

作者内一貫性と音楽的典型性を考慮した作曲支援インタフェース

中村 裕美^{†1} 中野 倫靖^{†1} 深山 覚^{†1} 後藤 真孝^{†1}

概要: 本稿では、自身の作品の一貫性と、楽曲としての典型性を考慮したメロディ作曲および作曲方針支援インタフェースを提案する。提案インタフェースは自身の作品集合から作者内一貫性を、様々な作曲家の作品集合から音楽的典型性を階層 Pitman-Yor 言語モデル (HPYLM) を用いてありがち度として求め、これまでの作品の作者内一貫性と音楽的典型性をユーザーにフィードバックする。このインタフェースによって従来では対象楽曲すべて聴きなおして行っていたこれまでの作品傾向の俯瞰的把握を行うことができ、作者による作成する作品の方針決定を支援できる。これによって、これまでの作風やスタイルに沿った支援では選択しにくい自身のスタイルや音楽的典型からの逸脱も選択肢に含めることができる。また、作曲中の作品の一貫性・典型性をリアルタイムで表示する機能、次に入力する音列の推薦機能によって、作成中の楽曲の現状把握と自身が定めた目標を達成するための支援を行う。

1. はじめに

作曲者は作曲行為を続けていく中で、楽曲に対し様々な視点でその方向性を考慮することがある。例えば、作曲中の作品のメロディやリズムといった1曲単位・1曲の一部に対する単体の曲を対象とした局所的な視点もあれば、作曲者自身の作品群の中にどう位置付けるかという俯瞰した視点も必要となる。特に楽曲の位置づけを考えるためには、自身の作品群を聴き直すことも求められるが、楽曲数が増えるほど全容を把握する時間は増える。また、特定のフレーズやリズムについて意識できても、作品全体を俯瞰して判断することが困難になる。そのため、作曲者が自身の記憶力だけでこれらの作業を行うには多大な労力を要する。そこで本研究では、作曲者が作品を作り続けていく中で直面する、自身の次の作品を作る際の方針決定と、その決定に基づいた作曲を支援するインタフェースを提案する。

本稿では自身の作品群に対する俯瞰的な把握の支援と、実際の作品への支援を行うために、作者内の一貫性と、楽曲の典型性という指標を定義し(図1)、その計算手法と提示方法を提案する。本稿では、一貫性の高い楽曲は、作品群に対して楽曲の形式、使用楽器、使用する調やメロディ・リズムのパターンなどに共通の要素が含まれる比率が高いと仮定し、1作品と自身の作品集合とを比較して計算可能なありがち度[1][2]を作者内一貫性の指標として用いる。

本指標は、一貫性を高くしたい場合のみでなく、自身がまだ挑戦したことのない作風に挑みたい場合には一貫性が低くなるよう意識するためにも用いることができる。しかし、一貫性を低くするように作曲すると、新しい作風やメロディのパターンへの挑戦につながる反面、作曲者がこれまでに培った経験を活用しにくくなる可能性が生じる。そこで、作成している楽曲が音楽的に許容できるものか、作りたいジャンルと適合するか等の考慮を助けるために、その作品とさまざまな作曲家の作品集合とを比較した音楽的

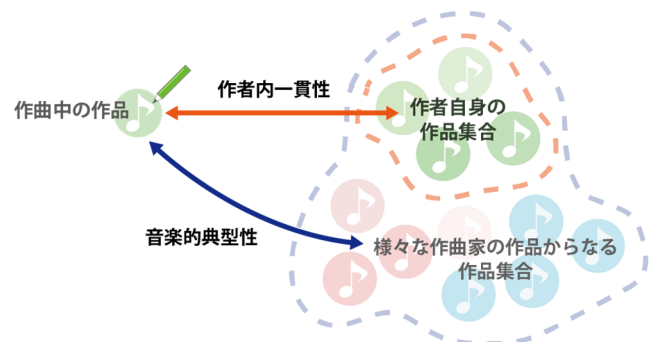


図1 作者内一貫性と音楽的典型性

典型性も同時に計算する。音楽的典型性が高いほどジャンル内で確立した表現がよく使用されている可能性がある。

本稿では楽曲の典型性を、その作品とさまざまな作曲者の作品集合と比較した際の類似度として定義し、一貫性と同じく1作品と作品集合とを比較して計算可能なありがち度[1][2]を用いて算出する。この典型性が高いほど楽曲としてジャンル内で確立した表現であると考えられる。

この二つの指標によって、作曲者は作品の位置づけに以下の様な指標を持つことができる。一貫性の高い曲を作ろうとした場合、自身のスタイルやまとめて聴取したときのその人らしさは上がる反面、あまりにも一貫性の高い作品を作り続けると、作曲者の創作による満足感が弱まる可能性も高まる。そのため、一貫性の低い曲を作ることで、作曲者本人の作品の傾向として新たなバリエーションを加えられ、経験の幅が広がる可能性が高まる。また、典型性が高い曲を作る場合は、音楽の共通的な知を活用できているため音楽的な許容度は高まるが、自身の作品の中では新規な方向性でも、音楽的にはありきたりと思われる可能性が生ずる。逆に、低すぎると、聞き手にとっては音楽として許容しにくくなるのが懸念される。

上記のような点を踏まえると、「新しい作風にトライするため一貫性は低くてもよいが、ジャンルに沿った音楽を作りたいので典型性は高めにした」「自分の作品感を出し

^{†1} 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology (AIST)

たいから一貫性は高くしたいが、ジャンルの見たら新奇なものを作りたいから典型性は低くしたい」といったように、自身の作風やスタイルの維持度（一貫性）と音楽的な許容度（典型性）を鑑みつつ、自身の次の作品の位置づけを決められることは重要である。

本研究では、作品に対して作者内一貫性（以下、一貫性と略す）と音楽的典型性（以下、典型性と略す）を計算する手法の提案と、作曲の方向性や進行を決めるための支援インタフェースの提案を行う。また、過去の作品から機械学習を行った作曲者の傾向を用いて音高列の推薦を行い、作曲者が定めた方針を考慮した作曲を支援する。

2. 関連研究

作曲支援システムはこれまでも数多く提案されている。ここでは特に作曲者のスタイルを考慮した作曲システム、推薦候補生成と提示インタフェース、コンテンツ生成過程にかかわる俯瞰的情報を提示する研究について述べる。

2.1 作曲者のスタイルを考慮した作曲システム

作曲者の作品群には類似性があるという前提のもと、その類似性を活用した作曲を行うシステムは以前から提案されている。たとえば、Pachet らによる FlowComposer [3] や、Cope が提案したシステム[4]は、作曲者の過去作品を用いて学習することで、その作曲者のスタイルを反映させた作品を自動生成している。

この考え方は本稿の提案と近いが、先行研究が過去作品からその人らしいもの、本稿の定義でいえば一貫性の高いものを生成するのに対し、本稿では学習結果や推薦を提示するもの、それを採用してその人らしさ（一貫性）を高めるか、無視するかは作曲者に委ねられている。

2.2 音楽生成・分類・推薦に基づく作曲支援

本稿のインタフェースによる支援では、作品集から学習した結果を用いた音列の推薦を行っている。このように学習結果から推薦や自動生成を行う研究や、ユーザの作曲行為を支援する研究は多く存在する。

白井らは可変長 Pitman-Yor 言語モデル (VPYLM) [5] を用い、歌詞の入力情報とコード進行入力情報からメロディを生成する手法を提案している[6]。N-gram 言語モデルは作曲者の分類にも活用されており[7]、Jacek らはメロディ及びリズムのデータを用いた学習において、モーツァルトとハイドンの作品の傾向が別のものであることを算出している[8]。Kitahara et al.はループ音楽に対し、隠れマルコフモデル (HMM) を用いてその盛り上がりユーザに輸入させることでループ内の音楽構造を変化させる提案を行っている[9]。

また、こういった推薦・生成をユーザの作曲行為の支援に用いる研究も数多く提案されている。Kitahara et al.の OrpheusBB では、こういった作曲形態を「Human-in-the-loop

型の自動作曲」と定義し、歌詞の制約に基づいたメロディおよび伴奏の推薦を行いつつ、ユーザの決定を優先させている[12]。推薦が棄却された時点でユーザの入力を元に再計算を行い、再度推薦候補を選出している。また、谷口らはユーザ側の推薦確認の負担や解の探索の複雑化を避けるため、ギブスサンプリングを用いたインタラクティブ作曲支援システムを提案している[13]。本研究も、算出結果に対するユーザからのインタラクティブ性を有する。

本研究では一貫性・典型性の算出に階層 Pitman-Yor 言語モデル (HPYLM) [10][11]を用いる。また、提案インタフェースは[12]と同じくユーザの推薦を無視した場合には、そのユーザの入力を優先し、新たに一貫性と典型性の計算と推薦候補の提案を行う点は、OrpheusBB などと近い思想であると考えられる。

2.3 創作時の対象コンテンツ作成と作成にかかわる俯瞰的情報の提示

コンテンツ作成において、実際に創作する対象の閲覧・編集の様に局所的な作業と、その方向性や指針などの俯瞰的思考の双方を支援するものも提案されている。中小路らの Representational Talkback があげられる[14]。この研究では、テキスト作成やイラスト作成において、実際に編集するコンテンツ生成画面とともに、そこで記述する内容を検討するための思考ツールを有している。また、網谷らは作者のメンタルスペースをシステム上に外在化させ、音楽的特徴と結び付ける創作支援手法を提案している[15]。メンタルスペースの外在化によって、自分自身との対話を具体的・客観的にとらえる、メンタルスペースの全体を見渡すといったことが可能になり、これが実際の作品を作ることの支援につながると述べている。

本研究のインタフェースは、実際のコンテンツを編集する部分と、その方針を検討するための俯瞰的情報提示を有している点がこれらの既存研究と共通している。

3. 提案手法

本章では一貫性と典型性の定義とその算出手法について述べる。また、その結果を活用した作曲支援インタフェースについて説明する。

3.1 楽曲におけるありがち度の算出

本稿では一貫性・典型性として楽曲間の類似度・ありがち度を算出する。この指標を求めるために、Nakano et al.による音楽的ありがち度 (Musical typicality) を応用する[1][2]。Nakano et al.は、音色などの (bag-of-words で表現可能な) 特徴量を潜在的ディリクレ配分法 (LDA) で、コード進行を VPYLM[5]でモデル化した上で、楽曲集合モデルからの各楽曲の生成確率としてありがち度を定義した[2]。その後、情報理論における典型性の概念を導入することで、LDA に

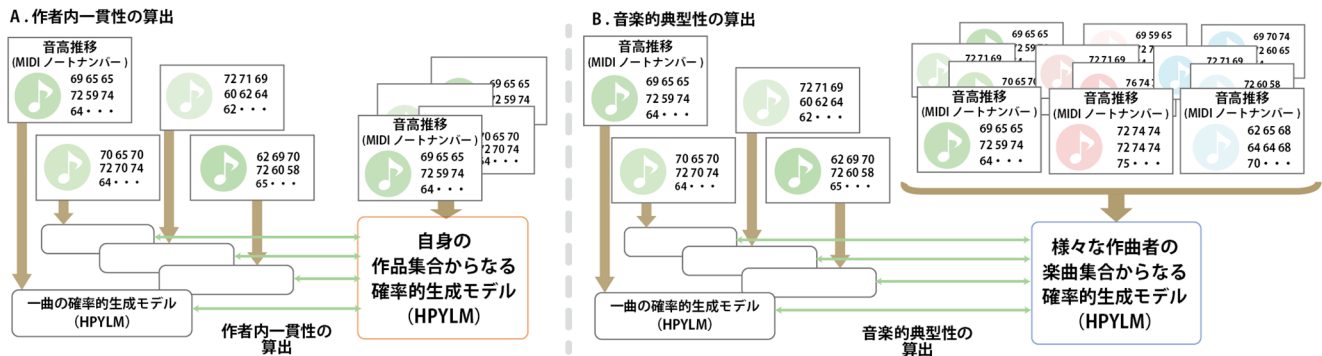


図 2 A. 作者内一貫性の算出手法 B. 音楽的典型性の算出手法

基づくありがち度の計算を改善した[1].また、楽曲全体に対して算出を行い、学習データとして「ボーカルの歌声」「楽曲中の音色」「リズム」「和声進行」「歌詞」を対象とした。

3.2 ありがち度の一貫性と典型性への応用

本稿では一貫性と典型性の算出、インタフェース上でのユーザのインタラクションを考慮し、3.1 で述べた中野らのありがち度算出手法を以下のように応用した。

まず、一貫性は作曲者が自身の作品の傾向を把握し、方向性を決定する指標となる。すなわち自身の作品の一曲一曲、または作曲中の楽曲が自身の中でどれぐらいありがちか、自身の各作品が作品集合に対してどの程度一貫しているかを求める必要がある。そのため、対象楽曲と比較する楽曲集合に、作曲者の作品集合を用いる (図 2A)。

それに対し典型性は、さまざまな楽曲に対して自身の作品がどれぐらい類似しているか、典型的な作品であるかを計算する必要がある。そのため、中野らの提案と同じく、さまざまな作曲家の作品集合と、対象とする楽曲とでありがち度を求める (図 2B)。

加えて、インタフェースは作曲中の支援も行うため、完成した楽曲だけでなく、作成中の楽曲に対してその時点までの情報を元にリアルタイムに一貫性と典型性を計算し、ユーザにフィードバックする必要がある。そのため本稿では、新たにメロディー (音高遷移) を対象とし、サイズ n -gram 言語モデルとして HPYLM を用いて、モデル間の Kulback-Leibler (KL) 尺度の指数をとったものの期待値を計算し、ありがち度を計算する[1]。この分析結果をユーザに提示することで、作曲中にも自身が目標として定めた一貫度と典型性に基づく指標に近づける選択ができたかフィードバックを受けることができる。

3.3 一貫性・典型性の算出

本稿の一貫性・典型性を算出にはメロディー (音高遷移) を学習データとするため、MIDI シーケンス内のメロディの MIDI ノートナンバーを使用した。移調して共通の調に揃えることは行わず、音価などの情報も現在は利用していない。今後音価やフレーズ、構成情報などの音楽的特徴の

利用することを検討している。

一貫性・典型性を算出するための確率モデルにはバイズ n -gram 言語モデルである階層的 Pitman-Yor 言語モデル (HPYLM) [10][11]を用いた。本言語モデルでは階層的バイズスムージングによってゼロ頻度問題が緩和される。また、次の音高の予測確率は直前 $n-1$ 語の履歴から計算することができる。 n は事前に $n=3$ と設定し、言語モデルの語彙サイズ W は 40 とした。

モデル間類似度 (ありがち度) は、同時分布で計算する。ありがち度の計算[1]は定常無記憶情報源を仮定しているため、 n 個の音 (n 個の確率変数) の同時分布を定常無記憶情報源だとみなして、同様の計算を行う (n 個の音を独立にサンプリングするモンテカルロ近似を行う)。この結果から求められるありがち度 (P) は、学習して得られた確率モデル Q との間で以下のように計算される。 $D(P \parallel Q)$ を PQ 間の KL 尺度として。

$$\text{Typicality}(P|Q) = \exp\{-D(P \parallel Q)\} \quad (1)$$

上記の定義を n -gram に適用し、連続する n 音の同時確率に基づいて、ありがち度 (一貫性・典型性) を以下のように定義する。

$$\text{Typicality}(P_n|Q_n) = \exp\left\{-D\left(p_P(x_t, x_{t-1}, x_{t-2}) \parallel p_Q(x_t, x_{t-1}, x_{t-2})\right)\right\} \quad (2)$$

$$p(x_t, x_{t-1}, x_{t-2}) = p(x_t|x_{t-1}, x_{t-2})p(x_{t-1}|x_{t-2})p(x_{t-2}) \quad (3)$$

ここで P_n は対象楽曲の n -gram モデル、 Q_n は楽曲集合の n -gram モデルである (ともに $n=3$)。 Q_n のモデルパラメータ学習において、一貫性では作曲者の作品集合、典型性では様々な作曲者の作品集合を用いる。そして、最終的に、モンテカルロ近似[16]で以下の処理を行うことで、計算処理を軽減させる。

$$\begin{aligned} D\left(p_P(x_t, x_{t-1}, x_{t-2}) \parallel p_Q(x_t, x_{t-1}, x_{t-2})\right) \\ = \sum_{x_t \in W} \sum_{x_{t-1} \in W} \sum_{x_{t-2} \in W} p \log \left(\frac{p(x_t, x_{t-1}, x_{t-2} | P_n)}{q(x_t, x_{t-1}, x_{t-2} | Q_n)} \right) \end{aligned}$$

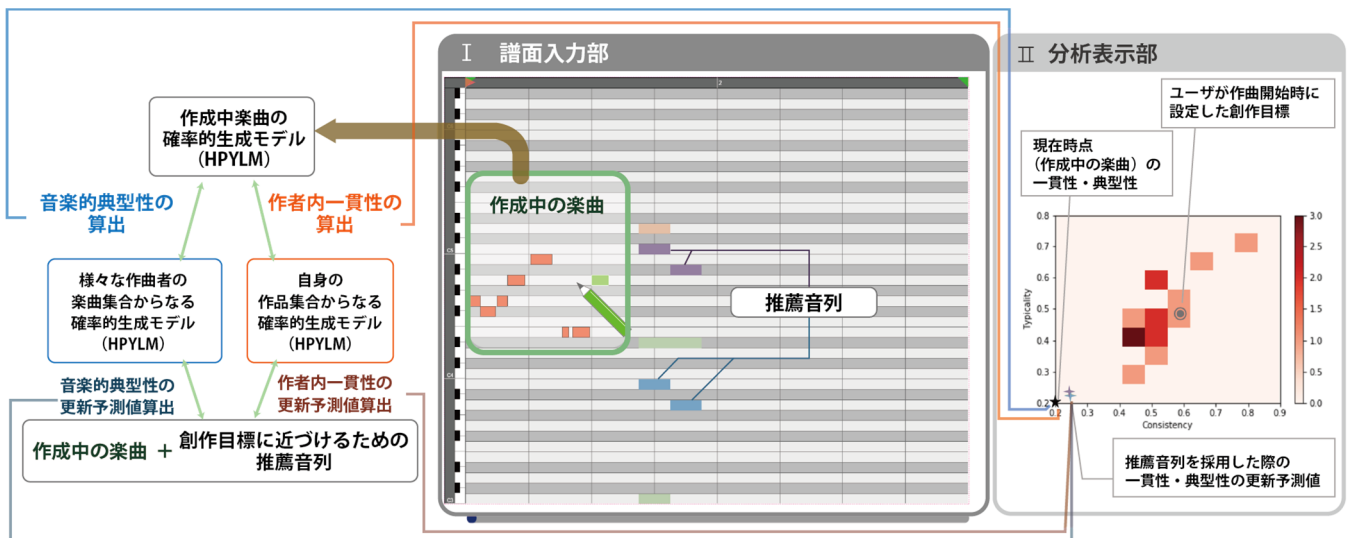


図 3 作曲支援インターフェースの構成 I. 譜面入力部と II. 分析表示部がインターフェース上に配置される。

$$\approx \frac{1}{R} \sum_{r=1}^R \log \frac{p(x_r | P_n)}{p(x_r | Q_n)} \quad (4)$$

この式における $x_r (= x_{t-2:t}^{(r)})$ は $P_p(x_t, x_{t1}, x_{t2})$ からサンプルされた 3 組の音列, R はサンプル数を指す. $P_p(x_t, x_{t1}, x_{t2})$ からのサンプルにおいては, 式(5)のように, 確率の乗法定理に基づいて展開してサンプリングした. 具体的には, x_{t-2} に対し uni-gram でサンプリングした結果を使用して x_{t-1} を bi-gram でサンプリングし, その結果を用いて x_t を tri-gram でサンプリングする. このように, $1 \sim n-1$ までの確率分布を順に求める際に, HPYLM のスムージングで得られた結果を用いた.

4. 一貫性と典型性を用いた作曲支援インターフェース

本章では提案インターフェースについて説明する. ユーザはこのインターフェースを用いることで, 以下の点について支援を受けながら作曲を進められる.

まず, ユーザは 1 章に述べたように自身がこれから作曲する楽曲に対して, 一貫性と典型性の観点から自身の作品集合を俯瞰した上で創作目標を決定できる. また, その創作目標に作曲中の楽曲が近づいているかリアルタイムで把握できる. そして, 目標に近づくための具体的な支援 (音列の推薦) を受けることができる.

インターフェースは作曲とこれらの支援を行うため, 譜面入力部と分析表示部を持つ (図 3). 以下では, その構成と想定するユーザの操作, ユーザの目標を考慮した推薦について説明する.

4.1 作曲支援インターフェースの構成

分析表示部 (図 3 I) には, ユーザ自身がこれまでに作

曲した作品から計算した一貫性・典型性を軸とした分析結果が表示される. 提案インターフェースでは一貫性・典型性双方から過去作品を分析する必要があること, 作成曲で目標とする値の設定が行いやすいことから, 2 次元ヒストグラムを用いている. この 2 次元ヒストグラムは x 軸に一貫性, y 軸に典型性が配置されており, 該当の一貫性・典型性の作品が多いほど, ヒストグラム上の色の濃度が高く表示される. また, 譜面入力部 (図 3 II) については, メロディの入力を対象とするためピアノロール形式を用いた. ユーザはここに作成中の楽曲を入力することで, 一貫性・典型性のリアルタイムフィードバックと, 音列の推薦を受けることができる.

4.2 想定するユーザの操作

まず, ユーザは楽曲作成前に分析表示部の 2 次元ヒストグラムを閲覧する. このヒストグラムでは自身の過去の作品の傾向を, 一貫性・典型性の観点から俯瞰的に振り返ることができる. 自身の傾向を把握した上で, ユーザは作曲する楽曲の創作目標を定める. 創作目標はヒストグラム上で希望する一貫性と典型性の値の交点をクリックすることで設定できる. この値は作成曲の一貫性・典型性の目標値として保持され, 推薦音列の提案の際にも活用される.

次に, ユーザは譜面入力部を用いて実際にメロディを入力する. ユーザが入力するメロディは, 選択中は緑色, 決定後は赤色で表示される. インターフェースはこのユーザによる入力を作曲中の楽曲とみなし, 一定音数に達した時点で (本稿では便宜上 8 音としている) 作成中の楽曲を基に現時点での一貫性・典型性を算出する. この算出結果も 2 次元ヒストグラム上に星印で反映される. ユーザは, このリアルタイム表示の推移と, 先に入力した目標の座標を確認することで, 自身の現状までの入力結果が先に定めた目標に向かっているかを確認することができる.

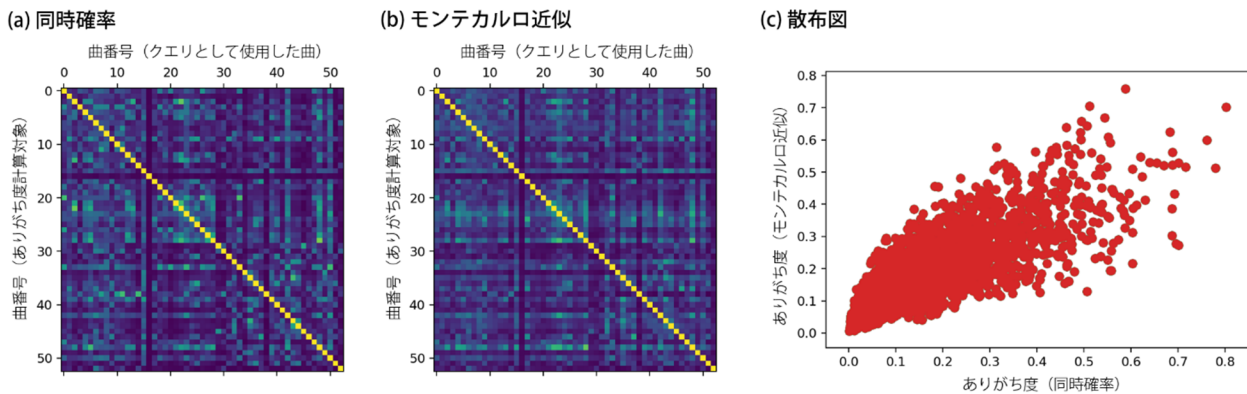


図 4 (a)式 2 に基づいたありがち度マトリクス (b)式 4 に基づいたありがち度マトリクス (c)式 2 および式 4 から得られたありがち度の散布図

しかし、現在の入力と目標の差異を比較できても、その目標に近づくような次の音列を検討するのは難しい場合がある。そのため、インタフェースでは作品集合データを用いて行った学習結果を基に、目標値に近づくような音列を推薦し、ユーザの目標値達成を支援する。この音列は、複数推薦され、譜面入力部の作曲中の楽曲（赤色部分）に引き続くように異なる色で複数表示される。

また、推薦された音列を採用した場合に一貫性・典型性がどの程度目標値に近づくのかはユーザにとって推測しにくい。そのため、推薦された音列を採用した場合の一貫性・典型性の更新値も算出する。これは、作成中の楽曲に推薦候補を追加した音列に対して一貫性と典型性を算出することで求められる。この更新値は、譜面入力部の音列と対応した色の十字印で分析表示部に表示される。ユーザは実際の音列に加え推薦を採用した場合の一貫性・典型性の更新値の表示を確認することで、推薦結果がそれまでの入力に続く音列としてふさわしいか、更新値の変化が望ましいかの双方を参考に次の音の入力を決定することができる。

推薦される音列はユーザの希望と大きく異なり、ユーザが考えていたメロディの進行とは一致しない場合も生ずる。その場合、ユーザは推薦を無視して入力を行うことも可能である。インタフェースは入力した音高を優先させた状態で再度一貫性・典型性の計算を行い、現時点の一貫性の値と、それに基づいた推薦内容が更新される。

4.3 ユーザの入力と目標値を用いた推薦

本節では、インタフェースで提供する推薦候補を生成する 2 種類の方法を説明する。

4.3.1 推薦生成手法 1. 同時分布に基づく推薦

一つ目はまとまった数(3 つ)の音高の組合せを同時分布 $p(x_t, x_{t-1}, x_{t-2})$ に基づいて生成する方法である。一貫性・典型性の算出には同時分布が用いられているため、この同時分布から音列を生成することで、一貫性を効率的に高められる可能性が高くなる。

インタフェースでは、事前に同時分布による音列を 20 種

生成する。ただしこれをすべて提示すると譜面入力部・分析表示部ともに表示が煩雑になり、ユーザの選択の負荷も高まる。そのため、ユーザが入力した達成目標の値を活用し、その達成目標と距離の近い一貫性・典型性更新値になる 5 種を選出し、インタフェース上に表示する。

4.3.2 推薦生成手法 2. 条件つき確率分布に基づく推薦

二つ目は条件つき確率分布 $p(x_t | x_{t-1}, x_{t-2})$ に基づいた生成する方法である。同時分布を用いる方法では、それまでの音列とは独立に引き続く音列を生成するため、すでに入力された音列との接続部分で不自然なつながりが提供される可能性がある。それに対し、条件つき確率分布を用いる方法では、推薦音列の直前にある音列を考慮して生成するため、ユーザの入力した音列に自然に接続する音列を推薦できる。推薦する音列の生成方法を工夫することで、一貫性・典型性の更新値にバリエーションを持たせることができる。

具体的には、生成確率の高い音列ばかりを候補として複数生成すると、作品集合の中でよく用いられる音列が続きやすくなり、一貫性・典型性の更新値は高くなると考えられる。逆に、生成確率が上位のものをあえて用いずに推薦候補を生成することで、作品集合の中で用いられる頻度が低い音列が生成され、一貫性・典型性の更新値は、生成確率の高い音列ばかりを生成した場合と比べて低くなると考えられる。

これによって、一貫性や典型性の上昇率を制御できるため、目標値として定めた一貫性の値と現時点での一貫性の値の差が小さい場合には、緩やかに一貫性を上昇させるような推薦を採用することができる。

5. 実験

本章ではインタフェースで提供される一貫性・典型性高速化での近似性能と、ユーザへ推薦する音列生成に関する実験について述べる。本稿ではありがち度算出の際にモンテカルロ近似を使用している。この近似性能について、

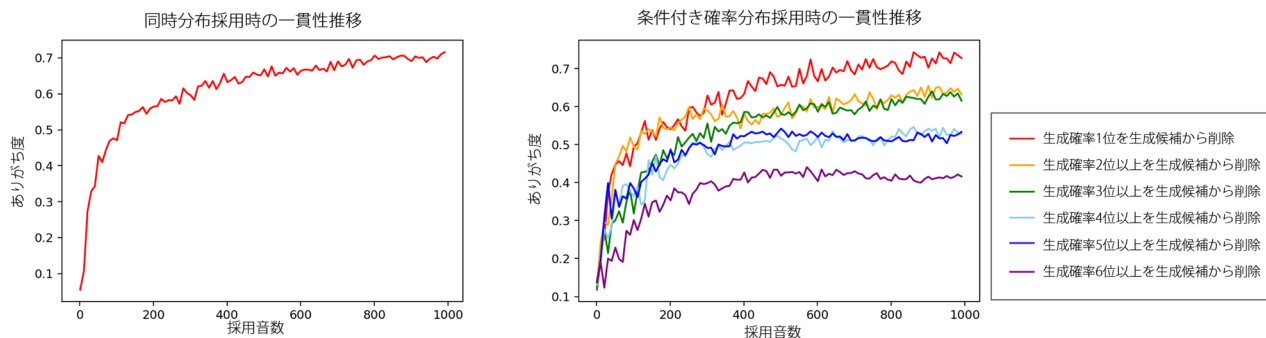


図 5 左：同時分布を採用し続けた場合の一貫性推移 右：条件付き分布を採用し続けた場合の一貫性推移

同時確率を用いた場合と比較した。また、インタフェースで提供される推薦音列のバリエーションによって更新値の制御が可能か確認するために、推薦を採用し続けた場合の一貫性の変化を求めた。

5.1 実験に用いたデータセット

ユーザがインタフェースに自作品の楽曲群を登録し、一貫性を参照しながら創作することを模擬するために、既存の特定の作曲者の作品集を登録して実験を行った。また典型性の算出のために、複数の作曲者による作品を追加した。本稿では歌曲の単旋律データを使用し、データセットとしてモーツァルト 16 曲、ブラームス 16 曲、シューベルト 21 曲の計 53 曲を採用した。これらの楽曲の MIDI データからメロディの音列を抽出し、学習データとして用いた。

5.2 一貫性と典型性の推定についての評価

まず、モンテカルロ法に基づくありがち度の近似計算の妥当性を評価した。

5.2.1 実験手法

実験では、式 2 を用いた同時確率の計算と、式 4 を用いたモンテカルロ近似を用いて、作品ごとに典型性を求めた。サンプル数は $n=100$ とし、HPYLM の学習においては 500 回のギブスサンプリングを行った後の単一サンプルをありがち度計算に使用した。

5.2.2 実験結果・考察

図 4(a)は式 3 をもとに同時確率を用いてありがち度を求めた結果、図 4(b)はモンテカルロ近似を行ってありがち度を求めた結果である。この双方の結果の対角要素を除いた要素の相関係数は $p = 0.807$ となり、双方の値の散布図は図 4(c)のようになつた。このことから、モンテカルロ法による近似計算は同時確率と相関の強い結果を出力することが示唆されたため、計算高速化のためにモンテカルロ近似を用いる影響はほぼ生じないことがわかった。

5.3 一貫性・典型性に基づいた推薦を採用した場合の一貫性の変化

インタフェースはユーザが設定した創作目標の達成を支援するために音列を推薦する機能を持つ。この創作目標は、ユーザが決定した一貫性・典型性のデザイン方針を基

に入力される為、それぞれが 2 次元ヒストグラムの範囲内の様々な値(0~1.0)をとりうる。そのため、その創作目標の達成を支援する推薦の生成では、目標値に合わせて更新値による変化量を制御する必要がある。

この推薦による更新値の制御を検討するため、まず本稿では、推薦生成に用いた 2 つの確率分布での推薦候補を採用し続けた場合に、一貫性が実際にどのように変化するかを確認した。また、条件付き確率分布では生成確率を基にした音列生成のコントロールによって、一貫性の変化に差異が見られるかを確認した。

5.3.1 実験手法

この実験では、4.3.1 で述べた同時分布を用いて生成した音列と、4.3.2 で述べた条件付き確率分布を基に生成した音列をそれぞれユーザが採用し続けたと仮定した場合の一貫性の推移を算出した。算出には 5.1 に上げたデータセットのうち、ブラームスの 16 曲を用いた。このようなシミュレーションによる評価手法は、先行研究においても採用されている[17][18]。これらの研究ではタグやアーティスト分類などを制約とした上でその中から推薦されたものを採用し続けた場合のシミュレーションを行い、そのスキップ率を算出して評価している。

同時分布を用いた推薦による算出では、 $n = 3$ とし 3 音 $\times 1000$ 個生成したものを 10 音ずつ元の音列の最後尾に追加し、そのたびに一貫性の計算を行った。条件付き確率分布を基にした推薦による算出では、生成のパラメータとして、生成確率の上位をどの程度採用するかを変化させて算出した。生成確率で推薦候補を順位付けし、1 位からすべてを候補として採用し生成したものから、上位から 6 位までを削除して生成したものの計 6 パターンを用意した。3 音 $\times 1000$ 個生成したものを 10 音ずつ元の音列の最後尾に追加し、一貫性の計算を行った。

5.3.2 実験結果

図 5 左は同時分布による推薦を採用し続けた結果、図 5 右は条件付き確率分布による推薦を採用し続けた結果である。条件付確率分布を用いた推薦において、生成確率上位を削除した生成を行うことで、一貫性の上昇度を制御でき

ることが明らかとなった。これによって、目指す一貫性と現在の値との距離に応じて、生成確率上位の採用数を変更することで、一貫性の上昇度を制御できることが示唆された。また、同時分布を用いた場合も、生成確率上位のものをすべて採用した状態とほぼ同程度の一貫性の上昇が見受けられた。

しかし、400音程度までは生成確率の上位採用数を変化させても一貫性の上昇度が意図した通りに制御できない傾向が示された。実際のインタフェースの利用に即して考えると、入力したメロディの音数が400音以上に到達するのは楽曲をほぼすべて入力し終わっている可能性も生じる。そのため、音を入力している初期の段階では、この一貫性の上昇度の制御方法はうまく機能しないと考えられる。

6. 終わりに

本稿では、楽曲作品に対して作曲者の一貫性と、典型性を算出し、作成する作曲の方針決定と方針に基づいた作曲を助ける推薦手法を提案した。また、この算出された一貫性・典型性と推薦結果をユーザに提示し、作曲を支援するインタフェースを実装した。実験として一貫性・典型性計算における高速化で用いたモンテカルロ近似の妥当性の評価と、推薦の生成手法とその調整によって更新値の制御を行えるか検討した。結果、確率分布から得られた生成確率上位の音列候補の採用数を変更することで一貫性の上昇率を制御できることが明らかとなった。しかし、安定した一貫性の制御には一定の音数の入力が必要であることも明らかとなった。今後、この点においては計算手法の改良や、一貫性・典型性を計算する範囲の再設定などを行い、入力音数が少ない状態でも機能するよう改良を検討している。

音列の推薦に関しては、今後剽窃や盗作を助長しないような配慮が必要である。Papadopoulosらの論文でもマルコフモデルを用いた自動生成で剽窃を防ぐ手法を取り上げている[19]。本稿で用いた一貫性に基づいた推薦の場合、学習データは自身の作品のため、剽窃・盗作にはならない。また、典型性に基づいた推薦は、多くの作品にありがちな系列となるので、それらは共有知として使える可能性がある。ただし現状では、ありがち度の計算にあたって音高の出現回数のみが考慮され、複数の作曲者が使っている「ありがちな」系列かどうかまで判断ができないため、そのような判断のできる機構を導入することが今後の課題である。そして、インタフェースを用いたユーザスタディは未実施である。一貫性・典型性の提示手法、推薦の提示手法においては作曲経験者へのヒアリングや提案インタフェースの使用実験を通して改良する予定である。

謝辞 本研究の一部はJST ACCEL「OngaACCELプロジェクト」(No.JPMJAC1602)の支援を受けました。

参考文献

[1] Nakano, T. Yoshii, K. and Goto, M Musical Typicality: How Many

- Similar Songs Exist?. In Proceedings of The International Society of Music Information Retrieval, 2016, pp.695-701.
- [2] Nakano, T. Yoshii, K. and Goto, M.. Musical Similarity and Commonness Estimation based on Probabilistic Generative Models of Musical Elements, International Journal of Semantic Computing, 2016, Vol.10, No.1, pp.27-52.
- [3] Pachet, F. and Roy, P.. Imitative Leadsheet Generation with User Constraints. 21st European Conference on Artificial Intelligence (ECAI 2014), 2014, pp.1077-1078.
- [4] Cope, D.. An Expert System for Computer-Assisted Composition. Computer Music Journal, 1987, vol.11, no.4, pp. 30-46.
- [5] 持橋大地, 隅田英一郎. 階層 Pitman-Yor 過程に基づく可変長 n-gram 言語モデル. 情報処理学会論文誌, 2007, Vol.48, No.12, pp.4023-4032.
- [6] 白井亨, 谷口忠大. 可変長階層 Pitman-Yor 言語モデルを用いたメロディー生成手法の提案.知能と情報 (日本知能情報フュージ学会誌兼論文誌), 2013, Vol.25, no.6, pp. 901-913.
- [7] Lo, M. Y. and Lucas, S.M.. Evolving Musical Sequences with n-gram based Trainable Fitness Functions. In Evolutionary Computation, 2006, pp. 601-608.
- [8] Wołkiewicz, J. and Kešelj, V.. Evaluation of n-gram-based Classification Approaches on Classical Music Corpora. In Conference on Mathematics and Computation in Music, 2013, Springer Berlin Heidelberg, pp. 213-225.
- [9] Kitahara, T.. Smart loop sequencer: An Audio-based Approach for Ease of Music Creation. The Journal of the Acoustical Society of America, 2016, 140:4, p.3090.
- [10] Teh, Y.W.. A Bayesian Interpretation of Interpolated Kneser-Ney, Technical Report TRA2/06, 2006, NUS School of Computing.
- [11] Teh, Y.W.. A Hierarchical Bayesian Language Model based on Pitman-Yor Processes. In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2006, pp. 985-992.
- [12] Kitahara, T. Fukayama, S. Sagayama, S. Katayose, H. and Nagata, N.. An Interactive Music Composition System based on Autonomous Maintenance of Musical Consistency. In Proceedings of Sound and Music Computing, 2011.
- [13] Shirai, A. and Taniguchi, T.. A Proposal of an Interactive Music Composition System using Gibbs Sampler. Human-Computer Interaction. Design and Development Approaches, 2011, pp. 490-497.
- [14] Yamamoto, Y. Takada, S. Gross, M. and K. Nakakoji.. Representational talkback: An Approach to Support Writing as Design. In Proceedings of the 3rd Asia Pacific Conference on Computer Human Interaction, 1998, pp.125-131.
- [15] 網谷重紀, 堀浩一. 作曲者のメンタルスペースの外在化による作曲支援環境の研究. 情報処理学会論文誌, 2001, vol. 42, No. 10, pp. 2369-2378.
- [16] Virtanen, T. and Helen, M.. Probabilistic Model based Similarity Measures for Audio Query-by-example, in Applications of Signal Processing to Audio and Acoustics, 2007, pp.82-85.
- [17] Pampalk, E. Pohle, T. and Widmer, G.. Dynamic Playlist Generation based on Skipping Behavior. In Proceedings of The International Society of Music Information Retrieval, 2005, Vol. 5, pp. 634-637.
- [18] Cardoso, J. P. V. Pontello, L. F. Holanda, P. H. F. Guilherme, B. Goussevskaia, O. and da Silva, A. P. C.. Mixtape: Direction-based Navigation in Large Media Collections. In Proceedings of The International Society of Music Information Retrieval, 2016, pp. 454-460.
- [19] Papadopoulos, A., Roy, P., & Pachet, F.. Avoiding Plagiarism in Markov Sequence Generation. In Association for the Advancement of Artificial Intelligence, 2014, pp. 2731-2737.