

## アルゴリズムック・インプロヴィゼーション用ソフトウェアの開発

### —リアルタイム AV パフォーマンスによるデジタル美学の探究—

#### 1. 背景

昨今のハードウェアの性能の向上により、小型のパーソナル・コンピュータ(主にラップトップ/ノートブック型のもの)による音響と映像の実時間同時生成テクノロジーが急速に普及しはじめた。そのため従来、例えば DJ と VJ のセッションのように、音響と映像が役割分担して行っていた AV パフォーマンスが、1 台ないしは相互にデータの送受信が可能な数台のマシンによる同一のプラットフォーム上で行えるようになってきた。本プロジェクトでは、そうした背景を踏まえて、インタラクティブなリアルタイム AV パフォーマンスが持っている本質的な可能性とその具体的なストラテジーを即興パフォーマンスの実践を通じて探求した。そのために、可能な限りシンプルなハードウェアによる(すなわちソフトウェアのもつ可能性を最大限に活用した)音響/音楽演奏用プログラムの開発を行うことを目的とした。

プログラム開発に際して、特に着目したのは、

1. 離散的な時間領域と連続的な時間領域によって区別されていた音楽構成と音響合成を、1 つのシームレスな枠組みの中で捉える。
2. デジタルという素材がもつ形式の抽象という特徴を生かして、音響と映像という 2 つの形式の表現を数値という共通の素材から同時並列的に生成する「メタメディア表現」に取り組む。このメディア抽象には「パラメータ抽象」「スペクトル抽象」「マテリアル抽象」の 3 つのレベルがあるが、本テーマでは特に抽象度の高い「マテリアル抽象」をベースにしたプログラム開発を行う。
3. デフォルトのキーボードとマウスというハードウェアの枠組みの中で、集中的で視覚的なメニュー型のインターフェイスではなく、分散的かつ身体的なコックピット型のインターフェイスを実現する。

という 3 点であった。

## 2. 目的

本プロジェクトで開発するのは、ポータブルなパーソナル・コンピュータ上で音響と映像の双方をリアルタイムかつ即興的に生成するためのソフトウェアである。このソフトウェアの上でスキルを積み重ねることで、コンピュータやネットワークを活用した音響映像即興演奏（リアルタイム AV インプロヴィゼーション）の新たな可能性を探索し、生まれつつあるデジタル美学の実践と探究を行った。

前項にあげた本プロジェクトの 3 つのポイントのうち、パフォーマンス実践の際に特に重要になるのが、3 番目のインターフェイスに対する考え方である。通常のソフトウェアの場合、ユーザーとしてスキルを持たない初心者、あるいは素人を対象とし、だれもがすぐに使えることを目標とすることが多い。しかし本プロジェクトの場合は、スキルを重ねる必要があるが、その見返りとして個々の使用者によって異なる、个性的かつ創造的な表現が生まれる余地を持ったプログラムを開発することを目的とした。

## 3. 開発内容

### 3.1 メタメディア表現を可能にするプログラムの基幹部分の設計と実装

Pd+pdp、pd+framestein、max/msp/jitter それぞれのプログラム言語を用いて、素材としての数字列を直接組織化することで、映像と音響を同時生成する「マテリアル抽象」を実装し、パフォーマンスの実践の中から柔軟性や将来性を評価した。max/msp/jitter においては、データ型として実装されている「マトリックス」というデータフォーマットを用い、プロトタイプとして、マトリックスのフィードバックループを活用したパフォーマンス用プログラムを制作した。サウンドやビデオのフィードバックループは既存の音響映像表現においてもさまざまなアプローチで使用されてきたが、今回はマトリクスという数字列をループさせることで、同一の数値データから映像と音響を同時に生成させている。

### 3.2 インプロヴィゼーションとプログラミングの整合性に関する設計と実装

従来から行われてきたアルゴリズムック・コンポジションの場合、プログラムによる構造の静的記述が中心であったが、アルゴリズムック・インプロヴィゼーションにおいては、リアルタイム性、選択肢の多様性、状況への対応、逸脱の可能性といった即興演奏の特徴を実現するため、プログラムによる動的な構造の生成が必要となる。すると通常のプログラミングに必要な、統合的プログラミングによる全体の整合性と明確な境界の記述に代わって、構造の柔軟性、すなわち選択肢の多様性と状況へ対応が求められる。そのために今回はプログラムをモノシリックなデザインにせず、細かなモジュールを動的に再構成する非統合的なアプローチによって、プログラムの境界を曖昧にし、仕様からの逸脱の可能性

を模索した。

本プロジェクトにおいては、こうしたひとつひとつのプログラム・モジュールを、練習によって獲得される人間のスキルのディテールに相当するものと捉え、このモジュール（マイクロスキル）の集合体としてプログラムの仮設的な全体を表現する。さらにプログラム言語がインタープリタであることを利用して、パフォーマンス中にプログラムを書き換える、インスタント・プログラミングも行う。

### 3.3 複数のモジュールやマシン間の通信に関する設計と実装

前項で述べたアルゴリズムを実現するためには、部分としてのモジュールの起動と停止、書き換え、生成と消滅といったダイナミックな振る舞いと、モジュール間のコミュニケーションといった分散的な処理のインプリメントが必要となる。前者については、pd や max のスクリプティング機能とダイナミックなオブジェクト管理機能を用いる。後者については、同一マシン内のコミュニケーションには通常の send/receive 機能を用い、複数のプログラム言語やマシン間の通信には OSC (Open Sound Control) を用いたコネクションを用いた。OSC は、異なるプログラミング言語間の通信に有効である。

### 3.4 音響映像生成システムの構築

コンピュータ内の数値から生成された音響や映像に原音や生映像は存在しない。AD コンバータや AV システムはすでに何らかの形で存在しているものを再生するのではなく、それ自身の特性によってかたちづくられる表現の生成装置である。デジタルサウンドの美学的特徴は、音のエッジの立ち上がりが鮮明であることと、低周波から高周波まで、低音から高音まで、人間の可聴域をフルに使えることに依拠している。その美学を知覚に結びつけていくためには、通常のポピュラー音楽やクラシック音楽以上に、再生システムの音響的性能が必要とされる。そこで次項のパフォーマンスを実践する際には、使用するソフトウェアのみならず、音響と映像を生成するためのシステムや環境にも十分配慮した。

### 3.5 パフォーマンスの実践

開発プロジェクト期間を通じて、幾度かパフォーマンスとインスタレーションの実践を行った。

1. 初台の NTT インターコミュニケーション・センター[ICC]で開催された「サウンディング・スペース 9 つの音響空間」にインスタレーション作品「materialAV—共鳴するインターフェイス」を出展（2003年7～9月）

[http://www.ntticc.or.jp/Calendar/2003/Sounding\\_Spaces/Works/material\\_j.html](http://www.ntticc.or.jp/Calendar/2003/Sounding_Spaces/Works/material_j.html)

<http://www.idd.tamabi.ac.jp/~kubotaa/materialAV/materialAV.mov>

2. 中京大学でレクチャーパフォーマンスを開催 (2003 年 10 月)
3. 多摩美術大学で AV パフォーマンスを実践 (2003 年 10 月)  
<http://www.idd.tamabi.ac.jp/~kubotaa/nu/031027.mov>
4. 静岡 mixed media におけるサウンドパフォーマンスを実践 (2003 年 11 月)  
<http://www.idd.tamabi.ac.jp/~kubotaa/solo/solo031103.mp3>
5. 武蔵野美術大学で AV パフォーマンスを実践 (2003 年 12 月)
6. 渋谷 Uplink Factory とせんだいメディアテークにおいて「ブラッケージ・アイズ2003-2004 リスpondダンス音楽企画 Visual Music for the Silent Film Maker」を開催 (2003 年 12 月, 2004 年 2 月)  
<http://www.visualmusic-for-sfm.net/info/livej.html>  
<http://www.idd.tamabi.ac.jp/~kubotaa/brakhage/031227tamabi.mov>  
[http://www.idd.tamabi.ac.jp/~kubotaa/nu/lovesong\\_t1.mp3](http://www.idd.tamabi.ac.jp/~kubotaa/nu/lovesong_t1.mp3),  
[http://www.idd.tamabi.ac.jp/~kubotaa/nu/lovesong\\_t2.mp3](http://www.idd.tamabi.ac.jp/~kubotaa/nu/lovesong_t2.mp3)

### 3.5 最終成果

上記のようなプロセスを経て、「pulsebeat」と「noisescan」という、アルゴリズムック・インプロヴィゼーション用のプログラムを開発した。

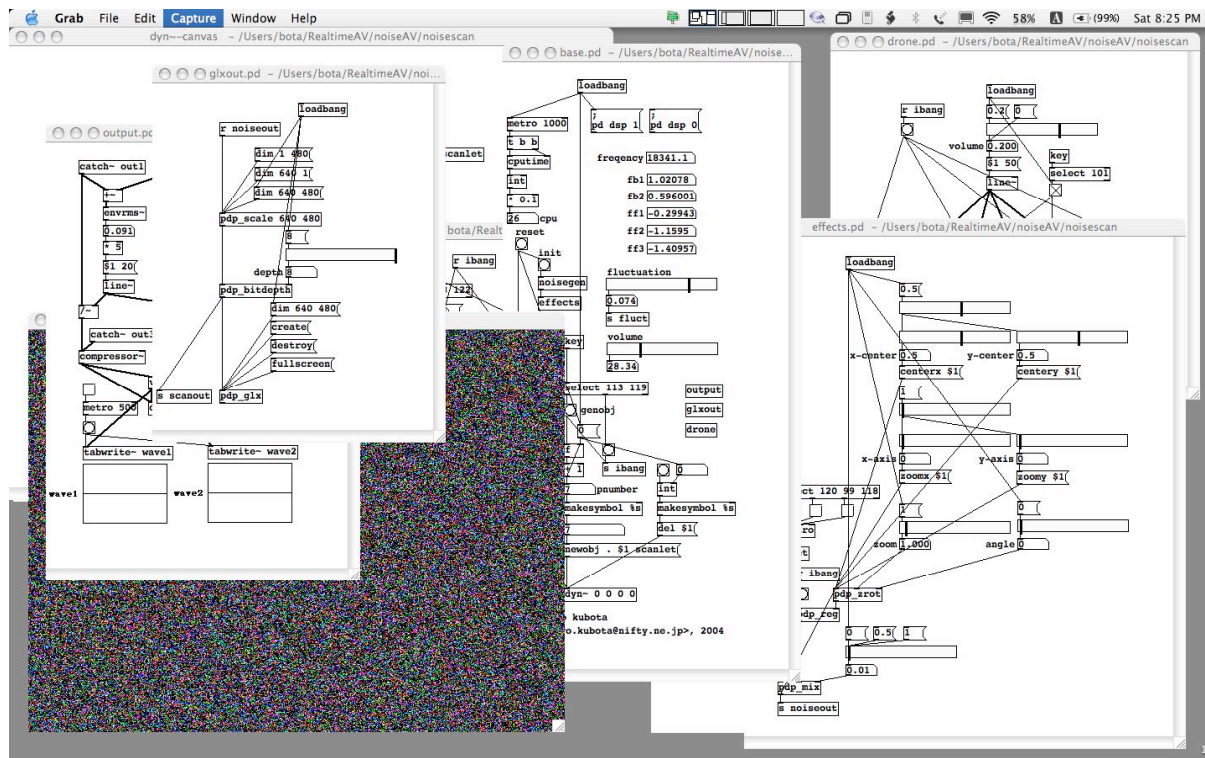
<http://www.idd.tamabi.ac.jp/~kubotaa/nu/pbns.mp3>

「Pulsebeat」はオーディオ専用ソフトウェアである。グラニューラ・シンセシスの一種の パルサー・シンセシス (<http://www.granularsynthesis.live.com.au/what.html>) と古典的な振幅変調 (アンプリチュード・モジュレーション) を用いて、異なる時間スケールによる音響合成モジュールと音楽構築モジュールをシームレスに統合しながらコントロールすることを目指したプログラムである。プログラムは pd で書かれている。これまで述べられてきた、ダイナミックなオブジェクト管理、オブジェクト間のダイナミックなルーティングといった手法を用いて、演奏中にモジュールを生成消滅させることで、動的な加算合成 (オーディティヴ・シンセシス) を行う。パルスをコントロールするテーブルは共通の変数 (グローバル変数) 領域におかれ、そのスキャン速度をオブジェクトごとに変化させている。

「Noisescan」は、図 4 の noiseSB プログラムをベースに、上述の「Pulsebeat」プログラムと同一のインプロヴィゼーション用フレームワークで書かれた、マテリアル抽象による AV パフォーマンス用プログラムである。数字列としての映像を、複数のスキャンレットがスキャンすることでサウンドを合成する。物理モデルによる音響合成法のひとつとし

て最近開発された、スキャンド・シンセシス (<http://www.csounds.com/scanned/>) の AV 版 (物理モデルの代わりに映像データをスキャンする) といってもいいだろう。

「Noisescan」演奏中のデスクトップのスナップショットを以下に示す。



## 5. 開発成果の特徴

ラップトップ AV パフォーマンスが行われる頻度は高くなってきたが、これらのほとんどは、アドホックな手法や、パフォーマンスに先立って制作していたムービーやサウンドファイルベースにしていたり、伝統的なアルゴリズム・コンポジションの延長線上で行われている。本プロジェクトのように、プログラミングとインプロヴィゼーション、GUI とリアルタイム操作といった、ラップトップ・パフォーマンス自体が有している本質的なパラドックスに対する考察や、スキルやインターフェイスに対する新たなアプローチの提案をしているものはほとんどない。

同様に、オーディオとビジュアルという2つの (代表的な) 表現形式を同時に使用することの意味についても、エンターテイメントにおける臨場感といった商業的な視点からではなく、聴覚と視覚、空間と時間といった芸術表現における本質的な問題まで掘り下げて考察と実践を行っている例は少ない。

哲学や思想なき技術や製品が革新的になることはない。本プロジェクトでは、こうしたイ

ノヴェーション（未踏性）の原則に立ち返って、コンピュータという新たな素材から生み出し得る表現の可能性を、実践と思考/思索の両面から総合的に探索する、という意味で既存のアプローチとは一線を画している。

## 6. 今後の課題、展望

本プロジェクトを通じて、アルゴリズムック・インプロヴィゼーションが含む、さまざまな可能性と問題点が明らかになった。楽器のインターフェイスの第一の特徴は、楽器を「わざと使いにくくデザインする」ということにある。誰もがすぐにできるようにすることも時と場合によっては必要となるが、「練習することによってできないことができるようになる」ということも人間にとって根源的な喜びのひとつである。インターフェイス・デザインについて考える、ということは人間について考えることと同じである。今日世界同時多発的に生まれつつある、コンピュータを素材とする新たなメタ・カルチャーやデジタル美学を、一人でも多くの人と共有していきたいと願っている。

開発者

久保田晃弘（多摩美術大学）

akihiro.kubota@nifty.ne.jp

<http://homepage2.nifty.com/~bota/>