

# 旋律パターンを利用した音楽インタフェースとパターンの獲得 に関する研究

出口幸子

## Research on the musical interface using melodic patterns and the acquisition of melodic patterns

Sachiko Deguchi

### Abstract

This paper describes the analysis of melodic patterns and the improvisation system using 4-note melodic patterns. This improvisation system has a graphical user interface which is composed of pattern area and note area. A user selects a pattern in the pattern area and inputs 4 times in the note area, and then 4-note melodic pattern is generated. This system also has an assistant mode for the rhythm and an automatic playing mode. This system provides a learning function for melodic patterns and a learning function for rhythmic patterns. Both learning functions are based on Genetic Algorithm. Several learning parameters can be set by the users of this system. By using these learning functions, melodic patterns and rhythmic patterns are acquired based on the preference of users.

Keywords: improvisation system, melodic pattern, rhythmic pattern, Genetic Algorithm, learning

### 1. はじめに

現在、コンピュータ、タブレット端末、ゲーム機および携帯電話等の上で動く音楽演奏システムが多くある。また、新しい電子楽器<sup>1)</sup>や演奏用の新しいインタフェースの研究<sup>2) 3)</sup>も盛んである。しかし、楽器演奏経験の少ない人が即興演奏をして他の人と合奏することは難しい。筆者は、音楽演奏経験のない人でも即興演奏が可能なシステムの開発を目指して幾つかのプロトタイプを作成してきた<sup>4)</sup>。その1つとしてユーザインタフェースに液晶タブレットを使用し、旋律パターンをユーザが指定することにより旋律が合成できるシ

ステムを試作した<sup>4) 5)</sup>。これは、種々のジャンルの音楽の旋律を分析して抽出した旋律パターンを利用している<sup>6)</sup>。その後、旋律パターンとリズムパターンを学習により獲得できるように拡張した<sup>7)</sup>。また、ユーザインタフェースとして、液晶タブレットより普及しているタッチパネルを使用することにした。本稿では、旋律分析による旋律パターンの抽出、旋律パターンを用いた即興演奏システム、および旋律パターンとリズムパターンの学習について説明する。

### 2. 旋律パターンの抽出

## 2.1. 楽譜データベース

旋律分析では、日本ポピュラー音楽の4名の作曲家の楽譜データベース(計96曲)を使用した<sup>8)</sup>。48曲については、楽譜から声のパートの音高と音価をHumdrum kern<sup>9)</sup>形式で入力した。他の48曲については、スキャナと楽譜認識ソフトウェアを用いて声のパートを入力し、MusicXML<sup>10)</sup>形式からHumdrum kern形式に変換した。

## 2.2. 特徴抽出

本研究では、分析のためHumdrum kern形式で記述された音高をMIDIノートナンバーに変換している。ここで、中央C(ド)は60、半音の音程は1である。次に、n音旋律(n-gram, n=3~7)を抽出する。n音の長さの旋律を1音ずつずらして順に切り出し、音程列として表現する。ここで、音程は、物理的には2音の音高の周波数比であるが、本分析では音高のMIDIノートナンバーの差とする。例えば、(ド レ ミ レ ド)という音の列は、音高列(60 62 64 62 60)であり、この音高列から、3音旋律(3-gram)として(60 62 64)(62 64 62)(64 62 60)が切り出される。これを音程列として表現すると、(2 2), (2 -2), (-2 -2)が抽出される。実際には、先に音高列を音程列に変換し、それから旋律パターンを抽出している。先の例では、音高列(60 62 64 62 60)を音程列(2 2 -2 -2)に変換して3音旋律パターンを抽出する。このようにパターンを抽出し、各パターンの出現回数を数えて全体に対する割合を計算し、多用される上位パターンを求める。4~7音旋律パターンについても同様に分析する。このように、本研究では、旋律パターンはリズムを含まない音程の並びとしている。リズムについては後述するように音程列と分けて検討している。

分析結果より、3音旋律の上位パターンは作曲家によらず、日本ポピュラー音楽で共通であることが確認された<sup>8)</sup>。また、旋律が長くなると共通性が低くなるが、5音旋律には作曲家の特徴が現れ、それより長い旋律では曲に固有のパターンが現れて来ることが分かった<sup>8)</sup>。

そこで、即興演奏システムの旋律合成に使用する旋律パターンとして、4音旋律を使用することにした。その理由は、以下の通りである。

- ・3音旋律では上位パターンが決まっており、それらを組み合わせて生成される旋律に変化がつかない。
- ・5音旋律では作曲家の特徴が現れ、上位パターンが作曲家により異なり、共通のパターンを用意することが難しい。
- ・4音旋律は、3音旋律と5音旋律の中間で、適度にばらつきがあり、且つある程度共通のパターンを用

意することが可能である。また、4/4拍子は演奏が容易であり、演奏の単位としても4音旋律が好ましい。

## 3. 即興演奏システム

本システムは、Windows XPおよびWindows 7上のVisual C++ 2008・2010を開発環境としている。PCに液晶タブレットあるいはタッチパネルを接続して使用している。当初、液晶タブレットの筆圧情報を使用していたが、筆圧にばらつきがあり、音の強弱等に使用することが難しいことがわかった<sup>4)</sup>。そこで、現在のシステムでは単純にペンの位置情報のみを使用している。ペン入力はマウス入力として処理しているため、ペン、指、およびマウスを使用可能であるが、画面の左右の両領域を頻繁に使用するため、マウスは適していない。また、音の生成にはMIDIを用いており、音源はMIDI音源を使用している。

図1に本システムのユーザインタフェースを示す。本システムでは、画面上で旋律パターンを選択し、その開始音および各音の生成タイミングを指定することにより、旋律を合成する。図1の画面左側が旋律パターンの領域であり、4音旋律を配置している。ここで、4音旋律は当初、前述した旋律分析で用いた音程列で表現していたが、ユーザにとって分かりにくいという意見があり、音高に基づく表現にしている。例えば、0は開始音、1は開始音より1音上(音楽では2度上。開始音がドであれば2度上はレ)、2は開始音より2音上(音楽では3度上)、-1は開始音より1音下(音楽では2度下)、等のように表現している。一方、図1の画面右側が開始音の領域であり、1オクターブ中の5音を2オクターブ分配置している。一般に音楽では1オクターブ中の7音が主に使用されるが、本システムでは開始音としては代表的な5音(ハ長調ではド・レ・ミ・ソ・ラ)を使用している。開始音にない2音(ハ長調ではファ・シ)も旋律合成の過程で生成されて使われている。

画面左側で1つの旋律パターンを選択しておき、画面右側の1つの開始音領域に4回入力すると、選択された旋律パターンの音が順次生成される。例えば、旋律パターン(0 1 0 1)を選択しておき、開始音ドの領域にペンや指で4回入力を行うと、(ド レ ド レ)という旋律が生成される。旋律パターン(0 -1 0 -1)を選択しておき、開始音ドの領域にペンや指で4回入力を行うと、(ド シ ド シ)という旋律が生成される。

開始音は、設定により変更できる。デフォルトではハ長調の音の組となるが、ト長調等の他の長調も設定できる。また、楽音はデフォルトではピアノ音であるが、MIDI音源の楽器音からユーザが選択できるようにしている。

当初のシステムでは、4音旋律パターンは既存の楽曲から抽出したもの、あるいは筆者らが作成したものをを用いており、ユーザの好みのパターンを用意することが難しかった。そこで、学習モードを用意して、ユーザが自分の好みのパターンを設定できるように改良した<sup>7)</sup>。第4章で旋律パターンの学習について説明する。一方、当初のシステムにおいて、リズムはユーザが指定していたが、リズムを取ることが難しいユーザもいる。そこで、補助モードを用意して、ユーザに入力タイミングを示す機能を実現した<sup>8)</sup>。また、リズムに関しても学習モードを用意し、ユーザの好みのリズムを設定しておき、それを用いた旋律の生成ができるようにした<sup>7)</sup>。第5章と6章でリズムパターンの利用と学習について説明する。

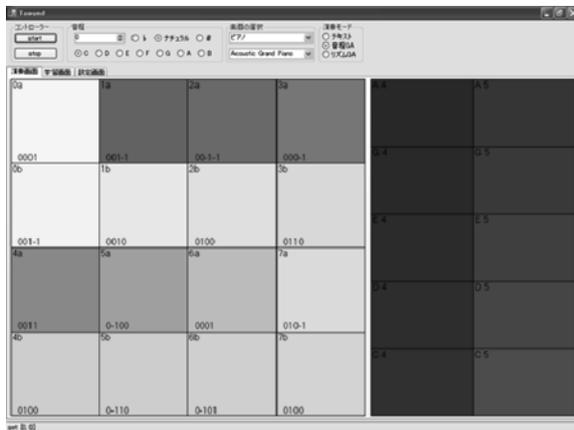


図1 即興演奏システムのユーザインタフェース

#### 4. 旋律パターンの学習

4音旋律にユーザの好みのパターンを追加する方式について検討した。好みのパターンがあれば学習により収束すると考え、GA (遺伝的アルゴリズム) を用いて4音旋律パターンを学習する機能を実現した。GAを作曲に応用する研究は複数あるが<sup>11)</sup>、本研究では旋律パターン (音程列パターン) とリズムパターンを分離し、それぞれをGAで学習している。また旋律パターンとして4音旋律パターンを対象を限定している。GAを使用するメリットは、ユーザの好みを反映するのみでなく、交叉や突然変異によりユーザが思いつかない新しいパターンを提示することができる点である。

##### 4.1. 遺伝子型と表現型

本研究では、GAで使用する旋律パターンの表現型と遺伝子型を以下のように規定した。

表現型：音高に基づく記述であり、3章で説明した即興演奏システムの画面に表示される記述である。例えば、(0 1 0 1)は開始音がドであれば(ドレドレ)を表している。

遺伝子型：音程に基づく記述である。表現型の隣接する数値の差分となっている。例えば、表現型が(0 1 0 1)の場合、遺伝子型は1-0, 0-1, 1-0を順に並べた(1 -1 1)となる。これは、音が順に、1音(2度)上がり、1音(2度)下がり、1音(2度)上がることを示している。

2章で述べた旋律分析では、音高列(ドレドレ)は(60 62 60 62)と表し、それに対応する音程列は(2 -2 2)と表した。これに対して、遺伝子型の(1 -1 1)は、開始音がド・レ・ファ・ソ・ラの何れかであれば図2に示すように、音程列(2 -2 2)に対応し全音(2)の幅で音が変化する。一方、同じ遺伝子型の(1 -1 1)は、開始音がミ・シの何れかであれば音程列(1 -1 1)に対応し半音(1)の幅で音が変化する。このように音程列ではなく、全音と半音の区別をせずに遺伝子型を記述する理由は、{0, ±1~±7}の数値の組合せで上下1オクターブの音の変化を表すためである。遺伝子型を音程列そのものとする、上下1オクターブの音の変化は{0, ±1~±12}の数値の組合せになるため、組み合わせの爆発を起こしてしまう。



図2 音と音程

##### 4.2. GAの手順

本節では、本研究におけるGAによる旋律パターンの学習の手順を説明する。

###### (1) 初期世代



図3 旋律学習における設定画面

初期世代は{0, 1, -1}の各値をそれぞれ40%, 30%, 30%としてランダムに16個体生成する。例えば(0 -1 1)や(1 0 0)等の4音旋律が生成される。初期世代を{0, 1, -1}の値の組合せに限った理由は、旋律分析の結果から、実際の楽曲には同音(0)および隣接する音の間の遷移(±1)が多いことによる。ここで、{0, 1, -1}の各値の出現確率は図3に示

す設定画面でユーザが変更することができる。

(2) 次世代の生成

第  $n$  世代の集団から、ユーザが好みの固体（旋律パターン）を任意個選択する（親の候補が求まる）。学習画面を図 4 に示す。ユーザが各パターンの Play ボタンを押すと旋律が生成されるので、ユーザは自分の好みであればチェックを入れる。画面に表示された全 16 パターンに対してユーザの選択が終了すると、システムは、親の候補の中から任意の 2 つを親として、1 点交叉および突然変異で子を生成する。



図 4 旋律パターンの学習画面

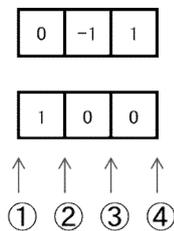


図 5 遺伝子の交叉点

交叉点は、図 5 の①～④の箇所において等確率（各 25%）で生じる。①と④の場合は実質的に交叉しない。例えば、(0 -1 1) と (1 0 0) が親として選択され、③の位置で交叉が生じると、子の候補として (0 -1 0) と (1 0 1) が生成され、この内の 1 つをランダムに選択して子とする。

突然変異は、子の各遺伝子において 20% の確率で生じ、突然変異が生じる際の新しい値の出現割合を、以下のように規定している。

- 0: 25%, ±1: 各 25%, ±2: 各 6%, ±3: 各 2.25%, ±4: 各 2.25%, ±5: 各 0.5%, ±6: 各 0.5%, ±7: 各 1%

このように、各値の出現に異なる重みを与えている理由は、旋律分析の結果から実際の楽曲では近い音の間の遷移が多いことによる。また、旋律分析結果から ±

7 (1 オクターブの音の遷移) は ±5 や ±6 より多くしている。なお、交叉点の数、突然変異の確率、および変異が起きる際の各値の重み付けは図 3 に示す設定画面でユーザが変更することができる。

以上述べたように、親の候補から 2 親の選択、および交叉と突然変異による子の生成を、次世代の個体数が 16 になるまで繰り返す。ここで、当初のシステムでは、親と同じ遺伝子の個体は、確率的に残ることはあったが、確実に残す仕組みはなかった。そのため、気に入った旋律パターンが次世代で消えてしまうこともあった。そこで、エリートを残せる（遺伝子をそのまま残す）ように改良した。エリートを残す方法は以下の通りである。

- ・ユーザが複数回選択したパターンを、選択回数により重み付けする。
- ・ユーザが選択した全パターンを分析し、{0, ±1 ~ ±7} の数値の出現回数を調べ、出現回数の多い数値を含むパターンを重み付けする。
- ・重み付けられたパターンの上位から、ユーザが設定したエリートの数に従い、パターンを選択する。

(3) パターンの収束

上記(2)で述べたように第  $n$  世代のパターンからユーザが好みのパターンを選択して親の候補を決め、その中から次世代の個体（パターン）が生成される。この世代を繰り返すことにより、ユーザに好みのパターン群があれば、パターン群が収束すると考えられる。本研究では、予備実験をした結果、3 世代目にユーザが選択した任意個の個体（パターン）を旋律合成に用いることとした。

(4) 旋律合成に用いる旋律パターン

上記(1)(2)(3)の学習を実施することにより旋律合成に用いる旋律パターンが数個得られる。そこで、旋律パターンが 16 個得られるまで上記の学習を繰り返す。学習の実施状況によって異なるが、3~4 回の学習で 16 のパターンが得られている。得られたパターンはテキストファイルに保存され、後の即興演奏でそのファイルを読み込み、旋律パターンとして使用することができる。

音の遷移は、1 オクターブ以内に限っても 0 ~ ±7 の 15 通りであり、これらを単純に 3 つ組み合わせると 3375 通りとなる。一方、現実には {0, ±1} が多く用いられ、他の出現頻度は低い。本方式は現実的に即してユーザの好みを反映させる仕組を提供している。ユーザは 3375 個のパターンから好みのパターンを見つけ出すのではなく、現実的な 16 個のパターンから好みのパターンを選び、それを 3 回繰り返す。この 16 × 3 個のパターンからの選択を 3~4 回行うことで(計 144 ~ 192 パターンからの選択で)、好みのパターンを決めることができる。

#### 4.3. 学習の実験

学生 12 名に対し、旋律パターンの学習の実験を行った。5 名は各 10 回実施し、7 名は各 2 回実施した。その結果、半数強の被験者の実験において、学習で最終的に得られる 16 パターンの中に、幾つかの特定のパターンが現れる傾向があった。例えば、以下のような極端な結果もあった。

(-1 1 -1) が 13 個で、他は以下の 3 パターン：  
(-1 -1 -1), (-1 1 2), (2 1 -1)

残りの被験者では、適度にパターンが分散した。例えば、以下のような結果となった。

(0 1 0), (0 -1 0), (0 -1 -1), (0 1 4),  
(0 -2 -1), (-1 0 2), (-1 0 -2),  
(1 -1 0), (-1 1 -1), (-1 1 2),  
(-1 -1 2), (-1 -2 0), (-1 2 -1),  
(1 -2 -1), (-2 0 -1), (-2 1 0)

後者の結果はそのまま即興演奏システムの旋律パターンとして利用することができる。しかし、前者のパターンは特定のパターンに収束しすぎており、そのまま即興演奏システムに利用すると、同じパターンが多く曲の生成が難しい。このような場合は、そのユーザの別の学習結果と合わせるか、あるいはシステムの用意する標準的なパターンにユーザの好みのパターンを混ぜる等の仕組みを用意し、旋律合成に用いるパターンを作成する必要がある。

### 5. リズムの利用

#### 5.1. リズム補助機能

本研究の即興演奏システムでは、基本的にユーザがリズムを取って演奏するが、リズムを取ることが難しいユーザがいることを想定し、補助機能を実現した。演奏画面の右側の開始音領域の色を変えることにより、入力のタイミングを示す方式を実現した。この方式は市販の電子楽器に実装されているが、実際に使ってみた所、この方式では、知っている曲については入力の補助になるが、未知の曲の入力をタイミングよく行うことができないことが分かった。

そこで、入力のタイミングを明示するため、以下のようにタイミングを示す絵を開始音領域に提示する方式を実現した。

- ・外側に黒い円、および内側に小さい赤い円を提示する。
- ・赤い円を徐々に大きくし、入力のタイミングで外側の黒い円と重なるようにする。

この方式がどの程度有効であるかを定量的に示す方法を検討している。

#### 5.2. 自動演奏

本システムでは、4 章で述べた旋律パターンの学習

結果と、6 章で述べるリズムパターンの学習結果を使って、自動演奏する機能を実現している。学習で得られた旋律パターンとリズムパターンの中から、それぞれランダムに選んだパターンを使用してメロディを生成している。

### 6. リズムパターンの学習

本説では、リズムパターンを GA を用いて学習する方式について説明する。

#### 6.1. 遺伝子型と表現型

リズムパターンを 16 ビットの 2 進数で表して遺伝子型とする。これは、4/4 拍子の 1 小節に対応している。各ビットを 16 分音符の長さとし、1 の箇所が発音されるとする。休符は表現しない。表現型と遺伝子型は、例えば以下のように対応している。

表現型： 4 分音符 4 つ ⇒  
遺伝子型： 1000100010001000  
表現型： 8 分音符 8 つ ⇒  
遺伝子型： 1010101010101010

#### 6.2. GA の手順

##### (1) 初期世代

初期世代は  $n$  が偶数の  $2^n$  桁のビットを 0 として (16 分音符を生成しないため)、 $n$  が奇数の  $2^n$  桁のビットを 0 か 1 にランダムに生成し、16 個体生成する。

##### (2) 次世代の生成



図 6 リズム学習における設定画面

第  $n$  世代の集団から、ユーザが好みの固体を任意個選択する。システムは、その中から任意の 2 つを親として、1 点交叉および突然変異で子を生成する。1 点交叉は旋律パターンの学習と同様に、各交叉点候補が等確率で選択される。突然変異は、遺伝子の値 1/0 を 20% の確率でビット反転することにより実現する。なお、交叉点の数、突然変異の確率、および変異が起きる位置は図 6 に示す設定画面でユーザが変更することができる。

上記のように 2 親の選択、および交叉と突然変異による子の生成を、次の世代の個体数が 16 になるまで繰り返す。

##### (3) パターンの収束

リズム学習では、予備実験をした結果、6世代目にユーザが選択した任意個の個体（リズムパターン）を旋律合成に用いることとした。

#### (4) 旋律合成に用いるリズムパターン

旋律合成に用いるリズムパターンが16になるまで上記の学習を繰り返す。得られたパターンはテキストファイルに保存され、リズムの補助や自動演奏に使用することができる。

リズムに関する本学習方式も音程列の学習と同様に、現実には即してユーザの好みを反映させる仕組みを提供している。

#### 7. おわりに

本研究では、楽曲の旋律分析の結果から4音旋律パターンを用いて即興演奏するシステムを開発した。ユーザが旋律パターンを選択し、開始音領域に4回入力すると、指定された開始音から4音旋律が順に生成される。本システムにはリズムの補助機能や自動演奏機能もある。さらにユーザの好みを反映するため、旋律パターンとリズムパターンを遺伝的アルゴリズムにより学習するシステムを実現した。学習のパラメータをユーザが設定できるようにしている。現実にも多用されるパターンに基づき、かつユーザに合わせて学習することが可能となった。学習された旋律パターンは即興演奏システムや自動演奏機能で使用される。また学習されたリズムパターンはリズム補助機能や自動演奏機能で使用される。

旋律パターンの学習の実験結果より、パターンの収束が強すぎる場合があることがわかった。そこで、そのような場合のパターンの利用方法を検討する必要がある。また、リズムパターンの学習についてはまだ実験が十分ではないため、今後実験と改良を行う予定である。また、リズム補助機能についても有効性の評価実験が必要である。現在、拡張性の観点からシステム全体の作り変えを行っており、即興演奏、旋律パターン学習、リズム補助および自動演奏の機能まで終了している。今後リズム学習を実験と並行して作り変える予定である。現在、本即興演奏システムは研究室内で利用しているが、システムの作り変えの完了後、外部のユーザに使用してもらって評価する方法を検討したい。

#### 参考文献

- 1) Yu Nishibori, Toshio Iwai, "TENORI-ON," Proc. of the 2006 International Conference on New Interfaces for Musical Expression, pp.172-175, 2006.
- 2) Jieun Oh, Jorge Herrera, Nicholas J. Bryan, Luke Dahl, Ge Wang, "Evolving The Mobile Phone Orchestra," Proc. of the 2010 Conference on New Interfaces for Musical Expression, pp.82-87, 2010.
- 3) 土下竜人, 中村滋延, 西田紘子, "Kinect を用いた音楽演奏ツールの制作," 情報処理学会研究報告, 2011-MUS-93(4), pp.1-3, 2011.
- 4) 出口幸子, "液晶タブレットを使用して旋律を生成するシステムに関する研究," 情報処理学会研究報告, 2007-MUS-70, pp.25-30, 2007.
- 5) 出口幸子, 平岡晋也, 弘田達夫, "液晶タブレットの入力実験と応用システム," 人工知能学会全国大会(第21回)論文集, 3B5-2, 2007.
- 6) 出口幸子, C. S. Sapp, "旋律合成システムの試作—旋律パターンの利用—," 電子情報通信学会論文誌, Vol.J89-D, No.3, pp.595-597, 2006.
- 7) 出口幸子, 青木太一, "音楽即興演奏インタフェースのための旋律パターンの学習," 電子情報通信学会2009年総合大会講演論文集, D-12-50, 2009.
- 8) 出口幸子, 三家本祥平, 小迫隆大, 黒瀬能幸, "日本ポピュラー音楽の3作曲家の旋律分析と楽曲分類," 日本音楽知覚認知学会, 平成20年度春季研究発表会資料, pp.59-64, 2008.
- 9) D. Huron, The Humdrum Toolkit Reference Manual, CCARH, Stanford University, 1994.
- 10) Michael Good, MusicXML, <http://michaelgood.info/publications/music/musicxml-an-internet-friendly-format-for-sheet-music/>
- 11) 安藤大地, 丹治信, 伊庭斉志, "ECを用いた作曲支援システムと作曲モデルの客観的評価手法," 情報処理学会研究報告, 2006-MUS-068, pp.29-34, 2006.