

Beyond GTTMism - 音楽の意味論と計算体系

平田 圭二^{1,a)} 東条 敏^{2,b)} 浜中 雅俊^{3,c)} 松原 正樹^{4,d)}

概要: 本発表では、現在我々が進めている計算論的音楽理論に関するプロジェクトについて述べる。我々が提案する計算機が存在を前提とした音楽の意味論とその意味論に基づいて構築された音楽に対する代数的な計算体系について、ゴール、設計思想を議論し、これまでの研究成果を紹介し、今後の課題や展望を述べる。

キーワード: 音楽理論, 認知的リアリティ, GTTM, タイムスパン木, *join/meet*, 類似度, モーフィング

Beyond GTTMism - Semantics and Calculus of Music

HIRATA KEIJI^{1,a)} TOJO SATOSHI^{2,b)} HAMANAKA MASATOSHI^{3,c)} MATSUBARA MASAKI^{4,d)}

1. はじめに

音楽の認知科学は、楽曲の持つ意味を純粋に音楽自体に帰着するアプローチを採用することで、さまざまなことを明らかにしてきた。例えば、聴取された旋律、和声、拍節におけるゲシュタルトの形成、旋律や旋律輪郭の記憶と予測、調性と期待、音楽能力の発達過程などである [1][19]。しかし、これらの要素を明らかにする実験条件は比較的単純であり、複雑で大規模な現実の楽曲全体を聴取する時の認知過程の解明はまだ不十分である。

音楽理論は、基本的に作曲のために、楽譜に記述された楽曲全体を分析する手法として発達してきた [4][22]。つまり、他人の作品を分析しその結果を次の作曲に活かす。しかし分析の手順や判断に関して主観的あるいは属人的な要素が強かったり、理論の記述が曖昧であるため、第三者(学習者)がその分析手法を理解し再現するには十分な訓練を必要とする [2]。また、プログラムとして記述するには概念定義が曖昧だったり情報が不足しているため、計算機に実

装するのは一般に難しい [6]。

アルゴリズム作曲は、作曲手法をアルゴリズムあるいはソフトウェアの形で厳密に記述して第三者に提供する手法である [17]。しかし、その楽曲生成手法は作曲家の審美的判断に基づいて音楽の表層的な構造あるいは形式を操作するものが多く、音楽認知的な根拠に根差しているものは少ない。また、人が発揮する芸術性や創造性と機械が人を支援する部分との分担がアドホックである。

楽曲の創作や生成に対し、楽曲認識や識別の問題も大きな研究分野を形成している。現実の楽曲を認識、識別する場合は、問題自体に音楽家や聴取者の審美的価値観や社会的価値観が含まれているが、音楽の認知科学が採用した楽曲の持つ意味を純粋に音楽自体に帰着するアプローチは、それらを取り扱うのが難しい。結果的に、楽曲の音響信号的特徴や楽譜という表層的な記述を対象にして、大量の事例から規則性を抽出する統計的機械学習の手法が現時点では最も高い性能を示している。

本稿では、上のような背景から、我々はどうのような問題意識を持ち、どのようなアプローチを採用し、どのような音楽の理論を構築してきたか、そして最後に今後どのような方向に研究を進めていくかについて述べる。

2. 問題意識

我々の問題意識を以下に挙げる:

¹ 公立はこだて未来大学 システム情報科学部
² 北陸先端科学技術大学院大学 情報科学研究科
³ 京都大学大学院 医学研究科
⁴ 筑波大学 図書館情報メディア系
a) hirata@fun.ac.jp
b) tojo@jasit.ac.jp
c) masatosh@kuhp.kyoto-u.ac.jp
d) masaki@slis.tsukuba.ac.jp

- (1) 計算機の存在を前提とした音楽の理論とはどのようなものか。音楽の認知科学に根差した音楽の意味を、データ構造やアルゴリズムとして過不足なく精密に記述する理論あるいは計算体系とはどのようなものか。
- (2) 音楽の理論の妥当性はどのように評価すればよいのか。もし音楽の理論の妥当性を判定する指標があるとなれば、それはどのようなものか。
- (3) 楽曲生成・創作のための音楽の理論とはどのようなものか。音楽の理論を実装した計算機はどの範囲までどのように楽曲生成・創作する人を支援すべきか。

まず(1)に関して、音楽における推論、判断、予測、期待などを形式化するために、まず音楽の意味を情報学的に定義しなければならない。そのため我々は、楽譜という表層的な記述からでなく認知的根拠を持つ音楽理論を出発点とすべきと考える。そして、音楽理論に含まれる概念や関係を適切に形式化するが、その際、本質的な事柄まで捨象しないよう注意する。^{*1} 一般に、形式化によってその理論体系は計算できる(プログラム化できる)ようになり、実用性と検証性(反駁可能性)が高まる。すると音楽分野では、例えば、楽曲識別、作曲、演奏、検索、推薦などに応用される。

次に(2)について、我々は、理論が導く類似度と人が認識する類似度がどれくらい類似しているかをその理論の妥当性の指標とすることを提案する[21]。認知科学辞典[23]によれば、類似度を考えるためには、関連するモノやコトの特徴、概念、構造を正しく考慮する必要がある。従って、ある理論が一貫した類似度を導けるのであれば、その理論には一定の妥当性を認めようという立場である。また、類似度を比較する実験は、他の感性評価やカテゴリ識別の実験と比べて結果が安定しているというメリットもある。この時、理論に含まれる関連するモノやコトの特徴、概念、構造を用いて現象を理解するのが合理的であるという意味で、その理論には認知的リアリティがあると言う。これまで認知的リアリティのある理論を探求してきた人工知能や言語学からも知見を借用すべきと考える。

最後に(3)に関して、計算機では複雑な楽曲全体を扱うのが難しく、また音楽家や聴取者の審美的価値観や社会的価値観を取り扱うのも難しいとしたら、その部分は人に任せそれ以外を計算機が扱うという分業をすることになろう。表現生成あるいはコンテンツ生成の問題には、人の意図を表出することに関して、表現の正確さと人にとっての記述コストのトレードオフが存在する。^{*2} 表現の正確さとは、人

がコンテンツ作成に関連する作業をしている時点で、最終的に出力されるコンテンツをどの程度意図通りに予測し制御できるかという特性である。

正確ではあるが記述コストのかかる手段として、例えば、エディタがある。人はエディタを利用してコンテンツの構成要素を1つずつ入力していく。generate&test法は、正確さのある程度犠牲にして記述コストを下げる手法であり、直接それを生成するシステムの実現は難しいが評価したり識別する関数を与えることは容易であるという一般的な性質を利用している。事例やモデルのシミュレーション(コーパスからの学習、事例ベース推論、アルゴリズム作曲など)は、記述コストを抑えて比較的容易に全自動化を達成できるが、表現能力や表現範囲、表現の正確さの点で制約がある。以上は表現あるいはコンテンツそのものを生成する手法であったが、性質が良く知られている基本演算あるいは使い慣れた基本機能を人が自由に組み合わせ実行するというプログラミング的な手法も考えられる。

我々は、楽曲生成・創作のための理論として、四則演算程度に容易に使いこなせる基本演算を考案し、それらの自由な組み合わせを可能とするような枠組の提供を目標とする。我々は音楽理論をベースとしているため、扱う対象は記号として記述された楽曲(譜面に記された楽曲)とする。^{*3}

3. 音楽の意味論

3.1 音楽の意味の分類

音楽の認知心理学や音楽理論は音楽の持つ意味を明らかにしてきた。一方、記号論によれば次の3要素によって意味が生じる: 表現あるいは記号(signifiant), 表現が指し示すもの(表現した結果, signifié), 表現の結果に対する意識的な観察者。認知心理学や音楽理論におけるどのような概念がこれら3要素に対応付けられるかを議論する。

表現は楽譜として記された楽曲に対応する。表現が指し示すものに関して、Leonard B. Meyer[14]は記号論に従って内在的か外在的かという観点と形式か情動かという観点から次のように分類した。

内在的	音楽の意味はその音楽に内在する。絶対的。
外在的	音楽以外の指し示された世界に存在する。参照的。
形式	音楽そのものが作る抽象的な形
情動	情動的経験を与える手段

ここで参照^{*4}はさらに指示的な参照(音楽とは種類の異なるモノ・コトを指し示す)と具現的な参照(音楽と同種の

^{*1} Albert Einstein の格言(1950): "Everything should be made as simple as possible, but not simpler."

^{*2} Donald Normanはこのトレードオフに対処するために、簡潔性デザイン公理 "The complexity of the information appliance is that of the task, not the tool. The technology is invisible" を与えた[18]。つまり、多くの表現者にとってより無意識に利用できる方法や手段の方が望ましい。

^{*3} Scriptism ともいう。対して、音響信号として記述された楽曲に対する同様の問題設定も興味深い。

^{*4} 音楽には指示代名詞が存在しないので、音楽における具現的な参照とは、旋律、リズム、和音等の並置、繰り返し、変形等によって暗示される関係性を指す。

モノ・コトを指し示す)に分かれる。前者は John B. Davis が「Darling, they are playing our tune 理論」と呼んだものであり、形式的な意味を論じる立場からは重要ではない。さらに Leonard B. Meyer によれば、意識的な観察者にとって表現の結果は情動 (emotion) として現れると言う [14]。音楽における情動とは、今聴いている音楽的事象が過去の経験に基づいて期待されていたものと同じか違うかを推論し、その結果生じる驚き、不安、安定、緊張、弛緩などの心理的变化を指す。情動は一時的ですぐ消える間主観的な心理現象であり、審美的な要素を含まない。^{*5}例えば、絶対的表現主義とは、音楽の意味は音楽を聴くその過程に限定的に備わっており、非指示的に音楽を聴いても情動の意味が現れると考える立場である。参照的表現主義とは、音楽の参照的な内容を理解できるかどうか情動的表現を左右すると考える立場である。例えば、Claude Debussy 作曲アラベスク冒頭の数小節の旋律概形はモスク壁面装飾に見られるアラベスク模様と同じであり、これは形式としての指示参照的な意味に分類される。

我々は、従来の音楽理論のアプローチを踏襲し、情動としての具現参照的な意味を採用する。つまり、ある音楽的事象 (1つの音、フレーズ、和音など) が他の音楽的事象を参照することで生じる情動をもって意味とする立場である。

3.2 意味を与えるための理論構造

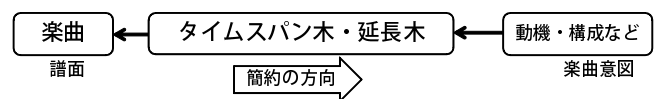
音楽の意味を探るため、言語と音楽の間のさまざまな共通点や差異が比較、議論されてきた。これは、人が同じ脳と同じ機能を使い分けて言語や音楽を理解し生成するからである [1][3]。さらに、言語進化論と音楽進化論^{*6}、言語発達と音楽発達の比較も盛んである。Lerdahl と Jackendoff は、音楽と言語の真の対応関係は、音楽理論を言語理論とは独立に構築した後に議論すべしと主張している [10], p.6。

我々は、音楽と言語の比較だけでなく、音楽理論と言語理論を比較することも有意義であると考え。言語では、表層の記号と明示されていない意味の区別が明確に意識されるため、表層の記号と意味がどのように関係しているかを説明する理論が古くから多数提案されてきた。音楽では、Leonard B. Meyer 以降に同様の動きが起き、チョムスキー (Noam Chomsky) の生成文法の枠組に則って、音楽理論 Generative Theory of Tonal Music (GTTM)[10] が提案された。^{*7}

まず、チョムスキー生成文法の枠組において人が言語コミュニケーションして理解するというのは、表出側の人が受取側の人にメッセージ (コンテンツ) を送り、受取側の人が表出側と同様なある構造をメッセージの記号列に割り当ててくることを指す。その構造は、階層的順序構造 (木構造) としてチョムスキー以降長年論じられてきたものである [9]。図 1(a) に示すように、発話意図が意味要素によって表現され、句構造規則と変形規則によって深層構造と表層構造に変形される。これらの規則が変形される木構造に意味を与えていく。意味を与える過程は文を産出する方向と同じである。一方 GTTM でも、楽曲の意図が動機や大域的な構成 (Schenker 理論の Ursatz に相当する) によって表現され、そこから簡約 (reduction) の逆操作である具体化 (elaboration) によって、タイムスパン木と延長木が産出される (図 1(b))。この具体化の操作が 2つの木構造に意味を与えていく。意味を与える過程は GTTM 分析過程と逆向きである。GTTM は生成文法の枠組を踏襲していると言われているが、規則の位置付けや、木構造の役割が異なっている。



(a) チョムスキー生成文法の枠組み



(b) GTTMが想定する意味の枠組み

図 1 文と楽曲に意味を与える枠組

4. 音楽の計算体系

4.1 表示的意味論

一般に計算するとは、その対象に対して定義された代数系を実行することである。我々が採用する、基本演算を用意しそれを自由に組み合わせるプログラミング的な手法 (第 2 章 問題意識 (3)) は代数的である。また、前章で議論したように、情動としての具現参照的な意味は木構造によって表現する。この枠組は、表示的意味論 [15] として捉えたと分かりやすい (図 2)。プログラミング言語における

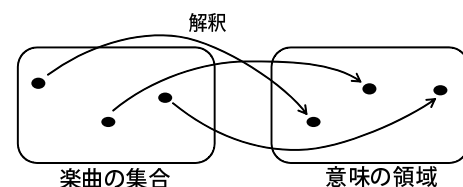


図 2 表示的意味論としての音楽の計算体系

表示的意味論では、意味の領域の各要素が各プログラムの

^{*5} 比較的永続的で安定した個人的現象は気分 (mood) と呼ばれる。
^{*6} 音楽の起源の議論をする際、今の音楽と昔の音楽はその意義もスタイルも異っている点に注意すべきである。太古の昔に中世や現代のような音楽が突然鳴り響いたりはない [5]。
^{*7} Leonard B. Meyer 直系の Eugene Narmour による Implication-Realization Model (IRM)[16] には、意味を与える枠組みが用意されていない。これは、Narmour の興味が楽曲の idiosyncratic な側面を顕在化させることに向いていて、帰納的に普遍的な意味論を構築する方に向いていなかったためであろう。

意味を表示 (denote) する。意味の領域に束などの代数系を仮定して、プログラムの等価性や妥当性などの性質を議論する。

音楽の場合、意味の領域の各要素が各楽曲の暗黙的な意味や構造を表示する。解釈という写像によって、楽譜として明示的に記述された楽曲と楽譜上に明示されていない旋律、リズム、和声の構造が対応づけられる。楽曲どうしの類似性や変形操作などを意味の領域での関数とみなして、楽曲の性質を議論することができる。

4.2 意味の領域上での演算とレンダリング

ここで我々は、GTTM が音楽聴取時のゲシュタルトに基づいて導入したタイムスパン木における簡約 (reduction) という概念に着目する。簡約を包摂 (subsumption) とみなすことで、意味の領域の要素間に半順序関係を導入でき、意味の領域を木構造を要素とする束とすることができる。つまり、解釈は楽曲からタイムスパン木を得ること (GTTM 分析) に対応する。束上の演算として *join* と *meet* が良く知られているので、ある楽曲が与えられた時、GTTM 分析によって得られた木構造が、*join*, *meet*, 包摂の関係にあるかどうかを調べることができる。

計算するとは何かを産出することである。GTTM では意味の領域に写像された音符をピッチイベントと呼ぶが、新しい木構造を産出する際、ピッチイベントを増やそうと思えば *join* を用い、減らそうと思えば *meet* を用いば良いだろう。しかし、意味の領域で *join* や *meet* などの演算を実行したとしてもそれは楽譜に記述された楽曲を産出しているわけではない。そこで、我々は GTTM 分析の逆操作を導入しそれをレンダリングと呼ぶ (図 3)。自然言語の

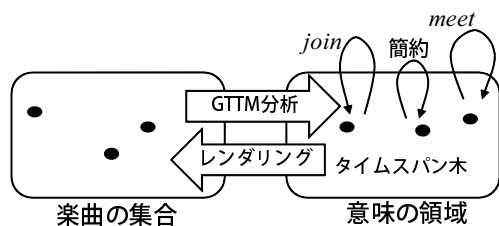


図 3 楽曲生成・創作のための枠組

場合は生成的な変形規則が陽に存在するので、虚辞、相槌音、間 (ま) のような削除可能な重要でない要素が極端に少ない。プログラミング言語の場合も厳格な文法と意味論が陽に存在するので、表層には冗長な要素が殆どない。これらに対して、音楽の「擬」変形規則は遥かに自由度が高い構文を生み出すので、表層構造を直接操作するような演算を定義するのは困難であると考え。より重要でない枝を削除する簡約という操作が音楽分析において意味を持つのはその傍証である [9]。ここで導入したレンダリングは、構文に含まれる自由度の高さを吸収する役割を果たす。

4.3 分析手法としての GTTM に対する指摘

意味の領域で議論した性質や産出された木構造が妥当な意味を持つためには、まず楽曲から正しいタイムスパン木を得る必要がある。しかし現在、GTTM 分析には不備が指摘されており、その内特に我々の枠組にとって重要なものを挙げる。

音楽的概念の記述が不完全である。例えば、曖昧な記述 (“... may be heard/placed”, “Prefer ...”), 未定義の概念 (安定性, 旋律類似性, 和声分析), 循環した規則 (GPR6, GPR7, MPR9) などである。従って、一般には計算機実装には適さないと考えられている [6][20]。

同一/類似の楽句やリズムが出現する所には同一/類似の構造が割当てられるという規則があり (GPR6, MPR1), Parallelism と呼ばれる。しかし、実際の楽曲に複数回現れる同一/類似の楽句に対し、異なる木構造を割当てて方が人の解釈として妥当な場合がある [12]。ここで 2 つの状況が考えられる。1 つは例えば 16 小節を越えるような長い楽節や楽章の場合である。この場合、そのように長い楽節に割当てられた 1 つの木構造自体にそもそも認知的な根拠があるのかどうか疑問である [3]。認知的リアリティを維持した理論構築を目指す立場からは、タイムスパン木の時間幅に上限を設けることが考えられる。延長木に関して、それをトップダウンに与えられる形式あるいは様式だと考えれば、延長木がタイムスパン木より長大になることはある程度容認できる。もう 1 つはより短い楽節の場合である。この場合、前後の楽句、出現する拍節位置、全体の中での出現位置が分析結果の木の形に影響を与える (文脈依存性) [12]。しかし GTTM にはそのような文脈を扱う規則は含まれていない。

タイムスパン木および延長木の位置付けと構成法が曖昧である。まず、安定したタイムスパン木を作るには、グルーピング分析と拍節分析に基づいてリズム的安定性 (rhythmic stability) を満たす必要がある。同時に、カデンツを 1 つの重要なピッチイベントにまとめる処理のため (cadential retention), 和声分析に基づいて音程的安定性 (pitch stability) も満たす必要がある [10], p.119。この 2 つの安定性を同時に満たすため、タイムスパン簡約分析の規則には、ローカルなグルーピング分析および拍節分析の結果を踏まえてボトムアップに 2 つの枝の優位性を比較するための規則と、トップダウンに得られる和声分析の結果から木の形を決める規則 (TSWFR4, TSRPR2, TSRPR5, TSRPR6b, TSRPR7) が含まれる。このボトムアップな規則とトップダウンな規則の混在は、タイムスパン木をアルゴリズム的に構成することを困難にし、タイムスパン木の位置付けを曖昧にする。同様の困難と曖昧さは延長木簡約分析においても現れる (PRPR1~PRPR4 が和声分析に関連する)。

そこで、我々はトップダウン処理とボトムアップ処理を

より厳密に区別し、タイムスパン木=リズムの安定性、延長木=音程的安定性というより単純で見通しのよい理論に再構成することを考える。そこに和声分析の方法と和声分析の結果を考慮する規則を明示的に導入する。さらに、文脈に関する規則を導入することで、上述の文脈依存性にも対応できるのではないかと考えている [12].

5. タイムスパン木の形式化

本章以降で我々のこれまでの研究成果を概観する。

5.1 最大タイムスパン

あるピッチイベント A が隣接するピッチイベント B を従属させると、A は等価的に A+B の時間を支配する。そうやって階層的に従属させるピッチイベントが増えていく時、A が支配する最長スパンのことを A の最大タイムスパン (maximal time-span) と呼ぶ [21]。これは、タイムスパン木上で最も上位 (幹に近い場所) で現れた時のタイムスパンに等しい。我々は、ピッチイベントの持つ情報あるいは複雑さはその最大タイムスパン長に比例すると仮定する。すると、簡約によってそのピッチイベントが削除されると、最大タイムスパン長分の情報が失われることになる。

図 4 (a) では e1~e4 という 4 つのピッチイベントが並んでおり、s1~s4 という細い線はピッチイベントの音価を表す。図 4 (b) における太い灰色の線は各最大タイムスパンを表す。最下層のピッチイベントに関して、音価と最大タイムスパン長は等しく、 $mt_2 = s_2$, $mt_3 = s_3$ である。それより上位では、 $mt_1 = s_1 + mt_2$, $mt_4 = mt_1 + mt_3 + s_4 = s_1 + s_2 + s_3 + s_4$ となる。つまり、 s_n の各最大タイムスパン長は、それより下位のタイムスパンをタイムスパン木の形に沿って足し合わせたものである。最大タイムスパンを導入したことで、タイムスパン木の簡約を半順序関係として定義できるようになった。

5.2 妥当な join 演算と meet 演算

次に、タイムスパン木どうしの join と meet を計算するには、木の形に沿って、対応する音の最大タイムスパンどうしの join と meet を計算する必要がある。ある音の持つ情報量はその最大タイムスパン長に比例するので、join と meet の値を図 5 のように定義する (時相論理の interval semantics を参照)。これら join と meet は吸収律を満たす: $(\sigma_A \sqcup \sigma_B) \sqcap \sigma_A = \sigma_A$ および $(\sigma_A \sqcap \sigma_B) \sqcup \sigma_A = \sigma_A$ 。

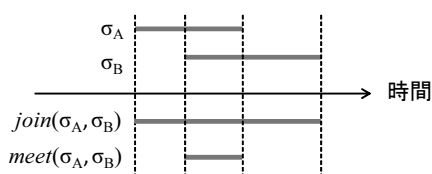


図 5 最大タイムスパンに対する join 演算と meet 演算

5.3 タイムスパン木の表現の拡張

2つのタイムスパン木が \wedge と \vee という形をしている時、吸収律を満たすようにそれらの meet と join を計算するため、意味の領域におけるタイムスパン木の表現として三分木を導入する (図 6) [7]。図中 \perp は空要素 (bottom) を表

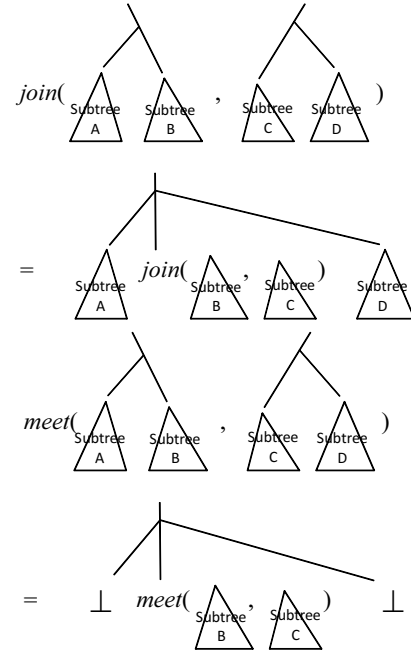


図 6 分岐の向きが異なるタイムスパン木の join と meet

す。三分木の意味は、分岐の向きが異なる 2 つのタイムスパン木ノードの重畳 (superposition) であり、三分木のタイムスパン木ノードを導入することではない。

6. 楽曲間の類似度を比較する実験

6.1 Join 演算と meet 演算に基づく距離

ある音が簡約によって削除された時に失われる情報量は、削除された音の音価ではなく、その音の最大タイムスパンに等しいと定義する [21]。そして、半順序関係 $\sigma_A \sqsubseteq \sigma_B$ なる 2 つのタイムスパン木 σ_A, σ_B がある時、 σ_A, σ_B 間の距離は、 σ_B から σ_A になるまで削除される最大タイムスパンの時間幅の総和と定義する ($|\sigma_B - \sigma_A|$ と書く)。タイムスパン木 T 全体が持つ情報量は \perp との距離であるから、全最大タイムスパンの総和 $|T - \perp|$ となる ($|T|$ と書く)。

任意のタイムスパン木 σ_A, σ_B 間の meet 経路の距離 d_{\sqcap} を $|\sigma_A - \sigma_A \sqcap \sigma_B| + |\sigma_B - \sigma_A \sqcap \sigma_B|$ と定義し、同様に join 経路の距離 d_{\sqcup} を $|\sigma_A \sqcup \sigma_B - \sigma_A| + |\sigma_A \sqcup \sigma_B - \sigma_B|$ と定義する。すると、join 距離と meet 距離は等しく、 $d_{\sqcup}(\sigma_A, \sigma_B) = d_{\sqcap}(\sigma_A, \sigma_B)$ である (証明は [21] 参照)。さらに、我々が定義した距離は、距離の公理の 1 つである三角不等式を満たすことも分かっている。

図 7 に例を示す。モーツァルト作曲きらきら星変奏曲 K.265/300e より、変奏曲 No.2 と No.5. の間の d_{\sqcap} と

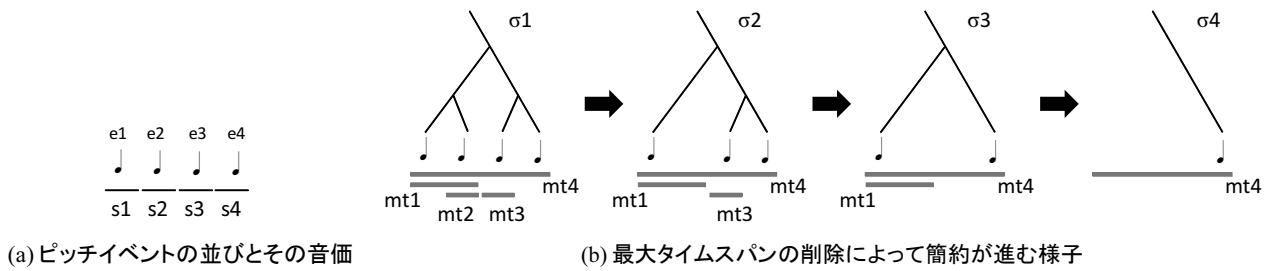


図 4 タイムスパン木の簡約と最大タイムスパンの作る階層

join(No.2, No.5) (822)

No.2 (744)

No.5 (654)

meet(No.2, No.5) (576)

図 7 変奏曲 No.2 と No.5 の間の *join* 経由距離と *meet* 経由距離

d_{\square} を計算する (実験のため単旋律に編曲してある)。図中、カッコ内の数値は全最大タイムスパンである。すると、 $d_{\square} = (822 - 744) + (822 - 654) = 246$ および $d_{\square} = (744 - 576) + (654 - 576) = 246$ である。

6.2 実験結果と分析

被験者実験により楽曲間の距離を測定し理論的に算出した距離と比較する。類似度は距離の逆数として定義し、理論的な値と被験者による値がどれだけ類似しているかによって認知的リアリティがあることを確認する。

モーツァルト作曲きらきら星変奏曲 K.265/300e の 12 の変奏曲の各冒頭 8 小節をテスト曲として用いた (図 8)。テーマも含めた 13 曲に対し対比較を行う (試行回数 ${}_{13}C_2 = 78$ 回)。被験者は大学生 11 名である。実験結果を多次元尺度構成法 (Torgerson scaling) を用いて表示する (図 9)。変奏曲 No.12 まで比較するとあまり良い対応がとれなかったため、変奏曲 No.9 までの比較を示す。寄与率は、理論的な結果 (a) 第 1 軸=0.23, 第 2 軸=0.21, 被験者の結果 (b) 第 1 軸=0.33, 第 2 軸=0.17 であった。図中 (a) と (b) の類似点としては、No.2, No.7 と No.3, No.4 と Theme, No.5, No.8, No.9 がそれぞれ集団を形成している, (b) を 90 度右回転させるとほぼ (a) の位置関係と重なる等が挙げられる。食い違っているのは、No.6 の位置である。詳細な実験方法, 結果, 分析は [7] に譲るが, 理論的な結果 (a) と被験者の結果 (b) の間には十分な対応関係が確認された。以上より, 我々が提案している枠組にはある一定の認知的リアリティがあると考えても良いだろう。

Theme

Variations

No. 1

No. 2

No. 3

No. 4

No. 5

No. 6

No. 7

No. 8

No. 9

No.10

No.11

No.12

図 8 変奏曲を基に実験のために編曲した単旋律

7. 楽曲生成に関する実験

我々は第 5 章にて四則演算程度に容易に使いこなせる基

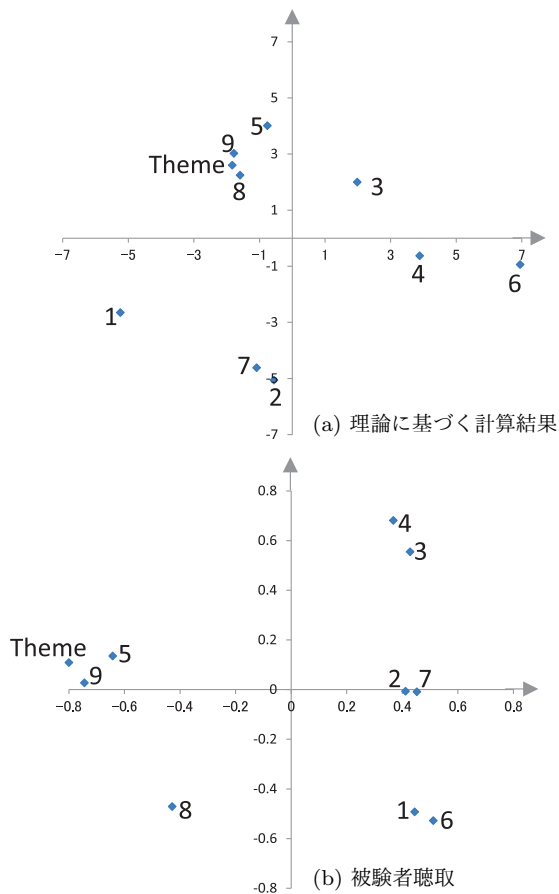


図 9 多次元尺度構成法で表示された変奏曲間の類似度

本演算として $join$, $meet$ を与えた。本章ではそれらを組合せて、楽曲生成の例としてモーフィングアルゴリズムを実現する。

7.1 モーフィングアルゴリズム

モーフィングとは、2つの楽曲が与えられた時、その任意の比率の内分点を計算することである。我々は以下5ステップから成るアルゴリズムを提案する (図 10) [8]。

- (1) 2つの楽曲のタイムスパン木 σ_A, σ_B が与えられるとまず $meet(\sigma_A, \sigma_B)$ を計算する。
- (2) σ_A からピッチイベントを1つずつ削除していくことで、 σ_A と $meet(\sigma_A, \sigma_B)$ を $N:M$ に内分する点 α を求める。
- (3) 同様に、 σ_B からピッチイベントを1つずつ削除していくことで、 σ_B と $meet(\sigma_A, \sigma_B)$ を $M:N$ に内分する点 β を求める。
- (4) α と β の $join$ を求める
- (5) $join$ をレンダリングして実際の楽曲を求める

図中、 $\alpha, \beta, meet(\sigma_A, \sigma_B), join(\alpha, \beta)$ の作る平行四辺形は、図 7 に見られる平行四辺形と同様の関係を表している。ここで図からも明らかなように、 $d(\sigma_A, \sigma_B) = d(\sigma_A, join(\alpha, \beta)) + d(join(\alpha, \beta), \sigma_B)$ である。また、 $|\sigma_A| \leq |\sigma_B|$ の場合 $|\sigma_A| \leq |join(\alpha, \beta)| \leq |\sigma_B|$ も成立する。

レンダリングのステップ (5) には簡易な方法を採用した。

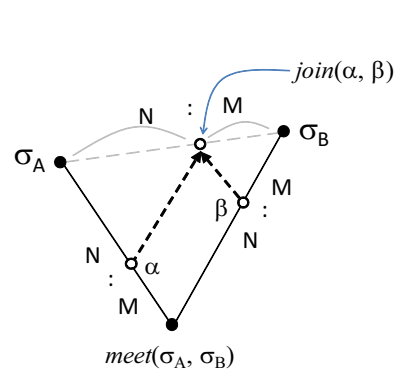


図 10 モーフィングアルゴリズム

それは、タイムスパン木の上位にあるタイムスパンから順に楽譜上の音符に変換し、後で変換された音符が上書きする方法である。5.3 節で導入した三分木の場合は、2つの木構造が重畳していると考え、単純に和音を生成する。

7.2 実験結果と分析

テスト曲は、前章の実験と同じく、モーツァルト作曲きらきら星変奏曲 K.265/300e の 12 の変奏曲の内 No.1, No.2, No.5 の各冒頭 8 小節を用いる (図 8)。上述のモーフィングアルゴリズムで実際に各変奏曲の 1:1 内分点の旋律を計算した結果を図 11 に示す。図中、例えば “No.2&No.5” は変奏曲 No.2 と No.5 の 1:1 内分点の旋律を表す。



図 11 変奏曲 No.1, No2, No.5 からモーフィングで生成された旋律

被験者実験の方法も前章とほぼ同様である。今回は変奏曲とモーフィングされた旋律合計 6 曲に対し対比較を行い (試行回数 ${}_6C_2 = 15$ 回)、被験者は大学生 6 名である。実験結果を多次元尺度構成法 (Torgerson scaling) を用いて表示する (図 12)。図中、No.1&No.2 は No.1 と No.2 のほぼ中央に、No.5&No.1 は No.5 と No.1 のほぼ中央に位置しているが、No.2&No.5 は中央から大きく外れ No.5 に近い。図 11 の No.2&No.5 を見ると、多くの音が 8 分音符であるのに対し、図 8 の No.2 は殆どが 16 分音符である。その No.2 の音が $join$ によって No.5 の同音高の音に吸収され、No.2&No.5 の聴取印象が No.5 に近くなってしまったと思われる。以上より意図を持って基本演算を組合せるとほぼ意図通りの出力が得られることが確認できた。一方で、被

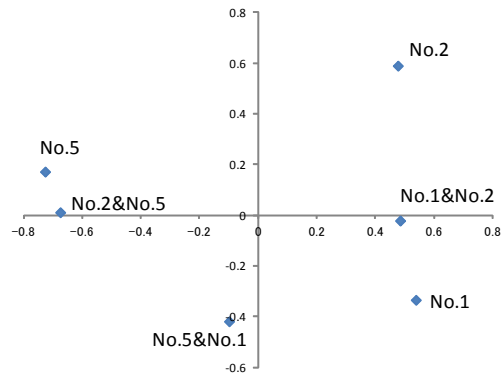


図 12 モーフィングされた楽曲に対する類似度聴取実験結果

験者の聴取印象が旋律の細部に影響を受けることも分かった。詳細な実験方法、結果、分析は [8] に譲る。

8. おわりに

現在 GTTM は最も信頼のおける音楽理論の 1 つと考えられている。しかし、GTTM を基に音楽の意味論や音楽の計算体系を構築しようとするとき幾つかの問題点があることを述べた。本稿では、GTTM を超克する 1 つの方法を示し、応用例としてモーフィングを実現し、認知的リアリティを確認した。我々が提案する理論枠組の利点は、楽曲生成アルゴリズムが、定規とコンパスで作図するように幾何学的に記述できる点と考えている。これは第 2 章 問題意識 (3) で指摘したトレードオフに対する 1 つの回答である。

レンダリングは意味の領域から楽曲への写像であり、様々な方法が考えられる。7.2 節で実装した方法以外にも、そのタイムスパンの由来となった実際の元のピッチイベントを保存しておく方法、旋律とその GTTM 分析によって得られたタイムスパン木を保存しておく事例ベース推論あるいは機械学習を行う方法がある。またレンダリングは GTTM 分析と逆方向の処理と見なせるので、レンダリングの妥当性の指標として、ある楽曲を GTTM 分析して得られたタイムスパン木をレンダリングした時に元と同一の楽曲に戻せるかどうか、どれだけ近い楽曲が生成されるかが使えるかも知れない。また、妥当なレンダリング結果を得るには妥当な GTTM 分析結果が前提となるので、レンダリングは分析過程と常に対で考えるべきであろう。

第 5 章以降に概観した研究成果では、木の形のみ考慮しピッチイベントの音高情報は無視していた。現在、Tonal Pitch Space 理論 [11] に基づいてより認知的リアリティの高い枠組の構築も進めている [13]。

謝辞

本研究は JSPS 科研費 23500145, 25330434, 26280089 の助成を受けたものです。

参考文献

- [1] Aiello, R. (Ed): 音楽の認知心理学, 誠信書房, 大串健吾 (監訳) (1998).
- [2] Cadwallader, A. and Gagné, D.: 調性音楽のシェンカー分析, 音楽之友社, 角倉一朗 (訳) (2013).
- [3] Cook, N.: 知覚 - 音楽理論からの展望, リタ・アイエロ編, 大串健吾監訳, 音楽の認知心理学, pp.72-110 (1998).
- [4] Duckles, V. et al.: Musicology §II, 2 Disciplines - Theory and Analytical Method, *The New Grove Dictionary of Music, Vol.12*, Sadie, S. (Ed), pp.840-842 (1980).
- [5] Goodall, H.: 音楽の進化史, 河出書房新社, 夏目大 (訳) (2014).
- [6] Hamanaka, M., Hirata, K. and Tojo, S.: Implementing "A Generative Theory of Tonal Music", *Journal of New Music Research*, Vol. 35, No. 4, pp. 249-277, 2007.
- [7] Hirata, K., Tojo, S., Hamanaka, M.: Cognitive Similarity grounded by tree distance from the analysis of K.265/300e, *Proc. of CMMR 2013*, pp.415-430 (2013).
- [8] Hirata, K., Tojo, S., Hamanaka, M.: Algebraic Mozart by Tree Synthesis, to appear in *Proc. of Joint Conference of ICMC and SMC 2014*.
- [9] 平田圭二, 東条敏: パーンスタインの「答えのない質問」再考: 計算論的音楽の理論の枠組みについて, 人工知能学会全国大会 (第 28 回) 論文集, 1K4-OS-07a-1 (2014).
- [10] Lerdahl, F., Jackendoff, R.: *A Generative Theory of Tonal Music*, The MIT Press (1983).
- [11] Lerdahl, F.: *Tonal Pitch Space*, Oxford University Press (2001).
- [12] Marsden, A., Hirata, K., Tojo, S.: Towards Computable Procedures for Deriving Tree Structures in Music: Context Dependency in GTTM and Schenkerian Theory, *Proc. of SMC 2013*, pp.360-367 (2013).
- [13] Matsubara, M., Hirata, K., Tojo, S.: Distance in Pitch Sensitive Time-span Tree, to appear in *Proc. of Joint Conference of ICMC and SMC 2014*.
- [14] Meyer, L.B.: 音楽における情動と意味, リタ・アイエロ編, 大串健吾監訳, 音楽の認知心理学, pp.3-45, 誠信書房 (1997). 原典: *Emotion and Meaning in Music*, University of Chicago Press (1956).
- [15] Mosses, P.D.: Denotational Semantics, *Handbook of Theoretical Computer Science, Volume B*, The MIT Press, Chapter 11, pp.575-631 (1994).
- [16] Narmour, E., *The Analysis and Cognition of Basic Melodic Structure - The Implication-Realization Model*, The University of Chicago Press (1990).
- [17] Nierhaus, G.: *Algorithmic Composition*, SpringerWien-NewYork (2009).
- [18] Norman, D.: *The Invisible Computer*, The MIT Press (1999).
- [19] Snyder, R.: 音楽と記憶 - 認知心理学と情報理論からのアプローチ, 音楽之友社 (2003).
- [20] Temperley, D.: *The Cognition of Basic Musical Structures*, The MIT Press, Cambridge, 2001.
- [21] Tojo, S. and Hirata, K.: Structural Similarity Based on Time-span Tree, *Proc. of CMMR 2012*, pp.645-660 (2012).
- [22] Wiggins, G.A., Müllensiefen, D., Pearce, M.T.: On the non-existence of music: Why music theory is a figment of the imagination. In: *Musicae Scientiae, Discussion Forum 5*, 231-255 (2010).
- [23] Wilson, R.A. and Keil, F. (Eds): *The MIT Encyclopedia of the Cognitive Sciences*, The MIT Press, 1999.