

# メロディスロットマシン

浜中雅俊<sup>†1</sup>

**概要:** 本稿では、仮想演奏者の未来の演奏のコントロールを可能とするインタラクティブな音楽システム「メロディスロットマシン」について述べる。従来のセッションシステムなどにおける仮想演奏者は、人間による入力演奏に対して仮想演奏者がどのような出力演奏を生成するか焦点があてられており、出力演奏を直接コントロールすることは困難であった。これに対しメロディスロットマシンでは、ユーザがダイヤルを回すことで、仮想演奏者が次に演奏するメロディを選択することを可能とする。また、演奏者の姿をペッパーゴーストディスプレイに投影し、仮想演奏者が目の前にいるように感じることを可能とする。

## 1. はじめに

過去 15 年に渡り音楽理論 GTTM (Generative Theory of Tonal Music) に基づいたメロディ操作の研究を行ってきた [1-4]。最終目的は、GTTM を援用し、人間のように作曲や演奏をすることが可能な仮想ミュージシャンを実現することである。これを達成するには多くの課題があるが、現在の技術で可能なシステム「メロディスロットマシン」を構築し、今後の研究開発の方向性を探究する。メロディスロットマシンは、仮想演奏者の未来の演奏をユーザが操作することを可能とする。その特長は以下の 3 つである。

### (1) ユーザフレンドリなインターフェース

仮想演奏者が演奏するメロディはあらかじめ用意したメロディ断片をつなげ合わせることで生成する。そのメロディ断片の一部を別の断片に置き換えることができるダイヤル型インターフェースを構築し、ユーザが演奏を容易にコントロール可能にした (図 1)。

ユーザは、楽譜を印刷したアクリル板の一部に開けた長方形の穴から、裏に置いたタブレットの上に表示されたダイヤルインターフェースを操作できる。楽譜の右側にある赤いレバーを引き下げると、すべてのダイヤルが回転し、ダ

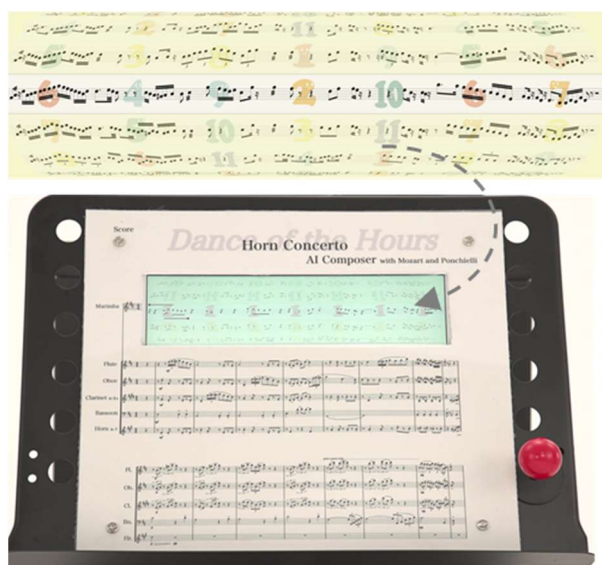


図 1 スロットダイヤルとレバー

イヤル上のメロディ断片がランダムに選択される。メロディ断片のバリエーションは、GTTM[1]に基づいて構成されているため、断片をどのように切り替えても、装飾的な音符のみが変化しメロディの全体的な構造は維持されるため音楽的な破綻は生じない。

### (2) 操作結果の視覚的理解

メロディの変化に気が付けない可能性もあるため、ダイヤル型インターフェースを操作した結果を、聴覚的にだけでなく視覚的にも確認できるようペッパーゴーストディスプレイに仮想演奏者を表示するようにした (図 2)。

使用したペッパーゴーストディスプレイ (Realfiction Dreamoc HD3) は 3 方向から見る事ができるタイプであったため、3 方向の背景が黒色のスタジオで 3 つのカメラを用いて各方向の映像を撮影し、撮影されたデータの時系列を合わせてペッパーゴーストディスプレイ用に合成した。複数の演奏者を表示するためには、それぞれ別々に撮影した後合成する。

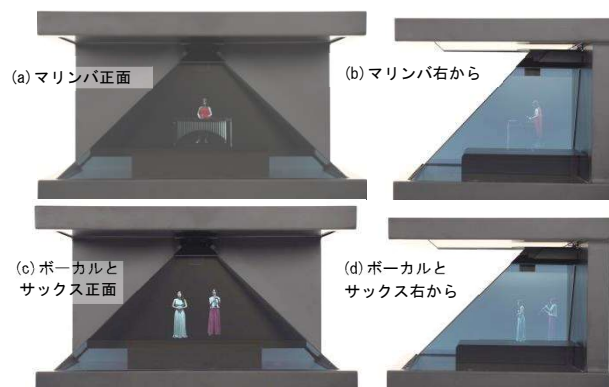


図 2 ペッパーゴーストディスプレイ

### (3) 音楽的な臨場感

音楽的にも高い臨場感で提供できるよう、演奏音はレコーディングを行った。本稿での臨場感とは、あたかも本当に目の前に演奏者がいるかのように自然に感じられることを呼ぶ。レコーディングには、残響音がほとんどないスタジオを使用した。録音されたメロディは断片に分割されるため、残響音があるスタジオでは、前の断片の残響のみが後ろの断片の先頭に入ってしまうことがあるためである。

<sup>†1</sup> 理化学研究所

残響は、仮想演奏者の演奏が出力する段階でプラグインを用いて付加した。

臨場感を高める工夫として、ディスプレイの周りに、前と左右の方向ごとに3組のスピーカを設置した(図3)。そして、サウンドエンジニアが方向ごとにパンポットとリバーブの設定を、目の前の(ペッパーゴーストディスプレイ上)の演奏者から演奏音が聴こえているように感じられるよう調整した。ユーザの頭を2組のスピーカの間に入れると、音楽的にも映像的にも最も臨場感が高まる。

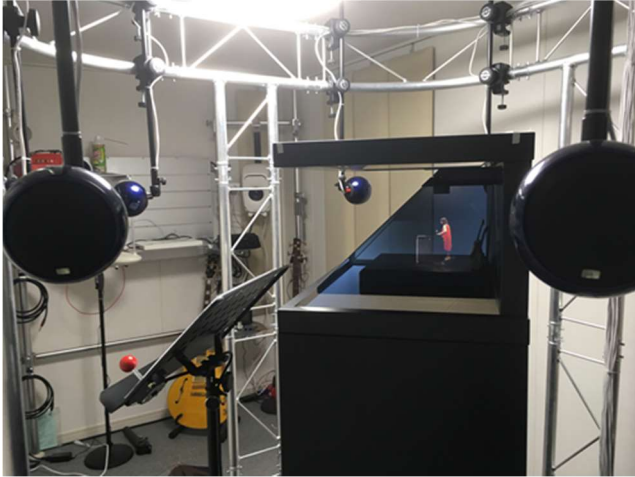


図3 方向ごとのスピーカ

## 2. 関連研究

Band-out-of-the-Box は、即興演奏する演奏者とコンピュータ上の仮想演奏者との音楽的なインタラクションを可能とした[5]。Virtual Musical Multi-Agent System [6]およびソフトウェア環境 OMax Brothers[7]は、マルチエージェントアーキテクチャに基づく仮想演奏者による即興演奏を可能とした。Continuator[8]は、演奏スタイルをリアルタイムで学習するためにマルコフモデルを拡張し、ユーザと仮想演奏者がユーザのスタイルでインタラクションすることを可能とした。Guitarist Simulator は、実在する人間の演奏者から振舞いのモデルを統計的に学習した仮想演奏者とのジャムセッションを可能とした[9]。これらのシステム[5-9]の出力は演奏音のみであり、仮想演奏者の視覚的な表示はなされていなかった。

VirJa Session では、仮想演奏者を視覚的にも確認可能として点が優れていたが、コンピュータグラフィックによって視覚化されたのは、リズムに合わせた足踏みや、身体の揺れなど比較的単純なものであった。これに対しメロディスロットマシンでは、ビデオカメラで撮影した演奏者の映像を使用することで臨場感を高めることを狙っている。

メロディ要約システム[11]、演奏レンダリングシステム[12]、およびメロディ予測システム[13]は、GTTM 分析の結果求まるタイムスパン木を使用したものである。たとえば、メロディ予測システムは、ピアノの天板にシステムが予測

された音符を縦スクロールするピアノロール形式で LED 表示することで、即興演奏の支援を可能としていた。即興演奏中に初心者、次にどの音を弾くべきか迷う場合があるが、天板に示された鍵盤を弾くことで、GTTM に基づく安定度の高いメロディを弾くことが可能となる。メロディ予測システムでは、鍵盤に慣れている人を対象としていたのに対し、メロディスロットマシンでは、ダイヤル型のインタフェースを採用し、寄り多くの人を対象とすることを目指した。

メロディモーフィング手法[14,15]は、GTTM による楽曲構造分析の結果求まるタイムスパン木を用いてメロディの操作を可能とした。しかし、タイムスパン木の自動分析精度[2-4, 13-15]は低いため、モーフィング操作の自動化は実現していない。そこで本研究では、GTTM 分析が可能な3人の音楽家が、メロディモーフィング法を使用してメロディ生成を行った。

モーツァルトによる「音楽のさいころ遊び」は、サイコロの出目に応じてあらかじめ作曲してあるメロディ断片を順番に結合していくことで曲が完成する[16]。メロディを切り替えるという点で、メロディスロットマシンは音楽のさいころ遊びのアイデアに触発されていると言える。ただし、音楽のさいころ遊びとメロディスロットマシンのメロディ作成方法は大きく異なっている。さいころ遊びでは、どのような順序で結合していても自然なメロディとなるように非常に精緻に設計されている。一方で、メロディモーフィング手法は GTTM の分析の結果として求まるタイムスパン木を使用しているため、メロディを切り替えた場合でも、基本的な構造は同じであり、装飾音のみが変更される。

## 3. メロディモーフィング手法

本節では、メロディ A と B を入力し、A と B それぞれの特徴を反映させる度合いを変化することで、メロディ A と B の間にある複数個のメロディ C をある尺度のもとで順序付けて生成するメロディモーフィング手法について述べる。このとき、メロディ A, B, C は以下の条件を満たすものとする。

1. A と B より A と C のほうが類似している、かつ、A と B より B と C のほうが類似している。
2. B が A と同じ場合、C も A となる。
3. A と B それぞれの特徴を反映させる度合いを変化させることで、C が複数出力される。
4. A と B がいずれもモノフォニー（和音を含まない単旋律）の場合、C もモノフォニーとなる。

条件1と2は、メロディモーフィングに限らず、画像等のモーフィングにも一般にあてはまる条件である。一方、条件3と4は、メロディモーフィングの条件として我々が新たに考えたものである。

タイムスパン木は GTTM による楽曲構造分析の結果求

まったものである。GTTMは、グルーピング構造分析、拍節構造分析、タイムスパン簡約、プロロンゲーション簡約という4つのサブ理論から構成されており、楽譜に分析を加えることで、楽譜に内在する様々な階層構造を深層構造として顕在化させる。タイムスパン簡約は、あるメロディを簡約化することによって、そのメロディの装飾的な部分が削ぎ落とされ、本質的なメロディが抽出されるという直観を表したもので、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)を求める分析である(図4)。

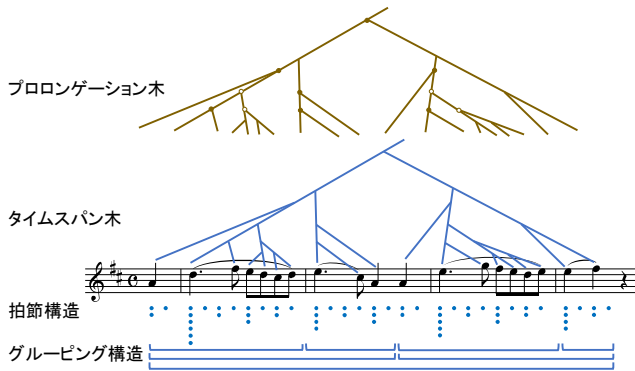


図4 グルーピング構造、拍節構造、タイムスパン木、プロロンゲーション木

### 3.1 メロディの簡約

図5は、タイムスパン木を用いたメロディの簡約の例である。図のメロディAの上にある木構造は、メロディAをタイムスパン簡約した結果得られたタイムスパン木である。タイムスパン木のレベルBより下にある枝の音符を簡約(省略)するとメロディBようになる。さらに、レベルCより下にある枝の音符を簡約するとメロディCようになる。このとき、メロディBはメロディAとCの間のメロディであることから、メロディの簡約もメロディのモーフィングの一種と考えることができる。

### 3.2 タイムスパン木に基づく基本演算

メロディのモーフィングを実現するため、文献[10-13]で定義されている演算である、包摂関係 $\sqsubseteq$ 、meet(最大下界) $\sqcap$ とjoin(最小上界) $\sqcup$ を用いる。包摂関係 $\sqsubseteq$ は、F1を下位の構造、F2を上位の構造(下位の構造を含んでそれ以上の構造を持つ)としたとき、 $F1 \sqsubseteq F2$ と表記し、F2はF1を包摂すると言う。

たとえば、図5のメロディA、B、Cのタイムスパン木(簡約されたタイムスパン木)、 $T_A$ 、 $T_B$ 、 $T_C$ の包摂関係は、以下のように表せる。

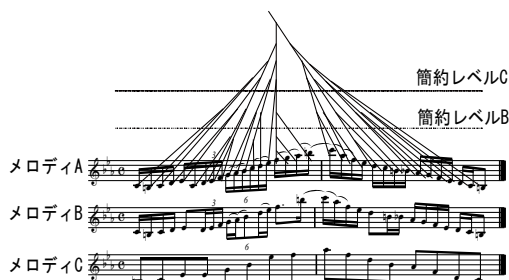


図5 メロディの簡約

$$T_C \sqsubseteq T_B \sqsubseteq T_A$$

meet(最大下界)は、 $T_A$ 、 $T_B$ の共通部分のタイムスパン木  $T_A \sqcap T_B$ である。join(最小上界)は、メロディA、Bのタイムスパン木  $T_A$ 、 $T_B$ が矛盾を起こさない限り統合したタイムスパン木  $T_A \sqcup T_B$ である(図6)。

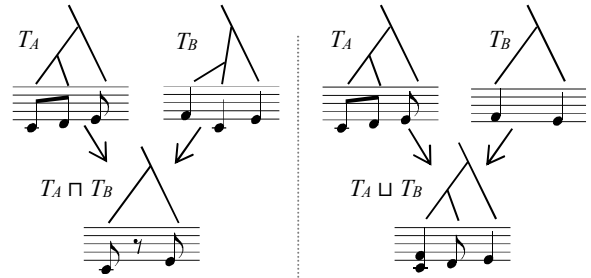


図6 meet  $\sqcap$ とjoin  $\sqcup$ の例

### 3.3 GTTMに基づくメロディモーフィング手法

モーフィングという用語は、通常、二つの画像がある場合に、片方の画像からもう一つの画像へ滑らかに変化していくよう、その間を補うための画像を作成することをいう。2次元の顔画像のモーフィングの場合、たとえば以下のような操作で中間的な画像の生成が実現できる。

- 1) 目や鼻など2つの画像の特徴点の対応づけ(図7a)。
- 2) 各画像の形状(位置)や色の重み付け。
- 3) 2つの画像の重ね合わせ。

一方、本研究で提案するメロディモーフィングでは、以下のような操作で中間的なメロディの生成を実現する。

- 1) 2つのメロディの共通部分の対応づけ(図7b)。
- 2) 各メロディについてメロディの部分簡約。
- 3) 両方のメロディの重ね合わせ。

#### (1) メロディの共通部分の対応づけ

2つのメロディA、Bのタイムスパン木  $T_A$ 、 $T_B$ を求め、その共通部分(最大下界)  $T_A \sqcap T_B$ を求める。これにより、タイムスパン木  $T_A$ 、 $T_B$ は、それぞれ共通部分と非共通部分に分けることができる。



図7 2つの画像/メロディの対応づけの例

$T_A \sqcap T_B$ は、タイムスパン木  $T_A$ 、 $T_B$ をトップダウンに見て、最も大きく共通する部分を取り出すが、その際、オクターブの異なる2音(たとえばC4とC3)を異なる音とみなす場合と、同じ音とみなす場合で結果が異なる。異なる音とみなす場合には、 $C4 \sqcap C3$ の解は空疎 $\perp$ となる。一方、同じ音とみな



す場合には、オクターブの情報が捨象され解はCとなる。オクターブ情報が未定義の場合、(2)以降の処理が困難となることから、オクターブの異なる2音は、異なる音として扱うことにした。

## (2) メロディの部分簡約

(1) で求めたメロディ  $A$  と  $B$  のタイムスパン木  $T_A, T_B$  の非共通部分には、それぞれ相手のメロディにはない特徴が表われていると考えられる。したがって、メロディのモーフィングを実現するためには、それら非共通部分の特徴をなめらかに増減させ、中間的なメロディを生成するような手法が必要となる。

本研究では3.1で述べたメロディの簡約を応用し、メロディの非共通部分についてのみ簡約を実行するメロディ部分簡約法を提案する。メロディ部分簡約法では、メロディ  $A$  のタイムスパン木  $T_A$  と、メロディ  $A, B$  のタイムスパン木の共通部分  $T_A \cap T_B$  から、次のアルゴリズムでメロディ  $C_m$  ( $m=1,2,\dots,n$ ) を生成する。 $C_m$  の添え字  $m$  は、 $C_m$  に含まれるが、 $T_A \cap T_B$  には含まれない部分(メロディの非共通部分)に表われる音符の数である。

図8の場合では、 $T_A$  には、 $T_A \cap T_B$  にはない音符が9個含まれているため、 $T_A$  と  $T_A \cap T_B$  の間のメロディが8種類得られることになる。

上記のようにして求めたメロディ  $C$  は、メロディ  $B$  にはないメロディ  $A$  のみが持つ特徴の一部を減衰させたものと考えられる。同様に、 $T_B$  と  $T_A \cap T_B$  から下記を満たすメロディ  $D$  を生成する。

$$T_A \cap T_B \not\subseteq T_D \not\subseteq T_B$$

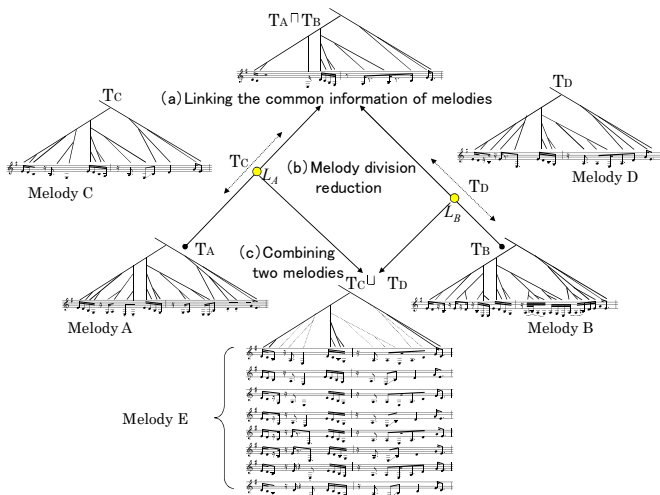


図8 メロディモーフィング手法

## (3) メロディの合成

メロディ  $A$  と  $B$  のそれぞれのタイムスパン木の一部を簡約したメロディ  $C$  とメロディ  $D$  を統合(最小上界)し、合成したメロディ  $E$  を生成する。

## (4) 音楽家による調整

メロディモーフィング手法は自動化を目指しているが、現時点では音楽家による作業なしでは実現できない。5節の実験では、音楽家は以下の3つの作業を行う。第一の作業では、GTTMに基づきメロディの構造分析を行い適切な

タイムスパン木を取得する。第二の作業では、3.2節(2)のメロディの部分簡約において音符を抽象化する順序を決定する。第一と第二の作業の後、 $[N1, N2]$ のようになっている箇所では、どちらかを選択していくことで、モーフィングメロディは自動的に取得可能である。しかし、この方法でメロディをモーフィングすると、メロディのバリエーションごとの変化が極めて少なくなることがある。そこで、メロディの変化が少なかった場合、第三の作業として、音楽家が手動でメロディの調整を行う。調整の詳細については5節で説明する。

## 4. 実装

ペッパーゴーストディスプレイ (Realfiction Dremoc HD3) は、半透過フィルムの貼られた3枚のガラスがデバイス上部に下向きに設置したディスプレイの映像を反射することで、ペッパーズゴーストにより3つの方向で異なる映像を表示することが可能である。

ユーザは、タブレット (iPad Pro 12.9) 上に表示されたスロットダイヤルと譜面台に取り付けられたスロットレバーを操作する。タブレットとメインのコンピュータ (Apple Mac mini 2018) は、MIDI インタフェースを介して通信している。どのメロディを再生するかを決定するスロットダイヤルのダイヤル番号は、随時タブレットからコンピュータに送信される。逆に、コンピュータは現在の再生位置をタブレットに通知し、再生位置がタブレット上に表示される。スロットレバーはレバーを倒している時だけONになるスイッチであり、スイッチの接続状態はマイコン (Arduino) を使用して検出する。検出した結果は、Arduino に接続した MIDI シールドと MIDI インタフェース (iConnectMIDI4+) を介してコンピュータに送られる。

### 4.1 音楽音響処理

使用したペッパーゴーストディスプレイは半透過フィルムの貼られたガラスが45度の角度で設置されているため、水平な高さから見た場合が最も臨場感が高く感じられるようになっている。逆に、上方や下方から見た場合には、ペッパーズゴーストによる映像であることがわかってしまいやすい。そこで、方向ごと水平となる高さにスピーカ対を設置して、ユーザが自然にスピーカの間に入頭を入れて聴くことで、ペッパーゴーストディスプレイに対して適切な位置で視聴できるようにした。

システムは、開発環境 Max/MSP 上に構築した。サウンドファイルは全バリエーションを同時に再生するが、選択されたスロットの番号に対応したバリエーションの音量だけを上げるようにした。つまり、選択されていないバリエーションは音量を0とする。その際、メロディが切り替わる10ミリ秒前から10ミリ秒後までに、フェードインおよびフェードアウトを入れることで滑らかに連結されるようにした。これにより、ユーザがタブレットでダイ

ヤルを設定すると、指定したメロディが順次出力される。

## 4.2 映像処理

映像処理では、ユーザが設定したスロット番号に従って映像を再生する。

### (1) 映像ファイルの同時再生

音楽音響の処理と同じように、すべてのバリエーションのファイルを同時再生して HDMI に出力される映像を切り替えることを試みたが、映像ファイルのサイズはオーディオに比べて巨大で不可能であった。

### (2) 映像断片の再生

映像を断片に分割し、スロット番号に対応した映像ファイルを次々再生することを試みたが、API などが提供されている多くの映像再生エンジンではファイルを開き再生を開始するまでに 0.5 秒以上の時間がかかるという問題が生じた。この場合、映像の再生開始が予定される 0.5 秒前にファイルを開き、映像ストリームの再生を準備する必要があるが、その 0.5 秒の間にもユーザはスロットを操作する可能性があり、次に再生されるスロットの番号が変化すれば、タブレット上に表示された内容と、実際の演奏音に食い違うという問題が生じた。そこで映像再生エンジンを新たに構築し 0.5 秒以内に処理が終わるようにした。

### (3) 補完映像の挟み込み処理

メロディ断片に対応する映像を接続して再生すると、映像と映像の境界で、演奏者の動きがなくなってしまう。そこで、前の映像の終わり 10 フレームと、後の映像の開始 10 フレームを削除し、そこに 20 フレームの補完映像を挟み込む。補完映像は、2 つの映像が滑らかにつながるようにあらかじめ深層学習により生成したものである。

## 5. 実験結果

3 名の音楽家にモーフィングメロディの作成を依頼した。1 名の音楽家は、GTTM 分析について 10 年以上の経験があり、モーフィング手法についても深い理解がある。残りの 2 名は GTTM 分析の経験は 2 年である。上記 3 名は分析結果のクロスチェックを行っており、ほぼ同等の能力であることを確認している。モーフィングの元となるメロディの長さや曲名は指定せず、音楽家が自身で考え制作を行った。

### 5.1 モーツァルト きらきら星変奏曲 K. 265/300e

10 年以上の GTTM 分析の経験を持つ音楽家 A は、モーツァルト きらきら星変奏曲 K. 265/300e のテーマと変奏曲 No. 1 をモーフィング元のメロディとして、9 個のモーフィングメロディを作成した。

表 1 は、総音符数、モーフィング手法で生成された音符数、音楽家により追加された音符数を示している。音楽家により追加された音符は以下のいずれかであった。

- 倚音、刺繍音、経過音の追加
- タイムスパン木で隣接する枝からの借用音もしくは枝内の音順変更

- 1 つの音符を同じ音程の 2 つの音符に時系列方向の分割
- メロディの拡大あるいは縮小
- メロディの引用

メロディ 1 および 2 のすべての音符はモーフィング手法により生成された。これは、1 も 2 も、テーマと変奏曲 No.1 の共通部分(meet)と、情報が近く、メロディの合成(join)の結果、[N1, N2]のような演算子が付いている音符からどちらかを選ぶだけでメロディが完成したためである。また、部分簡約により適度に音符数が減った 2 つのメロディを合成して生成されたメロディ 5 の全ての音符もモーフィング手法により生成された。

テーマに近い、モーフィングメロディ 1, 2, 3 では平均 1.5% の音符が音楽家により追加されたのに対し、変奏曲 No.1 に近いモーフィングメロディ 8, 9 では、音楽家により 37.9% の音符が追加された。これは、8 分音符の多いメロディと 16 分音符の多いメロディを合成した結果として不自然なメロディとなった場合に、調整が必要であったためである。

表 1 各メロディの音符数

	総音符数	手法により生成された音符数	音楽家が追加した音符数
テーマ	82	-	-
メロディ 1	51	51 (100%)	0 (0%)
メロディ 2	51	51 (100%)	0 (0%)
メロディ 3	93	90 (96.8%)	3 (3.2%)
メロディ 4	121	96 (79.3%)	25 (20.7%)
メロディ 5	94	94 (100%)	0 (0%)
メロディ 6	157	129 (82.2%)	28 (17.8%)
メロディ 7	176	157 (89.2%)	19 (10.8%)
メロディ 8	253	164 (64.8%)	89 (35.1%)
メロディ 9	267	159 (59.6%)	108 (40.4%)
変奏曲 No. 1	271	-	-

### 5.2 モーツァルト ホルン協奏曲第 1 番およびポンキエリ 歌劇「ラ・ジョコンダ」時の踊り

音楽家 B は、ホルン協奏曲と時の踊りのモーフィングメロディを作成した。ホルン協奏曲、時の踊りおよび 9 通り生成したモーフィングメロディのうち 4 つについて、生成された曲を再度分析した。そして、分析の結果得られたタイムスパン木を用いてモーフィングメロディ  $M$  がモーフィング元の 2 つのメロディの中間に位置するかを確認した。 $R(A, B)$  を曲  $A$  と  $B$  との類似度とすると、 $M$  が中間に位置すると次式が成立する。

$$\{R(A, M) < R(A, B) \text{ and } R(A, M) < R(B, M)\}, \text{ or } \{R(B, M) < R(A, B) \text{ and } R(B, M) < R(A, M)\}, \quad (1)$$

類似度は平田ら[11]が提案した次式を用いた。

$$R_N(X, Y) = \frac{|meet(X, Y)|_N}{\max(|X|_N, |Y|_N)}, \quad (2)$$

この式は、meetにより2つのメロディからどれだけ情報が減少したかを示す。 $|X|_N$ はXに含まれる音符数を表す。

ホルン協奏曲、時の踊り、および9個のモーフィングメロディについて確認したところ、すべての組み合わせで式(1)が成立した。これにより、3節冒頭で述べた条件1「AとBよりAとCのほうが類似している、かつ、AとBよりBとCのほうが類似している」が成立していることが確認できた。

### 5.3 リスト ラ・カンパネラ

音楽家Cは、5個のモーフィングメロディ(M1, M2, M4, M5, M6)をリストのラ・カンパネラから生成した。音楽家Cのモーフィング手法の使い方は、AとBとは異なり、M3がラ・カンパネラがとなるよう残りの5つのメロディを構成した。すなわち、M1からM6の間には以下のような包摂関係が成立する。

$$M1 \sqsubseteq M2 \sqsubseteq M3 \sqsubseteq M4 \sqsubseteq M5 \sqsubseteq M6 \quad (3)$$

式(6)の包摂関係がすべての枝(音符)について成立することが確認できた。

## 6. おわりに

本稿では、メロディスロットマシンの構築について説明した。概要映像は以下に公開している。

<https://www.youtube.com/watch?v=DE8j6DtAU5g>

本研究の貢献は以下の5点である。

1. スロットダイヤルやレバーなど、誰でも簡単に操作できるインタフェースにより、ユーザが仮想演奏者の未来の演奏をコントロール可能なシステムを開発した。
2. 従来の仮想演奏者は、音の出力のみであるか、コンピュータグラフィックによる表示であったのに対し、仮想演奏者をペッパーゴーストディスプレイに表示し、臨場感の高い音楽システムを構築した。
3. メロディモーフィング手法はこれまで作曲に使用されたことはなかったのに対し、本研究では3名の音楽家がメロディモーフィング手法を使用して制作を行った。
4. 2つのメロディの類似度 $R_N(X, Y)$ に基づいてモーフィングメロディを評価した。AとBからモーフィングメロディCを生成した場合に、AとCの類似度はAとBの類似度より高く、BとCの類似度はAとBの類似度より高いことを確認した。

今後、メロディスロットマシンのコンテンツを充実していく予定である。また、GTTMを使用してあたかも人間のように作曲や演奏ができる仮想ミュージシャンの実現を目指していく。

**謝辞** 本研究の一部は、JSPS 科研費 17H01847, 16H01744

の助成を受けたものです。

## 参考文献

- [1] Fred Lerdahl and Ray S. Jackendoff, 1983. A Generative Theory of Tonal Music, MIT Press, Cambridge, Massachusetts, USA.
- [2] Masatoshi Hamanaka, Keiji Hirata, and Satoshi Tojo. 2006. Implementing “A generative theory of tonal music,” Journal of New Music Research 35 (4), 249–277. DOI: 10.1080/09298210701563238
- [3] Masatoshi Hamanaka, Keiji Hirata, and Satoshi Tojo. 2016. σGTTM III: Learning-Based Time-Span Tree Generator Based on PCFG, Lecture Notes in Computer Science, Vol. 9617, pp. 387–404. DOI: 10.1007/978-3-319-46282-0\_25
- [4] Masatoshi Hamanaka, Keiji Hirata, and Satoshi Tojo. 2018. deepGTTM-III: Multi-task Learning with Grouping and Metrical Structures, Lecture Notes in Computer Science, Vol. 11265, pp. 238–251. DOI: 10.1007/978-3-030-01692-0\_17
- [5] Belinda Thom, 2003. Interactive Improvisational Music Companionship: A User-Modeling Approach, The User Modeling and User-Adapted Interaction Journal, Special Issue on User Modeling and Intelligent Agents, 13(1): 133–177.
- [6] Rodolfo Daniel Wulforst, Lauro Nakayama, and Rosa Maria Vicari, 2003. A Multiagent Approach for Musical Interactive Systems, In AAMAS '03: Proceedings of the second international joint conference on autonomous agents and multiagent systems, pp. 584–591. DOI: 10.1145/860575.860669
- [7] Gérard Assayag, Georges Bloch, Marc Chemillier, Arshia Cont, and Shlomo Dubnov, 2006. OMax brothers: a Dynamic Topology of Agents for Improvisation Learning, In AMCMM '06: Proceedings of the 1st ACM workshop on audio and music computing multimedia, pp. 125–132. DOI: 10.1145/1178723.1178742
- [8] Francois Pachet, 2003. The Continuator: Musical Interaction with Style, Journal of New Music Research, 32(3): 333–341.
- [9] Masatoshi Hamanaka, Masataka Goto, Hideki Asoh, and Nobuyuki Otsu, 2003. A Learning-Based Jam Session System that Imitates a Player’s Personality Model, In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI2003), vol. 18, pp. 51–58.
- [10] Masataka Goto, Isao Hidaka, Hideaki Matsumoto, Yosuke Kuroda, and Yoichi Muraoka, 1996. A Jazz Session System for Interplay among All Players – VirJa Session (Virtual Jazz Session System), In Proceedings of the 1996 International Computer Music Conference (ICMC1996), pp. 346–349.
- [11] Keiji Hirata and Shu Matsuda, 2003. Interactive Music Summarization based on Generative Theory of Tonal Music, Journal of New Music Research, 32(2): 165–177.
- [12] Keiji Hirata and Rumi Hiraga, 2003. Ha-Hi-Hun plays Chopin’s Etude. In Working Notes of IJCAI-03 Workshop on methods for automatic music performance and their applications in a public rendering contest, 2 pages.
- [13] Masatoshi Hamanaka, Keiji Hirata, and Satoshi Tojo, 2008. Melody Expectation Method Based on GTTM and TPS, In Proceedings of the 9th International Conference on Music Information Retrieval conference (ISMIR2008), pp. 107–112.
- [14] Masatoshi Hamanaka, Keiji Hirata, and Satoshi Tojo, 2008. Melody Morphing Method Based on GTTM, In Proceedings of the 2008 International Computer Music Conference (ICMC2008), pp. 155–158.
- [15] Masatoshi Hamanaka, Keiji Hirata, and Satoshi Tojo, 2009. Melody extrapolation in GTTM approach, In Proceedings of the 2008 International Computer Music Conference (ICMC2009), pp. 89–92.
- [16] Stephen A. Hedges, 1978. Dice Music in the Eighteenth Century, Music & Letters, 59(2): 180–187.