

第4回

音楽理論 GTTM の定式化と実装の試み

平田圭二 (NTT コミュニケーション科学基礎研究所)

浜中雅俊 (筑波大学システム情報工学研究科)

音楽の構文解析器を作る

本連載第1回目では、計算論的音楽理論の主要な研究テーマの1つは音楽理論に基づいた分析手法や作曲手法を計算機上に実装する試みであると述べました。また本連載の第3回目では、自然言語理解において文というマクロな構造を理解する構文解析が重要な役割を果たし、その自動化が長年研究されてきたことを紹介しました。その自然言語処理との対比から、音楽を理解するためには、楽曲のマクロな構造に対する構文解析が重要な役割を果たすと述べ、その具体例として、音楽理論 GTTM¹⁾ のタイムスパン簡約や HPSG によるカデンツ (終止感をもたらす和音進行) 分析法を紹介しました。

もしこれら「音楽の構文解析」が自動化できたら、計算機は音楽をより深く理解することができるようになり、より知的な作曲、編曲、演奏や、質の高いマルチメディア作品制作などが実現するでしょう。さらに、工学的に有用であるばかりでなく、音楽理論を検証可能な形に改良し、音楽そのものの理解を深めることにもつながるでしょう。

今回は、GTTM を計算機上に実装する筆者らの試みについて、その課題と解決法をご紹介します^{2), 3)}。GTTM 全体を扱うと紙面が足りなくなってしまうので、本稿ではグルーピング構造分析の部分に着目して筆を進めていきたいと思います。

● 音楽分析における 2 種類の曖昧性

非常に当たり前ですが、音楽分析をする人は音楽分析の本を読んで内容を理解して、音楽分析を実行します。音楽分析とは、音楽の 3 要素と呼ばれる旋律、リズム (拍節)、和音のそれぞれに関して、区切りや繰り返しの個所を見いだしてグルーピングを行い、階層的にグループのグループ (グループ階層構造) を作り、特徴的な個所どうしを関連

付けてネットワークを構成することを言います。

英語で書かれた GTTM 本は音楽分析者が読んで理解するものです。音楽分析者に、グルーピングや階層の作り方に関する宣言的^{☆1}で部分的な情報さえ与えておけば、音楽分析の厳密な手順などを示さずとも、音楽分析者は試行錯誤しながらさまざまなルールを上手に適用して、グループ階層構造を作り出してしまいます。宣言的で部分的な情報しか与えられていないということは、グループ階層構造の具体的な作成手順が書かれていないということであり、計算機プログラムに翻訳しようとする何らかの情報が足りないということです。

ここで、音楽分析には 2 種類の曖昧性が生じています。

- 音楽分析手法や概念を記述するときに生じる曖昧性。
- 音楽分析を実行する人の主観的な判断や考え方に起因する曖昧性。

前者の曖昧性は、音楽理論 GTTM の場合ですと、GTTM のルール定義において何らかのパラメータやアルゴリズムが欠けていたり不完全だったりすることです。たとえば以下のルールをご覧ください。

GPR3c (Articulation) : Consider a sequence of four notes n_1, n_2, n_3, n_4 . All else being equal, the transition $n_2 - n_3$ may be heard as a group boundary if the transition $n_2 - n_3$ involves a change in articulation and $n_1 - n_2$ and $n_3 - n_4$ do not.

(筆者訳: 音列 n_1, n_2, n_3, n_4 について、すべての音が等しい場合を除き、遷移 $n_2 - n_3$ でアーティキュレーションが変化し、 $n_1 - n_2$ と $n_3 - n_4$ で変化しないとき、 $n_2 - n_3$ はグループ境界に聞こえることがある)

このルールは、ある条件が満たされると旋律中のある個所がグループ境界の候補になることを主張しています。ルール中の音楽用語「アーティキュレーション」というのは、演奏に感情を込めるため、ある音と後続の音をどのように分離したり・つなげたりするかという演奏法を意味します。短く分離する staccato やスムーズにつなげる legato 演奏記号が、その一例です。しかし、譜面上の legato がすべて同じタイミングや音量になっているとは限りませんし、legato の意味は演奏する楽器ご

☆1 ある対象に関して、その満たすべき性質や条件だけを記述する方法論のこと。対する概念は「手続的」と呼ばれ、その性質を実現したり条件を満たすための具体的な手順を記述する方法論です。

とに異なっています。また、ルール中の“may be”も、どのように解釈していいのかがよく分かりません。

音楽分析における2つ目の曖昧性に関して、たとえば音楽分析者の嗜好や属する文化、時代、地域が異なる場合、同じルールに従って分析したつもりでも、音楽分析者によって異なる分析結果が出てくることがあります⁴⁾。音楽分析者も人間ですから、分析結果がその時々的心情にまったく左右されないとも限りません。

前者と後者の曖昧性の境界にはトレードオフがあります。もし、音楽分析者の主観や考え方に依存する部分をできるだけ排除した厳密な理論を提案しようとする、その理論の適用範囲を狭めてしまうというジレンマをもたらします。しかし、音楽というものを対象としている以上、どうしても文章化できない部分が残ります。音楽分析者に依存する部分が生じてしまうのではないのでしょうか。数学のように厳密な体系を構築する方法論は、ここではあまり適切ではないと思います。たとえそのために、少々、不正確な定義、循環定義、用語使用の不統一などが紛れ込んでしまったとしても、それは音楽を音楽たらしめている何ものかの表出かもしれません。

● 音楽の構文解析器

筆者らは音楽の構文解析器を作るにあたり、これら2種類の曖昧性に対し以下のような設計方針で臨みました(図-1)。まず、音楽分析の記述に由来する曖昧性に対しては、不足しているパラメータやアルゴリズムを可能な限り補充し明示化し、それらの値や動作を音楽分析者から制御可能とすることを目指しました。これは、知的な音楽システムを実現するには、最初から楽譜上に明示的に記述されている情報だけでなく、暗黙的な背景知識も同様に明示化することが重要であるという主張に通じるものがあります⁵⁾。パラメータが明示化されれば、その次は、音楽分析を音楽の構文解析器のパラメータ値を最適化したり制御する問題として定式化できます。

2番目の音楽分析者の主観や考え方に由来する曖昧性に対して、人間の生み出した分析結果はそのいずれもが“正しい”と認めることとします。よって、音楽の構造分析器に適切なパラメータ値を与えさえすれば、構造分析器は各々の正しい分析結果を出力するということを目指しました(高い再現率)。つまり、そのような動作が可能となるように不足しているパラメータを補足し、アルゴリズムを設計するということです。本連載の第1回目でも触れましたが、GTTMの分析手順では、ある楽曲が与えられると、まずwell-formedな音楽構造を規定し、その中からより好ましい音楽構造を選択します。複数の正しい分析結果が存在すると考えるのはwell-formedな音楽構造が複数通り存在することに対応し、

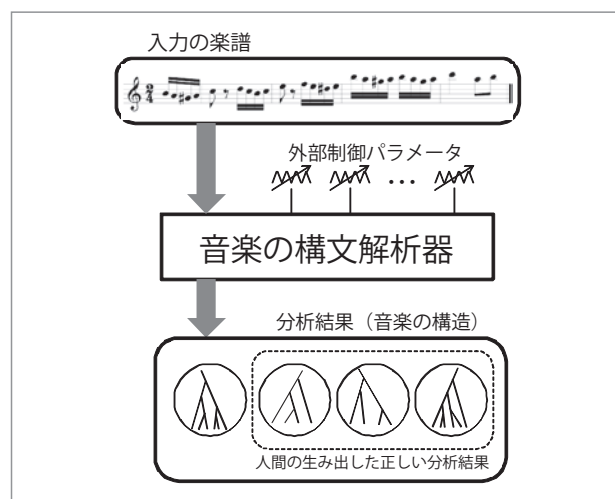


図-1 音楽の構文解析器

構文解析器に適切なパラメータ値を与えることはより好ましい音楽構造を選択する尺度や閾値を変更することに対応します。

筆者らは、音楽理論 GTTM の分析手法を計算機上に実装するため、上述の方針に従って GTTM を拡張しました^{2), 3)}。それを exGTTM と呼びます。では次に、具体的に GTTM ルールのどこにどんな曖昧性があるのかを見て、それから exGTTM に至る道のりの一端をご紹介します。

GTTM ルールの曖昧性

GTTM ルール記述に含まれる曖昧性は、さらに大きく次の3種類に分けられます。

- ルール間競合の解消が曖昧。
- ルール適用の範囲(時間区間)や適用タイミングが曖昧。
- 定式化に必要な概念やルール定義が欠けている。

では、1つずつ説明していきましょう。

● ルール間競合の解消が曖昧なケース

まず、グループ境界を決めるためのルールを示します:

GPR4: Where the effects picked out by GPRs 2 and 3 are relatively more pronounced, a large-level group boundary may be placed.

(GPR2, 3の示す効果が相対的に顕著な場所は、より上位レベルでもグループ境界となる可能性がある)

ここで GPR というのは Grouping Preference Rule、つまり、グルーピング構造分析において、well-formedな音楽構造を複数規定したあと、その中からより好ましい音楽構造を選択するための選好ルールのことです。GTTM には GPR1 から7まであって、GPR2, 3は、連続した4音に関して、音高やタイミングの近接性や変

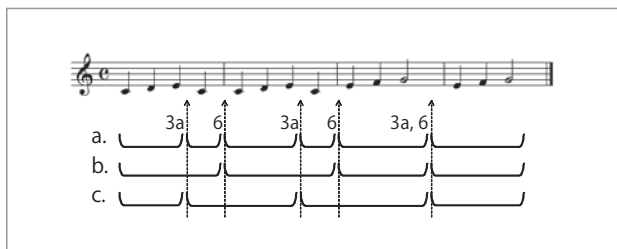



図-2 ルール間競合解消が曖昧な例



図-3 グループ階層構造の生成順序

化に基づいてグループ境界の可能性を指摘する局所的なルール群です（前掲のGPR3cはその1つです）。これらに対して、GPR4はメタルールの役割を担っています。このGPR4の記述中のrelativelyやmay beをどのように解釈すべきかは、音楽分析者の主観や考え方に委ねられています。

もう1つグループ境界を決めるためのルールを示します（図-2）：図中、「3a」と「6」は、各々GPR3a（音高の跳躍するところはグループ境界になりやすい）とGPR6（並列な旋律断片の始端と終端はグループ境界になりやすい。詳細は後述）というルールによってグループ境界の可能性を指摘された場所を表しています。また、が1つのグループを表しています。これだけですと図中aのようにグループ境界が5カ所生じそうです。ところがGTTMにはGPR1というルールもあります：

GPR1 : Avoid analyses with very small groups
-the smaller, the less preferable

（非常に小さいグループを含む分析は避ける—小さいグループほど望ましくない）

このGPR1より、図-2の3aか6は排他的に成立することが強く要請されます。つまりルール間競合を解消し図中bかcのいずれかのグルーピングに決めなくてはなりません。しかし、GPR1にはルール間競合解消に関する情報は含まれていません。

実際に競合を起こす可能性があるルールはもっと多数あります。GPR2は2つのサブルールからなり、各々、連続した4音に関して、スラーと休符（2a）および発音時刻（2b）の近接性からグループ境界の可能性を指摘します。同じくGPR3は4つのサブルールからなり、各々、連続した4音に関して、音高（3a）、音量（3b）、アーティキュレーション（3c）、音価（3d）の変化からグループ境界の可能性を指摘します。これら各サブグループすべてがルール間競合を起こす可能性がありますし、後述するようにexGTTMではGPR6も競合を起こす可能性があります。

これら多数の要因から局所的なグループ境界の可能性が決まってくるので、多数のルールが競合を起こすのです。しかし、どの要因が強くてどの要因が弱いのか

かは一概に決められません。GTTM本でもLerdahlとJackendoffは、各ルールの重みにはあえて触れないと述べています。

● ルールの適用範囲やタイミングが曖昧なケース

グルーピングに関するルールは、局所的なもの（GPR1, 2, 3, 4）と大域的なもの（GPR5, 6）に分けられます^{☆2}。これらルールから得られる各グループ境界の可能性に基づいて、全体として一貫したグループ階層構造を作り出さねばなりません（図-3）。このとき、GPR1, 2, 3, 4を用いて、局所的に弱いグルーピング境界を発見し、そこに接しているグループどうしをボトムアップに合体させます。一方、GPR5, 6を用いて、トップダウンにグループを分割したり、並行な旋律断片を発見してその始端と終端をグループ境界候補とします。音楽分析者はボトムアップとトップダウンの間を適当に行き来しつつグループ階層構造を作り出していきます。

この大域的なルールの1つであるGPR5は次のように与えられています：

GPR5 : Prefer grouping analyses that most closely approach the ideal subdivision of groups into two parts of equal length

（あるグループをできるだけ等しい長さに二分するようなグルーピングが好ましい）

局所的なルール群は、範囲や適用タイミングを考慮せずに適用することができますが、GPR5, 6といった大域的なルールは、そうはいきません。GPR5を楽曲のどの範囲にどのタイミングで適用するかが、生成されるグループ階層構造を大きく左右することは容易に想像できると思います。GTTMでは、GPR1～6全部を同時に考慮して最も妥当なグループ階層構造を作るためのルール（アルゴリズム）が与えられていません。

前述したように、GPR5に含まれる「できるだけ等しい長さ」や「好ましい」という表現の曖昧さにももちろん対処しなければなりません。

^{☆2} GPR7は後段からのフィードバックに関するルールです。

● 必要な概念やルール定義が欠けているケース

同一あるいは類似した断片部分を含む旋律のグルーピングに関するルール GPR6 を挙げます：

GPR6 : Where two or more segments of the music can be construed as parallel, they preferably form parallel parts of groups

(2つ以上の旋律断片が平行であれば、その部分ではグルーピング構造も平行となるのが好ましい)

しかし GTTM では平行 (parallel^{☆3}) の定義、つまり旋律間の類似性の定義が与えられていません。実際に、類似性に関する提案や実験は多数にのびります^{7), 8)}、音楽認知科学的にも未解明な部分があります。理論家の立場からは、定義をどれか1つに固定してしまうのはあまり現実的ではないでしょう。

ここで問題のそもそものに立ち戻ってみると、旋律断片が平行であるとは、その音楽的な構造が同一あるいは類似しているから平行と認識されるのではないのでしょうか。本連載第2回目でも指摘しましたが、認知科学的な視点を取り入れて、このような循環定義に陥らない妥当な理論を構築する研究も必要であると思います。

もう1つのケースを挙げたいと思います。GTTM 分析ではグルーピング構造分析と拍節構造分析の結果を受けて、タイムスパン簡約を行いタイムスパン木を得ます。さらに連載第2回目で触れた延長的簡約を行い、ある木構造 (Normative Prolongational Structure, 規範的な延長的構造) を得て分析全体が完結します。ここで、GPR7 はタイムスパン木と規範的な延長的構造がより安定^{☆4} するようなグループ階層構造を要請します。同様に、拍節構造分析のルール (MPR9) も後段の処理結果が安定することを要請します。このように、後段の処理結果からのフィードバックがあると一般に解が収束しなくなる可能性が生じます。このフィードバックの問題も、認知科学的な視点を取り入れて理論の枠組み自体から再構築して解消する道があるかもしれません。

GTTM から exGTTM へ

ここまで、グループ構造分析ルールの曖昧性のみに触れてきましたが、拍節構造分析やタイムスパン簡約にお

☆3 音楽用語としての parallel の意味は、2声が完全5度あるいはオクターブで並進すること、それらは一般に平行5度、平行8度と呼ばれています⁶⁾。しかし GTTM では、同発音時刻に限らず、同一あるいは類似した旋律の進行が出現しているときに parallel と呼びます。

☆4 GTTM 本では、安定したタイムスパン木の定義、安定した規範的な延長的構造の定義も曖昧です。

いても同様の曖昧性が含まれています。GTTM を計算機上に実装するには、それらにも対処しなければなりません。筆者らは GTTM を規範的に見る立場から、以下の点について GTTM を拡張し定式化を行いました。

- 不足していたり暗黙的なパラメータの明示化、
- 音楽構造を生成するアルゴリズムの導入。

そうやって定式化した GTTM を exGTTM と呼びましょう。GTTM を規範的に見る立場というのは、これまで見てきたように GTTM ルールには曖昧な個所があるものの、基本的な概念や規則は取り込まれており、全体としては正しく意味のある理論であるという意味です。ですから、GTTM ルールが十分かつ的確かという問題は棚上げし、足りない部分を補うというアプローチで定式化しています。また、ここでは計算機上で動作することを優先して、exGTTM が与えるアルゴリズムと人間の認知過程との対応はあまり考慮していません。

では以下、引き続きグルーピングのルールを取り上げて各ルールをどのように定式化し実装したかを具体的に説明していきたいと思います。

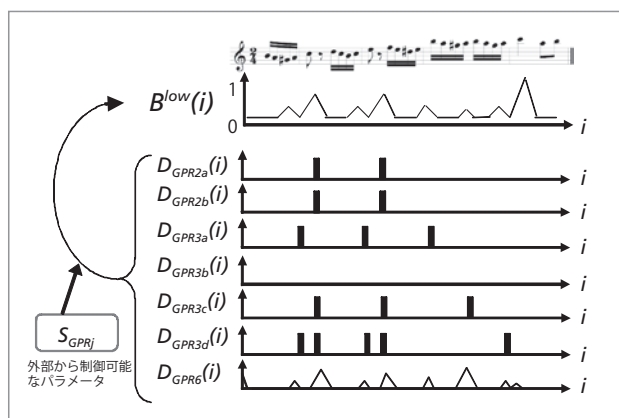
● ルール間競合の解消

先に挙げた例では GPR1, 3a, 6 が競合していました (図-2)。このようなとき、各ルールを重み付けするパラメータを補って、競合を解消することを考えます。

そこでまず GTTM のルールごとに、そのルールが場所 i においてどの程度成立しているかを示す値域が $0 \sim 1$ の変数 $D_{\text{ルール名}}(i)$ を導入します。たとえば GPR2, 3 の場合は、ルールが成立するか否かの2通りに分かれるので、 $D_{\text{ルール名}}(i)$ の値は離散値 $0, 1$ を取ります。次に各 $D_{\text{ルール名}}(i)$ に対して重み付けのパラメータ $S_{\text{ルール名}}$ (ルール名 = GPR2a, 2b, 3a, 3b, 3c, 3d) を導入し、各サブルールの重みを外部から制御できるようにしました (図-4)。

各 $D_{\text{ルール名}}(i)$ を算出する際に、また別のパラメータが必要になる場合もあります。たとえば、 $D_{\text{GPR2x}}(i)$ や $D_{\text{GPR3x}}(i)$ は 0 か 1 の離散値を取るので、その値から $D_{\text{GPR4}}(i)$ (GPR2, 3 の示す効果がどの程度相対的に顕著か) を算出するわけにはいきません。そこで exGTTM では、 $D_{\text{GPR2x}}(i)$ と $D_{\text{GPR3x}}(i)$ を算出するために用いられた発音時刻の間隔、音の跳躍、休符の長さ、アーティキュレーション等の値を用いて「効果」の大きさを算出し、ある閾値を超えたかどうかで $D_{\text{GPR4}}(i)$ が成立するかどうかを判定しました。このときに導入された閾値もまた外部から制御できるパラメータの1つです。

このようにしてルールごとに導入した変数 $D_{\text{ルール名}}(i)$ とそのルールの重みを外部から制御するパラメータ $S_{\text{ルール名}}$ から、 i 番目の音符間遷移がグループ境界となる

図-4 GPR2, 3の各サブルールを制御するパラメータ $S_{\text{ルール名}}$

可能性を表現する変数 $B^{\text{low}}(i)$ が得られます：

$$B^{\text{low}}(i) = \sum_j S_{\text{GPR}j} \times D_{\text{GPR}j}(i)$$

$$j=2a, 2b, 3a, 3b, 3c, 3d, 6$$

ここで変数 $D_{\text{ルール名}}(i)$ の値域が 0 ~ 1 と正規化されていたことに留意してください。exGTTM では、上記 $B^{\text{low}}(i)$ の算出法に限らず、今後さまざまなルールをさまざまに組み合わせてグループ境界の算出方法を検討しようと考えています。そのとき、あらゆる組合せを試行錯誤するためには正規化が必要です。

● GPR6 の定式化と実装

先にも述べたように、GPR6 は、楽曲中の平行な旋律断片の始端、終端がグループ境界になる可能性があることを主張しているルールで、楽曲中に平行な旋律断片が現れているかどうかをくまなく探す操作を含む大域的なルールです。ところが、図-4の最下段に書いてあるように、exGTTM では GPR6 に関してある前処理を施すことで、局所ルールと同列に扱っています。そのポイントとなるのは、平行な旋律断片を探索する範囲です。大域的なルールから算出される値というのは、グループを分割して大域的な状況が変化するたびにその値を計算し直さなければなりません。これに対し exGTTM では、平行な旋律断片の探索範囲を常に楽曲全体とし、異なるグループに出現する旋律断片どうしても平行な場合があるという前提を置きました。これより、楽曲全体に対して最初に 1 度だけ平行な旋律断片を全探索しておけば、トップダウンにグループを分割していくたびに平行な旋律断片を探索し直す必要がなくなります。

このような前提を置いて GPR6 を定式化するということは、楽曲全体にわたって平行な旋律断片を探索し、その断片の始端、終端において値が高くなるような変数 $D_{\text{GPR6}}(i)$ のアルゴリズムを具体的に与えるということです（図-5）。このとき、 $D_{\text{GPR6}}(i)$ の値に影響を与

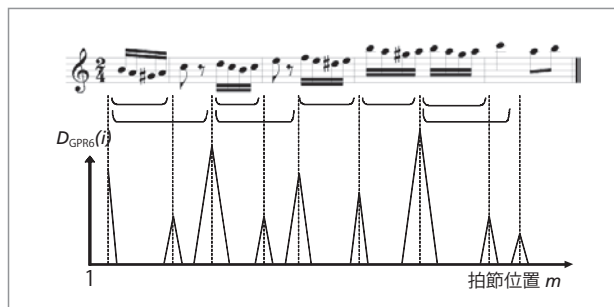
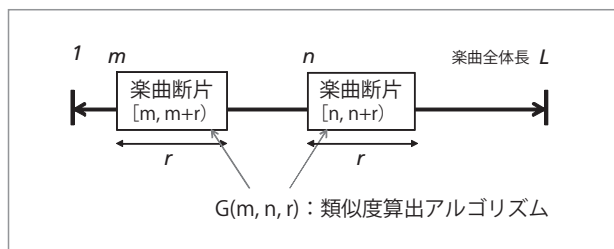
図-5 平行な旋律断片の始端、終端を指摘する変数 $D_{\text{GPR6}}(i)$ 

図-6 楽曲中の平行な旋律断片を発見するアルゴリズム

える要因として少なくとも以下の 3 点が考えられます：

1) 平行な旋律断片の長さ、2) 旋律断片が平行であることを判定する際に音高の一致に重みを置くか発音時刻の一致に重みを置くか、3) 旋律断片の始端に重みを置くか終端に重みを置くか。GTTM ではこれらの点については何も議論されていませんが、少なくともこれらの要因をパラメータとして明示してアルゴリズムに取り込む必要があります。

exGTTM では、この変数 $D_{\text{GPR6}}(i)$ の値を基本的に全探索で求めます（図-6）。まず、四分音符単位の拍位置 m から始まる長さ r 拍の旋律断片 $[m, m+r]$ ともう 1 つの旋律断片 $[n, n+r]$ の間の類似度を算出するアルゴリズムがあるとします（図中 $G(m, n, r)$ で表現）。楽曲全体にわたって、 m, n, r の全組合せについて類似度を算出して、各音符間遷移 i ごとに集計するのです。旋律断片の始端と終端を四分音符単位の拍位置に限定するのは、exGTTM では、四分音符より短い旋律断片は平行かどうか探索する必要がないと考えるからです。この類似度算出手法は、類似マトリクス法⁹⁾と同じ原理です。

では次に、上の要因 1)、2) を考慮した旋律断片どうしの類似度 $G(m, n, r)$ を算出するアルゴリズムを考えましょう（図-7）。図では、4 つの発音時刻のうち 3 つが一致しています ($A_{m, n, r}$)。さらにその 3 つの発音時刻において、2 つの音高が一致しています ($B_{m, n, r}$)。この発音時刻と音高が一致する音符が多ければ多いほど 2 つの旋律は類似していると考えられます。ここまです算出された $A_{m, n, r}$ と $B_{m, n, r}$ を組み合わせて旋律断片 $[m, m+r]$ と $[n, n+r]$ の間の類似度 $G(m, n, r)$ を定義しま

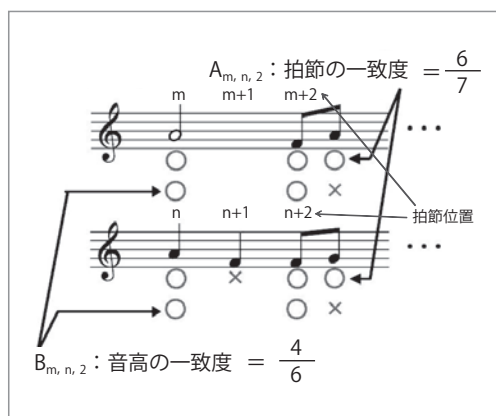


図-7 旋律断片どうしの類似度の算出

す。このとき、類似している旋律断片のうち、より長いものに重みを与えるパラメータ W_l と、音高列の概形 (contour) が似ていることに重みを与えるか発音時刻が似ていることに重みを与えるかを調整するパラメータ W_m を導入します。

最終的にこの $G(m, n, r)$ を使って図-5のように値が変化する変数を構成します (図-8)。exGTTM では、パラレルな旋律断片は四分音符で数えた拍位置から始まる／終わるとしているので、音符 i が旋律断片の始端や終端になり得るかを判定する必要があります。音符 i が旋律断片の始端になり得るとき $b(i)$ が成立し、 i が終端になり得るとき $e(i)$ が成立するとします。beat(i) は音符 i の出現する拍位置を表します。 m が与えられたときのすべての n, r を調べるのが $\sum_{n=m+1}^L \sum_{r=1}^{L-n}$ です (L は楽曲全体長) ^{☆5}。最後に、要因3)を考慮して、旋律断片の始端と終端のいずれがより強くグループ境界の可能性に影響するかを調整するパラメータ W_s を与えます。そして図-8 最下段に示したグラフのような $D_{GPR6}(i)$ を得ます。

ここまで長々と GPR6 の定式化とその実装法を説明してきました。その過程で、新しく3つのパラメータ W_m, W_l, W_s を導入しました。それは、本章の冒頭で述べたグループ境界の可能性に影響を与える3つの要因にそれぞれ対応しています。GTTM 本の中ではさまざまな関連事項が議論されていますが、いずれも GTTM 本ではまったく着目されていなかった要因ばかりです。前述のパラレルの探索範囲の問題も、GTTM の自動化という試みを通じて初めて提出された論点の一つだと思います。

将来もし他の優れた類似度アルゴリズムが提案された場合、exGTTM の類似度算出の部分は、その新しいア

☆5 音ただし m と n から始まる旋律断片どうしが重複した場合を除く処理は説明が煩雑になるので省略してあります。

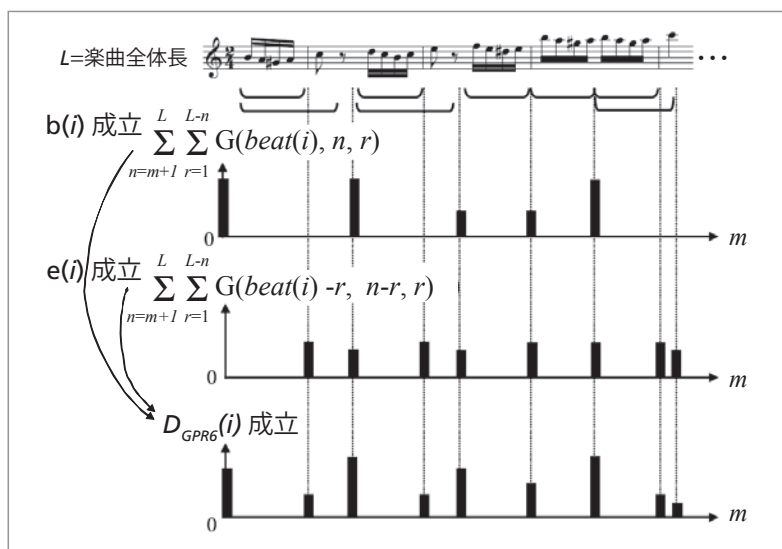


図-8 パラレルな旋律断片の始端あるいは終端がグループ境界になる可能性とその足し合わせ

ルゴリズムに置き換わることになるでしょう。そのときは本稿で紹介したパラメータとはまったく異なるものを使うことになるかもしれません。

● グループ階層構造生成アルゴリズム

何度か述べてきましたように、GTTM 本にはグループ階層構造を作るための部品となるルールは与えられているものの、それらを組み合わせるルールやアルゴリズムは記述されていません。しかし、音楽分析者はボトムアップとトップダウンの間を適当に行き来しつつ妥当なグループ階層構造を作り出していきます。拍節構造、タイムスパン木、規範的な延長的構造についても同様です。では exGTTM でのグループ階層構造を作り出すアルゴリズムについて見ていきましょう。

アルゴリズムの概略は、トップダウンにグループを分割するたびに、局所的なグループ境界の情報と大域的なグループ境界の情報を再計算するというものです (図-9)。ここで、大域的なルール GPR6 は、前の章で導入した前提「パラレルな旋律断片の探索範囲を常に楽曲全体とし」により局所的なルールとして扱えば十分となりました。よって大域的なルールとして考慮すべきは GPR5 だけです。exGTTM では以下のステップでグループ階層構造を生成します：

[ステップ1] 楽曲全体の音符間遷移 i ごとに局所的なグループ境界の強さを算出します。

ここでは GPR2, GPR3, GPR6 の効果を計算します。

[ステップ2] ある対象区間 $[s, e]$ に関して、大域的な情報も加味してグループ境界の強さを算出します (図-9 c)。

図中 c の $D^{high}(i)$ は、ステップ1で算出したグルー

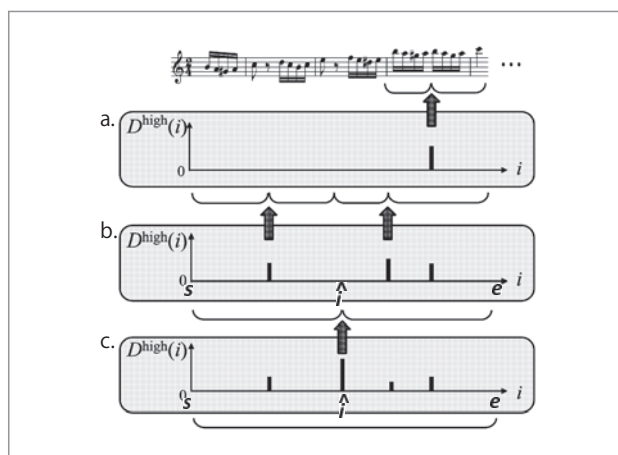


図-9 トップダウン処理とボトムアップ処理の統合

ブ境界の強さに、GPR1, GPR4, GPR5 の効果を加味したものです。4カ所のグループ境界の候補があります。

[ステップ3] $D^{high}(i)$ が最大となる場所 \hat{i} を選びグループを二分割します (図-9 b)。

対象区間 $[s, e]$ 中、最もグループ境界の可能性が高い i をグループ境界とします (図中の上向き矢印の位置, \hat{i} と記します)。

[ステップ4] 分割した対象区間ごとにステップ2, 3を繰り返します (図-9 a)。

$[s, e]$ を $[s, \hat{i}]$ と $[\hat{i}, e]$ に分割しステップ2に戻ります。

そして、対象区間に局所的グルーピング境界がなくなるまでステップ2～4を再帰的に繰り返します。

アルゴリズムの詳細については文献2)

をご覧ください。

このようなグループ階層構造を生成するアルゴリズムが GTTM で与えられていないのは、旋律断片の類似度を算出するアルゴリズムと同様に、音楽認知科学的に未解明部分があるということと、実際のアルゴリズムが何通りも考えられるためと思われます。筆者らが提案した exGTTM のグループ階層構造生成アルゴリズムもその1つの実現方法にすぎないと考えています。

本稿では紹介しませんでした、GPR5 についても同程度の拡張が必要です。さらに、拍節構造分析やタイムスパン簡約に関するルールについてもさまざまな拡張が必要です。それら GTTM ルールの定式化とその実装法の詳細についても文献2) をご覧ください。

exGTTM の実装

筆者らは exGTTM を計算機上に実装しました。そのシステムを ATTA (Automatic Time-span Tree Analyzer) と呼んでいます。exGTTM で定式化し実装したのはグルーピング構造、拍節構造、タイムスパン簡約の3つのサブ理論であり、それは GTTM 全26ルール中、17ルールでした。その際、全部で46個のパラメータを導入しました。

ATTA を起動すると、グルーピング構造分析、拍節構造分析、タイムスパン簡約ごとに、パラメータ値調節パネルと分析結果の簡易表示パネルが開かれます (図-10)。音楽分析者がパネル左半分のスライダを動かして、ATTA が正しい結果を表示するようにパラメータ値を調節します。

それでも、どうしても ATTA が正しい結果を表示できない場合は、図-11 のようなエディタを起動して手動で正しい分析結果を入力し表示することができます (譜例はモーツァルトのトルコ行進曲です)。GUI ウィンドウの上半分がタイムスパン木の表示であり、下半分がグループ構造および拍節構造の表示です。

ATTA の評価実験について簡単に触れたいと思います。クラシック100曲から切り出した8小節の旋律断片を100個準備し、4人の GTTM の専門家が相互チェックをしながらそれらの正解の分析結果を作成しました。次に、100個の旋律断片の各々に対して ATTA が正解の分析結果を出力するように手動でパラメータ調整を行い

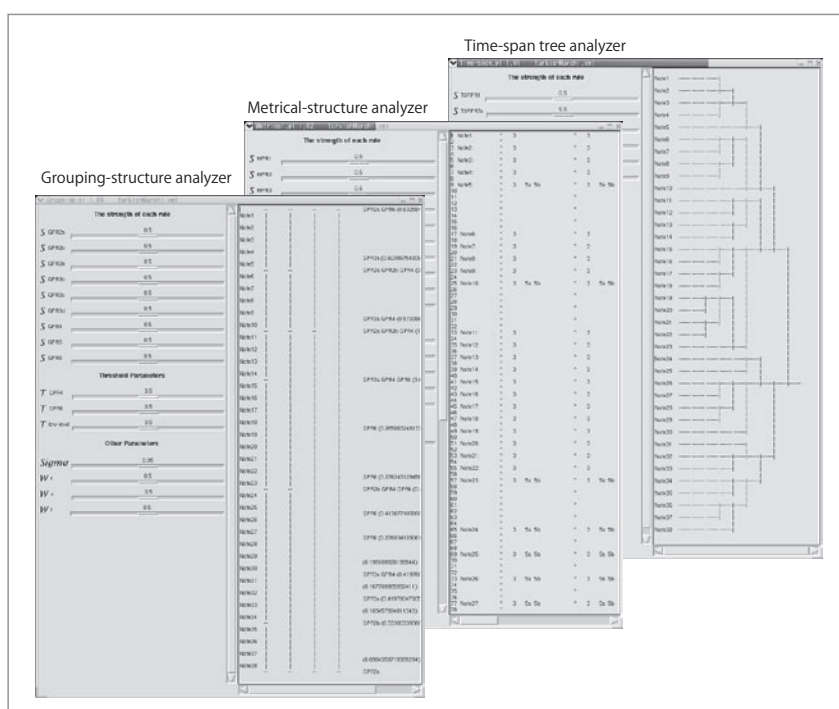


図-10 ATTAのパラメータ値調節パネルと分析結果表示パネル

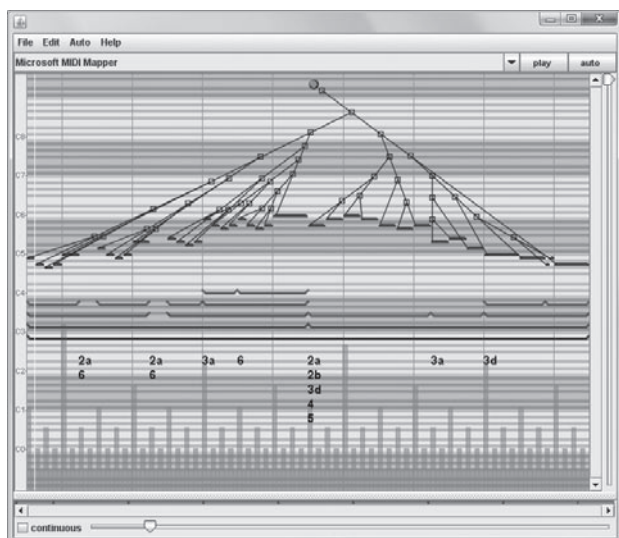


図-11 ATTAのGUI

ました。どんなにパラメータ値を調整しても正解が出力できないケースがいくつか残りましたが、それでも、部分的には正しい分析結果を含んでいました。グループ構造分析、拍節構造分析、タイムスパン簡約の各正解率のF値^{☆6}は全100曲の平均で0.77, 0.90, 0.60となりました。

むすびに代えて

最初に、音楽の構造解析器の設計目標として、不足しているパラメータやアルゴリズムを可能な限り補充し明示化し、それらの値や動作を音楽分析者から制御可能とすること、そして音楽の構造分析器に適切なパラメータ値を与えれば、構造分析器は各々の正しい分析結果を出力できることを掲げました。GTTMを規範的に見る立場からは、GTTMをよく理解する音楽学の専門家の協力を仰いで、exGTTMの分析結果が設計目標に近づくよう改良していくことになると思います。一方、現実には、従来では考えられなかったような新しい解釈を発見して、それに基づく素晴らしい演奏を次々と披露する演奏家があります。これは、従来の規範を越えて新たな規範を提案する作業です。そのような演奏に触れるにつけ、固定された規範を持たない音楽理論とはどのような理論なのか興味が湧くところです。

exGTTMは、GTTMのルールごとに定式化を行いました。その結果、グループ境界の可能性を算出するモジュール、旋律断片の類似度アルゴリズム、階層構造を生成するアルゴリズム等のモジュラリティの高い部品が

らexGTTM全体を構成することができました。今後はそれら部品ごとに、より性能の良いものを設計し交換していけば、exGTTMの効率の良い改良ができるのではないかと期待しています。

GTTM本は、音楽分析者が読んで理解し音楽分析を実践するための本です。厳密な手順などを示さずとも、グループ階層構造に関する宣言的で部分的な情報さえ与えておけば、音楽分析者は試行錯誤しながら、局所的なルールや大域的なルールグループ階層構造を作り出してしまいます。exGTTMの設計を通じて一番痛感したのが、音楽に関する人間知性の奥深さでした。先にGTTMの定式化においては人間の認知プロセスをあまり考慮しないと述べましたが、実際に暗黙的な処理を明示化するには内省によるところが大でした。

今回は触れなかったGPR7(フィードバックのルール)の定式化は、次回(本連載最終回)で紹介したいと思います。どうぞお楽しみにしてください。

参考文献

- 1) Lerdahl, F. and Jackendoff, R. : *A Generative Theory of Tonal Music*, The MIT Press (1983).
- 2) Hamanaka, M., Hirata, K. and Tojo, S. : Implementing "A Generative Theory of Tonal Music", *Journal of New Music Research*, 35:4, pp.249-277 (2007).
- 3) Hirata, K., Tojo, S. and Hamanaka, M. : Techniques for Implementing the Generative Theory of Tonal Music, ISMIR 2007 Tutorial. チュートリアルスライドはhttp://ismir2007.ismir.net/proceedings/ISMIR2007_tutorial_hirata.pdf からダウンロード可。
- 4) 片寄, 橋田, 野池 : 演奏上での頂点とグループ境界の聴取モデルについて, 情報処理学会 音楽情報科学研究会, 2003-MUS-52, pp.95-102.
- 5) Widmer, G. : The Importance of Basic Musical Knowledge for Effective Learning, In (Eds.) Mira Balaban, Kemal Ebcioglu, and Otto Laske, *Understanding Music with AI : Perspectives on Music Cognition*, pp.491-507, The AAAI Press/The MIT Press (1992).
- 6) 新音楽辞典 楽語, 音楽之友社 (1977).
- 7) Hewlett, W. B. and Selfridge-Field, E. (Eds.), *Melodic Similarity : Concepts, Procedures, and Application. Computing in Musicology*, Vol.11, The MIT press, Cambridge (1998).
- 8) MIREX Symbolic Melodic Similarity Contest, http://www.music-ir.org/mirex/2007/index.php/Symbolic_Melodic_Similarity
- 9) Cooper, M. and Foote, J. : Automatic Music Summarization via Similarity Analysis, In *Proc. of ISMIR 2002*, pp.81-85.

(平成20年8月13日受付)

平田圭二(正会員) hirata@bri.ntt.co.jp

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基礎研究所入所。1990～93年(財)新世代コンピュータ技術開発機構(ICOT)。1999年よりNTTコミュニケーション科学基礎研究所。2001年本会論文賞, 2003年山下記念研究賞。音楽情報処理に興味を持つ。ビデオコミュニケーションシステムtRoomのプロジェクトに取り組む。

浜中雅俊(正会員) hamanaka@iit.tsukuba.ac.jp

2003年筑波大学大学院工学研究科博士課程修了。同年日本学術振興会特別研究員PD, さきがけ研究員(専任)などを経て2007年より筑波大学大学院システム情報工学研究科講師。音楽情報処理の研究に従事。博士(工学)。2001年山下記念研究賞, 2001年SCI 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), ICMC2005 Best Paper Award受賞。

☆6 F値とは、適合率R, 再現率Pとしたとき $F = 2RP/(R+P)$ で計算される値。