

音楽理論 GTTM に基づくグルーピング構造獲得システム

浜中 雅俊[†] 平田 圭二^{††} 東条 敏^{†††}

本論文では、音楽理論 Generative Theory of Tonal Music (GTTM) に基づき、曲をフレーズ、モチーフなどに自動でグルーピングするシステムについて述べる。楽曲の切れ目を発見する手法は従来からも検討されてきたが、それらは主にメロディの局所的な境界を求めることが主眼であった。GTTM によるグルーピングでは、そのような局所的な境界を求めると同時に、メロディの繰返しなどを発見し、階層的な全体構造を獲得することを目的とする。しかしながら、GTTM は多数のルールから構成されており、その適用に関して優先順序が定義されていないため、グルーピング構造を獲得するためにはこれまで恣意的・非手続的な手作業が必要であった。この問題を解決するため、本研究では、計算機実装用にルールを再形式化した GTTM の計算機モデル exGTTM を提案する。exGTTM の特長は、ルールの優先順位を決めるためのパラメータを導入したことである。計算機上に実装した exGTTM を用いてグルーピング構造分析を行い性能を評価した。

Grouping Structure Generator Based on Music Theory GTTM

MASATOSHI HAMANAKA,[†] KEIJI HIRATA^{††} and SATOSHI TOJO^{†††}

This paper describes a grouping system which segments a music piece into units such as phrases or motives, based on the Generative Theory of Tonal Music (GTTM). Previous melody segmentation methods have only focused on detecting local boundaries of melodies, while the grouping analysis of GTTM aims at building a hierarchical structure including melodic repetition as well as such local boundaries. However, as the theory consists of a number of structuring rules among which the priority is not given, groups are acquired only by the ad hoc order of rule application. To solve this problem, we propose a novel computational model exGTTM in which those rules are reformalized for computer implementation. The main advantage of our approach is that we attach a weight on each rule as an adjustable parameter, which enables us to assign priority to the application of rules. In this paper, we show the process of grouping analysis by exGTTM, and show the experimental results.

1. はじめに

本論文では、音楽理論 Generative Theory of Tonal Music (GTTM)¹⁾ のグルーピング構造分析を計算機上に実装する手法について述べる。

音楽というメディアの認識や表現は曖昧なため、ユーザの思いどおりに計算機に作曲させたり演奏させたりすることは一般に困難である。市販の楽譜エディタやシーケンサが操作できる対象は、音符、休符、和音名など曖昧性が低い表層的な構造に限定されている。我々の目的は、専門家の持つ音楽知識を計算機上に形式的に記述することで音楽知識の乏しいユーザを支援

し、旋律、リズム、和声といった高次の音楽的な構造を適切に操作できる音楽システムを実現することである。そのための第1歩として、現在我々は、音楽理論 GTTM の計算機上への実装を試みている。

音楽理論は、音楽に関する知識、経験、技能を利用して楽曲を分析、解釈する方法論を我々に与える。音楽は様々な側面から分析、解釈することができるため、これまで多くの音楽理論^{2)~4)} が提案されており、楽曲の分析や解釈に用いられる様々な音楽的な概念が抽出され、様々な手順が議論されてきた。

音楽理論 GTTM の特徴は、音楽が備える多様な側面を包括的に表象しているという点である。音楽知識の乏しいユーザを支援し音楽的な構造を適切に操作するという我々の目標と照らし合わせると、音楽の持つ旋律、リズム、和声という3つの側面に関して一貫性のある操作を実現する必要があると考える。たとえば、楽曲を2つに分割するという単純な操作を考えたとき、着目する音楽的な構造によってその操作の実現

[†] 科学技術振興機構さきがけ研究員

PRESTO Japan Science and Technology Agency

^{††} NTT コミュニケーション科学基礎研究所

NTT Communication Science Laboratories

^{†††} 北陸先端科学技術大学院大学

Japan Advanced Institute of Science and Technology

は異なってくるが、装飾が付いた楽曲とそうでない楽曲に対して2つに分割する箇所は本質的に同じであることが望ましい。GTTMでは、旋律の区切りを表現するグルーピング構造とリズムや韻律を表現する拍節構造をもとに、旋律や和声を本質的な部分と装飾的な部分に区別する簡約構造を抽出する手順が提案されている。GTTMに従えば、旋律、リズム、和声という3つの側面に関して一貫性のある操作の実現が期待できよう。

GTTMのもう1つの特徴は、楽曲中に現れる音楽的な構造や関係を詳細に検討し得られた知識や手順をルールとして記述している点である。他の多くの音楽理論が楽曲の分析例を通じてその考え方を記述する傾向があるのに対し、GTTMのルールは、音楽に関する基本的現象や概念の定義から積み上げ、満たすべき制約や望ましい状態を規定している。これより、計算機上のプログラムとして記述しやすい、音楽知識を人間が理解しやすい大きさに分割して整理され表現しているなどの利点が得られる。

音楽知識を計算機上に形式的に記述する観点から、我々はGTTMが最も有望であると考えている。GTTMに基づいて音楽的な構造を機械的に得られれば、音楽の意味構造の分析^{5),6)}が可能となり、たとえば、演奏の表情づけ⁷⁾⁻⁹⁾、音楽情報検索システムの出力結果を提示する際などに有用な音楽要約^{10),11)}などの音楽システムに応用できるであろう。

本論文では、GTTM全体の機械化への第1歩として、グルーピング構造分析の計算機上への実装法について述べる。GTTMのルールを計算機上へ実装する際の課題は次の2つに大別される：(1)すでにルールや概念として定性的に記述されているが、計算機上のプログラムとして実現するために定量的に記述し直すこと、(2)ルールや概念として明示的に記述されていないが、計算機上のプログラムとして実現するために必要なアルゴリズムやパラメータを発見し補うこと。まず(1)に関して、パラメータの存在は示されているがその値が不明な場合は、パラメータの値域を定義することとした。音楽の解釈は曖昧なので、一般に、人間がGTTMに従ってグルーピング構造分析した結果は唯一には決まらない。よって、我々のグルーピング構造分析器の分析結果が、人間の分析結果全体を包含すれば十分であると考え、そのような分析結果を生成できるような値域を決定することを目標とする。次に(2)に関しても同様に、その値を調整すれば人間の分析結果全体を包含できるような新しいパラメータとその値域を導入する。

我々は上述の方針に従い、GTTMを拡張したexGTTMを提案する。その拡張の具体例として、ルール間競合を解決するためにルールの優先順位を制御するパラメータを導入したことや、階層的なグルーピング構造を獲得するためにボトムアップ的に求めた局所的グループ境界を用いてトップダウン的に階層構造を求めるアルゴリズムとその実行を制御するパラメータを導入したことなどがあげられる。exGTTMの複数のパラメータ値を変化させると、それにつれて得られる階層的グルーピング構造も変化し、人間が正しいと考える階層的グルーピング構造を生成することを目指す。

グルーピング構造分析の関連研究としては楽曲の切れ目を発見する旋律分割の研究があり、これまで多数のアルゴリズムが提案されている^{4),12)-14)}。従来の旋律分割手法では旋律の局所的境界を求めることに主眼が置かれており、GTTMのようにタイムスパン木の獲得に必要な階層的なグルーピング構造を得ることは目標とされていなかった。文献15)では、ピアノロール譜面上に描いたポロノイ線図を用いて、多声音楽の階層的なグルーピングを実現した。しかし音楽的な論拠に乏しく、階層的なグルーピング構造が適切にもとまらない場合もあった。また、GTTMの計算機上への実装もいくつか試みられている。文献12)ではグルーピング構造分析の基本的な2つのルールについてのみ実装を行っており、階層的グルーピング構造は生成できなかった。文献20)ではタイムスパン木の獲得まで実装していたが、ルールの適用を手作業によって制御していた。

以下、2章では音楽理論GTTMを概観し、GTTMを計算機上へ実装するうえでの課題と解決法について議論し、GTTMを拡張したexGTTMを提案する。3章ではexGTTMのグルーピング構造分析のアルゴリズムの詳細について述べる。そして4章では、実装したグルーピング構造分析の性能を評価し、5章でまとめと今後の課題について述べる。

2. GTTMの計算機上への実装

GTTMは、音楽に関して専門知識のある聴取者の直観を形式的に記述するための理論で、グルーピング構造分析、拍節構造分析、タイムスパン簡約、プロロングーション簡約という4つのサブ理論から構成されている。

グルーピング構造分析は、連続したメロディをフレーズやモチーフなどに階層的に分割するもので、長いメロディを歌うときにどこで息継ぎすべきかを見つける

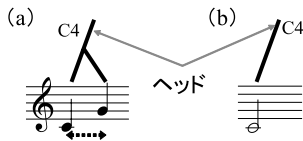


図1 タイムスパン木の例
Fig. 1 Simple example of time-span tree.

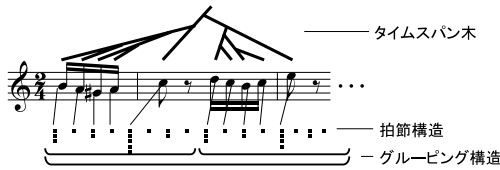


図2 グルーピング構造, 拍節構造, タイムスパン木
Fig. 2 Grouping structure, metrical structure and time-span tree.

ような分析である。拍節構造分析は、4分音符/2分音符/1小節/2小節/4小節など各拍節レベルにおける強拍と弱拍を同定するもので、聴取者が曲に合わせて手拍子を打つタイミングや指揮者がタクトを振るタイミングを求めよう分析である。タイムスパン簡約は、メロディの重要な部分と装飾的な部分を分離するもので、構造的に重要な音が幹になるような2分木(タイムスパン木)を求めよう分析である。図1(a)は、メロディとそのタイムスパン木を描いたものであるが、タイムスパン(←---→で表された部分)は、図1(b)のように2音をヘッドと呼ばれる1つの音で代表させることができる(ここではC4の音)。プロロンゲーション簡約は、和声的に持続的な部分・変化する部分を明示し、和声間の従属関係を明示する木構造を作る処理である。GTTMの各サブ理論は次の2種類のルールで表現されている: 構成ルールはグルーピング構造や拍節構造が満たすべき条件や制約を記述したルールであり、選好ルールは構成ルールを満たす複数の構造のいずれかが好ましいかを示すルールである。

我々は上記の4つのサブ理論のうち、プロロンゲーション簡約を除く3つのサブ理論の計算機上への実装を試みている¹⁶⁾⁻¹⁹⁾(図2)。本論文においては、このうちのグルーピング構造解析について扱う。そして本章では、このうち特にグルーピング構造解析の選好ルール(Grouping Preference Rule; GPR)に関する問題とその解決法を議論する。GPRは、GPR1(単音の非グルーピング化), GPR2(音の近接箇所), GPR3(音の断絶箇所), GPR4(GPR2および3による境界判定の有意性), GPR5(音符列長の対称性), GPR6(メロディの並列性), GPR7(木の安定性)という7つのルールからなる。GPR2は、(a)(スラー/休符)

と(b)(アタックポイント)の2通りに分かれ、GPR3は(a)(音高差), (b)(強弱), (c)(アーティキュレーション), (d)(音価)の4通りに分かれる。

2.1 GPR実装上の問題点

GTTMは、他の音楽理論と比べて比較的厳密なルールで記述されており、音楽知識を形式化するうえで最も有望であると我々は考えている。しかし、それでもなお、実装を実現するためには困難な問題がある。1章では、GTTMの各ルールを実装するうえでの2つの問題点(定性的な記述を定量的に記述し直すことと、必要なパラメータを補うこと)とその解決の方針について述べた。本節ではこれらをさらに詳しく議論する。

2.1.1 楽曲分析における曖昧性の峻別

一般に、GTTMを含む音楽理論の分析結果は唯一に決まらないことが多い。このうち(i)音楽理論自体に曖昧さがあるため分析結果が曖昧になることと、(ii)楽曲の解釈自体に曖昧性が内在していることで、分析結果も曖昧になることは区別すべきであると考えられる。本研究では、(ii)の内在する曖昧性は積極的に認めつつ、(i)の音楽理論の曖昧性を解消することを主たる目標とする。ただし、(i)と(ii)を厳密に区別することが困難な場合も存在する。たとえば、GTTMのGPR6(メロディの並列性)では、類似したメロディが類似したグルーピングになることを述べている。ここで、GTTMでは、並列性についての詳細な定義が与えられていない点は(i)の理論自体の曖昧さに対応する。同時に、様々な並列性の定義が考えられる点は(ii)の解釈の曖昧さに対応する。

2.1.2 階層構造に関するアルゴリズムの欠落

GTTMの選好ルールには、局所的な構造からより高次の構造を生成するボトムアップ方向に働くルールと大局的な構造をより低次の構造に分解するトップダウン方向に働くルールが混在している。しかし、その両者をどのように組み合わせる適切な階層構造を生成するかに関するアルゴリズムは欠落している。素朴に、最小単位(最も下位)のグループを生成した後にグループを結合していくことで次々と上位のグループを作っていくボトムアップな方法では、大局的な構造に関するルールを適用することが困難であり、一方、上位のグループを分割していくことで下位のグループを作っていくトップダウンな方法では局所的な構造に関するルールを適用することが困難である。したがって、大局的な構造に関するルールと局所的な構造に関するルールの両方を適切に組み合わせるアルゴリズムを構築する必要がある。



図 3 ルールの競合の例

Fig. 3 Simple example of conflict between rules.

2.1.3 選好ルール間の競合

選好ルールを適用する際には、複数のルール間での優先度が決まっていないため、競合が起きることがある。図 3 は、音程における跳躍点をグループの境界とするルール GPR3a (図中の a の箇所) とメロディの繰返し時点を境界とするルール GPR6 (図中の b の箇所) が競合する場合である。この両方を切れ目とすると単音 (第 4 音や第 8 音) が 1 グループになってしまい、これは GPR1 では避けるべきとされている。したがって GPR3a, GPR6 の一方しか適用されず、両者は競合関係にある。

2.2 解決法: exGTTM の提案

本節では、計算機上で実行可能となるよう理論を拡張した exGTTM を提案する。

2.2.1 exGTTM 設計上の方針

我々は、楽曲の正しい解釈は複数あるという前提のもとで以下に述べる 2 つの方針に基づき exGTTM を設計する。まず、1 つ目の方針は、パラメータを導入して曖昧さをできる限り排除することである。exGTTM の目的は、これらのパラメータを手動で動かすことによって、楽曲の多様な解釈をすべて出力できるような分析器を実現することである。導入したパラメータは、以下の 3 つのカテゴリに分類できる。

まず第 1 のカテゴリは、GTTM でその存在は明らかであったものの具体的な値が与えられていなかったパラメータである。たとえば、GPR2a というルールは成立するか否かが排他的に決まるので、成立すれば D_{2a} というパラメータの値を 1 に、不成立ならば 0 に対応付ける。また GPR5 というルールの場合、ルールが成立する度合いは連続的なので、 D_5 というパラメータの値も 0 から 1 の間の連続値をとる。第 2 のカテゴリは、GTTM ではその存在自体が暗黙であったものを明示化したパラメータである。この例として、GTTM の各選好ルールの強さを重み付けするパラメータがあげられる。2.1.3 項で述べたように、GTTM ではルールの適用に際して競合が起きることは認識されており、何らかの方法で解決しなければならなかった。このパラメータを導入することによって、ルールの強

さの制御が可能となり、競合を解消することが可能となる。第 3 のカテゴリとして、GTTM ではその存在自体が議論されていなかったパラメータがある。この例として、並列性に関するルール GPR6 においてフレーズ間で並列性がどの程度で成立するかを表すパラメータがあげられる。GTTM では並列性について詳細な定義が与えられていなかった。以上、exGTTM のそれぞれのパラメータの詳細については 3 章で詳述する。

exGTTM を設計するにあたっての 2 つ目の方針は、並列性など詳細な定義がないまま用いられている用語に、直感的で分かりやすい定義を与えることである。たとえば、メロディの並列性に関しては、様々な定義²¹⁾ が可能である。本研究では、ユーザが並列性に関するパラメータを手動で調節することで 2 つのメロディが並列的であるかどうかを自由にコントロールできるように、できる限り直感的で分かりやすい定義を与えるを試みた。

2.2.2 階層構造獲得のためのアルゴリズム

局所的な構造に関するボトムアップなルールと、大局的な構造に関するトップダウンのルールを組み合わせ、階層的な構造を獲得する手法を提案する。グルーピング構造分析では、以下のようなアルゴリズムによって階層的な構造を獲得する。

- (1) 楽曲全体を 1 つのグループとする。
- (2) 局所的な構造に関するルールを適用。
- (3) 局所的な境界の強さを算出し、局所的境界を検出する。
- (4) 大局的な構造に関するルールを適用。
- (5) 高次の境界の強さを算出する。
- (6) 最も強い境界でグループを 2 つに分割する。
- (7) グループ中に局所的境界がある限り (4), (5), (6) を再帰的に繰り返す。

このアルゴリズムによりルール適用に関する順序の規範を与えることができ、前節で問題にした大局的・局所的な処理を組み合わせる構造化が実現される。

3. exGTTM に基づくグルーピング手法

階層的なグルーピング構造は、ボトムアップ処理により求めた局所的境界を用いて、トップダウンに獲得する。図 4 に、システムの構成を示す。システムの入力形式には、楽譜作成や、分析、検索ツールが普及しており、フォーマットの相互変換が容易な MusicXML²²⁾ を採用した。獲得したグルーピング結果の出力形式としては、GroupingXML を新たに設計した。GTTM ではすべての楽曲をホモフォニー (homophony) とし

exGTTM は、extended-GTTM (拡張した GTTM) と executable-GTTM (実行可能な GTTM) の両方の意味をあわせ持っている。

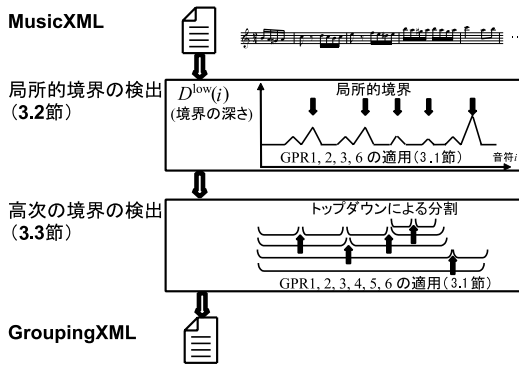


図 4 システムの構成

Fig. 4 Processing flow of the system.

て扱っているが、本研究では、分析の対象をモノフォニー (monophony) に限定する。

3.1 グルーピング選好ルールの適用

本節では、 GPR_R ($R \in \{1, 2a, 2b, 3a, 3b, 3c, 3d, 4, 5, 6\}$) の 10 個のルールの適用について述べる。各ルールにより生じるグループの境界の深さは、 $D_R(i)$ ($0 \leq D_R(i) \leq 1$, $R \in \{1, 2a, 2b, 3a, 3b, 3c, 3d, 4, 5, 6\}$) で表される。本研究では表 1 の 15 個の調節可能なパラメータを導入する。図 5 は、グルーピング選好ルールの適用とパラメータの関係を表したもので、式番号を括弧付き数字で、変数を四角、パラメータを角の丸い四角で示している。

グルーピング構造分析には、上記 10 個のルールのほかに、「タイムスパン簡約やプロロンゲーション簡約が安定するグルーピング構造を選択する」と定義されている GPR7 (time-span and prolongational stability) というルールがある。しかし、高次の構造から低次の構造へのフィードバックする方法について詳細な説明がなく、またルールの適用例も少ないため今回は実装することができなかった。GPR7 の実装については今後の課題とする。

3.1.1 基本変数の算出

MusicXML から 6 つの基本変数を算出する。6 つの変数はそれぞれ、消音時刻から次の発音時刻までの間隔 ρ_i 、発音時刻間隔 l_i 、音高の差 (音程) η_i 、ダイナミクス (音量) の差の絶対値 δ_i 、楽譜上の音符の長さ実際に演奏された音の長さの比の差 α_i 、音価の差の絶対値 β_i である。

図 6 において、 τ_i は i 番目の音の楽譜上の正規の発音時刻、 ε_i は i 番目の音の楽譜上の正規の消音時刻、

表 1 15 個の調節可能なパラメータ
Table 1 Fifteen adjustable parameters.

パラメータ	説明
S_R	各ルールの強さを表すパラメータ。値が大きいくほど、ルールの強さが強くなる。 $R \in \{2a, 2b, 3a, 3b, 3c, 3d, 4, 5, 6\}$ 式 (19), (23) で使用。
$\hat{\sigma}$	GPR5 で用いる平均をグループの中心とする正規分布の標準偏差。値が大きくなるほど正規分布の裾野が広がる。 式 (14) で使用。
W_m	GPR6 で、各音の発音時刻の類似度と音高差の類似度のどちらを重視するか決めるパラメータ。値が大きいくほど、音高差の類似度を重視する。 式 (15) で使用。
W_l	GPR6 で、並列的な区間の長さをどのくらい重視するかを決めるパラメータ。値が大きいくほど、長い並列的な区間を重視する。式 (15) で使用。
W_s	GPR6 で、音符 i が並列な区間の始端あるいは終端のいずれになる方を重視するかを調整するパラメータ。値が大きいくほど終端を重視する。式 (16) で使用。
T_4	GPR4 で、GPR 2, 3 の効果が明白であるかどうかを決める閾値。値が小さいほど、GPR4 が成立しやすくなる。式 (13) で使用。
T^{low}	低レベルの境界であるかどうかを決める閾値。値が小さいほど、境界と認識されやすくなる。式 (20) で使用。

$\hat{\varepsilon}_i$ は i 番目の音の実際の消音時刻、 f_i は i 番目の音の音高、 v_i は i 番目の音のベロシティを表している。時間の単位は 4 分音符長、音高の単位は半音 (MIDI ノートナンバ) である。このとき、基本変数はそれぞれ以下のように表される。

$$\rho_i = \begin{cases} \tau_{i+1} - \varepsilon_i & \text{if } \tau_{i+1} - \varepsilon_i \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$l_i = \tau_{i+1} - \tau_i \quad (2)$$

$$\eta_i = f_{i+1} - f_i \quad (3)$$

$$\delta_i = |v_{i+1} - v_i| \quad (4)$$

$$\alpha_i = \left| \frac{\varepsilon_{i+1} - \tau_{i+1}}{\hat{\varepsilon}_{i+1} - \tau_{i+1}} - \frac{\varepsilon_i - \tau_i}{\hat{\varepsilon}_i - \tau_i} \right| \quad (5)$$

$$\beta_i = |l_{i+1} - l_i| \quad (6)$$

3.1.2 GPR2, 3, 4 の適用

GPR2, 3, 4 は連続する 4 音 (n_1, n_2, n_3, n_4) に関するルールである。 i 番目の遷移 (i 音目と $i+1$ 音目の間) が各ルールにより生じる境界の深さは、境界になりうる ($D_R(i) = 1$) かそうでない ($D_R(i) = 0$) かで表される。

GPR2a (スラー/休符) では、 n_2 の終わりから n_3

ホモフォニーは和音を含む単旋律、モノフォニーは和音を含まない単旋律である。

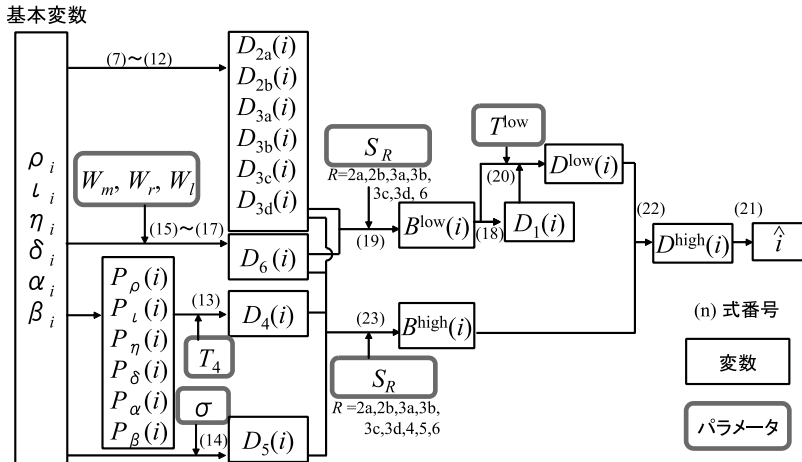


図 5 GPR の適用とパラメータの関係
Fig. 5 Relationship between parameters and GPRs.

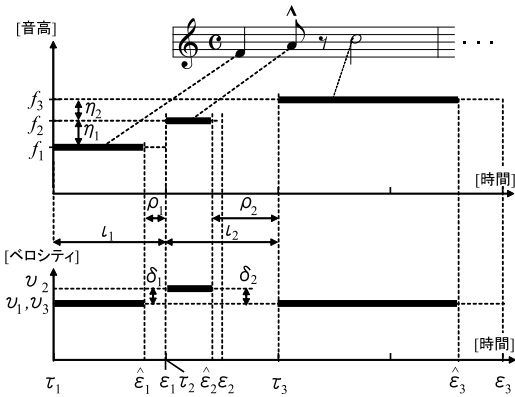


図 6 基本変数の算出
Fig. 6 Calculation of basic parameters.

の始まりまでの時間間隔が n_1 の終わりから n_2 の始まりまでの時間間隔および、 n_3 の終わりから n_4 の始まりまでの時間間隔よりも長い場合、グループの境界と認識される。GPR2a は次式のように定式化できる。

$$D_{2a}(i) = \begin{cases} 1 & \text{if } \rho_{i-1} < \rho_i \text{ and } \rho_i > \rho_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

GPR2b (アタックポイント) では、 n_2 の始まりから n_3 の始まりまでの時間間隔が n_1 の始まりから n_2 の始まりまでの時間間隔および、 n_3 の始まりから n_4 の始まりまでの時間間隔よりも長い場合、グループの境界と認識される。GPR2b は次式のように定式化できる。

$$D_{2b}(i) = \begin{cases} 1 & \text{if } l_{i-1} < l_i \text{ and } l_i > l_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

GPR3a (音高差) では、 n_2, n_3 の音高差が n_1, n_2 および n_3, n_4 の音高差よりも大きい場合、グループの境界と認識される。GPR3a は次式のように定式化できる。

$$D_{3a}(i) = \begin{cases} 1 & \text{if } |\eta_{i-1}| < |\eta_i| \text{ and } |\eta_i| > |\eta_{i+1}| \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

GPR3b (強弱) では、 n_2, n_3 にダイナミクス (音量) の変化があり、 n_1, n_2 および n_3, n_4 でダイナミクスの変化がない場合、グループの境界と認識される。GPR3b は次式のように定式化できる。

$$D_{3b}(i) = \begin{cases} 1 & \text{if } \delta_{i-1} = 0, \delta_i \neq 0, \text{ and } \delta_{i+1} = 0 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

GPR3c (アーティキュレーション) では、 n_2, n_3 でアーティキュレーションの変化があり、 n_1, n_2 および n_3, n_4 でアーティキュレーションの変化がない場合、グループの境界と認識される。GTTM では、アーティキュレーションの意味について明確な定義がない。ここでは、文献 1) での適用例から、楽譜上の音符の長さ与实际に演奏された音の長さの比という意味であると判断した。GPR3c は次式のように定式化できる。

$$D_{3c}(i) = \begin{cases} 1 & \text{if } \alpha_{i-1} = 0, \alpha_i \neq 0, \text{ and } \alpha_{i+1} = 0 \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

GPR3d (音価) では、 n_2 と n_3 が異なった音価で、 n_1 と n_2 および n_3 と n_4 が同じ音価の場合、グ

ループの境界と認識される．GPR3d は次式のように定式化できる．

$$D_{3d}(i) = \begin{cases} 1 & \text{if } \beta_{i-1} = 0, \beta_i \neq 0, \text{ and} \\ & \beta_{i+1} = 0 \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

GPR4 (**GPR2** および **3** による境界判定の有意性) では, **GPR2**, **3** で示される効果が比較的明白な場合, グループの境界と認識される．**GPR4** は次式のように定式化できる． $P_\rho(i), P_l(i), P_\eta(i), P_\delta(i), P_\alpha(i), P_\beta(i)$ それぞれ, **GPR2a**, **2b**, **3a**, **3b**, **3c**, **3d** が明白に成立するほど大きな値を示す． T_4 ($0 \leq T_4 \leq 1$) は, **GPR2a**, **2b**, **3a**, **3b**, **3c**, **3d** の効果が明白であるかどうかを決める閾値である．

$$D_4(i) = \begin{cases} 1 & \text{if } \max(P_\rho(i), P_l(i), P_\eta(i), \\ & P_\delta(i), P_\alpha(i), P_\beta(i)) > T_4 \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

ただし,

$$P_\rho(i) = \begin{cases} \rho_i / (\rho_{i-1} + \rho_i + \rho_{i+1}) \\ \text{if } \rho_{i-1} + \rho_i + \rho_{i+1} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$P_l(i) = l_i / (l_{i-1} + l_i + l_{i+1})$$

$$P_\eta(i) = \begin{cases} |\eta_i| / (|\eta_{i-1}| + |\eta_i| + |\eta_{i+1}|) \\ \text{if } |\eta_{i-1}| + |\eta_i| + |\eta_{i+1}| > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$P_\delta(i) = \begin{cases} \delta_i / (\delta_{i-1} + \delta_i + \delta_{i+1}) \\ \text{if } \delta_{i-1} + \delta_i + \delta_{i+1} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$P_\alpha(i) = \begin{cases} \alpha_i / (\alpha_{i-1} + \alpha_i + \alpha_{i+1}) \\ \text{if } \alpha_{i-1} + \alpha_i + \alpha_{i+1} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$P_\beta(i) = \begin{cases} \beta_i / (\beta_{i-1} + \beta_i + \beta_{i+1}) \\ \text{if } \beta_{i-1} + \beta_i + \beta_{i+1} > 0 \\ 0 & \text{otherwise} \end{cases}$$

3.1.3 GPR5 (音符列長の対称性) の適用

GPR5 は対称性 (symmetry) に関するルールで, 下位の階層のグループの境界が上位の階層にあるグループの中心に近くなるような構造を優先する．しかし, GTTM には対称的であるかどうかをどのように評価すればよいかについての定義がない．

本研究では, 上位の階層のグループを 2 つに分割し下位の階層のグループを生成する際, 分割された 2 つのグループの長さが等しいほど高い値を示す関数

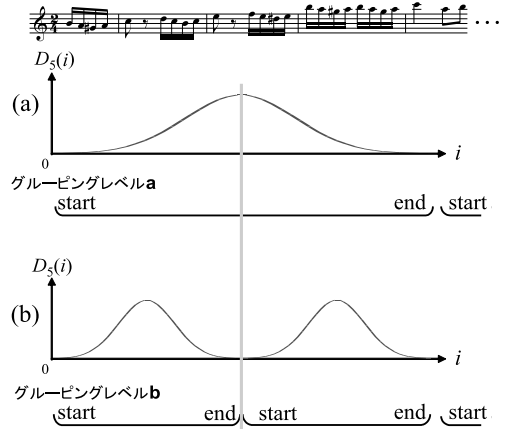


図 7 対称性の度合い $D_5(i)$
Fig. 7 Degree of symmetry $D_5(i)$.

$D_5(i)$ を定義し, その関数によって対称的である度合いを表現することにする．ここでは, そのような関数として平均をグループの中心, 標準偏差を σ とする正規分布を用いる． σ の単位は 4 分音符長である．裾野の広さを 1 つのパラメータで変えられる関数は, 正規分布のほかにも存在するが, 今回は正規分布を採用した．

図 7(a) は, グルーピングレベル a に対応した対称的な度合い $D_5(i)$ である．すべてのグルーピング選択ルールが適用された後, 次のレベルのグルーピング構造がグルーピングレベル b のようになった場合, 対称的な度合い $D_5(i)$ は図 7(b) のようになる．

標準偏差 σ は, 0 に近いほど正規分布の裾野が狭まり, グループの中心付近での $D_5(i)$ の値が大きくなる．したがって, より上位のグループの中心に近いほど下位のグループの境界になりやすくなる．逆に σ が大きくなった場合, 上位グループの中心付近以外の位置でも下位のグループの境界になりやすくなる．

$$D_5(i) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left\{ -\frac{(\tau_i - \tau_{mid})^2}{2\sigma^2} \right\} \quad (14)$$

ただし,

$$\tau_{mid} = \frac{\varepsilon_{end} - \tau_{start}}{2}$$

$start$: 対象としているグルーピングレベルにおけるグループの始まりの音

end : 対象としているグルーピングレベルにおけるグループの終わりの音

$start$ と end は上位のグループが分割され新たな下位のグループが生成されるごとに更新され $D_5(i)$ の値が計算される．

システムの調節可能なパラメータとしては, σ を直接

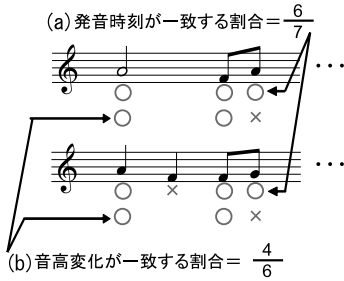


図 8 並列性のある音符列の例
Fig. 8 Similarity of parallel phrases.

与えるのではなく $\hat{\sigma}$ を与える ($\hat{\sigma} \times (\varepsilon_{end} - \tau_{start}) = \sigma$)。したがって、グループが下位レベルになるに従って、正規分布の裾野は狭くなる。

3.1.4 GPR6 (メロディの並列性) の適用

GPR6 は、並列性 (旋律の同一性, 類似性) に関するルールであり、旋律中の複数の部分間に並列性があるとき、それら部分のグルーピング構造も並列 (同一あるいは類似) になることを主張する。旋律中のある部分に並列性が認められると、その区間の始端と終端はグループ境界になる可能性が高くなる。本項では、音符 i が並列な区間の始端あるいは終端となる程度を表すパラメータ $D_6(i)$ ($0 \leq D_6(i) \leq 1$) を定義する。

$D_6(i)$ は以下のように計算される。まず、音符 i から始まる長さ r の区間と音符 j から始まる同じ長さの区間の類似度を求める。次に、音符 i に対し楽曲中のすべての音符 j とすべての長さ r に対する類似度を積算する。GTTM では、楽譜上に記述された 2 つの旋律間の類似度について特に記述がないので、我々は $D_6(i)$ を計算するための簡単な旋律間類似度を新たに定義する。旋律間の類似度を計算するとき、パラメータ W_m により各音の発音時刻の類似度と音高差の類似度のいずれを重視するかを調整する。さらに、並列な区間が重複している場合はより長い区間の始端と終端を優先し、パラメータ W_l によりその優先度を調整する。そして、類似度を積算するとき、パラメータ W_s により音符 i が並列な区間の始端あるいは終端のいずれかになる方に重みを置くかを調整する。これらパラメータ W_m, W_l, W_s の値域はいずれも 0 から 1 までである。ここで用いた旋律間類似度は文献 21) など述べてられている他の方法で代用することも可能である。

まず、図 8 のような 2 つの音符列間の類似性を考える。図中、発音時刻の一致する音符は 7 音中 6 音なので、発音時刻が一致する割合は $6/7$ である。さらに、発音時刻が一致している音のうち音高差が一致す

る割合は $4/6$ である。exGTTM では、同程度の長さの音符列があるとき、発音時刻が一致する音符が多いほど類似性は高く、その発音時刻が一致する音符に関し、隣接する音符間の音高差が同じものが多いほど類似性は高いと判定する。

上のような類似性を定式化するための準備を行う。まず、並列な区間の始端と終端はいずれかの拍位置に等しいと仮定し、並列な区間の長さ r は 1 拍の整数倍とする。また exGTTM ではある長さ未満の幅で平行移動した音符列どうしには並列性は存在しないと仮定し、その長さを 4 分音符 1 拍分とする。楽曲先頭からの拍番号 m (≥ 1) が与えられたとき、拍番号 m の位置から r 拍分の区間を $[m, m+r)$ と書く ($m+r$ 番目の拍は含まない)。

音符列間の類似度を計算するための基本的なパラメータ N, O, P を導入する。

$$N(m) = \text{区間 } [m, m+1) \text{ に存在する音符数}$$

$$O(m, n) = \text{区間 } [m, m+1) \text{ と } [n, n+1) \text{ の音符のうち、発音時刻が一致する音符数}$$

r 拍長の区間における音符数と発音時刻が一致する音符数はこれらの総和になる：

$$N(m, r) = \sum_{j=0}^{r-1} N(m+j)$$

$$O(m, n, r) = \sum_{j=0}^{r-1} O(m+j, n+j)$$

さらに、区間 $[m, m+r), [n, n+r)$ において発音時刻が一致する音符に関し、隣接する音符間で一致する音高差の個数を $P(m, n, r)$ とする。これより区間 $[m, m+r), [n, n+r)$ の類似度は以下のように定義できる：

$$G(m, n, r) = \left\{ \frac{O(m, n, r)}{N(m, r) + N(n, r)} \times (1 - W_m) + \frac{P(m, n, r)}{O(m, n, r)} \times W_m \right\} \times r^{W_l} \quad (15)$$

ただし曲全体の拍数を L として、 $G(m, n, r)$ の定義域は $1 \leq m, n \leq L - r + 1, 1 \leq r \leq L$ それ以外は $G(m, n, r) = 0$ とする。項 r^{W_l} は、 W_l が 0 のときにつねに 1 となり、 W_l が正 (> 0) のときに r とともに増大する。よって、 W_l を 0 から 1 に変化させることで、より長い区間の類似度ほど増大させることができる。

ここで、類似度を算出しても意味のない区間があることに留意されたい。上で、並列な区間の始端と終端はいずれかの拍位置に等しいと仮定したので、たとえば、拍の先頭でない音符から始まる音符列は、他の音

符列との類似度を算出しても無意味である．そこで，音符 i が並列な区間の始端や終端になれるか否かを判定する述語を導入する．音符 i が並列な区間の始端となりうる場合に真となる述語を b ，終端になりうる場合を e ，始端にも終端にもなりうる場合を t とする．関数 $head(m)$ は区間 $[m, m + 1)$ における最先頭の音符 i (i 番めの音) を返し，関数 $tail(m)$ は区間 $[m, m + 1)$ における最後尾の音符を返す．関数 $beat(i)$ は音符 i が区間 $[m, m + 1)$ 内に現れるときに m を返す．

$$b(i) \equiv (i = head(beat(i)) \wedge i \neq tail(beat(i)))$$

$$e(i) \equiv (i \neq head(beat(i)) \wedge i = tail(beat(i)))$$

$$t(i) \equiv (i = head(beat(i)) \wedge i = tail(beat(i)))$$

次に，音符 i から始まる区間に関する類似度の積算を行う：

$$A(i) = \begin{cases} G(beat(i), n, r) \times (1 - W_s) & \text{if } b(i) \text{ holds and } N(n) \geq 1 \\ G(beat(i) - r, n - r, r) \times W_s & \text{if } e(i) \text{ holds and } N(n) \geq 1 \\ G(beat(i), n, r) \times (1 - W_s) + G(beat(i) - r, n - r, r) \times W_s & \text{if } t(i) \text{ holds and } N(n) \geq 1 \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

並列な区間は重複しないので，2 つの区間の類似度を比較する際の区間長の最大は曲全体の拍数 L の $1/2$ とした．最後に $A(i)$ を正規化する．楽曲に含まれる $N(1, L)$ 個すべての音符に対する最大値を $A_{max} = \max(A(1), A(2), \dots, A(N(1, L)))$ とする：

$$D_6(i) = A(i) / A_{max} \quad (17)$$

ここで， $D_6(i)$ の値は曲全体の長さには無関係であり，音符 i を始端あるいは終端とする並列な音符列が楽曲中に繰り返し現れるほど $D_6(i)$ の値は大きくなる．

図 9 に実際の計算例を示す．最上段のグラフは各音符が並列区間の始端になる程度を表しており，中段のグラフは同様に終端になる程度を表している．両者を組み合わせて正規化を施したものが最下段のグラフであり，音符列の並列性から算出したグループ境界になる強さを表している．

3.1.5 GPR1 の適用

GPR1 (単音の非グループ化) では，非常に小さいグループ，特に単音からなるグループへの分割は避ける．したがって，グループの境界となるためには，局所的な境界の強さが前後の遷移より強い必要がある．GPR1 による境界の強さは，境界になりうる ($D_1(i) = 1$)

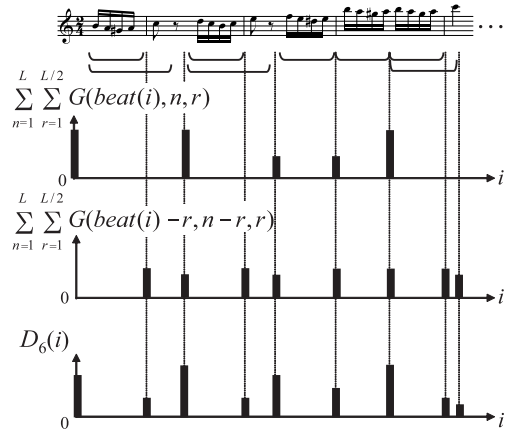


図 9 並列性の度合い $D_6(i)$
Fig. 9 Degree of parallelism $D_6(i)$.

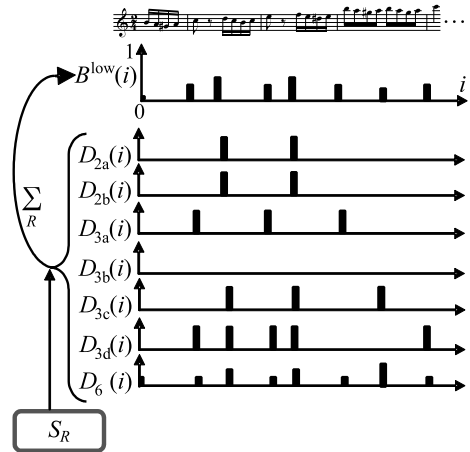
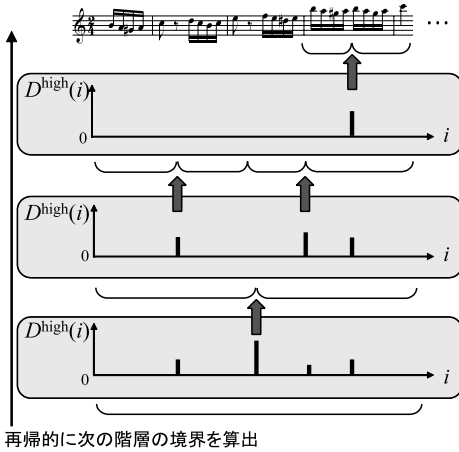


図 10 局所的なグルーピング境界の強さ $B^{low}(i)$
Fig. 10 Low-level strength of boundary $B^{low}(i)$.

かそうでない ($D_1(i) = 0$) かで表される (図 10)． $B^{low}(i)$ は，局所的な境界の強さを 0 から 1 の実数で表したものである． $B^{low}(i)$ の値が大きいくほど，局所的な境界が強いことを表す． $D_1(i)$ は，次式を用いて i を 1 から順に増加させながら算出していく．

$$D_1(i) = \begin{cases} 1 & \text{if } B^{low}(i-1) \leq B^{low}(i), \\ & B^{low}(i) \geq B^{low}(i+1), \\ & \text{and } D_1(i-1) = 0 \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

ただし，



再帰的に次の階層の境界を算出

図 11 階層的なグルーピング構造の生成

Fig. 11 Construction of hierarchical grouping structure.

$$B^{\text{low}}(i) = \frac{\sum_R D_R(i) \times S_R}{\max_{i'} \left(\sum_R D_R(i') \times S_R \right)} \quad (19)$$

ただし, $R = (2a, 2b, 3a, 3b, 3c, 3d, 6)$.

3.2 局所的境界の検出

局所的な境界は, $D_1(i), D_{2a}(i), D_{2b}(i), D_{3a}(i), D_{3b}(i), D_{3c}(i), D_{3d}(i), D_6(i)$ 用いて検出する. T^{low} は, 遷移 i がグループの境界となる ($D^{\text{low}}(i) = 1$) かそうでない ($D^{\text{low}}(i) = 0$) かを決めるための閾値である. $B^{\text{low}}(i)$ は, 式 (18) と同じである.

$$D^{\text{low}}(i) = \begin{cases} 1 & \text{if } B^{\text{low}}(i) > T^{\text{low}} \text{ and} \\ & D_i(1) = 1 \\ 0 & \text{otherwise} \end{cases} \quad (20)$$

3.3 階層的なグルーピング構造の獲得

階層的なグルーピング構造は, ボトムアップ処理により求めた局所的境界 $D^{\text{low}}(i)$ および, $D_1(i), D_{2a}(i), D_{2b}(i), D_{3a}(i), D_{3b}(i), D_{3c}(i), D_{3d}(i), D_4(i), D_5(i), D_6(i)$ を用いて, トップダウンに獲得する. $B^{\text{high}}(i)$ は, 高次の境界の強さを 0 から 1 の実数で表したものである. $B^{\text{high}}(i)$ と $B^{\text{low}}(i)$ との違いは, 大局的な構造に関する規則の適用結果である $D_4(i)$ および $D_5(i)$ が新たに加わる点である. グループがその内部に局所的境界を含んでいる場合, 次式によって, 次の階層の境界 \hat{i} が再帰的に求まる (図 11). $B^{\text{high}}(i)$ の値は, 各階層ごとに算出する.

$$\hat{i} = \underset{i}{\operatorname{argmax}} D^{\text{high}}(i) \quad (21)$$

ただし,

$$D^{\text{high}}(i) = D^{\text{low}}(i) \times B^{\text{high}}(i) \quad (22)$$

$$B^{\text{high}}(i) = \frac{\sum_R D_R(i) \times S_R}{\max_{i'} \left(\sum_R D_R(i') \times S_R \right)} \quad (23)$$

$$R \in \{2a, 2b, 3a, 3b, 3c, 3d, 4, 5, 6\}$$

i : グループに含まれるすべての遷移

(パラメータ計算の流れについては図 5 参照のこと)

4. 評価実験

システムがある音楽家のグルーピング結果と同じ分析結果をどの程度出力できるか適合率 (精度) P ($0 < P < 1$) と再現率 R ($0 < R < 1$) で評価した. 適合率 P と再現率 R の算出はグループが所属する階層に関係なく行う.

$$\text{適合率 } P = \frac{\text{システムの出力したグループが正解データにも含まれている数}}{\text{システムの出力したグループの数}} \quad (24)$$

$$\text{再現率 } R = \frac{\text{正解データのグループがシステムの出力にも含まれている数}}{\text{正解データのグループの数}} \quad (25)$$

4.1 評価用データ

GTTM に基づき音楽家がグルーピングした結果のデータベースは公開されておらず, 新たに評価用データを作成した. 評価用データは, GTTM をよく理解している 1 人の音楽家がクラシック曲から切り出した 8 小節の長さの 100 個のメロディの楽譜データと, それを GTTM に基づき手作業でグルーピング構造分析した正解データからなる.

4.1.1 楽譜データ

100 個のメロディの楽譜データは, 楽譜作成ソフトウェア *Finale*²³⁾ を用いて手作業で入力し, MusicXML 出力用のプラグイン *Dolet* でエクスポートして作成したものである. 手作業で入力したのは, 音符, 休符, スラー, 強弱記号 (アクセント), アーティキュレーションである.

3.1.1 項で MusicXML から基本変数を算出する際に, 正規の発音時刻 τ_i や消音時刻 ε_i 音高 f_i は, MusicXML に陽に示されているのに対し, 実際の消音時刻 $\hat{\varepsilon}_i$ やベロシティ v_i は陽には示されていない. 本研究では, スラーの部分では音符の正規の長さ但实际上に

8 小節で区切りの悪い場合, 8 小節以上で区切りの良いところまでとした.

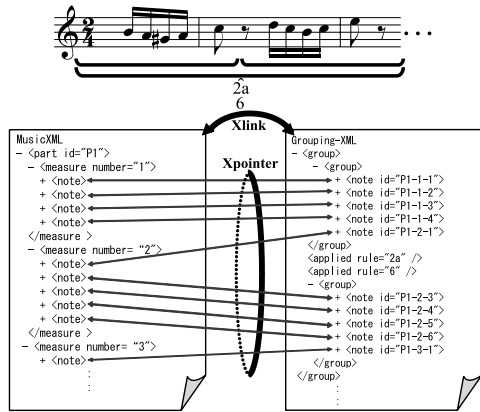


図 12 GroupingXML
Fig. 12 GroupingXML.

演奏された長さとのデュレーション比が 1 となり、それ以外の部分では 0.8 となるよう実際の消音時刻 ε_i を設定した。一方、ペロシティ v_i はアクセント記号のある部分で 1、それ以外の部分 0.8 となるように設定した。これらは、スラーやアクセント記号がある部分でルールが成立する（もしくは成立しないようにする）ための処理である。

4.1.2 正解データ

正解データは、1 人の音楽家の分析結果である階層的なグルーピング構造および、どの箇所でのルールが適用されたかを GroupingXML 形式で記録したものである。GroupingXML は、グループエレメント、ノートエレメント、アプライドエレメントからなる。ノートエレメントは、発音時刻順に並んでおり Xpointer²⁴⁾ と Xlink²⁵⁾ を用いて MusicXML 上の各音とリンクしている（図 12）。明らかに間違った解釈でないことを、3 人の GTTM の専門家がクロスチェックした。

4.2 パラメータの調節

グルーピング構造は、パラメータの調整によって変化する。1 人の実験担当者が 100 曲についてそれぞれ、パラメータの調節を行った。具体的には、楽譜およびその正解データを見ながら 1 曲 10 分で、システムの出力が正解データに近くなるようにパラメータを調節した。

実験中、正解データおよびシステムの出力した結果の参照は図 13 に示す GroupingXML ビューアで行った。GroupingXML ビューアは、音符をピアノロール形式で表示し、音符の下に階層的に並んだ円弧がグルーピング構造を表す。そして、グルーピング構造の

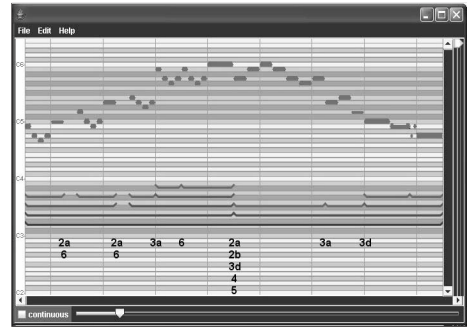


図 13 GroupingXML ビューア
Fig. 13 GroupingXML viewer.

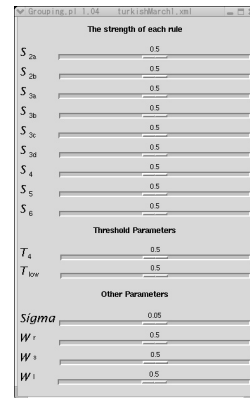


図 14 パラメータ調節のための GUI
Fig. 14 GUI for configuring parameters.

下には適用されたルールが表示される。パラメータの調節には、図 14 に示すグラフィカルユーザインタフェースを用いた。パラメータを変更することに新たなグルーピング結果が求まり、その結果が GroupingXML ビューアに反映されるようになっている。パラメータの初期値はそのパラメータの値域の中央値とした。 $S_R (R \in \{2a, 2b, 3a, 3b, 3c, 3d, 4, 5, 6\})$, W_m , W_s , W_l , T_4 , T^{low} の値域は、効果を消すための位置としての 0 からすべてのパラメータを均等に重み付けた位置を 1 とし、その間を 0.1 刻みとした。 $\hat{\sigma}$ については様々な値を試みたが、顕著に効果が現れた値として 0.01 から 0.10 までを 0.01 刻みとした。

調節の結果、適合率 P の平均値は 0.77、再現率 R の平均値は 0.79 であった。図 15 に、100 曲の適合率と再現率のヒストグラムを示す。適合率、再現率ともに右肩上がりの分布となり、適合率が 0.9 以上の曲は 51 曲、再現率が 0.9 以上の曲は 55 曲であった。

4.3 考察

本節では、分析結果をいくつか提示し、分析がうまくいった原因、もしくはうまくいかなかった原因につ

予備実験を繰り返した結果、十分な収束が得られる時間として 10 分を選択した。

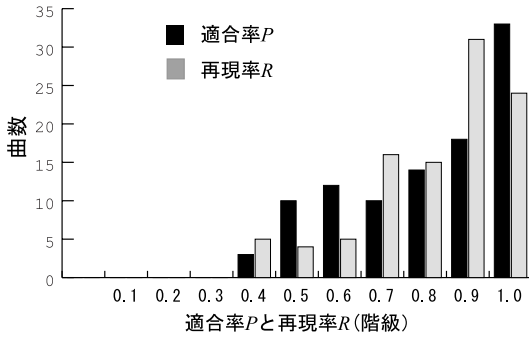
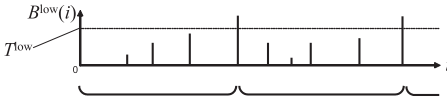


図 15 適合率 P と再現率 R のヒストグラム
Fig. 15 Histogram of precision and recall.



分析結果 (a) $S_{2a}=0.5, S_{2b}=0.5, S_{3a}=0.5$



分析結果 (b) $S_{2a}=0.3, S_{2b}=0.7, S_{3a}=0.3$

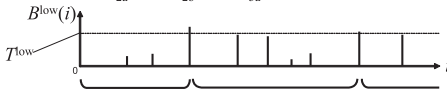


図 16 モーツァルトピアノソナタ K. 331 の分析例
Fig. 16 Analysis of Mozart Sonata K. 331.

いて考察する。

4.3.1 局所的境界の検出における問題

局所的境界の検出において選好ルール間の競合の問題が適切に解消できているかを検討する。ルール間競合を解消するパラメータは第 2 カテゴリーのパラメータであった (2.2.1 項)。ここで、適切に解消するというのは、我々のグルーピング構造分析器の分析結果が人間の分析結果全体を包含することを指す (1 章)。

GTTM では、楽曲に複数の解釈が考えられる例として、モーツァルトのピアノソナタ K. 331 (図 16) をあげている。この曲の冒頭部の解釈は、4 音-5 音間が境界となるようなグルーピングと、5 音-6 音間すなわち小節線が境界となるようなグルーピングの 2 通りが考えられる。システムを用いて分析した結果、 S_{2a} 、 S_{2b} 、 S_{3a} を調節することにより両方のグルーピングが実現できた。それ以外のパラメータの値は初期値である。適用されたルールは両方のグルーピング結果とも同じであった。この曲について局所的境界の検出が適切であった理由は、 S_{2a} 、 S_{2b} 、 S_{3a} を用いたルール間の優先度制御と並列性の度合いを表す $D_6(i)$ の値

が妥当であったためであろう。分析結果 (b) において、4 音-5 音間と 6 音-7 音間にはいずれも GPR2b のみが適応されているが、4 音-5 音間のほうが値が大きくなっているのは、この部分で $D_6(i)$ が大きな値となっているためである。

局所的境界の検出がうまくいかなかった例として、ショパンのバラード op. 23 (図 17) の冒頭部があげられる。この曲の特徴は、前半と後半で、曲調が大きく変わっていることである。図 17 の分析結果 (a) は、正解データに局所的境界が近くなるように、パラメータを調節した結果である。分析結果 (a) では、後半部分は正解に一致しているものの、前半部分は一致しない箇所が多い。そこで、図 17 の分析結果 (b) のように、前半部分がより正解データに一致するようにパラメータを調節すると、今度は曲の後半部分で一致しない箇所が増えてしまう。このような曲の場合には、曲の途中でパラメータを変化させる方法について検討する必要があると考えられる。途中で曲調が大きく変わる例はこの 1 曲だけであった。

局所的境界の検出がうまくいかなかったほかの例として、ビゼーのファランドール (図 18) があげられる。正解データの最後のグループが単音グループになっているが、今回実装した分析器では単音グループを禁ずる GPR1 が必ず成り立つような仕様としているため、適切にグルーピングすることができなかった。単音グループが正解データに含まれた曲は 2 曲であった。今後、単音グループを含む曲への対応について検討していく必要がある。

4.3.2 階層構造獲得における問題

本項では、階層的なグルーピング構造が適切に獲得できているかを検討する。図 19、図 20 は、100 曲の正解データとシステム出力に含まれるグループ数、グルーピング階層の数を比較したものである。グループ数とグルーピング階層の両方ともが、正解データに比べてシステムの出力のほうが値が大きくなる傾向があることが分かる。

システムの出力のほうが値が大きくなった原因の 1 つとして、今回の実装で、GPR5 が必ず成り立つような仕様としており、上位の階層のグループに含まれる下位階層のグループの数を 2 つに限定していることがあげられる。例として、チャイコフスキーのワルツ (図 21) を示す。正解データでは、最下位の階層の 3 つのグループ (第 1 音から第 3 音、第 4 音から第 6 音、第 7 音から第 9 音のグループ) がその上位の階層で 1 つのグループにまとめられているのに対し、システムの出力では、まず第 1 音から第 3 音、第 4 音から第

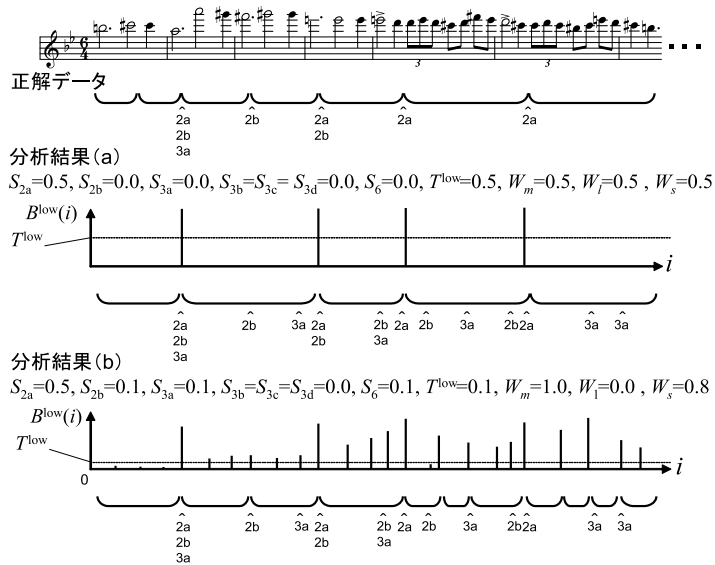


図 17 ショパンバラード op. 23 の分析例
Fig. 17 Analysis of Chopin, Ballade, op. 23.

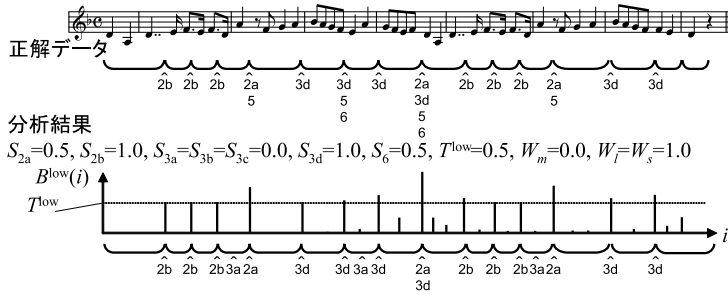


図 18 ビゼー「アルルの女」よりファランドールの分析例
Fig. 18 Analysis of Bizet, L'Arlesienne, Farandole.

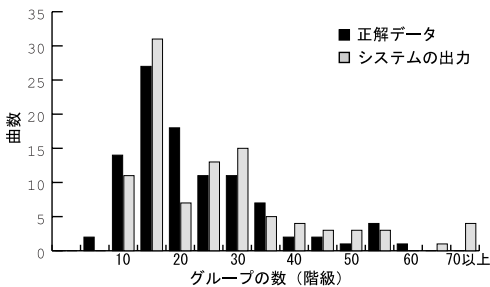


図 19 正解データと出力データのグループ数
Fig. 19 Numbers of groups in correct data and system outputs.

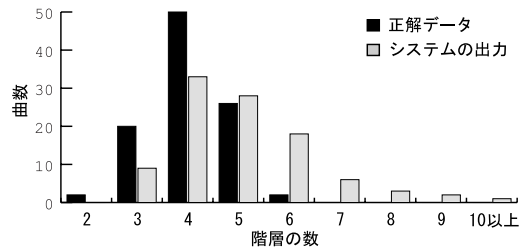


図 20 正解データと出力データのグルーピング階層の数
Fig. 20 Numbers of grouping hierarchies in correct data and system outputs.

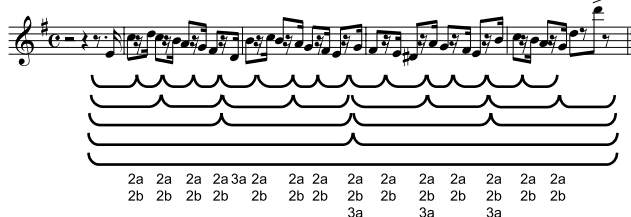
音のグループが1つのグループにまとめられ、その後第1音から第9音までが1つのグループにまとめられている。したがって、システムの出力のほうが、グループ数、グルーピング階層の数とも多くなっている。

る。ただし、システムの出力における第1音から第6音と第7音から第9音をまとめている階層を除けば、正解データと一致したグルーピング階層を生成している。今回の評価データでは、下位の3つ以上のグループが上位の1つのグループにまとめられている部分を



図 21 チャイコフスキー「子供のためのアルバム」よりワルツの分析例
 Fig. 21 Analysis of Tchaikovsky, Album pour enfants, Waltz.

分析結果(a) 適合率=1.0, 再現率=0.9
 $S_{2a}=S_{2b}=S_{3a}=1.0, S_{3b}=S_{3c}=S_{3d}=S_4=0.0, S_5=S_6=T_4=1.0, T_{low}=0.5, \sigma=0.05, W_m=W_l=W_s=0.5$



分析結果(b) 適合率=1.0, 再現率=1.0
 $S_{2a}=S_{2b}=S_{3a}=1.0, S_{3b}=S_{3c}=S_{3d}=S_4=0.0, S_5=S_6=T_4=1.0, T_{low}=0.5, \sigma=0.05, W_m=W_l=W_s=0.5$

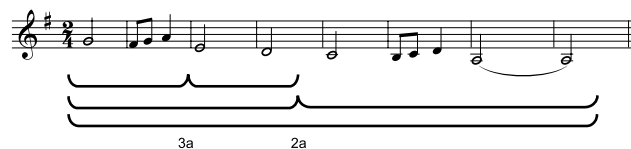


図 22 パラメータセットが同じであった 2 つの曲の分析結果 . (a) チャイコフスキー「くるみ割り人形」より行進曲 . (b) エルガー「威風堂々」第一番
 Fig. 22 Analysis of two songs which has same parameter sets. (a) Tchaikovsky, The Nutcracker, March. (b) Elgar, Pomp and Circumstance Marches, op. 39. No.1

含む曲は 58 曲あった . それら 58 曲での適合率の平均は 0.76 , 再現率の平均は 0.75 であった , 一方 58 曲を除く 42 曲での適合率の平均は 0.79 , 再現率の平均は 0.83 であった . 今後グループの分割が 3 以上の場合への対応についても検討していく必要がある .

4.3.3 パラメータの設定に関する考察

今回の実験では , パラメータの値は曲ごとに異なり , 統一的な傾向は認められなかった . すなわち , どの曲にも有効で最適なパラメータセットを見つけることはできなかった . このことは , グルーピングの重要パラメータは各曲の特性に応じて変化していることを表し , 調節後のパラメータの値が , 楽曲を分類する手がかりになる可能性があることを示唆している . 図 22 は , 調節後のパラメータの値が同じであった 2 曲 (チャイコフスキーの「くるみ割り人形」より行進曲とエルガー「威風堂々」第一番) の分析結果である . 両者を比較すると , 適用されているルールがほぼ一致していた .

特に , S_5 や S_6 が高い値となっており , 並列性や対称性が重要な要素となっている曲であることが分かる . 今後 , パラメータセットによるジャンル分け , 時代分け , 作曲家の特徴などの分類に利用できる可能性を検討することが考えられる .

5. まとめ

本研究では , exGTTM に基づきグルーピング構造を獲得するシステムについて述べた . 本研究の主な意義は以下の 3 点である .

- exGTTM の提案

計算機上で実行可能となるよう音楽理論 GTTM を拡張した exGTTM を提案した . GTTM の実装の困難さはこれまでたびたび指摘されてきたが⁽²⁶⁾ , その根本的な解決法は提案されてこなかった . 本研究では , ルールの定義の曖昧性と分析の曖昧性を分離するため , ルールを再形式化する際に ,

調節可能なパラメータを用いてルールを数式化した。

- グルーピング構造分析の計算機上への実装
実際に計算機上で動作するシステムを実現し、階層的なグルーピング構造の獲得を可能にした。システムは現在 Web 上で CGI アプリケーションとして公開しており、今後開発されるシステムとベンチマークすることが可能である。
- 正解データの作成と適合率と再現率による評価
正解データを作成し、実験により適合率、再現率を評価した。実験用に作成した 100 曲の正解データは、GTTM の分析結果のデータベースとしては最も多くの曲数を収めており、今後順次公開していく予定である。

今後、曲ごとに最適なパラメータの値を自動で設定する方法について考えていくとともに、今回実装できなかった GPR7、すなわち、高次の構造から低次の構造へのフィードバックに関するルールについて実装方法を検討する。また現在タイムスパン簡約の計算機上への実装を試みており、楽曲の表層的な構造を直接操作できない音楽初心者でもタイムスパン木を用いて自分の思いどおりに楽曲を操作できるようなシステムの構築を目指す。

参 考 文 献

- 1) Lerdahl, F. and Jackendoff, R.: *A Generative Theory of Tonal Music*, The MIT Press, Cambridge (1983).
- 2) Cooper, G. and Meyer, L.B.: *The Rhythmic Structure of Music*, The University of Chicago Press, Chicago (1960).
- 3) Narmour, E.: *The Analysis and Cognition of Basic Melodic Structure*, The University of Chicago Press, Chicago (1990).
- 4) Temperley, D.: *The Cognition of Basic Musical Structures*, The MIT Press, Cambridge (2001).
- 5) 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.1512-1526 (1992).
- 6) Hirata, K. and Aoyagi, T.: Computational Music Representation on the Generative Theory of Tonal Music and the Deductive Object-Oriented Database, *Computer Music Journal*, Vol.27, No.3, pp.73-89 (2003).
- 7) Todd, N.: A Model of Expressive Timing in Tonal Music, *Musical Perception*, Vol.3, No.1, pp.33-58 (1985).
- 8) Widmer, G.: Understanding and Learning Musical Expression, *Proc. ICMC1993*, pp.268-275 (1993).
- 9) Hirata, K. and Hiraga, R.: Ha-Hi-Hun plays Chopin's Etude, *Working Notes of IJCAI-03 Workshop on Methods for Automatic Music Performance and their Applications in a Public Rendering Contest*, pp.72-73 (2003).
- 10) 平田圭二, 松田 周: パピプーン: GTTM に基づく音楽要約システム, 情報処理学会研究報告 2002-MUS-46, pp.29-36 (2002).
- 11) Hirata, K. and Matsuda, S.: Interactive Music Summarization based on Generative Theory of Tonal Music, *Journal of New Music Research*, Vol.32, No.2, pp.165-177 (2003).
- 12) Stammen, D.R. and Pennycook, B.: Real-time Segmentation of Music using an Adaptation of Lerdahl and Jackendoff's Grouping Principles, *Proc. ICMPC1994*, pp.269-270 (1994).
- 13) Cambouropoulos, E.: The Local Boundary Detection Model (LBDM) and its application in the study of expressive timing, *Proc. ICMC2001*, pp.290-293 (2001).
- 14) Ferrand, M., Nelson, P. and Wiggins, G.: Memory and Melodic Density: A Model for Melody Segmentation, *Proc. XIV CIM 2003*, pp.95-98 (2003).
- 15) Hamanaka, M. and Hirata, K.: Applying Voronoi Diagrams in the Automatic Grouping of Polyphony, *Information Technology Letters*, Vol.1, No.1, pp.101-102 (2002).
- 16) Hamanaka, M., Hirata, K. and Tojo, S.: Automatic Generation of Grouping Structure based on the GTTM, *Proc. ICMC2004*, pp.141-144 (2004).
- 17) 浜中雅俊, 平田圭二, 東条 敏: ATTA: exGTTM に基づく自動タイムスパン木獲得システム, 情報処理学会研究報告 2005-MUS-61, pp.19-26 (2005).
- 18) Hamanaka, M., Hirata, K. and Tojo, S.: Automatic Generation of Metrical Structure based on the GTTM, *Proc. ICMC2005*, pp.53-56 (2005).
- 19) Hamanaka, M., Hirata, K. and Tojo, S.: ATTA: Automatic Time-span Tree Analyzer based on Extended GTTM, *Proc. ISMIR2005*, pp.358-365 (2005).
- 20) Nord, T.: Toward Theoretical Verification: Developing a Computer Model of Lerdahl and Jackendoff's Generative Theory of Tonal Music, Ph.D. Thesis, The University of Wisconsin, Madison (1992).
- 21) Hewlett, W.B. (Ed.): Melodic Similarity: Concepts, Procedures, and Application, *Computing*

in Musicology, Vol.11, The MIT press, Cambridge (1998).

- 22) Recordare L.L.C.: MusicXML 1.1 Tutorial (2006). <http://www.recordare.com/xml/musicxml-tutorial.pdf>
- 23) Finale: PG Music Inc. (2006). <http://www.pgmusic.com/finale.htm>
- 24) W3C, XML Pointer Language (XPointer) (2002). <http://www.w3.org/TR/xptr/>
- 25) W3C, XML Linking Language (XLink) Version 1.0 (2001). <http://www.w3.org/TR/xlink/>
- 26) Heikki, V.: Lerdahl and Jackendoff Revisited (2000). http://www.cc.jyu.fi/heivalko/articles/lehr_jack.htm

(平成 18 年 5 月 7 日受付)

(平成 18 年 10 月 3 日採録)



濱中 雅俊 (正会員)

2003 年筑波大学大学院工学研究科電子・情報工学専攻博士課程修了。2003～2004 年日本学術振興会特別研究員 PD。2004 年より現在まで科学技術振興機構さきがけ研究員 (専任) として独立行政法人産業技術総合研究所において音楽情報処理の研究に従事。2004～2005 年オランダ・ナイメヘン情報認知研究所 (NICI) 客員研究員。博士 (工学)。2001 年情報処理学会山下記念研究賞, 2001 年 SCI (5th World Multiconference on Systemics Cybernetics and Informatics) in Art 優秀論文賞, 2003 年筑波大学大学院優秀論文賞 (博士課程長賞), 2005 年 ICMC2005 Best Paper Award (Journal of New Music Research Distinguished Paper Award) 各賞受賞。



平田 圭二 (正会員)

1987 年東京大学大学院工学系研究科情報工学専門課程博士課程修了。工学博士。同年 NTT 基礎研究所入社。1990～1993 年 (財) 新世代コンピュータ技術開発機構 (ICOT)。平成 13 年度論文賞, 平成 15 年度山下記念研究賞。本会理事。音楽情報処理に興味を持つ。t-Room プロジェクトに取り組む。



東条 敏 (正会員)

1981 年東京大学工学部計数工学科卒業, 1983 年東京大学大学院工学系研究科修了。同年三菱総合研究所入社。1986～1988 年, 米国カーネギー・メロン大学機械翻訳センター客員研究員。1995 年北陸先端科学技術大学院大学情報科学研究科助教授, 2000 年同教授。1997～1998 年ドイツ・シュトゥットガルト大学客員研究員。博士 (工学)。自然言語の形式意味論, オーダーソート論理, マルチエージェントの研究に従事, その他人工知能一般に興味を持つ。情報処理学会, 人工知能学会, ソフトウェア科学会, 言語処理学会, 認知科学会, ACL, Folli 各会員。