

音群的音楽の表現を取り入れた演奏ソフトウェアの制作

土下竜人^{†1} 中村滋延^{†2}

音響テクスチュアの状態変化を音楽的内容とする音群的音楽の表現を取り入れ、ソフトウェアアート作品《雲霞の如し》、《げんげ》を制作した。魚の群れの動きを状態変化の主要要素として用い、音に対応した映像が聴取の手がかりとなることを目指した。

Design of a Rendition Tool Adopted the Expression of the Tone Cluster

TATSUTO TSUCHISHITA^{†1} SHIGENOBU NAKAMURA^{†2}

I created rendition tools named “unka-no-gotoshi” and “genge,” which adopt the expression of tone clusters. Generally speaking, tone cluster music is composed based on considering the change in texture state as a musical component. The motion of a school of fish was used for the software as the main factor of the change of state. I designed the software so that the image corresponding to sound would serve as a key, which is listening.

1. はじめに

本稿は筆者が制作したソフトウェアアート作品《雲霞の如し》《げんげ》の制作過程について解説・考察するものである。

1950、60年代の現代音楽シーンの主流を形成した総音列技法は、構造的・組織的視点が強調されるものであった。その矛盾点は、その緻密な組織や構造を耳でとらえることができないところにあり、聴取者がとらえるのは構成の結果として鳴り響く音の表面である音響テクスチュアであった。音群的音楽は、偶然の結果としての音響テクスチュアではなく、意図の結果として音響テクスチュアを作るという考えで出現してきた。本稿において取り上げる音群的音楽とは、いわゆるクラスター音楽よりも広い解釈をもつものである。音群的音楽の本質は、音響テクスチュアの変遷、音群の状態変化を音楽的内容としてとらえる所にある[1]。

本稿で紹介する二作品は、音群の状態変化のパラメーターとして魚の群れの動きを取り入れた音楽演奏ソフトウェアアート作品である。音群的音楽における音響テクスチュアの考え方は、楽譜などを見てみても視覚的なアプローチによってなされている部分があることが分かる(図 1)。映像を伴うような音楽演奏ソフトウェアアートにおいて音群的音楽の表現の発展の可能性を感じたため、作品制作に至った。

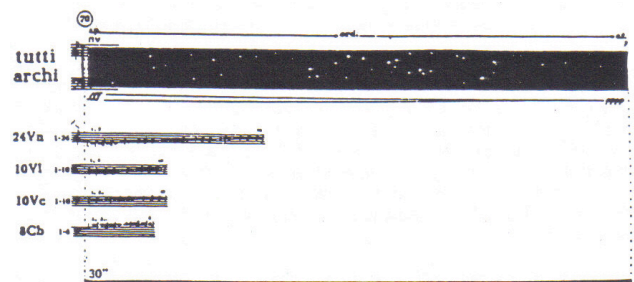


図 1. ペンデレツキ《広島犠牲者に捧ぐ哀歌》のスコア

2. 《雲霞の如し》の制作

2.1 作品概要

《雲霞の如し》は、筆者が 2010 年に制作したソフトウェアアート作品である(図 2)。演奏者はマウスを用い、PC 画面内の GUIa を操ることによって群れや音の状態を変化させ、演奏を行う。

2.2 システム

グラフィックやアニメーションの生成には Processing [2]、音響生成に Max [3]を用いた。

魚の形をしたオブジェクトが群れをなし、音を出しながら 3D の水槽の中を泳ぎまわる、というソフトウェアである。Craig Reynolds によって提唱された boids 理論[4]に基づき魚の群れの動きをシミュレートしている。

^{†1} 九州大学大学院芸術工学府
Graduate School of Design, Kyushu University
^{†2} 九州大学大学院芸術工学研究院
Faculty of Design, Kyushu University

a グラフィカルユーザーインターフェース。ユーザーに対する情報の表示にグラフィックを使用し、その操作をマウスなどのデバイスによって行えるインターフェースのこと。



図 2 雲霞の如し/操作画面

2.3 GUI

GUIの説明を通じて作品を解説していく。GUIには水槽全体に影響を及ぼす全体 GUI(図 2)と群れごとにパラメータを設定する群れ GUI(図 3)がある。

2.3.1 全体 GUI

水流速度スライダー／水流方向転換ボタン

水流の流れる向きと速さを設定する。値を大きくすると群れの動きよりも水流の動きの方が支配的になり、水槽内をめまぐるしい速度で動き回ることになる。

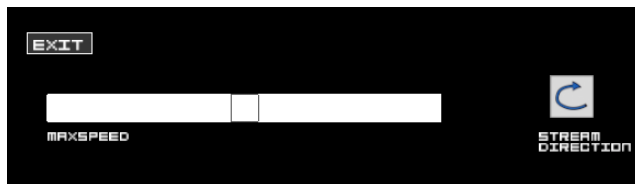


図 2 全体 GUI

2.3.2 群れ GUI

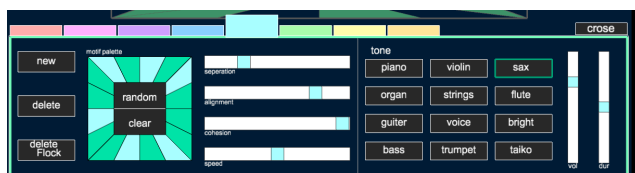


図 3 群れ GUI

群れ選択タブ(図 4)

群れに対応するタブを押すことによって、操作する群れを切り替えるという機能を持っている。群れは 8 つまで作成することができ、一つ一つに対して別々の群れパラメータや音色を設定することができる。



図 4 群れ選択タブ

オブジェクト増減ボタン(図 5)

3つのボタンから成り立っている。
 new/オブジェクトが水槽上部から一匹追加される。
 delete/オブジェクトが一匹削除される。
 delete flock/選択中の群れが削除される。

発音ゾーン設定ボタン(図 6)

オブジェクトが音を発生させるゾーンを設定するボタンである。ステップセンサーの要素を取り入れ、水槽を分割し、それを一つ一つのステップとして捉えた。水槽を上から見て、16のゾーンに分ける。ボタンのひとつひとつがその分けられたゾーンに対応している。ボタンを on しておくとオブジェクトがそのボタンに対応するゾーンに達したときオブジェクトの色が変わるエフェクトと共に音を発する。音の定位はオブジェクトの X 座標、音の高さはオブジェクトの Y 座標、音の大きさはオブジェクトと視点の距離に基づいている。また、ランダムにボタンの on/off を決める、発音ゾーンランダム決定ボタン、設定した発音ゾーンを消す clear ボタンも中央部に実装してある。

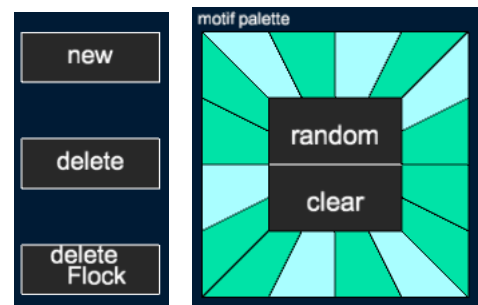


図 5.6 オブジェクト増減ボタン(左)、発音ゾーン設定ボタン(右)
 群れパラメータスライダー／巡航速度スライダー(図 7)

群れパラメータスライダーは、boids 理論に関する 3 つの重要なパラメータ、separation(分散)・cohesion(集合)・alignment(整列)を操作することができる。この数値の増減によって同じ群れに属するオブジェクト同士の関係性が変化する。例えば、separation の数値を高く、cohesion の数値を低くすると、群れの仲間との距離をとろうとする動きが高まり、それと同時に群れの仲間へ近づこうとする力が弱まるので、群れ全体としてみると分散してゆく、といったようにグラフィックや音がそのパラメータに対応して変容するのがわかる。

巡航速度スライダーはオブジェクトが移動するときの最大速度を操作することができる。値を大きくすると、オブジェクトの泳ぐ速度が上がる。

音色設定 GUI(図 8)

群れごとにオブジェクトが出す音を設定するボタンと、音量を司る Volume スライダー、音の持続時間を設定する duration スライダーによって構成されている。音色は全部で 12 種類ある(ピアノ、オルガン、ギター、ベース、ヴァイオリン、ストリングス、ボイス、トランペット、サクソフーン、フルート、ブライツネス、太鼓)。この音色は PC に内蔵されている MIDI 音源を使用した。

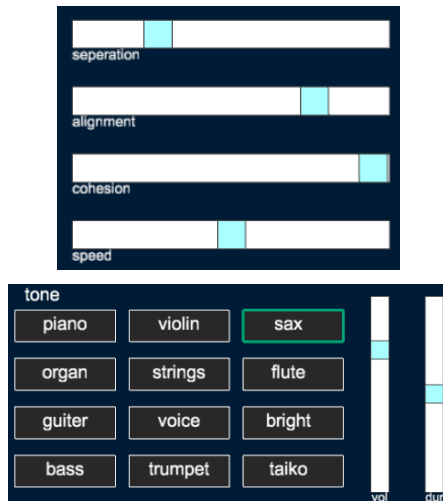


図 7.8 群れパラメーター(上)、音色設定 GUI(下)

2.4 演奏の手順

このソフトウェアの演奏の手順としては、

- ①群れ選択タブにおいて操作する群れを選択する。
- ②どこのゾーンで音が出るかを定める。
- ③その群れの魚を水槽内に入れてやると発音が始まる。
- ④オブジェクトが水槽内をぐるぐる泳ぎ回るのでそれがループのような働きをする。また8つの群れがミュージックシーケンサーで言うパート分けのような役割を果たす。
- ⑤状況や好みに応じて音色設定 GUI や boids パラメータースライダーといった GUI を操作し音楽を構成していく。となっている。

2.5 考察

群れが密集すると同じような高さの音が連続して出力されたり、オブジェクトが素早く画面を横切ると音も素早く横切ったりというような、音と映像の対応関係を実装することができた。そのため、次に音が鳴るタイミングを認識できたり、群れの形を見るだけで発される音がある程度予想できたりといった聴取を助ける効果が見られた。また、魚のようなオブジェクトを用いたことによって、オブジェクト自体に愛着が湧いたことや、音楽を演奏することを目的とするのではなく魚を動かして遊び、その結果として音楽が出力されるといった方向性も見られた。加えて、この作品は一人で楽しむ作品として制作していたが、人前で演奏するパフォーマンスツールとしての可能性も感じた。

問題点としては、パラメーターがオブジェクトの動きや音とどのように対応しているかが理解しづかったことや、群れのパラメーターが演奏に対して支配的で、演奏者が寄与できる部分が少なく自動的に音楽が演奏されているような感覚に陥ってしまったこと、オブジェクトが増えすぎると音に変化が感じられなくなることなどが挙げられた。

3. 《げんげ》の制作

3.1 作品概要

《げんげ》は筆者が2012年に現在作成中のソフトウェアアート作品である(図9)。前作《雲霞の如し》の反省点を踏まえ制作している。ここでは前作との相違点に着目しながら実装が完了している機能を紹介する。

3.2 システム

グラフィックやアニメーションの生成には openFrameworks [5]、音響生成に Max を用いた。

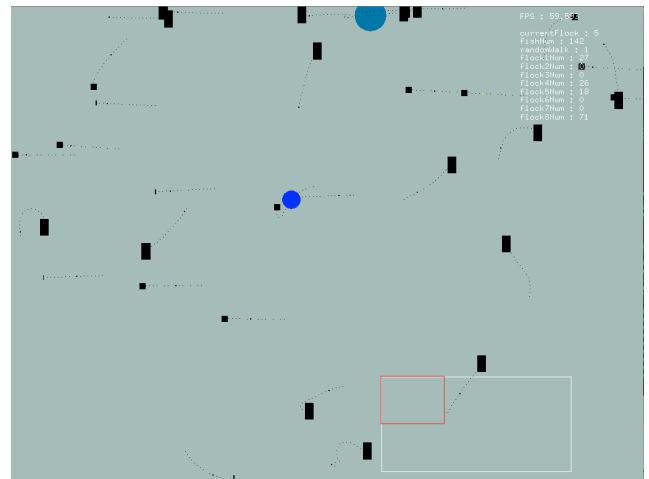


図 9. 《げんげ》の演奏画面

《げんげ》は基本的に前作《雲霞の如し》をベースに作られている。ここでは現時点で実装されている部分について述べることにする。

大きな変更点としては、3D 表現をやめ 2D 表現にしたことが挙げられる。前作よりも楽譜を意識したようなビジュアルにすることによって、群れの配置と音の関係がわかりやすくなるのではないかと考えたためである。また、マウスでオブジェクトを捕まえるといった動作も直感的にできるようになった。

前作では群れ全体に与えていたステップシーケンサーの要素をオブジェクト一匹一匹に与えた。群れのパラメーターが支配的だったため、リズムパターンなどを一匹一匹で制御できるようにするための変更である。

前作では水槽の境界線が画面内にあったことに対して、本作では画面よりも大きな枠で水槽を作ることになった。画面内にあるオブジェクトのみ発音をさせることによって、オブジェクトの増え過ぎによる音の変化の実感の低下を軽減することを目指した。

4. 今後の展望

作品《げんげ》は現時点では完成していないため、その

作品の今後の展望をまとめて替えて述べようと思う。

(1) UIの実装

作品《げんげ》ではまだUIが実装されていない。演奏してもらおう際にはもちろん必要であり、パフォーマンスとして演奏する際にも何を操作しているかを伝えるためにUIは必要だと考えている。その設計指針としては、①できるだけ操作するパラメーターを減らすこと、②操作の効果がわかりやすいこと、が挙げられる。

(2) 群れの外形描画システムの実装

《雲霞の如し》のときの反省点として挙げられた、パラメーターの操作が音にどう反映されているか分かりづらいという問題を解消するべく、群れの形を自分で描けるシステムを実装しようと画策している。その外形によって、群れ単位で操作しやすくなったり、群れが発音しうる音域を直感的に設定したりできるのではないかと考えている。

(3) 音源の選別

現在は前作《雲霞の如し》と同じくMIDI音源を鳴らす仕組みになっている。作品の世界観を構築するべく新たな音源の可能性を模索中である。

(4) パフォーマンスツールとしての可能性

前作《雲霞の如し》のときに見いだした、パフォーマンスツールとしての可能性も同時に模索していきたい。演奏に適度な自由度をもたらすことも重要であるが、時間経過によって音と背景が変化していくなどといった、音楽として展開するための機能を実装しようと考えている。

参考文献

- [1]中村滋延, 現代音楽×メディアアート音響と映像のシンセシス, 九州大学出版会, (2008)
- [2] Processing
<http://processing.org/>
- [3] Max
<http://cycling74.com/products/max/>
- [4]Reynolds, C. W., "Flocks, Herds, and Schools: A Distributed Behavioral Model!" *Computer Graphics*, Ver.21, Num.4, pp.25-34.,(1987)
- [5] openFrameworks
<http://www.openframeworks.cc/>