

多声音楽の旋律スタイルの文法的なモデル化および自動生成

田中 翼^{1,a)} 古川 聖¹

受付日 2012年7月4日, 採録日 2013年1月11日

概要: 本稿では, 多声音楽の自動作曲において新しい旋律スタイルを自動生成する手法を提案する. 本手法において旋律スタイルは, 書き換え規則の集合からなる文法として表され, その文法規則は遺伝的機械学習システムであるクラシファイアシステムを用いて生成される. これまでの文法的なアプローチの自動作曲研究においては, 多声音楽をいかに文法的に扱うかという問題や, オリジナルな旋律スタイル自体を自動生成する問題はあまり重点的に研究されてこなかった. そこで本稿では, 多声音楽の生成過程を, 各旋律が互いに参照し合いながら, 声部間での書き換え規則の非同期的な適用によって成長していくプロセスとしてモデル化を行う. そしてモデルの要素としての文法規則をクラシファイアシステムを用いて自動生成することで, 新しい旋律スタイルを創発させる手法を提案する. 生成楽曲の評価実験の結果, 文法ルール数を小さく設定することが複数の観点からの高評価につながることや, 各声部が別々のルールに基づく「複数スタイル」の楽曲が「旋律の動きの豊かさ」において高評価を得ることが明らかになった.

キーワード: 自動作曲, 多声音楽, 旋律スタイル, 文法, クラシファイアシステム

Grammatical Modeling and Automatic Generation of Polyphonic Melodic Styles

TSUBASA TANAKA^{1,a)} KIYOSHI FURUKAWA¹

Received: July 4, 2012, Accepted: January 11, 2013

Abstract: In this paper, we propose a method to generate new melodic styles in automatic composition of polyphonic music. In the proposed method, a melodic style is represented as a grammar that consists of rewriting rules, and the rewriting rules are generated by a classifier system, which is a genetics-based machine learning system. In the previous studies of grammatical approaches, the problem of how to treat polyphony and that of generating new melodic styles automatically haven't been studied very intensively. Therefore, we have chosen to tackle those problems. We modeled generative process of polyphonic music as asynchronous growth by applying rewriting rules in each voice separately. In addition, we developed a method to automatically generate grammar rules, which are the elements of the polyphony model. The evaluation experiment revealed that setting the number of grammar rules to a small number leads to high evaluations and that "multi-style" pieces, which have different melodic styles in respective voices, have higher scores than "single-style" pieces from the standpoint of "diversity of melodic movement."

Keywords: algorithmic composition, polyphony, melodic style, grammar, classifier system

1. はじめに

自動作曲における自動という概念は2つの側面に分けて考えることができる. 一方は, 既存の音楽理論のもとでの

楽曲のリアライズや, 既存の楽曲スタイルの模倣, 芸術家が新たに考えた音楽アルゴリズムの遂行作業を手計算ではなく自動化するといった, ルーティンワーク的作業をコンピュータが代行する「消極的な」自動化であり, 他方は, いまだ存在しない新しい形式の楽曲をコンピュータに生成

¹ 東京藝術大学大学院美術研究科
Graduate School of Fine Arts, Tokyo University of the Arts,
Toride, Ibaraki 302-0001, Japan

^{a)} t-tsubasa@y4.dion.ne.jp

本稿に関連する動画コンテンツを以下で視聴できる.
<http://www.youtube.com/watch?v=NBHnpA6Mt00>
<http://www.nicovideo.jp/watch/sm19919850>

させ、芸術的な創作の根幹となる部分をコンピュータに行わせようとする「積極的な」自動化である。大まかな方向性として、本稿^{*1}での我々の関心は、後者の自動化に踏み込むことにある。従来、具体的な音楽作品のメタレベルの構造に相当する楽曲スタイルあるいは音楽理論は、人間が考案し、作り出してきた面が大きく、自動作曲の研究においても人間が設定している場合が多い。だが、そのような領域を部分的にでも自動生成の対象に含めなければコンピュータを用いて初めて作り出されるオリジナルな楽曲の実現は難しいと考えられる。したがって、曲自体だけでなく、そうしたメタレベルの構造を自動生成することは自動作曲における重要な研究課題の1つだと我々は考える。

本稿ではそのような研究の手始めとして、旋律スタイル（旋律法）の自動生成に焦点を当てる。それは旋律が音楽的要素の中でも最も基本的で重要なものの1つだからである。旋律スタイルを明示的に形式化するにあたっては、旋律スタイルを文法として形式化するアプローチの研究が従来からなされており、有効であると考えられる。そこで我々にとっての課題は、旋律の文法の自動生成ということになる。ここで旋律の文法を自動生成し、本格的な音楽作品の生成を試みるにあたっては、解決すべき主要な問題が2つある。第1の問題は多声音楽をいかに文法で扱うかという問題である。多声音楽は言語や単旋律のように単純な1つの線状のシーケンスではないため、取扱いの困難が存在する。第2の問題は、人間が試行錯誤で文法を設定するのでも、既存の作品からの学習によるのでもなく、いまだ存在しないスタイルの文法をいかに生成するかという問題である。

これら2つの問題への取り組みとして、本稿では多声音楽の一般的な文法モデルを構築し、その具体的な文法規則を自動生成する手法を提案する。ここで我々は、本稿で取り扱う多声音楽を、ある声部間の音楽的制約を満たしながら、複数の独立した声部の旋律が同時進行する音楽として定義しておく。ただし各声部は同時に2音以上発音することはないものとする。また、声部間の音楽的制約は人間が手動で設定するものとし、自動生成の対象ではないことをあらかじめ断っておく。つまり対位法的（声部間的）な側面はモデル化のみで自動生成の対象とはせず、あくまで旋律的な側面に自動化の焦点を当てるということである。主に想定している楽器の編成は、人間の奏者またはMIDIによるピアノであり、曲のデータは声部数分の、音高（音階上の度数）と音価の組を要素とするリストとして与えられる。このリストは適宜MIDIや楽譜に変換される。

本稿の構成は以下のとおりである。2章では、本研究と既存研究との関係について述べる。3章では、単旋律を対

象とし、その生成過程を、書き換え規則に基づく成長プロセスとしてとらえる枠組みについて記述する。4章では、第1の問題（多声音楽の文法的な扱い方）に対し、単旋律の生成モデルを、非同期的、並列的に拡張した多声音楽のモデルを提案する。5章では、第2の問題（新しいスタイルの生成）に対し、クラシファイアシステムを用いて文法規則を獲得する手法を提案し、実際に生成した楽曲を紹介する。6章では生成曲の評価実験について述べる。7章ではまとめと今後の課題について述べる。

2. 関連研究

音楽の中に文法を見い出そうとする研究は多数存在している。特に言語学者Chomskyの生成文法は大きな影響力を持っており、音楽学者Schenkerの調性音楽の階層的な楽曲分析を生成文法と結び付けたGTTM[2]や、ジャズのコード進行のモデル化、民族音楽の旋律のモデル化等の応用がなされている[3]。

作曲に重点をおいた文法的なアプローチの研究としては、L-Systemを用いた旋律生成の研究がさかんに行われてきた[4], [5], [6], [7], [8]。L-Systemは、元々は生物の細胞分裂や植物の成長過程のモデル化に用いられた文法モデルであり、様々な目的対象を記号列としてコード化し、書き換えルールとして形式化された文法を記号列に反復適用することで、記号列を成長発展させていくシステムである。L-Systemは終端記号と非終端記号の区別のない一種の生成文法とも考えられ、決定論的なもの、確率的なもの、文脈依存的なもの、パラメータ付きのもの等、様々なバリエーションが考案され、音楽にも応用されている[3]。L-System等の書き換え規則を用いた方法はシンプルさ、高速性、汎用性等の美点があり、音楽以外にも様々な物事の生成モデルとして応用されている。特に近年ではコンピュータ・グラフィックスの需要の高まりから、自然景観や建物の外観等を高速に生成するプロシージャル技術としてさかんに研究されている[9], [10]。多重音を扱ったL-Systemの研究としてはDuBoisによる文献[8]がある。DuBoisは与えられた旋律に対して文法規則を適用することにより、リアルタイムに伴奏を導出するような枠組みを提案した。この枠組みでは、元の旋律に対して、規則による変形を施したのによって逐次的に応答するような伴奏部が生成される。しかしこの枠組みでは、元の旋律に伴奏部が従属するために伴奏部の独立性が弱かったり、伴奏部の声部数が変化したりするといった点で、ポリフォニーよりもホモフォニーに近いと考えられ、本稿とは対象とする形式が異なる。また、本稿では必ずしもリアルタイム性を重視せず、作品の楽譜を最終的に生成できればよいと考える。また、同じく多重音を扱った編曲の研究として、多重音を含む音楽オブジェクト間の包含関係や二項演算を定義し、編曲へ応用した文献[11], [12]がある。これらの研究では、ある楽曲およ

^{*1} 本稿は著者自身による国際会議の予稿[1]を元に加筆修正したものである。5章までの図8以外の図はすべてそこからの引用である。

びその編曲例を参照した演算によって未知の楽曲に対する編曲を行う枠組みが提示されているが、作曲の問題への応用の可能性については言及されていない。

そのほかに、楽曲コーパスの中から文法を自動的に見つけ出す文法推論やスタイルの模倣の研究がある。それらを楽曲の生成につなげる研究の例として、ジャズの旋律に内在するマルコフ的な文法を楽譜から統計学習する文献 [13] や、可変長の文脈を用いた文脈依存的文法により無用なルールの学習を最小限に抑え、原曲のスタイルへの忠実性を保とうとする研究 [14] 等があげられる。これらの研究は、旋律の文法の獲得を目的とする意味では本稿の立場に近いものである。ただし、これらは、学習対象の楽曲への依存度が強く、元の曲の断片がそのまま出現することが多い等、本稿の目的とするオリジナルな楽曲スタイルの生成には不向きだと考えられる。

文法的なアプローチの自動作曲の既存研究を見渡したとき、いまだ十分に解決されていないと思われる重要な問題がある。それは多声音楽をどのように扱うかという問題である。生成文法におけるツリー状の階層構造や、L-system における 1 本の記号列や、マルコフ的なシーケンスといったものは、単旋律やコード進行を表現するには適しているが、独立した複数の系列が同時進行する多声音楽の構造を扱うことは困難である。多くの研究が単旋律を主な対象としていることも、そこに起因している面があると思われる。また多声部を書き換え規則で表現する例が示されている [5] 等の研究も存在するが、そこでは後に論じるように、複数の独立した状態系列をひとまとめにして多声部を扱っている。このような方法をとると、旋律間の独立性が損なわれるか、状態の数が膨大化して文法が過度に複雑になるか、どちらかに陥る恐れがある。そのため、この問題を解決するには新しいモデルが必要だと考えられる。

文法的なアプローチ以外での多声音楽の生成研究としては、対位法のルールを満たす解を探索する研究 [15] や、確率モデルの統計学習によるもの [16] 等が存在するが、それらは古典的な音楽理論のもとで楽曲をリアライズする研究であり、オリジナルな音楽スタイルの生成を目的とする本稿とはスタンスが異なる。

本研究で文法規則の生成に用いるクラシファイアシステム (CS [17], [18]) は、if-then ルールの集合を遺伝アルゴリズム (GA) を用いて探索するシステムであり、視覚的な形態のデザインのための規則を自動生成する研究 [19] 等にも応用されている。書き換え規則は if-then ルールとして扱うことができるため、CS は本研究にとって有用性が高いと考えられる。遺伝的手法を旋律の文法の生成に用いた既存研究に文献 [7] がある。この研究では、突然変異および交配の演算が文法ルールの変形、多様化に用いられているのみであり、GA による文法ルールの探索は行われておらず、それについては今後の課題とされている。

以上をふまえ、本稿で提示する研究の独自性は次の 3 点にある。(1) 文法的アプローチと多声音楽の生成との間のギャップを埋めるための 1 つのモデルを提示すること。(2) 旋律の文法の生成にクラシファイアシステムを応用すること。(3) 生成文法や L-system とは異なり、旋律生成への文法的アプローチを、人間の認知的な側面や、既存の楽曲の分析や、自己相似性の観点からではなく、作曲家の旋律作成プロセスの実例から根拠づけること (次章を参照)。

3. 旋律生成への文法的アプローチ

3.1 単旋律の既存モデル

書き換え規則は、生成文法や L-System 等の文法において用いられる生成規則であり、様々な物事の生成モデルの文法に用いられてきた。図 1-A (上) に示されるように、書き換え規則は predecessor と呼ばれる左辺の記号列を successor と呼ばれる右辺の記号列に置き換えるものである。この規則を、図 1-A (下) のように、最初に与えられた Axiom と呼ばれる記号列に対して適用することで、次世代の記号列を生成する。そして、その新しい記号列に対し、さらに同じ規則を適用するというように、記号列を段階的に成長させていく。図 1-B は、生成された記号列の第 3 世代を左から順にグラフィカルに解釈したものである。この図形はヒルベルト曲線と呼ばれ、空間を効率的に充填する図形として数学者ヒルベルトが発案したものである。この解釈で、“F” は一定の長さ前進、“-” は左へ 90 度回転、“+” は右へ 90 度回転を表す。“X” および “Y” は解釈時には無視される。また、Prusinkiewicz は図 1-B のヒルベルト曲線を、左下をスタート地点として、X 軸の移動を発音、Y 軸の移動を音階上の移動と解釈し、図 1-C のような旋律を生成した [4]。このように、書き換え規則は旋律の生成のための文法としても用いることができる。

3.2 旋律文法と作曲プロセス

前節で示した L-System による旋律は、他の多くの研究

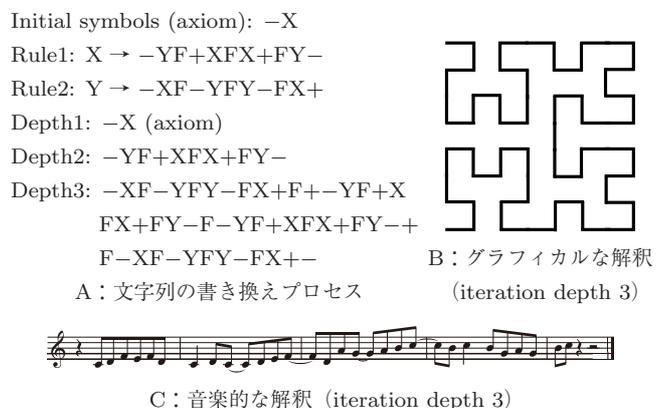


図 1 L-System 文法によるヒルベルト曲線とその音楽的解釈 [4]
 Fig. 1 Hilbert curve which is generated by a L-system grammar and its musical interpretation [4].

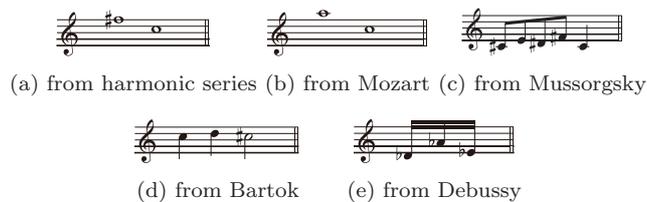


図 2 Messiaen の引用する短音型の例
Fig. 2 Short figures quoted by Messiaen.

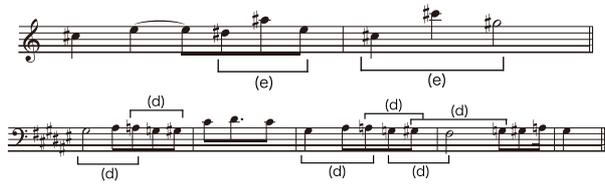


図 3 Messiaen の作品中での短音型の使用例。上：“Arc-en-ciel d'innocence”，下：“Les Mages”
Fig. 3 Examples of Messiaen's works in which the short figures are used. Upper: “Arc-en-ciel d'innocence.” Lower: “Les Mages.”

例と同様，もともと音符でない対象に音をマッピングする形で旋律を生成したものである。そのため，書き換え規則の音楽的な意義が必ずしも明らかではない。その一端を明らかにするため，作曲家による旋律の作曲プロセスと書き換え規則の関係について述べておく。

20 世紀の大作曲家の 1 人である Olivier Messiaen は，その著作 [20] の中で自らの作曲技法を，自身の作品の例を多数あげながら体系的に明らかにしている。我々は，その中の旋律論にあたる第 8 章に着目する。この章で Messiaen は，他の作曲家のイディオムや，民謡，グレゴリオ聖歌等を想起しつつ，これらの中に現れる特徴的な短い音型 (図 2) を，自分なりに引用して旋律の構成に用いる方法を解説している。それらの音型は Messiaen 自身の作品にたとえば図 3 のように用いられている。

これらの譜例では，1 つの短い音型が，1 つの旋律の中に頻繁に用いられ，その旋律を特徴づけている。さらに，Messiaen はそのような短音型を複数合成してより大きな音型を構成する方法も示しており，実際の作品中での合成音型の使用例を示している (図 4)。このような旋律の構成方法は，書き換え規則の用いられ方に通じるところがある。たとえば，図 4 の合成音型は，図 5 (上) に示す 3 つの書き換え規則 $R_1 \sim R_3$ を用いて図 5 (下) のような手順で段階的に生成した結果と見なすことができる。ここで， R_1 は図 2 の (a)， R_2 は図 2 の (d) から作られた規則であり， R_3 は R_1 を反行させて作られた規則である。このように，Messiaen の旋律の構成法は，書き換え規則の観点から自然にとらえ直すことができる。我々は，このような見方を全面化し，旋律法のモデルとする。少数の短い音型を個々の書き換え規則で表し，それらを次々と適用して旋律を合成していくこのモデルにおいては，音型の反復を生み出した

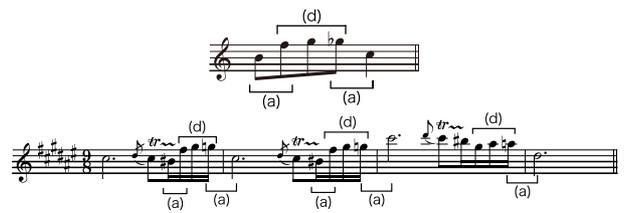


図 4 合成によって形成された音型 (上) と Messiaen の作品 “Chant d'extase dans un paysage triste” におけるその使用 (下)
Fig. 4 A compound figure (upper) and its use in Messiaen's work, “Chant d'extase dans un paysage triste” (lower).

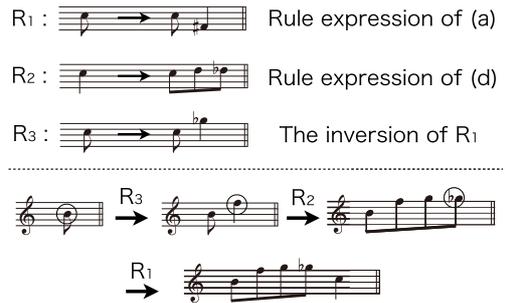


図 5 音型とその合成の書き換え規則による表現
Fig. 5 Rule expressions of short figures and their compound.

めに特徴を認知しやすい旋律が形成されるとともに，複数の音型を合成することで機械的な反復を防ぎ，有機的な旋律の形成が期待できる。そして何よりこのモデルは，有力な作曲家の実際の作曲プロセスに近いという点からも良いモデルであると考えられる。

4. 多声音楽のための非同期モデル

本章では，前章で述べた単旋律の生成モデルの多声部への拡張について議論する。4.1 節では単旋律の場合の書き換え規則の predecessor と successor の中身をポリフォニックなデータ構造に愚直に拡張するモデルでは不都合が生じる恐れがあることを示し，4.2 節では，その不都合を解消するためのモデルを提案する。

4.1 単旋律からの拡張の困難

多声部を書き換え規則によって扱う形式はすでに文献 [5] 等の研究で触れられている。そこでは次のような表記方法がとられている：

$$(CE)|(GC) \rightarrow D(CE)$$

括弧内の音は同時に演奏される音であり，predecessor 内の “|” は，和音 (CE) の後に和音 (GC) があるというコンテキストを表し，successor 内は単音 D の後に和音 (CE) がくるという意味である。この表記では，括弧内の音をつねに同時に鳴らすことになるため，独立したリズムを持った本格的な多声音楽が扱えない。これを愚直に解決するためには，各ノートのピッチのほか長さを導入し，

predecessor と successor を次のような類いの形式で表現する必要がある（2声の場合のみ記す）：

$$\left[\begin{array}{l} (p_{11}, d_{11}), (p_{12}, d_{12}), \dots, (p_{1n}, d_{1n}) \\ (p_{21}, d_{21}), (p_{22}, d_{22}), \dots, (p_{2m}, d_{2m}) \end{array} \right]$$

ここで、 p_{ij} は第 i 声部の第 j 音のピッチ、 d_{ij} は第 i 声部の第 j 音の長さである。 p_{ij} は具体的な数値ではなく、休符記号 r または、どのピッチであってもよいという意味のドントケア記号 $\#$ であってもよい。また、両声部の時間の関係として、両方の声部の音価の合計が等しいという条件

$$\sum_{k=1}^n d_{1k} = \sum_{l=1}^m d_{2l}$$

を課すこととする。

ここで問題は、このように愚直に多声部への対応をしようと、リズムのバリエーションの増加にともなって書き換えルール中に想定すべき音数が多くなり、必要なルールの数が声部数に対して指数関数的に増えることである。もし両声部の音の組合せのバリエーションに対応したルール数を確保しなければ、適用できるルールが足りなくなり、それ以上楽曲を成長させることができなくなる恐れがある。それを防ぐため、predecessor 内の音数を制限すると、声部間の組合せが同じルールを多用することになり、多声音楽にとって重要な、声部間の独立性が損なわれる恐れがある。また、この表記法では、1つ1つの規則が各声部内の規則どうしの単なる組合せの列挙になり、個々のルールの意義と可読性が弱まるという欠点も存在する。

4.2 非同期モデルの提案

前節に示した愚直な拡張モデルのような、全声部の情報をまとめて書き換える方法の困難を避けるため、我々は単旋律の書き換え規則を各声部ごとに別々に非同期に適用する図6(下)のモデルを提案する。また、図6(上)に前節で述べた愚直な拡張モデルを示し、両モデルにおける書き換え規則の適用の違いを示す。ただし、この非同期モデルのように声部ごとに別々に書き換え規則を適用する場合、

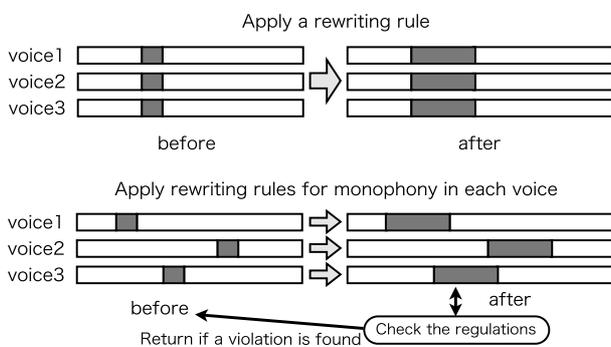


図6 多声部のための愚直な拡張モデル(上)と提案する非同期モデル(下)

Fig. 6 Naive extension (upper) and proposed asynchronous model (lower).

声部間の調和がとれない場合が出てくる。そこで、それを防ぐため、声部間に対して規制を設け、規制を満たしているかどうかを事後にチェックする重要な手順を導入する。ここでいう規制とは、多声音楽としての声部間の良い関係を保つための条件をルールとして事前に手動で与えるものとする（自動生成の対象はあくまで個々の旋律のスタイルである）。ただし本稿では、旋律の形を第1義に考える焦点化のため、和声についての規制は設けないこととし、それについては今後の課題とする。具体的に本稿では、次の4つの規制を設定する：

- 声部の上下関係を明確にするため交叉を禁止する。
- すべての声部に共通した音域の上限、下限を設ける。
- 拍節構造を保つため、小節の冒頭では過半数の声部で発音しなければならない。
- 声部間の独立性を保つため、同時に同じリズムを用いることを禁止する。

この提案モデルのもとでは、初期値 (Axiom) および書き換え規則の集合を与えたうえで、次の3ステップを、所望の長さになるまで反復 (T 回とする) することで楽曲が生成される：

T times iteration {

step 1 : 各声部に対して、ひとつずつ、書き換え規則を適用する対象をランダムに選ぶ。

step 2 : 各声部に1回ずつ、1で選択した対象に書き換え規則を適用し、各声部を同じ長さに成長させる。

step 3 : 延長した旋律に対し、旋律間の関係性の規則をチェックし、違反がなければ終了。もし違反があれば、step1 からやり直し。

}

ここで、すべての声部の旋律の長さをそろえるためすべての書き換え規則において、successor の長さは predecessor の長さよりも L だけ長いものとして単純化して考えることとする。またここで、後のため他の記号の定義も与えておく。 V を声部数とし、各声部の書き換え規則の数は R 個で共通とする。また、 C_i を第 i 声部に対する単旋律の書き換え規則の集合 ($1 \leq i \leq V$)、 c_{ij} ($\in C_i$) を第 i 声部の第 j ルールとする ($1 \leq j \leq R$)。 C_i は声部間で共通にすることもでき、別々にして「複数スタイル」の楽曲とすることもできる。上記の反復プロセスが終了すると、(最初の長さ + $L \times T$) の長さの楽曲が得られる。このモデルは、声部間の独立性を保ち、声部間の組合せの可能性を制限しすぎないため、多声音楽にとってふさわしいと考えられる。愚直な拡張モデルにおいては、書き換えルールに事前に声部間の規制が反映されることとなるが、提案する手法では、書き換え規則の適用後にチェックすることになる。書き換え規則に声部間の規制を織り込むことは、規則の適用が容易である反面、規制できる範囲が predecessor の内部に限

定されるという欠点が存在する。適用後のチェックは楽曲全体を範囲として行うことが可能であり、ローカルな範囲しか規制できない書き換えルールそのものの弱点を補うことができる。また声部数に対してルールの数は $V \times R$ と線型になり、ルール数の肥大化をおさえられる。

なお、複数の声部を並列的に生成するのではなく、1声ずつ順番に生成する方法も考えられるが、その場合、後に作曲する旋律ほど規制を強く受けるという声部間の不平等を生むため、ここでは同時に成長させるモデルを採用した。

4.3 より発展的な対位法的形式の扱いについて

多声音楽は、逆行、反行、拡大、縮小等の音型操作や、カノン、フーガ等の模倣による形式との関係が強い。それらの実現は次章以降で本稿が扱う範囲を超えるが、提案モデルの中でそれらを扱える可能性があるかどうかは重要な問題であるため、本節ではその見通しについて触れておく。

まず、逆行および反行については successor の音型を逆行、反行させた書き換えルールによって扱うことが考えられる。拡大、縮小については、書き換えルールの1回の適用で L だけ旋律を延ばすという前述の制限の中では長さが合わず、通常やり方では扱えない。ただし、たとえば2倍の拡大については、他の声部で複数回ルールを適用するときに successor を2倍に拡大したルールを1回適用することで声部間の拍数を合わせる、といった形で対応することができる。また、たとえば2分の1の縮小については、ある声部において長さ $2L$ の predecessor を半分に縮小するルールを用いたのち、通常ルールを1回適用することで長さを合わせる等の方法が考えられる。逆行、反行、拡大、縮小のルールを適用した後、その形を保つためには、後からその部分に対してさらに他のルールを適用して音型を壊してしまうことを禁じる措置が必要となろう。

カノンについては、声部ごとのルールの共有に加えて、複数の声部においてルールの適用を対応する位置に同時に指定することで実現できる可能性がある。ただし、各声部でのルールの適用位置が対応する箇所でなければならないという制限が加わるため、声部間の規制を満たすのが困難となる恐れもある。フーガについては、ルールの合成によって1声の長い主題を初めに作っておき、次にそれに合う対主題を、転回可能という条件や、主題と対主題の対比に関する条件を声部間の規制に追加して作るというやり方が考えられる。ここで、主題、対主題はルールの合成によって作られた、拡大と同様の新たな長いルールとして扱うことができる。ただし、この主題と対主題の形成ルールは同期的に適用しなければ共存することが難しいと考えられる。断片化された主題が出現するストレッタについては、主題形成ルールの合成を途中でやめたルールによって表現でき、曲の後半でのみ使用する等の制限をつけて用いることが考えられる。

以上のように、提案モデルはこれらの形式を統合できる可能性を持っている。ただし、これらを扱いながら実際に曲を生成できるルール集合を獲得するには、次章で述べる、扱わない場合のアルゴリズムよりも制約が厳しく、より解を見出すのが困難となる恐れがある。本稿ではその問題については扱わず、実現の可能性と困難の指摘にとどめる。

5. 旋律文法の生成手法

前章では、多声音楽を扱うための一般的な文法モデルを提示した。次の問題は、具体的な文法規則を自動生成することである。本章では、ルール集合を獲得するための遺伝的な機械学習方法として知られるクラシファイアシステム (CS [17], [18]) を応用した旋律文法の生成手法を提案する。

5.1 クラシファイアシステム

クラシファイアシステムとは、遺伝アルゴリズム (GA) の研究で有名な John Holland によって提唱された適応的システムである。システムは、クラシファイアと呼ばれる手持ちのルールの集合に基づいて行動を起こし、行動が環境に対して有効な働きかけをした場合に環境から報酬を受け取ることで、自らに有利な規則を学習、獲得し、環境に適応していく。さらに、信頼度の低いクラシファイアは GA によって生成される新しいクラシファイアと入れ替えられ、淘汰される。これにより、価値の高いルール集合が創発されていくことが期待できる。図7にその概念図を示す。

クラシファイアは、if (condition) then (action) という形式のルールである。環境からの情報が condition 部分を満たしているときに、システムは対応する action を環境に送り出す。各クラシファイアには信頼度 (credit) と呼ばれる数値が割り当てられており、報酬をもたらしたクラシファイアの信頼度を相対的に増加させるアルゴリズムによって随時更新される。条件部にあてはまるクラシファイアが複数あるときには、信頼度の高さに応じて確率的に選択される。本研究では、クラシファイアの condition と action に書き換え規則の predecessor と successor を対応させ、環境を生成中の楽曲に対応させることができるため、CS は目的にかなった手法だと考えられる。

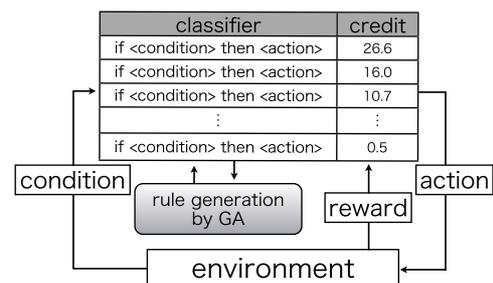


図7 クラシファイアシステム
Fig. 7 Classifier system.

5.2 文法規則の信頼度の評価方法

本研究においてCSを応用するには、旋律法としての望ましさを数値的に評価し、その値を信頼度に反映する必要がある。評価方法を設定するうえでは次のことを考慮する必要がある：

- (1) 少数の規則を多用することで旋律の特徴が際立つと考えられる (3.2節を参照)。
- (2) しかし、規則が少ないと適用できる規則がなくなりやすく、曲を長く成長させるのが困難になる。そのため適用可能なルールや他のルールの適用を可能にするのに役立つルールは高く評価すべきであろう。そのようなルールは、曲の成長への貢献度 (使用された頻度) を基準とすることで評価できると考えられる。
- (3) 規則の形に関する人為的な方向付けをしすぎると、旋律スタイルの自動生成の目的に反する恐れがある。

以上の点をふまえ、我々は次のような観点からルールの獲得に臨み、ルールの信頼度へのフィードバックを設定する。

- 各声部のクラシファイアの数 R をできるだけ小さくおさえる (アルゴリズムを遂行する前に固定する)。
- 個々のルールが曲の成長に貢献した度合い (そのルールが用いられた頻度) を報酬として、そのルールの信頼度に割り当てる。
- 規則の形を方向付ける人為的な評価基準はあえて導入しない。

5.3 非同期モデルへの適用

本節では、4.2節で提示した多声音楽の非同期モデルに対してCSを適用し、書き換え規則の集合を生成する具体的手順を述べる。この手順は次の5ステップからなる：

step1：初期値を設定。

step2：書き換えルールの反復適用による作曲 (規定の反復回数内に所望の長さまで曲を延長できた場合、プロセスを終了)。

step3：信頼度の更新。

step4：成長した長さが前回の最大値以上だった場合、そのルール集合および信頼度を保存しておく。そうでなければ前回まで保存されていたものに戻す。

step5：遺伝アルゴリズムでルール集合を更新し、step2に戻る。

各ステップの詳細は以下のとおりである。step1ではシードとなる楽譜の初期状態を与え、書き換えルール集合 C_i の初期値をランダムに与え、各書き換えルールの信頼度を初期化する。step2では、前章のステップ1から3のステップによって曲を1段階ずつ成長させる。各成長段階で規則の適用のトライアル回数の上限を設け、上限を超えたら手持ちのルール集合では曲の成長がこれ以上不可能であると判断し、step3に移る。各成長段階におけるルールの選択は、

各声部ごとに1回ずつ、信頼度によるランキング選択を行う。step2においてルールの選択、適用を繰り返した結果、曲の長さが所望の長さになった場合、すべてのプロセスが終了となる。このとき、求める旋律文法とそれに基づく楽曲が同時に生成されたことになる。step3では、各ルールの良し悪しを評価する。具体的には、step2での曲の成長に寄与した規則を、規則の使用頻度に応じて評価する。これによって、曲の成長を持続可能なものにし、目標となる楽曲の長さに近づけていく。各規則の信頼度は、(その規則が今回使用された回数 + 前回まで保存されていた信頼度 $\times \gamma$) で更新する。ここで γ は0と1の値をとる実数値であり、前回までの信頼度のある程度割り引いて保存しつつ信頼度の発散を防ぐためのものである。step4では、過去に最も良い結果を残したルールセットを保存しておく。これは、GAでのルールの入れ替えによって、結果が悪くなることを避けるための処置である。step5では、step3での信頼度評価に基づき、信頼度のランキングの低い規則を、遺伝的演算によって生成された新しいルールと置換し、規則の集合をアップデートする。

遺伝的演算には表1のような交叉と突然変異を用いる。ただし単純化のため predecessor の音数は1音とする。図8に一点交叉の手順を例示する。step1で定める初期値については、様々な設定法が考えられるが、本稿では次のように設定する。シードの楽譜の初期状態は、各声部ごとに長さ L の1音とし、各声部の初期値固有に設定した音域内からランダムな音高を選択する。声部間の上下関係を守るため音域は重複させない。 C_i の初期値は、predecessor の長さ、

表1 GAに用いる遺伝的演算のリスト
Table 1 Genetic operators used in GA.

交叉	
successor を一点交叉させ、successor の音価の変化に応じて predecessor の音価を伸縮調節する	
突然変異	
1	successor の内部で音高を1音変化させる
2	successor の内部で2音を入れ替える
3	successor の内部で2音の音高を入れ替える
4	successor の内部で2音の音価を入れ替える
5	successor の内部で1音削除して削除した音の分の音価をいずれかの音に分配する
6	successor の内部で1音の音価を2つに分割して新しい1音を挿入する
7	predecessor の音価を変化させ、successor のいずれかの音に変化した分の音価を分配する
8	successor のいずれかの音を削除し、predecessor の音価をそのぶん減らす
9	successor に音を挿入し、predecessor の音価をそのぶん増加させる
10	ランダムにルールを生成する

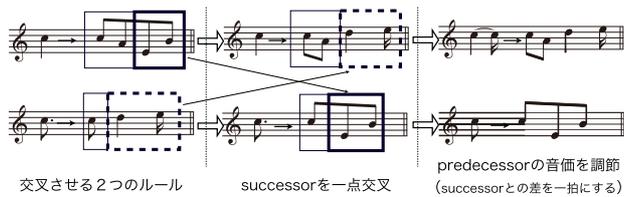


図 8 一点交叉の例

Fig. 8 An example of one-point crossover.

successor 内のオンセット数, オンセット位置, successor 内の各 2 音間の音程の順に可能な範囲内でランダムに選択する. ただし音程は, 上下ともに 1 オクターブを限度とし, 順次進行の確率のみ他の音程よりも高く設定する. 各書き換えルールの信頼度は 1.0 として初期化する.

5.4 楽曲の生成と観察

本節では, 提案手法による旋律文法と楽曲の生成実験について述べ, 生成楽曲サンプルを紹介する. 用いたコンピュータ環境は Mac OS 10.6.8, 2.4GHz Intel Core 2 Duo であり, プログラミング言語には Ruby を用いた.

生成の条件は次のように設定した. 声部数は 3 声とし ($V = 3$), 各声部の規則の数は 7 とした ($R = 7$). 旋法は, Messiaen が理論化し, 自作に用いたことで有名な, 移調の限られた旋法の第 2 番 (MLT2 と略記する) を用いた. 声部数は 3 声 ($V = 3$), 拍子は 4 分の 2 拍子, 曲の長さは 50 拍, 音価の最小単位は 16 分音符, 音域の上限および下限は MIDI 番号で 96 および 36 として統一した. 書き換え規則の形式は次のように限定した. successor のトータルの長さは predecessor に対して 1 拍分長くなるものとし, successor の開始音の音高は predecessor の音と同じとする. 休符は用いないものとし, 各声部の音符の長さは, 声部ごとに別々に設定した最小単位の倍数をとるものとする. これによってポリリズムが可能である. ピッチや書き換えルールは音階に基づくものとする. 書き換えルールとしては, 自動生成する通常のものほかに, 終止形を形成するルールを手動で設定した. このルールは, 任意の音階音から上または下に順次進行し, 二分音符の終止音を追加するものであり, 任意の音階音が終止音となりうる. 終止形ルールは 1 拍ごとの成長段階において 10 回に 1 回適用し, 旋律に区切りを与えるために用いた. 終止形ルールの適用方法は基本的に通常書き換え規則と同様である. ただし, 最終小節のみ全声部同期した終止形を付け加えた. ルール更新における GA の世代間で, 保存するルールの割合は 5/7 とし, それ以外の 2/7 を GA の交叉または突然変異によって置換した. 交叉と突然変異の比率は 7:3 とした.

図 9 および図 11 に生成された楽曲の 2 つのタイプの例を示す. 図 9 の曲は声部間でルールを共有した場合 (それにともない音価の最小単位も共有) であり, 図 11 の曲は声



図 9 生成楽曲例 1 (書き換え規則および音価の最小単位を声部間で共有する場合)

Fig. 9 Example 1 (the rewriting rules and the minimum duration unit are shared by all the voices).



図 10 図 9 の楽曲の生成ルール

Fig. 10 The rules that generated the piece of Fig. 9.



図 11 生成楽曲例 2 (書き換え規則および音価の最小単位は声部ごとに別々の場合)

Fig. 11 Example 2 (the rewriting rules and the minimum duration units are not shared between the voices).

部ごとに異なるルールを持つ場合である. 後者では, 音価の最小単位も声部ごとに異なるように設定した. 両例とも

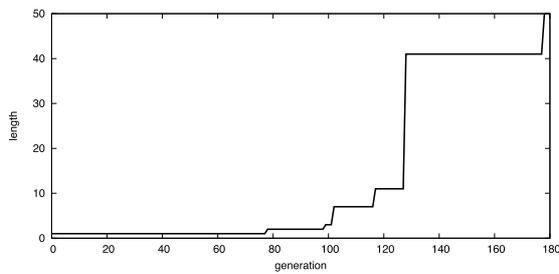


図 12 図 11 の楽曲生成における成長曲線

Fig. 12 Growth process of the piece of Fig. 11.

生成には数分を要した。まず、両例を観察すると、提案モデルにおける声部間の制約の効果を明確に確認することができる。それぞれの声部は音域の衝突を回避し、リズム的な独立や拍節構造が確保されていることが分かる。次に、両例を個別に観察する。図 10 は図 9 の楽曲を生成した文法である。この曲で特に目立つのは R_3 から R_6 にみられるような順次進行による 16 分音符の高速の下降や、 R_4 から R_7 にみられるような上方への 6 度の跳躍 (MLT2 内では 5 度ではなく 6 度)、 R_1 、 R_2 にみられるような 3 度の下降等を含む音型であり、声部間でのルールの共有により各声部で反復され、対位法音楽における摸倣のような声部間の呼応関係を生じさせていることが確認できる。他方、図 11 の曲は、各声部でルールを別々に設定した「複数スタイル」の楽曲の生成例である。上声は非常に細かい音価で激しく動いているが、スイングのような付点のリズムが特徴的な旋律スタイルとなっている。中声は、最小音符は 1 拍の 6 分の 1 と細かく設定されていたが、細かい音価は出現せず、1 拍を 2 分割する音価による、緩やかなリズムをとっている。下声部は 1 拍を 3 分割するリズムに基づいており、中声部とは 2:3 のポリリズムの関係になっている。それに加えて、上声部の細かい動きとの対比も感じさせ、全体として 3 つの旋律スタイルが共存している。この曲の上声部で扱われているような非常に複雑なリズムの領域は、古典的なポリフォニー音楽で扱われないため範例が少なく、コンピュータによる統計的な学習を方法論として自動作曲を行うことは困難である。このような領域に踏み込める点にも、本手法には意義があると考えられる。

図 12 に図 11 の楽曲の生成過程での、世代ごとの成長した曲の長さ (拍数) を図示する。遺伝アルゴリズムの初期世代においては、曲をほとんど成長させることができていないが、世代を経るごとに有効なルールを獲得し、成長の可能性が増大している。

6. 生成曲の評価実験

提案手法が旋律のスタイルと多声音楽の生成にどのような有効性を持つのかを調べるため、多数の楽曲を生成し、作曲の専門家 (学生も含める) による評価実験を行った。

6.1 評価基準

評価基準の設定は、旋律スタイルの評価のため「旋律スタイルの一貫性」、「旋律スタイルの独創性」、「旋律の動きの豊かさ」の 3 つの基準、また多声音楽としての声部間の関係の良さの評価のため「声部間の調和」、「声部間の独立性」の 2 つの基準、それに加えて「楽曲の総合的な良さ」を設定した。これら 6 項目を 7 段階で評価する。7 段階の意味は、7: きわめて高い、6: 高い、5: やや高い、4: どちらでもない、3: やや低い、2: 低い、1: きわめて低い、とする。

評価楽曲の種類の設定においては、文法ルールの数 R および、各声部でルールを共有するか否か (このパラメータを S とし、 $S = 1$ を共有、 $S = 0$ を非共有とする) の 2 つの要因を設定し、パラメータによる評価の変化を調べる。 R についてはモデルを立てる時点で、 R が小さいことが旋律スタイルにとって重要との予想をしていた。それを実証するために R による評価の変化を調べる必要がある。また、声部ごとに異なる旋律スタイルをとる場合と、すべての声部が同じスタイルの場合の違いを調べることは、多声音楽と旋律スタイルの関連性をみるうえで最も大きな差異だと考えられるため、 S の違いを評価対象に含める。

6.2 実験条件

実験参加者は、作曲家あるいは作曲を専攻する学生 26 名 (男性 15 名、女性 11 名、年齢: 20~41 歳) であった。実験に用いた楽曲は、 R について小中大の 3 水準 (7, 14, 21)、それぞれの R について $S = 1$ および $S = 0$ の 2 水準、合計 $3 \times 2 = 6$ 通りの条件設定で、それぞれ 5 曲ずつ、計 30 曲を生成したものをを用いた。すべての曲は約 40 秒の MIDI ファイルであり、聴取順序の影響を相殺するために実験参加者ごとの聴取順序はランダム化した。楽曲の生成の条件設定については、 $S = 0$ のときを含め、音価の最小単位を 16 分音符に固定した。このことと R を可変にしたこと以外は、前章に記述した条件と同じである。声部数や旋法の違い、声部ごとに異なる旋法を持つポリモーダルな楽曲、声部ごとに音価の最小音符が異なるポリリズム的な楽曲等を考えると、組合せの数が膨大になる。評価者が評価できる分量の限界を考慮してこれらは対象外とした。

6.3 実験結果

得られた 6 つの評価値の平均値を折れ線グラフにプロットし、図 13 に示す。まずこの図から、 $R = 7$ のときはすべての評価項目においてニュートラルな 4 の値を上回っており、 R が小であることが提案モデルにとって重要であることが把握できる。ただし、すべての項目において平均値は 4 から 5 付近に位置しており、その効果は非常に顕著というわけではないことが分かる。次に、効果を詳細に分析するため、各評価値に対して R と S を要因とした 2 元配置

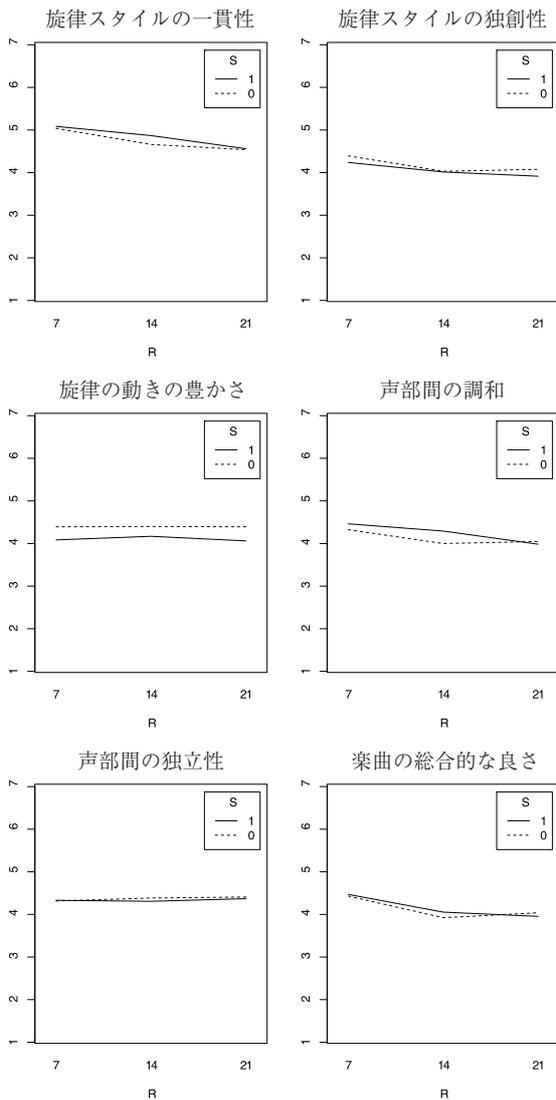


図 13 各評価項目の平均値

Fig. 13 Average scores.

分散分析を行った。その結果、すべての項目において交互作用は現れなかった。Sに関する主効果は「旋律の動きの豊かさ」において現れ、Rに関する主効果は「旋律スタイルの一貫性」、「旋律スタイルの独創性」、「声部間の調和」、「楽曲の総合的な良さ」において見いだされた。「声部間の独立性」においてのみ、いずれの主効果もみられなかった。

主効果のみられた項目について多重比較 (Sに関してはt検定)を行ったところ、表2のような結果となった。ここで $R = 7 > R = 14$ という表記は $R = 7$ のときの方が $R = 14$ のときよりも高評価であることを表すものとする。多重比較の検定方法は基本的に Tukey の HSD 法を用いたが、有意差がみられなかった「声部間の調和」においては、より焦点を絞った多重比較法である Dunnett 法を試みた。この結果から「旋律の動きの豊かさ」に関しては、 $S = 0$ のときの方が $S = 1$ のときよりも高評価だったことが分かる。このことは、複数の旋律スタイルの共存が、一定のポジティブな効果を持つことを示唆する。また、Rに関する

表 2 多重比較の結果

Table 2 The results of multiple comparisons.

評価項目	結果	検定方法	p 値
旋律スタイルの一貫性	$R = 7 > R = 14$	TukeyHSD	0.031
	$R = 7 > R = 21$		< 0.001
旋律スタイルの独創性	$R = 7 > R = 14$	TukeyHSD	0.030
	$R = 7 > R = 21$		0.015
旋律の動きの豊かさ	$S = 0 > S = 1$	t 検定	< 0.001
声部間の調和	$R = 7 > R = 14$	Dunnett	0.045
	$R = 7 > R = 21$		0.001
楽曲の総合的な良さ	$R = 7 > R = 14$	TukeyHSD	< 0.001
	$R = 7 > R = 21$		< 0.001

分析から「旋律スタイルの一貫性」、「旋律スタイルの独創性」、「声部間の調和」、「楽曲の総合的な良さ」についてはいずれも $R = 7$ (小) のときに最大となることが分かった。このことは、R が小さい方が旋律スタイルが向上するという当初の予想を裏づける結果である。

また、本実験では S について主効果が見い出せなかった項目が多いが、今後サンプル数を増やす等、より正確な検証の価値がありそうな仮説として以下のようなものがある：

仮説 1: 「旋律スタイルの一貫性」は共有のときの方が評価が高い。

仮説 2: 「旋律スタイルの独創性」は非共有のときの方が評価が高い。

仮説 3: 「声部間の調和」は共有のときの方が評価が高い。

仮説 4: 「声部間の独立性」は非共有のときの方が評価が高い。

仮説 1 はグラフ上ではつねに成り立っており、ルールの声部間での統一がスタイルの認知につながり、一貫性を高めると考えれば自然だと考えられる。仮説 2 もグラフ上ではつねに成り立っている。仮説 3 は、声部間でのルールの統一は声部どうしの関係を強め、調和を高めると考えれば意味として自然であり、グラフ上では $R = 7, 14$ のとき成り立っている、さらに $R = 14$ のときには t 検定で有意差が確認できている ($p = 0.034$)。仮説 4 に関して、声部ごとに異なるルールを持つと声部の独立性が高まることは意味として自然であり、グラフ上では $R = 14, 21$ のとき成り立っている。

7. まとめ

本稿では、多声音楽のための旋律法を自動生成する手法を提案した。そこではまず、複数の声部の旋律を非同期的に書き換え、成長させるような一般的な文法モデルを構築し、次に具体的な旋律文法規則の集合をクラシファイアシステムによって獲得し、旋律スタイルを生成する手法を提案した。また生成実験により、実際に文法規則と楽曲の生成が可能であることを確認したのち、多数の生成曲を作曲

の専門家によって評価する実験を行った。その結果、非常に顕著な効果というわけではないものの、文法ルールの数を少なく設定することが多くの評価項目を高めることが確認された。それらは、旋律スタイルの一貫性と独創性、声部間の調和、曲の総合評価の4項目である。このことは、少数のルールから旋律を生成するという、モデルを考案した当初の意図が支持されたことをも意味する。また本手法は、異なる旋律スタイルが共存する「複数スタイル」の楽曲を生成できるが、そのような生成曲は、声部間で同じルールを共有する場合よりも「旋律の動きの豊さ」を若干増す効果があることが確認できた。

今後の課題としては、4章で述べたモチーフ操作やカノン、フーガ等の形式を扱えるように本手法を拡張することがあげられる。また我々はこの研究を、楽曲のメタレベルのスタイルをも生成するというより深いレベルから楽曲を生成する研究の一環として位置づけているため、旋律スタイルだけでなく、楽曲のマクロな形式や、旋律間の関係性を規制する声部間的な規則（対位法や和声等）をも自動生成の対象に含め、楽曲スタイルをより包括的に自動生成できるようにしていくことが重要と考えている。また、旋律文法ルールの可読性を生かして人間のユーザが文法を部分的に指定し、それと整合する他の規則の集合を生成したり、提示された文法をユーザが好きなように修正した後、楽曲を再生成するようなインタラクティブな作曲支援システムに本手法を応用することも考えていきたい。

参考文献

- [1] Tanaka, T. and Furukawa, K.: Automatic Melodic Grammar Generation for Polyphonic Music Using a Classifier System, *Proc. SMC*, pp.150–156 (2012).
- [2] Lerdahl, F. and Jackendoff, R.: *A Generative Theory of Tonal Music*, MIT Press (1983).
- [3] Nierhaus, G.: *Algorithmic Composition: Paradigms of Automated Music Generation*, Springer (2009).
- [4] Prusinkiewicz, P.: Score generation with L-systems, *Proc. ICMC*, pp.455–457 (1986).
- [5] Worth, P. and Stepney, S.: Growing music: Musical interpretations of L-systems, *Proc. EuroGP*, pp.545–550 (2005).
- [6] Stocker, R., Jelinck, H., Burnota, B. and Bossomaier, T. (Eds.): *Complex Systems: From Local Interactions to Global Phenomena*, McCormack, J.: Grammar-Based Music Composition, pp.321–336, IOS Press (1996).
- [7] Lourenco, B.F., Ralha, J.C.L. and Brandao, M.C.P.: L-Systems, Scores, and Evolutionary Techniques, *Proc. SMC*, pp.113–118 (2009).
- [8] DuBois, R.L.: *Applications of generative string-substitution systems in computer music*, Ph.D. thesis, Columbia University (2003).
- [9] Ashlock, D.A., Gent, S.P. and Bryden, K.M.: Evolution of L-systems for compact virtual landscape generation, *Proc. CEC*, Vol.3, pp.2760–2767, IEEE Press (2005).
- [10] Mueller, P., Wonka, P., Haegler, S., et al.: Procedural Modeling of Buildings, *ACM Trans. Graphics (TOG) – Proc. ACM SIGGRAPH 2006*, Vol.25, No.3, pp.614–623

- (online), DOI: 10.1145/1141911.1141931 (2006).
- [11] 平田圭二, 青柳龍也: バービーブーン: 音符レベルでユーザ意図を把握して編曲を行う事例ベースシステム, 情報処理学会研究報告, Vol.2000, No.94, pp.17–23 (2000).
 - [12] 平田圭二, 青柳龍也: 音楽理論 GTTM に基づく多声音楽の表現手法と基本演算, 情報処理学会論文誌, Vol.43, No.2, pp.277–286 (2002).
 - [13] Gillick, J., Tang, K. and Keller, R.: Learning Jazz Grammars, *Proc. SMC*, pp.125–130 (2009).
 - [14] Kohonen, T.: A Self-Learning Musical Grammar, or “Associative Memory of the Second Kind,” *Proc. IJCNN*, pp.1–5 (1989).
 - [15] Mathews, M.V. and Pierce, J.R. (Eds.): *Current Directions in Computer Music Research*, Schottstaedt, W.: Automatic Counterpoint, pp.199–214, MIT Press (1989).
 - [16] Tanaka, T., Nishimoto, T., Ono, N., et al.: Automatic Music Composition Based on Counterpoint and Imitation Using Stochastic Models, *Proc. SMC*, pp.330–337 (2010).
 - [17] Holland, J.H.: *Adaptation in Natural and Artificial Systems*, pp.171–198, MIT press (1994).
 - [18] 電気学会 GA・ニューロを用いた学習とその応用調査専門委員会 (編): 学習とそのアルゴリズム, 玉置 久: 遺伝アルゴリズムと機械学習の基礎, pp.86–101, 森北出版株式会社 (2002).
 - [19] Taura, T., Nagasaka, I. and Yamagishi, A.: Application of evolutionary programming to shape design, *CAD*, Vol.30, No.1, pp.29–35 (1998).
 - [20] Messiaen, O.: *Technique de mon langage musical*, Alphonse Leduc (1944). Satterfield, J. (Translation): *The Technique of My Musical Language*, Alphonse Leduc (1956).



田中 翼 (学生会員)

1981年生。京都大学理学部卒業。東京大学大学院情報理工学系研究科修士課程修了。2011年より東京藝術大学大学院美術研究科博士課程在籍。日本学術振興会特別研究員。JSSA, IEEE各会員。



田川 聖

ベルリン芸術大学作曲科でユン・イサン、ハンブルク音楽大学でジェルジ・リゲティに学び、ハンブルク音楽大学作曲・音楽理論学部助手・講師、スタンフォード大学客員研究員・作曲家、Music Media Lab Hamburg 協会会長等を経て、2000年から東京藝術大学美術学部先端芸術表現科准教授。