

音楽知プログラミング試論

An Essay on Musical Knowledge and Intelligence Programming

平田 圭二¹

Keiji HIRATA

あらまし

本論文は音楽情報やデータをコンピュータ上で思い通りに操作するための新しいパラダイム「音楽知プログラミング」を提唱する。人は、コミュニケーションをするためのメディア技術や約束事を発明し活用してきた。音楽もそのようなコミュニケーションをするためのメディアの一つと考えられる。そこで、まずコミュニケーションメディアとしての音楽とは何か、どうあるべきかを歴史的背景とともに議論する。次に音楽学とコンピュータ科学の関連に留意しながら、コミュニケーションメディアとしての音楽を実現するために必要となる技術を検討する。本論文で提唱する音楽知プログラミングという新しいパラダイムは、音楽という曖昧で主観的なメディアを思い通りに操作し、ユーザの意図や楽曲の意味の伝達を可能にする。そして、音楽知プログラミングの有効性を検証するために試作した音楽システムを紹介する。読者の皆様にコミュニケーションメディアとしての音楽研究の持つ広がりを感じて頂けたらと思う。

Abstract

“Musical Knowledge and Intelligence programming” enables one to manipulate musical information and data on a computer as one intends. We have been inventing and developing many media technologies and protocols for communicating with each other. Music can be considered as one of the communication media. Since music has the features of ambiguity and subjectivity, it has not been considered as a communication media that can transfer information as measured in units of bits. The investigation into the possibility of music as a communication media may open up a new communication media that can carry information that has not been carryable so far. In light of the relationships between musicology and computer science, we discuss the technologies required for making the idea of music as a communication media a reality. Musical Knowledge and Information programming is a new paradigm that enables the user to manipulate music, which is an ambiguous and subjective media, and to transfer his/her intention and the meaning of a piece. We introduce prototype musical systems that verify the effectiveness of musical knowledge and intelligence programming. We hope that the readers of this paper would be interested in the broad perspective of studying music as a communication media.

1 技術の歩みと音楽

世界中に音を操る生き物はたくさんいるが、音楽を操る生き物は人間だけと言われている。人という生き物はそもそも創造性を発揮し自己表現したいという欲求を強く持っている。だから人は、コミュニケーションするために新しいメディアを積極的に開拓してきたし、そのメディアを使いこなす技術や約束事を生み出してきた。音楽はそのようなコミュニケーションメディアの1つとして位置づけられる。CDやインターネットどころかレコードさえもない昔の人々が音楽を聴くには自分で演奏するか身近な人が演奏するのを聴くしか方法がなかった。その頃の人々にとって、音楽を演奏することは今よりずっと当たり前で普通の行為であったろう。身の周りの物を何でも楽器として、音楽を自由に創り演奏することは、当時の人々のありふれたコミュニケーションの一部を担っていたに違いない。音楽による自己表現や創造性の発揮などと改めて言葉にするまでもなく、昔の多くの人々は表現者であり音楽家であった。音楽はコミュニケーションのメディアである。この命題は時と場所を選ばず真であろう。大昔もそうであったし、100年前もそうであったし、現在もそうである。

音楽の楽しみ方、音楽のスタイル、楽器の種類、演奏法等は複雑に絡みあいながら、その時々々の社会や文化の状況を反映しつつ変遷する。18世紀から19世紀にかけてピアノという楽器の誕生と進化は音楽を変えた。1877年にエジソンの発明した蓄音機は音楽だけでなく、人々の音楽との関わり方も劇的に変えた。時代が下り、作曲家、演奏

¹ NTT コミュニケーション科学基礎研究所 Communication Science Laboratories, NTT

家、一般のリスナという分業化が一段と進み、我々は音楽を楽しむ今のような状況を当然のように考えている。しかしそれもここ 100 年ほどの間に定着した状況に過ぎない。音楽と技術の関係は卵と鶏のようなものである。技術レベルが変われば音楽と音楽のあり方も変わる。現在では、より忠実に録音したり、より大量、高速、高品質にコピーを配布する技術が発達している。そのような技術的背景のもと、ごく少数の才能のある作曲家と演奏家に対し大勢の一般リスナという構図が成立し、より音楽性や技巧性の高い楽曲や演奏が創作されるようになっていく。現在の音楽のあり方を我々は所与で不変なものとして考えがちだが、実はそれは、今の技術レベルと音楽的な要請という制約の中での一過性的な平衡点に過ぎないのである。

音楽の世界にコンピュータが導入されたのはいつ頃だったのだろうか。Hiller と Isaacson がイリノイ大学のコンピュータで「イリアック組曲」を作曲したのは 1956 年であった。Max Mathews が ATT ベル研究所の IBM 704 を用いて「In the Silver scale」というほんの 17 秒の曲を合成したのは 1957 年であった。ちなみに、黎明期の汎用電子計算機として有名な ENIAC の誕生は 1946 年のことである。こうして、新しい道具を手にした音楽家たちはコンピュータに熱中した。そしてコンピュータ技術の進歩とともに、新しい音楽もどんどん生み出されていった。

ご承知のように、PC とインターネットの普及を基軸とする IT の発展が現在進行中である。IT は我々人類の文化を大きく変革すると言われているが、人類の短い歴史の中において変革と呼ばれるものはこれまで幾度となく訪れている。しかし、IT による変革が紙の発明や印刷技術の発明と並びことさら喧伝される理由は、IT が究極の個人対応を可能にするからである。それは、生活のあらゆる場面において、1 対多ではなく 1 対 1 のその人のためだけのサービスや支援を可能にし、個人に自分の生活環境をデザインする能力を与えるからである。これは、あらゆる応用分野において新しいビジネスを創出するヒントを与えてくれる。

音楽の分野も例外ではない。そして、音楽そのものだけでなく、我々の音楽に対する考え方や接し方まで変革してしまうと考えられている。これまで音楽は音楽家の手に渡り技巧の面において長足の進歩を遂げたが、IT は、その音楽を音楽家の手から再び一般大衆の手に取り戻すための重要な手段になり得るかも知れない。IT を有効に活用することで、高度な作曲、演奏、編曲等の音楽活動を個人単位で行うことができるようになるからである。あたかも、大昔の人々が思い思いに音楽に触れ楽しんだように。しかも、IT がもたらす高度な検索、集約、推論、配信等の機能を個人レベルで自由に活用することは、新たな平衡点をもたらし、そこでは、従来の枠組では捉えられないような新しいタイプの音楽や音楽の楽しみ方が生まれてくるだろう。すでに、携帯電話の着信メロディやインターネットカラオケなどにその萌芽が見られる。

2 音楽は非コンピュータ的か？

人はいずれ、コミュニケーションメディアとしての音楽を操る技術を必要とするに違いない。なぜなら、人はコミュニケーションに対する強い欲求を本来持っており、音楽によるコミュニケーションは人にとって大変大きな割合を占めているからである。コンピュータ技術の歴史は、人のタスクや機能の代替の歴史であったと言って良いだろう。数式を計算する、物を憶える、乗り物を運転する、ゲームをする等々。しかし、コンピュータはまだ音楽を操る技術を獲得していないし、コンピュータに人間並みの音楽的才能を持たせることにもまだ成功していない。これらの問題はどうか非常に難しいものらしい。

では、何が本質的に困難なのであろうか。音楽という応用分野において、高度で音楽的な検索、集約、推論、配信等の機能を実現するにはどのような問題から取り組めば良いのだろうか。

まず、コンピュータへの適用が比較的早く始った科学や工学の分野を考えると、それらの分野には共通して、予測、再現、制御、構造、精度、効率という特徴があることが分かる。これらの特徴を支えているのが、科学的な方法論による物理的な対象の仕組みや構造の解明、及び仕組みや構造の数式による表現である。コンピュータ上で知識を表現し操作する方法には簡潔さ、正確さ、直観に対する忠実さが要求されるが、科学や工学にはこのような下地があったからこそ早くからコンピュータへ適用することができたのである。ここで、簡潔さとは、ある情報を記述する際に、必要最低限のことだけ記述する、重複なく記述できる、可読性に優れる等を意味する。正確さとは、コンピュータを自動的な機械として動作させるにはプログラムを書かなければならないが、そのプログラムの記述には曖昧さが許されないということである。直観に対する忠実さとは次のようなことである。人があるタスクを実行する時、心の中にはそのタスクに関連する様々な概念が現れ、人はそれを直観として認識する。その様々な概念に相当するものをできるだけ正確にコンピュータにも持たせることを意味する。

一方、音楽という言葉から連想されることを列挙してみると、それは芸術性、直観、曖昧、主観、個性、感性、経験、暗黙的等になるだろうか。音楽が持つこれらの特徴は科学や工学の特徴とは正反対である。それゆえ簡潔さ、正確さ、直観に対する忠実さを達成することが難しく、音楽を科学的、工学的な対象とすることが難しいのである。音楽をコンピュータで操るといふ目標自体が矛盾しているようにさえ聞こえる。

コンピュータで情報を操作することを、簡単な例で考えてみよう。最初はある数式の計算についてである。机の

上にリンゴが 13 列 × 15 行にきちんと並べてあるとする。リンゴは全部で何個あるだろうか。殆んど人は $13 \times 15 = 195$ という計算によって合計 195 個という結果を得るに違いない。実際に並べられたリンゴを 1 から数え上げる人は殆んど居ないだろう。この時、リンゴの合計個数を得るには 13×15 という掛け算をすれば良いとすぐピンと来たのは、13 や 15 という数が何を意味し、 \times (掛ける) という演算が何を意味しているかが我々には明確に分かっているの、 13×15 という掛け算をすれば良いと分かったのである。

次の例は、自然言語の理解についてである。「I love music」という文の意味はどのようにして分かるのだろうか。この構文は SVO 構文であり、I, love, music の各々の単語の意味も良く知られている。だから文全体の意味は「私は音楽が好きだ」となる。構文の機能と単語の意味が明確に定義されているから文全体の意味が分かる。

これらの例から分かることは、コンピュータの上で正しく計算を進めるためには、コンピュータの上で操作する記号の意味とその組み合わせ方の機能を正確に曖昧なく定義する必要があるということである。この時に重要になる考え方が「抽象化」と「構造」である。人があるモノ (対象) を認識したり理解したり表現したりする時、意識的にしろ無意識的にしろ、そのモノはこういう枠組やテンプレートに従って組み立てられている筈だという仮定を使うことが多い。このモノを構成している枠組のことを本論文では **構造** と呼ぶ。構造を明らかにするとは、要素部分と全体はどういう関係にあるのか、部分を組み立てて全体を作るにはどうすればよいのか、逆に全体を部分に分解、還元するにはどうすればよいのかということとを説明することである。人は対象がある構造を持っていると認識する時、対象をその枠組に当てはめるために、細部を省略し本質的な部分だけ残すという操作を行う。これが抽象化である。抽象化したものどうしを集めさらに抽象化することで階層構造が発生する。人の認識の本質は、抽象化による構造の発見であると言って良いだろう。

同じレベルのものを集めること、まとまりを作ること、そこから共通の性質を抜き出すことは、人のあらゆる知覚、認知、理解における基本処理である。音楽を聴く時も例外ではない。つまり、コンピュータ上で音楽を操るためには、音楽に内在する構造を簡潔、正確、直観的に記述すれば良いのである。

3 抽象化 + 構造 = 音楽学

音楽において現れる抽象化と構造を概観してみよう。例えば、五線譜の中に見いだせる構造として、1 つの音、和音 (コード)、和音名、和音進行、旋律 (メロディ)、動機、楽節、形式、拍節、リズムパターン、などがある。1 つの音には、音の高さ、音の長さ、音の大きさ、音色などという要素から成る構造がある。

このように、楽曲に現れる様々な構造を抽象度の低いものから高いものへと階層的に並べたのが図 1 である。抽

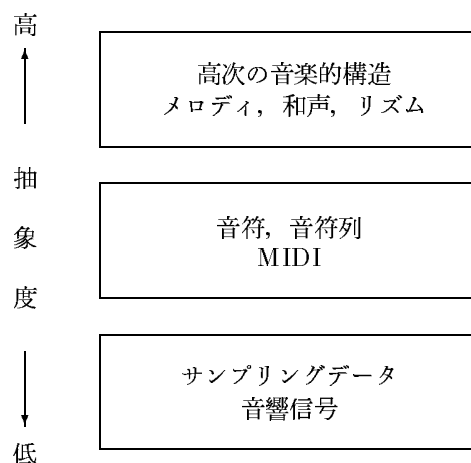


図 1: 音楽情報の階層構造

象度の最も低いレベルは、アナログ音響信号あるいはサンプリングデータのレベルである。このレベルで見られる構造は「サンプル値の列」(あるいは「電圧値の列」)という構造だけである。次に、1 つの音という構造が認識されるようなレベルがある。1 つの音は幾つかの部分的要素 (音の高さ、発音時刻、音色など) から成るのと同時に、さらに上位レベルの要素としても機能している。図 1 の最も抽象度の高いレベルでは、複数の音から階層的にメロディ、和音、楽曲と言った構造が構成される。この高次構造のレベルでは、個々の音に関して、音色の情報は例えば楽器名

としてのみ記され、ミリ秒単位の発音時刻は拍節構造上の点として単純化されている。例えば、モチーフの重心を考える時は音色に注意を払う必要は殆んどない。

音楽中にこのような構造を見出すことが音楽を認識し理解することであり、逆に音を使って構造を作り上げることが音楽を生み出すことである。

音楽学 (musicology) とは、内省や実験により、音楽という対象に潜む上述のような構造、普遍的な原理、概念を明らかにしようという学問である。音楽学は音楽に対する非常に豊かな内観を我々に与えてくれる。音楽理論とは楽曲が持つ意味やそこに暗黙的に込められた意図を分析、解釈するための理論である。

この音楽学が我々の研究に理論的な基盤を与えてくれることを期待したい所である。たしかに、音楽学が目指す楽曲分析の理論は我々にヒントを与えてはくれる。しかし、コミュニケーションメディアとしての音楽研究には、人が実際に音楽を聴取し創作し演奏する過程に関する理論が必要とされる。ところが、音楽学の興味はそのようなところにはない。さらに、音楽学から得られた知見や音楽理論は、自然言語によって定性的かつ分析的に提示されている。人間相手に研究成果を伝えたり議論する場合にはこれで十分であるが、コンピュータ上で音楽学の研究成果を利用しようとする、簡潔さ、正確さが不十分なため壁にぶち当たる。そのままアルゴリズムやプログラムとして書き下すことができないのである。

それでも我々は、音楽をコミュニケーションメディアとして操り、高度で音楽的な検索、集約、推論、配信等の機能を実現するために、音楽学の成果を最大限に活用するのが最も近道なのではないかと考える。飛行機を作る時には流体力学の知識を、病気を治す時には生理学の知識を利用するようなものである。従って、我々の研究の第一歩は、音楽理論を基礎としてその形式化を進めることであると考えられる。

4 音楽学から音楽知プログラミングへ

我々は音楽理論を積極的に援用するアプローチを採る。ところが、音楽理論は、コンピュータ上のアルゴリズムやプログラムとして記述することを考慮していない。そこで我々は、コンピュータによる処理を意識した音楽知識の体系をゼロから構築する必要があると考え、ここに新しい概念である「音楽知」を提唱する。

4.1 音楽知

コンピュータ上での形式化を前提とし、音楽理論に基づく構造によって意味が与えられている音楽全般に関する知識や知性の集まりを「音楽知」と呼ぶ。実際、音楽知には様々な形態や抽象度のものがある。五線譜に書かれている楽曲、音楽理論に基づく楽曲の分析結果、教科書で述べられている知識、音楽学から得られた知見や解明された事実、音楽家の頭の中のみある暗黙的な知識や知性等である。

音楽知をコンピュータ上で操作する際に考慮すべきは次の3点である：(1) 音楽知そのものに含まれる規則や構造、(2) 音楽知に含まれるその構造の表現、(3) 表現された音楽知に対する演算やアルゴリズム。

(1) に関して、我々は音楽理論に依拠する。楽曲を五線譜で表す場合、音符や休符等が五線の適切な場所に置かれる。これが人間の目に見える表層のレベルである。例えば和音進行も表層レベルの情報である。しかし五線譜の裏には何らかの音楽的な深層構造が隠されている(図2)。人は無意識的、意識的にかかわらずその深層構造の情報も加味

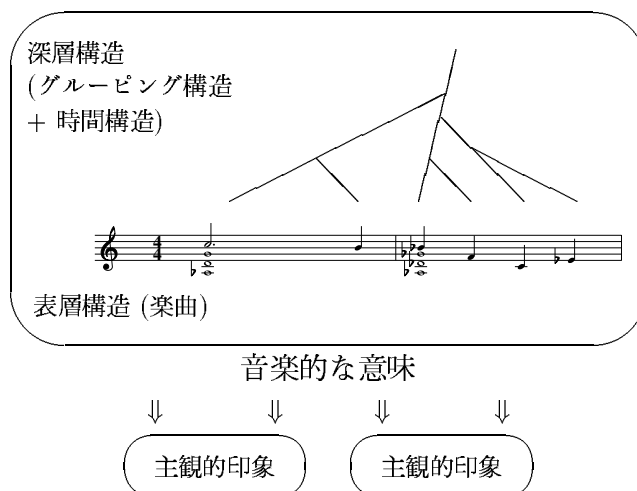


図2: 表層構造, 深層構造, 意味, 主観的印象

して音楽的活動を行っている。先の“I love music”の例文の意味を理解する時と同様である。これら深層構造と表層

構造を合わせて意味が構成される。従って、音楽知を取り扱うということは、表層構造だけでなく深層構造も含めてコンピュータ上で取り扱うことである。

音楽における深層構造とは何であろうか。深層構造を顕在化する基本的な考え方は、楽曲に含まれる個々の音に関してある種のまとまりを見い出すことである。まとまりを見い出すことは一般に、グルーピング、体制化、構造化、関係付け等と呼ばれる。つまり、楽曲分析を行い楽曲に含まれる音をグルーピングすることで深層構造を顕在化させるのである。それが音楽理論の役割である。そのような音楽理論に基づいて音楽知を取り扱う所まではコンピュータで形式的に取り扱える範囲であると考えられる。

主観的印象や情動の取り扱いはまた別の問題である(図2)。従来の音楽研究や音楽システム作成では、この主観的印象や情動を音楽知と混同して処理している事例が散見される。例えば「太郎は駅へ向った」という文を考えてみよう。この文を解釈することにより、太郎という人がどこかの駅へ向って移動するという文の意味が理解される。その認識に基づいて、この文を読んだ人には楽しいという情動やたまた悲しいという情動が恣意的に喚起あるいは連想されるかも知れない。この情動は現段階で体系的に理解できる対象のように思えない。従って、自然言語処理では、このように理解された文の意味と恣意的に喚起・連想された情動は区別して取り扱われる。音楽知を取り扱う場合もこの自然言語処理と同様にまず音楽知とそこから恣意的に喚起・連想される主観的印象や情動を区別し、それぞれの処理手法も区別して考えるべきである。

音楽はそもそも多角的な意味を持っていると考えられるので、複数の音楽理論を用いて多角的に楽曲分析する方がより直観に合致している。そうして、より精密な深層構造を顕在化することで、音楽家やユーザの意図を正確に認識できるようになる。

(2)に関して、我々は人工知能における知識表現技術を応用する。図2の表層構造や深層構造は、コンピュータ上の記号や項として表現される。この時、知識表現手法に求められるのは、表現すべきことを簡潔に述べられる記述力と正確さである。一階述語論理はこの基準を満たす多くの知識表現言語の基礎を成している。我々の研究においても、一階述語論理²(あるいはその発展形)を採用する。

(3)に関して、音楽知を表現する記号や項に対して適切で意味のある演算や推論体系を設計しなければならない。例えば、整数を対象とすれば、加減乗除の四則演算で実用上十分な関数が実現できる。あるいは、真偽の値を持つ命題を対象とすれば、論理和、論理積、否定の3種の演算ですべての論理関数が実現できる。同様に、音楽知を表現した記号や項を対象とした時、どのような種類の演算や推論を用意すれば、すべてのあるいは実用上十分な音楽的な関数が実現できるのだろうか。これは今後解決すべき重要な課題だと思われる。

また、(1)の所で、楽曲分析によって深層構造を顕在化する処理と、主観的印象と情動をその深層構造に関連付ける際の恣意性を混同すべきではないと述べた。主観的印象や情動はむしろ、音楽心理学の分野で積極的に扱われている。我々は、音楽システムの作成に際し、後者の恣意性に対しては事例やコーパスに基づく推論が有効であろうと考えている。その事例やコーパスはデータでもあり規則でもあるので、データと規則が同一の形式で表現されるような表現手法を採用することで、高い可読性と効率的な推論が可能となるだろう。その代表的なものの一つにやはり一階述語論理がある。

4.2 音楽知プログラミング

ここまでの検討より、音楽知をコンピュータ上の記号や項として表現する用途は付いた。次に、現実の音楽知を十分直観的に表現する記号や演算を過不足無く用意し、ユーザがそれらを適切に組み合わせる(プログラミングする)方法論を検討しよう。音楽知プログラミングとは、音楽知を対象とする推論を思い通りに制御し、所望の音楽タスクを実現することである。このような音楽知プログラミング環境が実現できれば、ユーザは所望の音楽システム、アプリケーション、機能を自由自在にデザインできるようになる。ここに初めてコミュニケーションメディアとしての音楽の成立を主張することができよう。

音楽的な状況において人が実行しているあらゆるタスクが音楽知プログラミングの対象となる。例えば、ハーモナイズ、ボーシング、リハーモナイズ、アドリブ、本番演奏のための練習、試行錯誤と自己評価、異なる形式/様式/形態で獲得した音楽知の統合、記憶、連想、学習、適用、データマイニング、音楽的に合理的なUIや可視化、実時間インタラクションなどである。

音楽知プログラミングは、与えられた基本要素を組合わせて所望のソフトウェアを構築するという点では従来のプログラミングと同様である。しかし、計算の対象となる音楽知は、1つの表層構造と何種類かの深層構造が組になっている。表層構造が同じでも異なる深層構造を持っている場合がある。従って、組になっている深層構造とユー

²取り扱う対象の上に定義される関数、対象の持つ性質や対象の間に成り立つ関係等を一階述語論理式で表現した論理。ここで、述語というのは、そこに含まれる変数の値が具体的に与えられると真偽が決まるような言明のこと。一階というのは、述語に含まれる変数の値域が項全体に限定され、述語を含まない場合を意味する。

ザが付与しようとしている深層構造が同じか異なるか、異なる場合はどのように異なるかをプログラミング中に検査、ユーザ提示する手段が必要となろう。

音楽知プログラミングの特徴をまとめる: (a) 音楽知を表現する記号や項には、複数の音楽理論に基づく多角的で一貫した意味が多重に付与されている, (b) 音楽的な意味を反映した基本演算を備えており, その意味は直感的に理解できる. (c) 楽曲等の表層には (a) で述べたような深層構造 (= 意味) が付与されるが, その (部分的な) 深層構造を最初に (自動的に) 表層構造に付与し, 後にその深層構造を追加, 修正, 更新する処理が必要である.

ここで注意すべきは, プログラミングという作業が成立するための条件である. その最も重要な条件は, どのようなプログラムを書くときどのような結果や振る舞いが現れるのかをユーザが精度良く予測できることである. 音楽知プログラミングはこの条件をクリアすることができるであろうか. もちろん, 我々はこの条件をクリアすべく音楽知プログラミングの枠組を構築しなければならない. 上の (c) の特徴が従来の情報処理には無かった新しい課題を提起するのではないかと思われる.

5 試作した音楽システム

我々は音楽知プログラミングの実現に向けて, これまで 4 台の音楽システムを試作した (表 1). いずれも共通の音楽理論, 音楽知表現手法, 推論手法を採用している. 音楽理論としては Generative Theory of Tonal Music

表 1: これまで試作した音楽システム

システム名	機能	開発年
ハービー君 [2]	和音から演奏生成	1996
パーピーブン [3]	和音進行から演奏生成	1999
バービーブン [4]	楽曲の編曲	2000
ハーヒーフン [5]	演奏の表情付け	2001

(GTTM) [1] を, 音楽知表現手法としては演繹オブジェクト指向データベース (Deductive Object-Oriented Database, DOOD³) [8] を, 推論手法としては事例に基づく推論 (Case-Based Reasoning, CBR) [7] を採用している. ここで, GTTM は F. Lerdahl 及び R. Jackendoff が 1983 年に提案した楽曲分析の理論であり, コンピュータ上への実装が最も有望視されているものの一つである. DOOD は簡潔で正確な知識表現を実現するために一階述語論理を拡張した知識表現手法である. CBR はある問題の解を生成する際, その問題に類似した事例を利用して解を構成する推論手法である.

我々は, これら音楽システムの試作を通じて一貫して音楽知の表現手法を追求した. そして, ハービー君, パーピーブンではコード進行の持つ構造を, バービーブン, ハーヒーフンでは多声の旋律の持つ構造を表現する手法を提案した. これら 4 台の音楽システムに関する資料やサウンド出力が web ページ <http://www.br1.ntt.co.jp/people/hirata> にまとめてあるので, ご興味のある方は参照されたい.

上の 4 つの音楽システムでは音楽知に対する 3 種類の基本演算を提供している. それは, 包摂関係, 2 つの項が与えられた時にその共通部分を取り出す演算 (lub), その重ね合わせを計算する演算 (glb) である. 包摂関係は, ある楽曲とそれより抽象的な楽曲を関連付ける, あるいは逆に, ある楽曲とそれより具体的な楽曲を関連付ける演算子である. 数学的には, lub は最小上界 (least upper bound) に, glb は最大下界 (greatest lower bound) に対応する. 現在は, これら包摂関係, lub, glb だけを用いてアルゴリズムを構成しているが, 実用上十分とは言えない. 実世界の複雑な音楽タスクの実現に十分有効なプリミティブの組を探し出す必要がある.

次に, パーピーブンとバービーブンの 2 つを取り上げてさらに詳しい説明を加える.

5.1 パーピーブン

パーピーブン [3] は, 入力として与えられた単純なコード進行をジャズ風の和音に自動的にアレンジする音楽システムである. 一般に, 段落感, 終止感を与えるような特定のコード進行パターンはケーデンス (ジャズやポップスの場合, 1 つのケーデンスは通常数個の和音列から成る) と呼ばれるが, パーピーブンはケーデンス単位で和声の流れ (和声的文脈) を考慮しながらアレンジを行う.

³DOOD という用語はもともと国際会議や研究分野の名称として用いられていたが, 本論文では知識表現手法の名称として用いる.

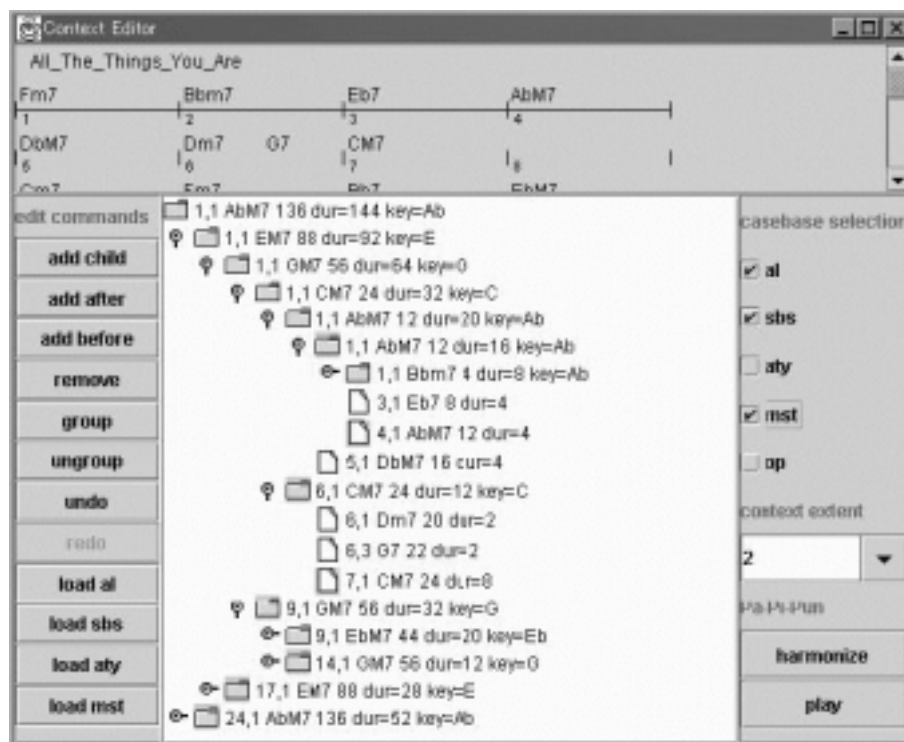


図 3: 和声文脈エディタのウィンドウ

パービーブンは、アレンジしたい楽曲に関する演奏家や作曲家の意図を表現するために、GTTM に基づく特別なデータ構造(ケーデンス木)と、lub, glb から構成される音楽的な類似度を計算するアルゴリズムを提案し実装した。パービーブンは専用の GUI (図 3, 和声文脈エディタ) を持つ。ユーザはこの和声文脈エディタを用いてケーデンス内の構造とケーデンス間の構造を指示し、ユーザの意図通りのケーデンス木を作成する。

図 3 中央の白い部分には、システムに対する問い合わせを表現するケーデンス木がその部分木の深さに対応した段下げとともに表示されている。ユーザの指示で部分木の表示を畳んだり展開したりできる。例えば、最も段下げされた部分の和音進行は和音を 3 つ含んでおり、その順序は $Bb_{m7} \rightarrow Eb_7 \rightarrow Ab_{M7}$ である。さらにその 1 行上に「1,1 Ab_{M7} 12 ...」という行があるが、これは、 $Bb_{m7} \rightarrow Eb_7 \rightarrow Ab_{M7}$ という和音進行がグルーピングされて Ab_{M7} という和音で代表されるということを意味している。ウィンドウ左側には、ケーデンス木の操作コマンド (add child, undo など) と組込みの問い合わせをロードするボタン (load all, load sbs など) がある。例えば group というコマンドは選択した複数の和音を 1 つのグループにまとめる役割を持っている。ユーザはこれら操作コマンドを用いて問い合わせのケーデンス木を自由に編集する。これが、表層構造としての和音進行に音楽知の深層構造を付与する作業に対応している。ウィンドウ右側には、事例ベースの選択ボタン (al, sbs など) や、harmonize ボタン、play ボタンなどがある。現在のパービーブンは、4 人のピアニストの長調 3 曲、短調 1 曲の演奏から採譜した 5 種類の事例ベースが提供されている。コード進行の深層構造をケーデンス木という形で顕在化した音楽システムはパービーブンが初めてである。

これより、パービーブンは、従来の音楽システムでは困難だった次のようなことを可能にした。まず、ある特定のユーザの個性的な音楽表現、例えば曲の流れやコードの押さえ方の癖等を明示的に表現できる。和声文脈エディタを使うとユーザの意図をシステムに伝達できるので、ユーザが好きなように出力の演奏を修正したり編集したりすることができる。推論に用いる事例ライブラリを選択することで、ユーザの希望する主観的印象や情動にうまく合致するような演奏を生成できる。

5.2 パービーブン

パービーブン [4] は、入力の単純なメロディから即興演奏を行うジャズピアニストのように編曲を行う音楽システムである。パービーブンは事例に基づいて出力を生成する。まず、ある単純なメロディが与えられるとこういふ風に編曲するというボキャブラリ (事例集) をあらかじめシステム内に蓄えておく。次に、実際に未知の単純なメロディ

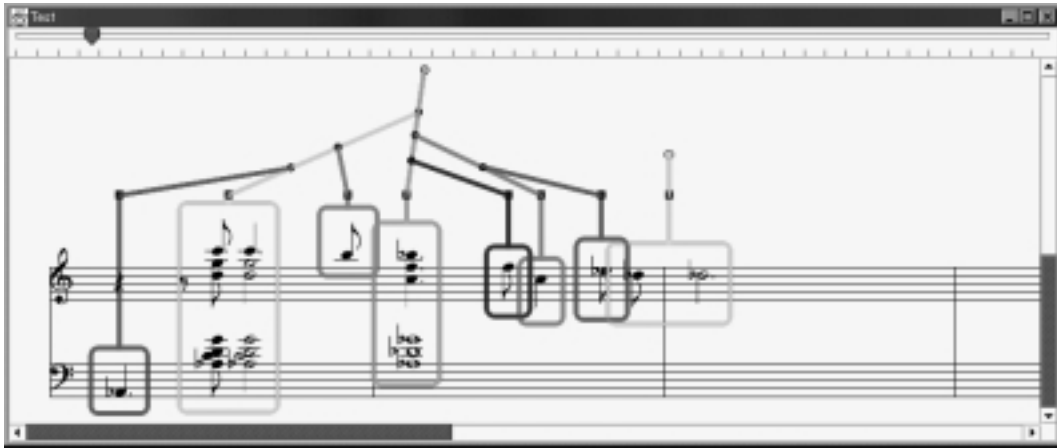


図 4: 楽曲分析し解釈を与えるためのエディタ

が与えられた時、そのボキャブラリの中から与えられたメロディに最も類似した事例を探し出し、それらを組合わせて編曲を行うのである。

従来の編曲システムの多くは、メロディを直接変形するような演算を用意していなかった。その代わりに、メロディを一旦特徴空間に写像し、そこで適当な処理を施し再び実際のメロディに戻す。ここで、特徴空間を構成するための特徴として例えば音域、音符数、メロディの平均音高や傾斜等が用いられることが多い。この特徴空間を設ける従来方式の欠点は、一旦特徴ベクトルに写像してしまうと元のメロディに含まれていた情報の一部が失われてしまうことである。従来方式では不完全な情報に基づいて処理を行わざるを得ないので、ユーザの意図や楽曲の意味を十分に反映した処理が実現できなかった。

一方、パービーブンは、五線譜に記述されたメロディを表層構造として、それが持つ深層構造を楽曲分析により顕在化し、システム内では常に表層構造と深層構造を組にして処理を行う。これにより、メロディに含まれる情報を一切捨象することなく直接メロディに演算を施すことが可能となった。

パービーブンは編曲という音楽タスクを、関数を外挿する問題として形式化している。まず、事例に基づいて編曲するためには、未知の単純なメロディ X 及び事例 (あるメロディ E とその編曲結果 A の組) というデータが必要である。編曲のための関数を f とすると、 X を編曲するとは、 $f(E) = A$ を満たすような f に対して、 $f(X)$ の値を推定することである。この時、ユーザの意図を $f(E) = A$ という条件として明示的に表現していることに注意されたい。一般にそのような f は多数存在するが、我々は音楽的に合理的であるために幾つかの条件を課し、前述の3種類の基本演算から構成できるものに限定した。その結果、事例の持つ編曲の雰囲気やうまく活かして、与えられたメロディを適切に編曲することができた。このような関数の設計が可能になったのは、音楽知の枠組に従いメロディをコンピュータ上に適切に表現できたからである。メロディの深層構造を顕在化し、メロディを直接演算の対象とした編曲システムはパービーブンの初めてである。

パービーブンは、メロディを分析し深層構造を付与するための専用エディタをユーザに提供している (図 4)。これは、楽曲分析のための一種のワープロみたいなものと言ってもよいだろう。本エディタを用いて隣り合う音どうしを次々とグルーピングし、1つの楽曲を最後には1つのグループにまとめる。人それぞれ、音楽の感じ方が異なるように、この音のまとめ方がそのユーザの楽曲の解釈を表しており、それが深層構造に対応する。本エディタを提供することで、楽曲分析の効率は飛躍的に向上することが期待される。

6 まとめ

本論文では、人と音楽と技術の歩みやコミュニケーションメディアとしての音楽研究の在り方を議論し、音楽知プログラミングを提唱し、我々の研究事例紹介を行った。コンピュータと音楽の研究は一般にはあまり馴染みがないせいも、その問題意識や取り組み方に関して、世の中から少なからず誤解を受けることが多い。本論文がその誤解を減らす一助になれば幸いである。

最後に、音楽知プログラミング及び関連する研究テーマについて触れたいと思う。

- 音楽知を表現する記号や項には、異なる音楽理論に基づく多角的で一貫した意味が多重に付与されているが、異なる音楽理論どうしがどのように共存、協調、競合するかは殆んど未解明である。

- コミュニケーションメディアとしての音楽を成り立たせる技術的要件は整いつつある。次は、実際にインターネット上のアプリケーション作成に音楽知プログラミングを応用する段階である。従来の音楽システムは、スタンドアロンのコンピュータ上で稼働し、作曲、編曲、即興、伴奏等の従来タスクを代行するものであった。音楽知プログラミングは、従来には無かった新しい発想の音楽タスクやアプリケーションを支える技術となることが期待される。
- 音楽というメディアの大きな特徴は曖昧で主観的という点である。ところが、音楽以外にも曖昧で主観的なメディアがある。例えば、絵、ジェスチャ、顔の表情などである。これら曖昧で主観的なメディアの処理技術はまだ未開拓であるが、音楽知プログラミングで培われた技術を一般化することで、このようなメディアの処理技術への貢献が期待される。
- 5節で紹介した音楽システムはいずれも CBR を採用していたが、より創造性の高い音楽システムを構築するには CBR では限界があるだろう。実際、本物のジャズピアニストは、その場の閃きで思いもかけない素晴らしい即興演奏をすることがある。このような、とても人間的で創造的な振る舞いを実現するにはどのような音楽知プログラミングの枠組を構築すれば良いのだろうか。或は、人間の創造性をかき立てるような音楽システムはどのように構築すれば良いのだろうか。
- コミュニケーションメディアとしての音楽研究を突き詰めて行くと、音楽的知性とは何かという根本的な問題に行き当たる。即興演奏を実現するにはフレーム問題を解かなければならないだろう。音楽を聴いて感動するコンピュータや感動する音楽を創作するコンピュータを実現するには記号接地問題と自意識の問題を解かなければならないだろう [6]。我々は、音楽という新しい応用領域で研究を行うことで、知性そのものに対して新しい解決の視点や実験場を提供できるのではないかと期待している。

音楽は人の知識活動の所産である。音楽学は、コンピュータという前提を置かずに、長年音楽における人の知的活動を記述し分析してきた。しかし、今後の技術や文化の発展を考えると、音楽知をコンピュータ上で表現し操作する技術、即ち音楽知プログラミングの確立が急務であると考えられる。我々の研究が、今後の音楽学とコンピュータ科学の橋渡しとなり、コミュニケーションメディアとしての音楽という新しい研究分野を切り拓く魁となれば幸いである。

謝辞: 本論文で紹介した音楽知プログラミング及び一連の音楽システム作成に関する研究は、津田塾大学 青柳龍也 助教授及び文教大学 平賀瑠美講師との共同研究として進めて参りました。日頃の貴重な議論やプログラム開発に対し感謝致します。

参考文献

- [1] Fred Lerdahl and Ray Jackendoff, *Generative Theory of Tonal Music*, The MIT Press (1983).
- [2] 後藤真孝, 平田 圭二, ハービー君: 演繹オブジェクト指向に基づいてジャズらしいコードにリハーモナイズするシステム, 情報処理学会 音楽情報科学研究会 研究報告 96-MUS-16, pp. 33-38 (1996).
- [3] 平田圭二, 青柳龍也, パーピーブ: ジャズ和音を生成する創作支援ツール, 情報処理学会論文誌, Vol.42, No.3, pp. 633-641 (2001).
- [4] 平田圭二, 青柳龍也, バーピーブ: 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 音楽情報科学研究会 研究報告 2000-MUS-37, pp. 17-23 (2000).
- [5] 平田圭二, 平賀瑠美, ハービーフン: 2 段階演奏表情付け法によるインクリメンタルな演奏生成システム, 情報処理学会 音楽情報科学研究会 研究報告 2001-MUS-39, pp. 19-26 (2001).
- [6] 星野力, ロボットにつけるクスリ, アスキー出版局 (2000).
- [7] Janet Kolodner, *Case-Based Reasoning*, Morgan Kaufmann (1993).
- [8] 横田一正, 演繹オブジェクト指向データベースについて, コンピュータソフトウェア, 岩波書店, Vol.9, No.4, pp. 3-18 (1992).