総数によって導き出される小節数が4以下だった場合: 思考回路(a)、記録された人間の演奏の bang の時間間隔を urn オブジェクトによってランダムに並べ替え、その間隔に則り bang を出力した。

人間の演奏の際の bang の総数によって導き出される小節数が5以上だった場合: 思考回路(b)、保存された人間の演奏の MIDI のデータを bach.roll オブジェクトで読み込むことにより、プロポーション・ノテーションの表記の楽譜に変換を行い、bach のオブジェクトを用いたアルゴリズムによって変化を加え演奏を行った。このシステムにおけるアルゴリズムは人間による楽器としてのコンピュータの演奏と連動しており、人間の演奏の際の bang の総数を scale オブジェクトを用いて 0.8~1.5 の範囲の値に変換し、MIDI からプロポーション・ノテーションの表記の楽譜へと変換を行った音符の MIDI セント全てに掛け合わせた。また、MIDI からプロポーション・ノテーションの表記の楽譜へと変換を行った楽譜に、bach によるアルゴリズムにより、逆行・反行・反逆行の3声部を追加する計算を行った。

### ③演奏者としてのコンピュータ

思考回路(a)から出力された bang は、演奏者としてのコンピュータに入力され、楽器としてのコンピュータの演奏を行うことを確認した。この際、Antescofo は以下の3つの役割を果たすことを確認した。

- ・人間による演奏の小節数による演奏者としてのコンピュータが出力する bang の回数の指定
- ・コンピュータによって不自然な間隔で出力される bang のタイミングの修正
- ・演奏者としてのコンピュータによる演奏の終了の合図

スコアフォローイングが最後まで終了するとアトリビュートの指定により第4アウトレットから bang が出力されるので、これをきっかけに楽器としてのコンピュータがリセットされ、人間による演奏の待機状態となり、(a)の思考回路の場合の円が一巡することを確認した。

思考回路(b)においては、bach.score オブジェクトにクラシカル・ノテーションの表記で楽

譜を表記させ、`bach.playkeys` オブジェクトを用いて楽譜を再生することにより演奏を行うことを確認した。この際、`pipe` オブジェクトと `random` オブジェクトを用い MIDI のノートナンバーが `makenote` オブジェクトに入力されるタイミングにズレを生じさせることにより、1-2 で言及したような不自然な演奏になることを回避していることを確認した。楽譜が最後まで再生されると `bach.score` オブジェクトから“end”が出力されるので、これをきっかけとした `bang` により、10 秒が経過した後、楽器としてのコンピュータがリセットされ、人間による演奏の待機状態となり、(b)の思考回路の場合の円が一巡することを確認した。

以上の分析を通して、人間とコンピュータによるインタラクティブな関係性によるシステムを構築する際、コンピュータに人間らしい自然な演奏をさせるようにすることは、人間の演奏者とのコミュニケーションの成立のために必要な要素ではあるが、コンピュータによる計算(思考)の段階においては、コンピュータに、人間には瞬時に計算し得ない計算をあえてさせることにより、コンピュータにしか果たし得ない計算機としてのコンピュータの役割を担わせることが可能であり、コンピュータを演奏者とする際の音楽の表現として有効な手段であることが示された。また、このシステムにおいては、②計算(思考)するプログラムを実行するツールとしてのコンピュータと③演奏者としてのコンピュータはシステム上一体となって機能していることが明らかとなった。

今後の展望として、`bach.score` オブジェクトに表記された楽譜の再生には今回 `bach.playkeys` オブジェクトが用いられたが、この楽譜の再生を人間が楽器を用いて、一人で、もしくは複数人によって行うことにより、コンピュータは②計算(思考)するプログラムを実行するツールとしてのコンピュータのみの役割を果たし、③演奏者としてのコンピュータの役割は人間が担うということが可能性として考えられる。また、“Response”のシステムにおいては人間の動きをセンサーを通して `bang` に変換し、①楽器としてのコンピュータを機能させたが、この `bang` は最初に③演奏者としてのコンピュータに出力させることも可能である。ただこれらの展望には現在のシステムと比較して、技術的な垣根は存在しないのであり、コンピュータと人間とが相互的な関係性によるコミュニケーションを行う際に `bang` を用いることによって、数の計算を行うコンピュータと拍を認識する人間との共通言語として有効に機能していることを示している。

## <注>

- 1) nslider オブジェクト。音価の表記には対応していない。
- 2) 音価が符頭から右側に伸びるバーの長さによって表記される。

## <引用文献>

大野雅夫(2017)「音楽におけるコンピュータと演奏者との関係性についての考察—Antescofo を応用した自作品の分析を通して—」『現代社会文化研究』第 64 号、新潟大学現代社会文化研究科、pp.37-52

大野雅夫(2018)「音楽創作における 2 つの関係性によるコンピュータと演奏者の役割についての考察—自作の Max パッチの分析を通して—」『現代社会文化研究』第 66 号、新潟大学現代社会文化研究科、pp.171-186

清水研作、大野雅夫(2015)「コンピュータ音楽とその作曲への応用」『新潟大学教育学部研究紀要 人文・社会科学編』第 8 巻第 1 号、pp.81-94

堀内靖雄、藤井敦、田中穂積(1995)「複数の人間と協調する演奏システム」『コンピュータソフトウェア』第 12 巻第 5 号、一般社団法人ソフトウェア科学会、pp.465-473

Andrea Agostini, Daniele Ghisi, “bach: an environment for computer-aided composition in Max”, *Proceedings of the International Computer Music Conference(ICMC 2012)*, Ljubljana, Slovenia. pp.373-378, 2012

Emmanuel Flety, “A comprehensive guide to using, programming & flashing the BITalino R-IoT WiFi sensor module”, *Prototypes & Engineering Team (PIP)*, IRCAM Edition 1.0, pp.1-2, March 2017

主指導教員（清水研作教授）、副指導教員（伊野義博教授・田中幸治准教授）