

## 事例に基づく演奏表情生成システムにおける 旋律断片自動生成アルゴリズムの改良と評価

清水 厚志<sup>†</sup> 鈴木 泰山<sup>†</sup> 野池 賢二<sup>††</sup> 金子 雄介<sup>‡</sup> 徳永 幸生<sup>†</sup> 杉山 精<sup>†</sup>

本報告では、事例に基づく演奏表情生成システムにおける楽譜の旋律と和音の構造を考慮した旋律断片自動生成アルゴリズムを提案する。具体的には、旋律と和声といった音楽の要素を楽譜上の演奏記号の情報を利用して構造化し、演奏事例となる旋律断片を生成する。また、本アルゴリズムと従来のアルゴリズムを用いたシステムの出力結果を統計的に分析し、それぞれを比較検討した結果について述べる。

### Improvement and Evaluation of Automatic Music Piece Segmentation Algorithm in a Case-Based Performance Rendering System

Atsushi SHIMIZU<sup>†</sup> Taizan SUZUKI<sup>†</sup> Kenzi NOIKE<sup>††</sup>  
Yusuke KANEKO<sup>‡‡</sup> Yukio TOKUNAGA<sup>†</sup> Kiyoshi SUGIYAMA<sup>†</sup>

This report proposes algorithm which analyzes structure of melody and harmony, and generates musical pieces automatically in a Case-Based Performance Rendering System. Specifically, it structure such musical elements as melody and harmony using symbol on the score and generates musical pieces as case of playing. We compared and evaluated this algorithm and conventional algorithm.

#### 1. はじめに

計算機による情緒あふれる演奏の自動生成は音楽情報科学の主要なテーマであり、表情付き演奏の自動生成システムが 1980 年代後半から多数発表されている。演奏表情生成手法は、規則に基づく手法と事例に基づく手法に大別される。事例に基づく演奏表情の自動生成では、あらかじめ用意された演奏データ集の中から対象曲に類似した楽曲を検索し、それらに見られる演奏表情を対象曲に適用することで演奏データを生成する。演奏表情を適用する際には、検索した演奏事例の中から対象曲に最も類似する事例を選択し、対象曲に適用する必要がある。そのため、演奏事例の品質は事例に基づく演奏表情生成システムの出力の品質に直接影響する。

現在我々が開発を進めている事例に基づく演奏表情生成システム“Kagurame Phase-II” [1][2]では、演奏事例として、楽譜を階層的に分割した“旋律断片”を用いる。

小節の分割方法として、人手で明示的に境界情報を与える方法と、小節構造を旋律断片自動生成アルゴリズムによって解析し、自動的に分割する方法が挙げられる。現在の Kagurame Phase-II では、後者を採用している。事例の選択は旋律断片単位で行われるため、旋律断片は楽譜の構造を反映していることが望ましい。また、旋律断片自動生成アルゴリズムの良し悪しが、演奏表情生成システムの能力に影響すると考えられる。本報告では、楽譜のまとまりを示す記号を利用し、楽譜の旋律と和声の構造を考慮した旋律断片自動生成アルゴリズムを提案する。また、本アルゴリズムと従来のアルゴリズムについて演奏表情生成の出力結果を分析し、比較検討した結果について述べる。

以下、第 2 章で事例に基づく演奏表情システム Kagurame Phase-II について概説する。第 3 章で、楽譜記述用の言語“MusicXML”[6]について概説する。第 4 章で、音符のまとまりを示す楽譜上の記号を利用した、MusicXML からの旋律断片自動生成について述べる。第 5 章・第 6 章で、今回実装した旋律断片自動生成アルゴリズムと従来のアルゴリズムによる演奏表情生成の比較と考察を行う。

#### 2. Kagurame システム

本章では、我々が作成した事例に基づく演奏表情生成システム Kagurame Phase-II の、概要および構成について説明する。

<sup>†</sup> 芝浦工業大学大学院 工学研究科  
電気電子情報工学専攻  
Department of Electrical Engineering and computer  
Science, Graduate School of Engineering,  
Shibaura Institute of Technology

<sup>‡</sup> 株式会社ピコラボ  
PicoLab Co., Ltd.

<sup>††</sup> 株式会社トランス・ニュー・テクノロジー  
Trans New Technology, Inc.

<sup>‡‡</sup> 株式会社日本総合研究所  
The Japan Research Institute, Limited

## 2.1 特徴

Kagurame Phase-II は、鍵盤楽器による複旋律の楽曲の対象とした演奏表情システムである。本システムでは演奏状況を加味した演奏表情の生成が可能であるが、本報告では演奏状況については対象としない。本システムの特徴として、対象曲や演奏事例を様々な長さの旋律断片に分割し、旋律断片を対象として参考事例の検索を行うことが挙げられる。これにより、事例の総数が増加するため、少数の演奏事例を効率的に利用することが可能である。また、旋律断片に分割することで個々の検索単位の長さが短くなるため、類似旋律の検索に成功する可能性が向上する。

## 2.2 構成

Kagurame Phase-II の構成を図 1 に示す。本システムでは、演奏表情の知識として演奏データ集を利用する。演奏データ集は、人間が演奏した実演奏を集めたものである。個々の演奏データは、演奏曲情報と演奏データからなる。演奏曲情報は、楽譜情報と境界情報からなる。楽譜情報は MusicXML で与えている。演奏データは、標準 MIDI フォーマット形式(SMF)のシーケンスデータで与えている。演奏表情の生成の際には、対象曲情報を入力として与える。対象曲情報は演奏曲データと同形式である。楽曲構成情報は、対象曲を旋律断片に分割する際に使用する。入力が与えられると、まず、演奏データ集から対象曲に類似した楽曲を検索する。類似した楽曲の検索は、対象曲全体だけではなく、対象曲を構成するフレーズや小節などの旋律断片に対しても行う。これによって、対象曲の旋律断片ごとに類似した旋律断片の集合が得られる。この集合を参考事例集とする。

次に、参考事例集の中の全ての事例について、重要度を評価する。重要度は、対象曲の演奏表情を生成する際に、その事例がどの程度参考になるかを表すスコアである。事例の重要度は、対象曲と事例との旋律断片の類似性から決定する。入力と参考事例とが類似しているほど、重要度は高くなる。旋律断片の事例の重要度は、1) 対象曲の断片との類似性、2) 前後に隣接する断片どうしの類似性、3) 親となる断片どうしの類似性、から求める。旋律断片の重要性は、参考事例の検索と同じ方法で評価する。旋律の重要性を正規化して、指数関数に適用したものを重要度とした。また、重要度の評価と同時に、全ての参考事例に対して演奏表情の分析を行う。演奏表情は、演奏データと楽譜とのずれという形で取り出すことができる。本手法では演奏表情の生成の際に、テンポや音の強さなどの絶対的な数値ではなく、これを相対的な変化量の比率に変換したものを利用している。この相対的な変化量を、演奏表情の大局比率と呼ぶ。

それぞれの参考事例に対して、重要度の評価と演奏表情の大局比率の分析を行うと、重要度でスコア付けされた演奏表情の大局比率の集合が得られる。この集合の中における個々の演奏表情の大局比率を重要度で加重平均して合成し、対象曲の演奏表情を生成する。最後にこの対象曲の演奏表情を対象曲の楽譜に適用し、演奏表情の与えられた対象曲の演奏データを作成する。生成された演奏データは、SMF形式のシーケンスデータで保存する。

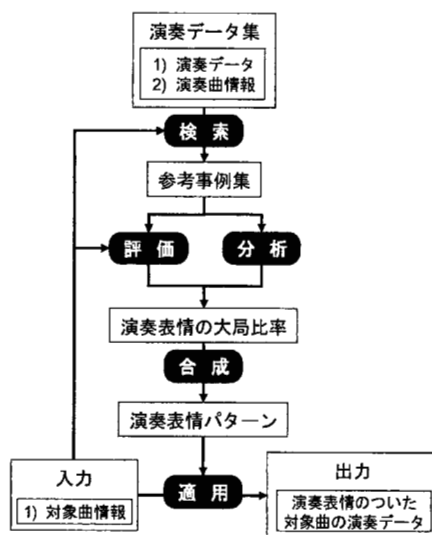


図 1 "Kagurame Phase-II" の構成

Kagurame Phase-II が類似事例を検索するときに必要なとする旋律断片の大きさはさまざまであるが、基本的には小節及び拍の 2 の累乗の長さである。例えば、1 小節、2 小節、4 小節、8 小節や、1 拍、2 拍、4 拍、8 拍、さらに半拍や 1/4 拍、1/8 拍などを分割の単位とする (図 2)。従って、ほとんどの断片は二つのより小さい断片を内包している。もっとも小さい断片は 1 音符である。

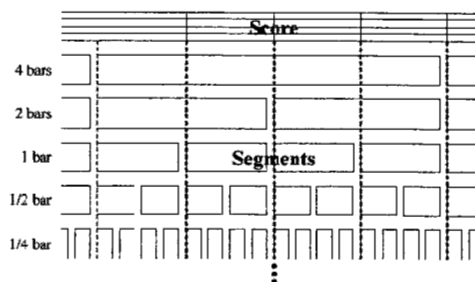


図 2 旋律断片への分割

### 3. MusicXML

MusicXML は、Recordare 社によって定義された、一般的な西洋音楽楽譜を記述するための XML 定義である。MusicXML は休符を明示的に記述することやスラーやタイなどの演奏情報を記述することが可能であるなど、一般的な西洋音楽の楽譜を記述するには十分な記述力をもっている。また、MusicXML はテキストファイルなので、テキストエディタで容易に閲覧・編集できるといった利点がある。

MusicXML の基本的な構造は、図 3 に示すように、パートを表わす <part>タグの下に小節を表わす <measure>、<measure>の下に音符を表わす <note>を列挙するという構造になっている。

MusicXML では、音符を表す <note>要素は <measure>要素の下に楽譜の音符の出現順に列挙されている。また、<measure>要素と違い、ID を持たない。<note>要素が何拍目であるかは、<note>要素の出現順と、それらの <note>要素の子である <duration>要素に依存する。したがって、「1 小節目の 1 拍目」のように、拍単位で <note>要素を直接に指定することはできない。

```
<-part id="P1">
  <-measure number="1">
    <+attributes>
    .....
    <+note>
    <+note>
    .....
  <-measure number="2">
    .....
```

図 3 MusicXML の基本的な構造の概略

<duration>要素は、この音符の実際の長さを、MusicXML の divisions (MusicXML 内部の解像度)の値に対する相対的な値として表している。

例えば、divisions 値が 24 のとき、ある <note>要素の <duration>要素の値が 12 の場合、その <note>要素が表す音符の長さは 4 分音符の  $12/24=0.5$  倍の長さ、すなわち 8 分音符の長さとなる。

MusicXML をサポートするソフトウェアは、市販品を含め、充実しつつある。中でも、カメオインタラクティブ社の楽譜作成ソフト“Finale”用に Recordare 社が提供する“Dolet for Finale”の完成度が高い。

### 4. 旋律断片自動生成アルゴリズム

Kagurame Phase-II は、演奏事例の検索や表情の適用を、楽曲を分割した旋律断片単位に行うことを特徴としている。そのため、楽曲をどのように旋律断片に分割するかによって、演奏表情の分析や適用が変化し、生成される演奏表情が異なると考えられる。

### 4.1 入力データの構成

現在の Kagurame Phase-II では、入力となる対象曲情報と演奏事例データベースとなる演奏データ集は、図 4 のように構成されている。

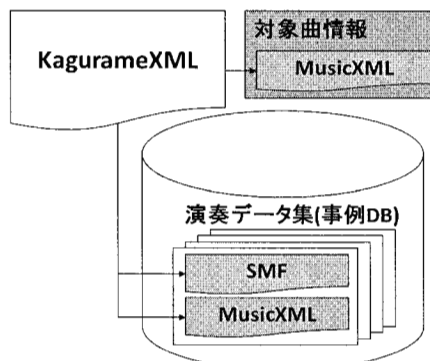


図 4 Kagurame Phase-II で使用するデータの構成

生成対象曲および演奏データ集は、Kagurame XML を用いて指定する。KagurameXML は以下の図 5 に示す文書型定義 (DTD, Document Type Definition) によって定義される。

```
<!ELEMENT kagurame (target, sampledataset)>
<!ELEMENT target (#PCDATA)>
<!ELEMENT sampledataset (sample+)>
<!ELEMENT sample (structure, smf)>
<!ELEMENT structure (#PCDATA)>
<!ELEMENT smf (#PCDATA)>
```

図 5 KagurameXML の文書型定義

<target>要素は、対象曲を表す。<sampledataset>要素は、演奏データ集を表す。

KagurameXML の記述例を図 6 に示す。

```
<?xml version="1.0" standalone="no"?>
<kagurame>
  <target>(演奏表情生成対象曲のMusicXMLのパス)</target>
  <sampledataset>
    <sample>
      <structure>(DB1曲目のMusicXMLのパス)</structure>
      <smf>(DB1曲目のSMFのパス)</smf>
    </sample>
    <sample>
      <structure>(DB2曲目のMusicXMLのパス)</structure>
      <smf>(DB2曲目のSMFのパス)</smf>
    </sample>
  </sampledataset>
</kagurame>
```

図 6 KagurameXML の記述例

従来アルゴリズムでは、小節のまとまりを示す情報を『DividingXML』として与えていたが[13]、本アルゴリズムによって境界情報をすべて自動的に

生成できるように改良したため、いままで手作業で境界情報を与えるために用いていた DividingXML は、与える必要がなくなった。

#### 4.2 楽譜のグルーピング構造化

現在、Kagurame は楽譜情報として MusicXML を使用している。あらかじめ与えた境界情報を用いて小節単位の旋律断片を生成し、小節より小さい旋律断片は、小節内の音符の出現位置に応じて自動生成するという手法をとっている。

しかし、この手法では、旋律断片の構造が小節の区切りに強く依存するため、小節をまたぐ旋律の流れがうまく表現できない。

また、右手と左手の音符を区別せずに旋律断片を生成しているため、現在の旋律断片の構造は、楽曲の和声の構造を正確に表現しているとは言えないと考えられる。

これらの問題点を改善し、旋律の構造を反映した旋律断片を生成することによって、出力の質が向上するのではないかと、という仮説を立てた。

仮説を検証するために、以下の方針でアルゴリズムを構築した。

一般的な西洋音楽の楽譜では、「音符のまとまり」を示す記号が使用されている。これらの記号は一つのフレーズを表していることが多く、旋律断片を生成する上での指標として有用であると考えられる。

音符のまとまりを示す記号には以下のようなものがある(図7)。

- (1) スラー：いくつかの音符を弧でくくり、音と音とを滑らかにつなげて演奏することを表す演奏記号
- (2) 連桁(れんこう)：8分音符より細かい音符が連なる時に符尾をつなげて書いたもの



図7 スラーと連桁

MusicXML では、連桁を表す<beam>要素は音符を表す<note>要素の子要素、スラーを表す<slur>要素は音符の装飾を表す<notations>要素の子要素として定義されている(図8)。

これらの情報を用いて MusicXML から旋律断片の木構造を構築するように、アルゴリズムの改良を行った。

```
<note>
  <pitch>
    <step>D</step>
    <octave>6</octave>
  </pitch>
  <duration>1</duration>
  <voice>1</voice>
  <type>eighth</type>
  <stem>down</stem>
  <staff>1</staff>
  <beam number="1">begin</beam>
  <notations>
    <slur type="start" number="1"/>
  </notations>
</note>
```

図8 MusicXML のスラーと連桁を表すタグ

#### 4.3 楽譜上の記号を用いたアルゴリズムの実装

- ① 楽譜を右手部分と左手部分に分割する。本アルゴリズムでは、楽譜の上段を右手、楽譜の下段を左手とみなして分割を行う(図9)。



図9 右手部分と左手部分の分割

- ② 右手部分と左手部分それぞれについて音符を走査し、各音符1つ分に相当する旋律断片を生成する。
- ③ 連桁を一つのまとまりとみなし、連桁で接続された音符を子要素とした旋律断片の部分木を構築する。
- ④ スラーの始点から終点までに含まれる旋律断片を内包する旋律断片の部分木を生成する。
- ⑤ ①から③で生成した旋律断片の部分木を包含する。
- ⑥ 以降、部分木の結合を繰返し、最終的に楽譜全体を内包する木構造をボトムアップで構築する。

従来のアルゴリズムと改良したアルゴリズムによって生成された旋律断片の比較を図10、図11に示す。

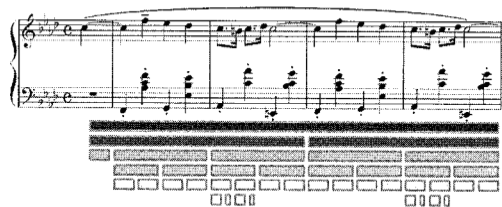


図10 従来のアルゴリズムによる旋律断片



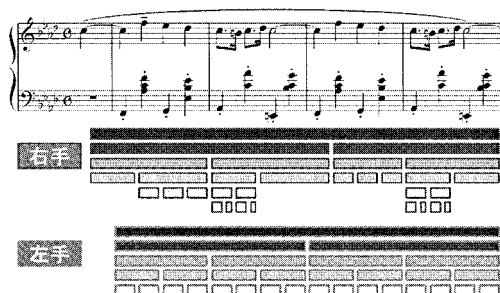


図 11 改良したアルゴリズムによる旋律断片

## 5. 生成結果の比較

旋律断片自動生成アルゴリズムを評価するために、改良前と後のそれぞれのシステムが生成した演奏を用いて、定量評価を行った。また、評価には以下の曲を用いた。

- ・ L.ボッケリーニ「メヌエット」
- ・ J.S.バッハ「メヌエット ト長調」
- ・ E.ローデ「あやつり人形」
- ・ R.シューマン「兵士の行進」
- ・ R.シューマン「メロディ」

システムが生成した演奏と人間の演奏との相異度  $ratio$  を、以下の式(1)により算出する。

$$ratio = \frac{1}{n} \sum_{i \leq n} \left| \ln \frac{E_{kp2}(i)}{E_h(i)} \right| \quad (1)$$

$E_{kp2}(i)$  はシステムが生成した  $i$  番目の音符の演奏表情の値であり、 $E_h(i)$  は人間の演奏の  $i$  番目の音符の演奏表情の値である。また、 $n$  は楽曲に含まれる音符の総数である。相異度  $ratio$  の値が小さいほど、DB をより正確に反映した演奏を生成しているといえる。なお、演奏表情の値は音符の打鍵強度 (Velocity) を用いている。

対象曲と同一曲のみを演奏事例 DB として用いた Closed-Test における相異度  $ratio$  を表 1 に示す。

表 1 Closed-Test における相異度  $ratio$  の比較

曲名	従来方式	改良方式
メヌエット(ボッケリーニ)	0.090	0.104
メヌエット(バッハ)	0.089	0.060
あやつり人形	0.061	0.073
兵士の行進	0.070	0.069
メロディ	0.065	0.054

また、交差検定法を用いた Open-Test における  $ratio$  の値の比較を表 2 に示す。演奏事例 DB には、表中の曲から対象曲以外の 4 曲を使用している。

表 2 Open-Test における相異度  $ratio$  の比較

曲名	従来方式	改良方式
メヌエット(ボッケリーニ)	0.110	0.120
メヌエット(バッハ)	0.102	0.078
あやつり人形	0.086	0.090
兵士の行進	0.091	0.084
メロディ	0.082	0.073

## 6. 考察

実験結果から、 $ratio$  の値が低下し、出力の品質が向上している曲もあれば、『メヌエット (ボッケリーニ)』や『あやつり人形』のように、逆に  $ratio$  の値が上昇した曲もあった。

これは、曲によって最適な旋律断片分割アルゴリズムが異なることを示しているといえる。

これらのことから、旋律断片自動生成アルゴリズムが、実際に出力 SMF に影響を及ぼしていることが明らかになった。

一部の曲で  $ratio$  の値が上昇した原因としては、以下のような原因が考えられる。

1. 旋律断片右手部分と左手部分に分けたことによって、一部の旋律断片において、音符の分布の類似性評価の際に悪影響を及ぼした可能性がある。
2. 連符とスラーの情報を用いることによって、より楽譜の構造を反映した旋律断片を生成することができたが、旋律断片の長さのパターンと旋律断片内部のリズムパターンが多くなり、類似した事例を検索することが難しくなった可能性がある。

このうち、2 に関しては、演奏事例 DB を充実させ、旋律断片のパリエーションを増やすことによって対応可能であると考えられる。

## 7. おわりに

本報告では、事例に基づく演奏表情生成システム Kagurame Phase-II における、MusicXML からの旋律断片自動生成アルゴリズムの改良とその評価について述べた。Closed-Test と Open-Test について演奏表情生成を行い、旋律断片生成アルゴリズムが演奏表情生成に影響を与えることを確かめた。

今後は、旋律断片自動生成アルゴリズムの改良を進めていくとともに、曲ごとの旋律断片生成アルゴリズムの最適化を行っていきたい。また、今回は生成結果の定量評価のみでアルゴリズムの評価を行ったが、今後は定性評価としてアンケートによる試聴評価も併せて行っていきたい。

また、CrestMusePEDB [12] のような良質で多量の曲情報を演奏事例 DB として採用し、有効性評価の精度と信頼性を上げることを計画している。

## 参 考 文 献

- [1] T. Suzuki: Kagurame Phase-I, musical expression generation system with case-based method, Proc. ICAD2002 Rencon Workshop, pp.13-20 (2002).
- [2] T. Suzuki: The second phase development of case based performance rendering system “Kagurame”, Proc. IJCAI03 Rencon Workshop (2003).
- [3] 金子雄介, 鈴木泰山, 徳永幸生: 事例に基づく演奏表情生成システムにおける表情生成式の最適化, 情報処理学会研究報告, 2004-MUS-58 (Dec 2004).
- [4] 金子雄介, 鈴木泰山, 徳永幸生: 事例に基づく演奏表情生成システムにおける演奏類似性と聴覚評価, 情報処理学会研究報告, 2005-MUS-59 (Feb 2005).
- [5] 鈴木泰山, 金子雄介, 徳永幸生: 事例に基づく演奏表情生成手法の演奏表情生成アルゴリズムの分析, 情報処理学会研究報告, 2005-MUS-59 (Feb 2005).
- [6] Recordare MusicXML Web Site:  
<http://www.recordare.com>
- [7] 平賀瑠美, 平田圭二, 片寄晴弘: 蓮根, めざせ世界一のピアニスト, 情報処理, Vol.43, No.2, pp.136-141 (2002).
- [8] 野池賢二, 平田圭二, 片寄晴弘: Rencon エントリーシート第1版の仕様の考察, 2003-MUS-50 (May 2003).
- [9] 浜中雅俊, 平田圭二, 東条敏: 音楽理論 GTTM に基づくグルーピング構造獲得システム, Transactions of Information Processing Society of Japan, Vol. 48, No. 1, pp. 284-299, (Jan 2007).
- [10] 橋田光代, 片寄晴弘: スラーに基づく演奏表現の一検討, 情報処理学会研究報告, 2003-MUS-49, pp.19-23, (2003).
- [11] 橋田光代, 片寄晴弘: スラーに基づく演奏表現: パラメータ推定と生成手法の考察, 情報処理学会研究報告 2003-MUS-51, pp.91-96, (2003).
- [12] 橋田光代, 片寄晴弘: 音楽演奏表情データベース構築に向けて, 第 21 回人工知能学会全国大会, (Jun 2007)
- [13] 清水厚志, 原田杏奈, 鈴木泰山, 野池賢二, 金子雄介, 徳永幸生: 事例に基づく演奏表情生成システムにおける MusicXML からの旋律断片自動生成, 情報処理学会研究報告 音楽情報科学, 2006-MUS-68, (Dec 2006).
- [14] 原田杏奈, 清水厚志, 鈴木泰山, 野池賢二, 金子雄介, 徳永幸生: 事例に基づく演奏表情生成システムにおける複旋律を考慮した表情生成式, 情報処理学会第 69 回全国大会 6N-7, (Mar 2007).
- [15] 清水厚志, 鈴木泰山, 野池賢二, 金子雄介, 徳永幸生, 事例に基づく演奏表情生成システムにおける旋律と和声の構造を考慮した旋律断片自動生成, FIT2007, 第 6 回情報科学技術フォーラム, 3E-028 (Sep 2007)