

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

鈴木泰山[†] 金子雄介^{††} 徳永幸生^{††}

本論文では事例に基づく演奏表情生成手法を複旋律の楽曲に対応させる手法を紹介し、その手法の有効性に関する分析を報告する。我々は、事例に基づく演奏表情生成手法を用いた演奏表情生成手法の研究を行っている。当初はアルゴリズムが複雑になることを避けるために、単旋律楽曲を対象として手法の検討や研究を行っていたが、実用的な演奏表情生成システムでは複旋律楽曲に対して自然な演奏表情を生成することが必要となることから、複旋律楽曲に対応したアルゴリズムの検討を行い、演奏表情生成システム Kagurame Phase-II に実装した。また、Kagurame Phase-II によって生成された演奏表情を分析することで、複旋律楽曲に対応したアルゴリズムの有効性を検証した。これにより、複旋律楽曲に対する事例に基づく演奏表情生成手法の有効性と課題が明らかになった。

Evaluation of Polyphonic Musical Expression Generation Algorithm for A Case Based Performance Rendering System

TAIZAN SUZUKI,[†] YUSUKE KANEKO^{††} and YUKIO TOKUNAGA^{††}

This paper describes about the expansion algorithm for a case-based approach to the generation of musical expression, and reports validity of the algorithm. We have been developing a case based approach to the generation of musical expression. Initially, to avoid the algorithm to be too much complicated, we limited the target pieces to monophonic pieces. However, this limitation prevents the system to be practical, because most common pieces are polyphonic. Hence, we developed expansion algorithm for the case-based method which can handle polyphonic pieces, and we have implemented the algorithm on a performance rendering system "Kagurame Phase-II". We have tested the new algorithm through the evaluation of performances rendered by Kagurame Phase-II. Results of tests made the validity and problems of the algorithm clear.

1. はじめに

本論文では、事例に基づく演奏表情生成手法の複旋律楽曲対応について報告する。

近年、人間が行うような自然な音楽演奏を自動的に行う演奏表情生成システムの研究が多数行われ、さまざまな手法が提案されている^{1),3),7),11)}。また、演奏表情生成システムが生成した演奏を評価する試みとして、人間による聴取実験によって演奏を評価するコンテスト Rencon が行われている^{2),5),6)}。

我々は、事例に基づく演奏表情生成手法を提案するとともに、この手法を利用した演奏表情生成システム Kagurame の構築を行っている^{8),9)}。以前の Kagurame は単旋律の楽曲のみを対象としていた。しかし、実用に耐えうるシステムとして Rencon に参加することを考えると、複旋律の楽曲に対応していることはきわめて重要である。ここでの複旋律楽曲とは同一の拍位置に2つ以上の音符が含まれる楽曲のことをさす。独唱曲や管楽器の単独演奏による楽曲を除けば、ほとんどの楽曲は複旋律の楽曲である。そこで、複旋律に対応した事例に基づく演奏表情生成手法の研究を進めると共に、この手法を実装した演奏表情生成システム Kagurame Phase-II の構築を行っている¹⁰⁾。

本論文では、2章で事例に基づく演奏表情生成手法の概

要を概観した後、複旋律楽曲への対応方法について3章で説明する。この手法を導入した Kagurame Phase-II による演奏表情の生成実験の結果、明らかになった問題点等の分析を4章で述べる。

2. 事例に基づく演奏表情生成手法

本章では、Kagurame Phase-I 及び Phase-II で利用している事例に基づく演奏表情生成手法について、両方のシステムで共通となる基本構成を中心に説明する。

2.1 概要

事例に基づく演奏表情生成は、人間によって行われた演奏を模倣することで対象曲の演奏表情を生成する手法である。演奏表情生成に関する知識として、人間が行った演奏の集合を利用する。この演奏の集合を演奏データ集と呼ぶ。

模倣するといっても対象曲と同一の楽曲が演奏データ集に含まれているわけではないので、事例に基づく手法では対象曲と類似した旋律の楽曲を選別し、その演奏データを模倣する。また、楽曲全体が対象曲に類似した事例が発見できることは稀なので、対象曲をさまざまな長さの断片に分割し、それぞれの断片について演奏データ集の楽曲中に類似した断片がないかどうかを検索する。このようにして得られた断片ごとの参考事例の演奏表情を分析し、これを対象曲に適用することで演奏データを生成する。

本手法は、演奏状況と組み合わせることによって、同一の楽曲に対して様々な演奏を生成できるという特徴を持つ。演奏状況とは、演奏データ集の個々の演奏データに対して与えられたメタデータで、その演奏が行われた際の状況の記

[†] 株式会社 デュオシステムズ
Duo Systems Co., Ltd.

^{††} 芝浦工業大学 工学部
Dept. of Information Science and Engineering, Shibaura Institute of Technology

録である．具体的には，演奏者や演奏日時，演奏場所や演奏のスタイルといったメタデータが相当する．

類似した旋律の演奏データを選別する際に，旋律そのものの類似性だけでなく，個々の事例に付与されたメタデータと演奏生成時に入力したメタデータ設定との類似性を考慮することで，同じ楽曲でもメタデータ設定によって選別される演奏データが変化する．これによって，同じ楽曲に対して様々な演奏表情を生成することが可能となる．なお，本論文では演奏状況に関する詳細な説明は割愛する．

2.2 生成手順

ここでは，事例に基づく演奏表情生成手法の全体構成を説明する．事例に基づく演奏表情生成手法の生成プロセスは図1に示すとおりである．本手法は，1) 検索，2) 分析，3) 評価，4) 合成，5) 適用，の5つの手続きから構成されている．

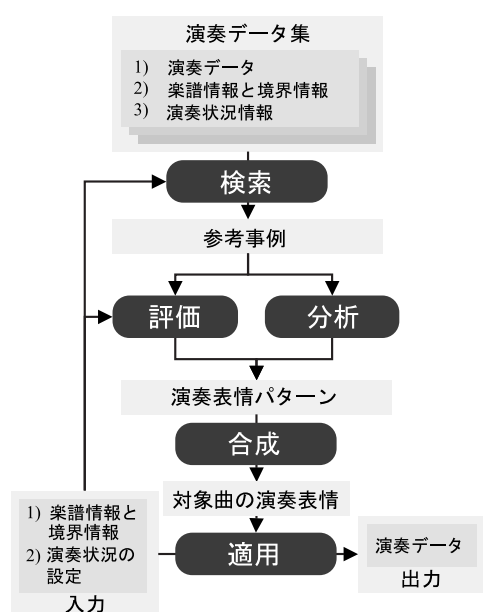


図1 事例に基づく演奏表情生成手法の構成

2.2.1 演奏データ集と入出力

事例に基づく演奏表情生成手法では，演奏表情の知識として演奏データ集を利用する．演奏データ集とは人間によって演奏された演奏データの集合である．演奏データ集の個々の演奏データは 1) 人間の演奏データ，2) 演奏曲の楽譜情報と境界情報，3) 演奏時の演奏状況情報，の3つから構成される．楽譜情報には音符の情報の他に，楽曲内のまとまりの境界の情報も与える必要がある．この境界情報は「検索」のプロセスで利用するものである．詳細は 2.2.2 項で説明する．

入力としては，1) 対象曲の楽譜情報と 2) 生成する演奏の演奏状況設定を与える．楽譜情報は演奏データ集の楽譜情報と同様に境界情報も与える必要がある．そして，出力は生成された演奏表情付きの演奏データである．

Kagurame では，演奏データ集の中の人間の演奏データは標準 MIDI フォーマット (SMF) ファイルを利用している．また演奏曲の楽譜情報と境界状況は独自フォーマットのテキストファイル (SEQ ファイル) で付与する．このほかに演奏

データ集の定義ファイルがあり，演奏状況についてはこのファイルに記述している．入力の楽譜情報は演奏データ集と同様に SEQ ファイルで与える．演奏状況はプログラムの引数で与えている．出力される演奏データも演奏データ集と同様に SMF ファイルである．

2.2.2 検索

検索は生成プロセスの中で最初に行われる手続きである．このプロセスでは演奏データ集の中から対象曲の演奏表情生成に利用できそうなものをピックアップする．

まず最初に対象曲及び参考データ集の各楽曲をそれぞれ断片に分割する．Kagurame Phase-I および Phase-II では，旋律の境界情報はあらかじめ楽譜情報とともに与えている．断片のサイズはさまざまではあるが，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 拍などを分割の単位とする (図2)．従って，ほとんどの断片は2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．

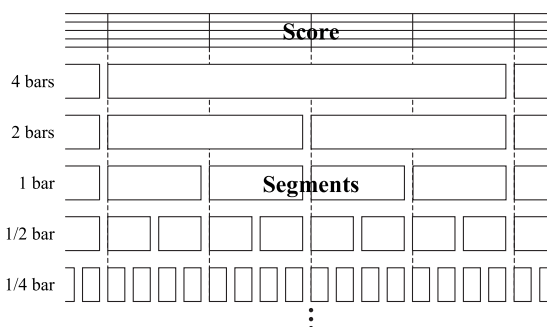


図2 旋律断片への分割

検索プロセスでは，断片に分割した後に，対象曲の個々の断片 (対象断片) に対して，演奏データ集中にある類似した断片 (参考事例) を検索する．このときに，旋律断片の類似性評価ロジックを利用する．旋律断片の類似性評価ロジックは対象断片と参考事例の候補との旋律を比較し，その類似度を評価するロジックである．類似性評価ロジックは後述の評価プロセスでも利用しており，生成される演奏表情に対して大きな影響力を持っている．このロジックは Phase-I と Phase-II との相違点のひとつである．詳細については 3 章で説明する．

検索プロセスでは，類似性評価ロジックの出力したスコアに基づいて演奏データ集の中の旋律断片をランキングし，スコアが閾値を超えているものを，その対象断片に対する参考事例として抽出する．従って，ひとつの対象断片に対して複数の参考事例が検索される．

2.2.3 分析

分析プロセスでは，個々の参考事例の演奏表情を分析する．参考事例ごとに楽譜情報と演奏データを比較し，テンポ，音の強さなどの演奏表情パラメータの変化率を計算する．この比率の集合を演奏表情パターンと呼ぶ．参考事例に対して，その断片全体での演奏表情種別 (テンポ，音量など) の平均値とその断片より一回り大きな断片 (これを親断片と呼ぶ) での平均値を計算する．そして，それぞれの演

演奏表情断片の平均値と親断片の平均値との比率を演奏表情種別ごとに計算して演奏表情パターンとする。

演奏表情パターンは、楽曲全体の演奏表情を旋律断片を基準として比率に分解した値であり、フーリエ変換にも似た機能を有する。演奏表情パターンと演奏表情値の間には可逆性がある。つまり、全ての旋律断片の演奏表情パターン情報があれば楽曲全体の演奏表情を再計算によって求めることが可能である。

2.2.4 評価

評価プロセスでは、個々の参考事例の重要度を計算する。参考事例の重要度とは、対象曲の演奏表情の生成の際に、その参考事例がどの程度参考になるのかを評価したスコアである。重要度はこの後に続く合成プロセスで利用する。合成プロセスでは参考事例に与えられた重要度が高いものほど、その参考事例の演奏表情パターンを重視する。

重要度は、旋律の類似度と演奏状況の近さから算出する。旋律の類似度は、対象断片と参考事例とを類似性評価ロジックで比較した結果である。これは検索の際に用いたスコアと同様である。演奏状況の近さは、入力として与えられた演奏状況設定と、参考事例が含まれる演奏データが持っている演奏状況情報との近さをベクトルの内積から計算したものである。入力として演奏状況設定を特に指定していない場合は、重要度は検索時のスコアとまったく同じ値になる。

2.2.5 合成

合成プロセスでは、対象断片に対する参考事例の演奏表情パターンを組み合わせることで対象断片ごとの演奏表情パターンを合成する。先に述べたとおり、演奏表情パターンは演奏表情種別ごとの比率値として表されている。合成プロセスではそれぞれの参考事例の比率値を、演奏表情種別ごとに重要度で加重平均をしたものである。加重平均の際には、重要度をそのまま利用するのではなく、重要度を指数関数に適用し分布を変化させた値を用いる(図3)。これにより、対象断片ごとの演奏表情パターンが得られる。

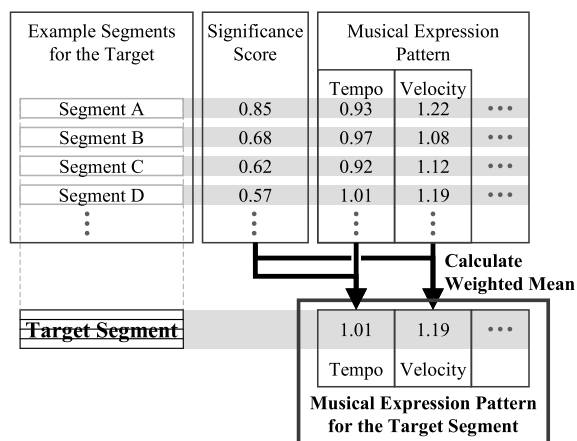


図3 対象断片に対する演奏表情パターンの合成

さらに、それぞれの対象断片の演奏表情パターンを掛け合わせることで合成して対象曲全体の演奏表情を生成する(図4)。先に述べたとおり、演奏表情パターンと演奏表情とは可逆変換が可能であることから、対象曲の全ての対象断

片に対して演奏表情パターンが得られれば、対象曲の演奏表情値に逆変換することが可能である。

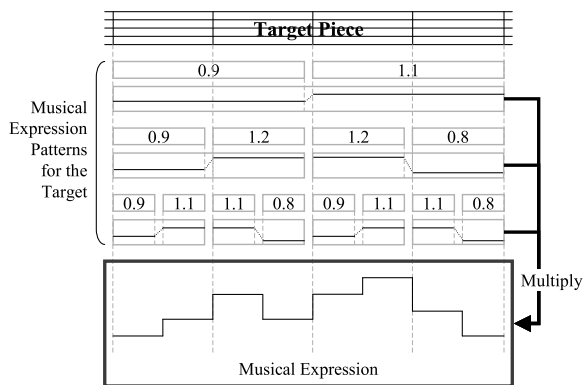


図4 対象曲の演奏表情値の合成

2.2.6 適用

適用プロセスは、演奏表情生成プロセスの最後に行うプロセスである。ここでは、合成プロセスで生成された演奏表情値を対象曲に適用し、対象曲の演奏データを生成する。一音ごとの演奏表情値はすでに合成プロセスで得られているので、このプロセスではその演奏表情値を対象曲に適用しSMFファイルに変換する。

3. Kagurame の複旋律対応

Kagurame は2章で説明した事例に基づく演奏表情生成手法を利用した演奏システムである。実装時期や目的の相違から Phase-I システムと Phase-II システムの2種類の実装がある。Kagurame Phase-I システムは本手法の有効性を検証することを目的としており、関連する実装をシンプルにするために対象を単旋律の楽曲に限定していた。Phase-II では実用性の検証と Rencon へのエンタリを目的とし、そのため複旋律楽曲への対応を中心とした拡張を行っている。

Kagurame Phase-I が単旋律の楽曲の演奏表情しか生成できない主な理由は下記の2点にある。

- 旋律断片の類似性評価ロジックが単旋律を前提としたアルゴリズムを採用している
- 演奏表情の分析プロセスが複旋律を考慮していない

事例に基づく演奏表情生成手法の基本構成は単旋律でも複旋律でも対応可能であるが、個々のプロセスで利用しているロジックが単旋律を前提としていたために複旋律の楽曲の演奏表情の生成ができなかった。そこで、Phase-II の実現にあたっては、これら2項目に関連するロジックの複旋律への対応を中心に行った。本章では、この2項目の拡張について説明する。

3.1 旋律断片の類似性評価ロジック

本節では、複旋律楽曲への対応のうち旋律断片の類似性評価ロジックに関する変更点について説明する。

3.1.1 旋律類似性評価ロジックの概要

旋律断片の類似性評価ロジックは、与えられた2つの旋律断片の類似度を評価するロジックである。生成の過程ではこのロジックは検索と評価の際に利用している。このロジックが出力するスコアは生成する演奏表情値を計算する際に

利用しているため、ロジックの内容によって生成される演奏表情は大きく変化する。

従来の手法では、複旋律の旋律断片の類似性について評価するには不十分であった。そこで、旋律類似性評価ロジックに関連して下記の拡張を行った。

- 素性情報の追加
 - 素性情報から類似性を算定する方法の改善
- 以降ではそれぞれについて説明する。

3.1.2 素性情報の追加

事例に基づく演奏表情生成手法での旋律類似性評価は、楽曲検索における類似性評価に比べると、類似していると判断できるものをより広範に検索できる必要がある。また、合成プロセスではスコア情報を用いて対象曲の演奏表情を生成することから、ランキングだけではなく個々の候補に対する連続的なスコアを算出できる必要がある。

これらの目的から、Kagurame では素性ベクトルを比較する方法によって旋律類似性の評価を行っている。単旋律のアルゴリズムでは素性情報を抽出するロジックとして 1) LengthEvaluator, 2)HarmonyEvaluator, 3) RhythmEvaluator の 3 種類を用いていた (図 5)。

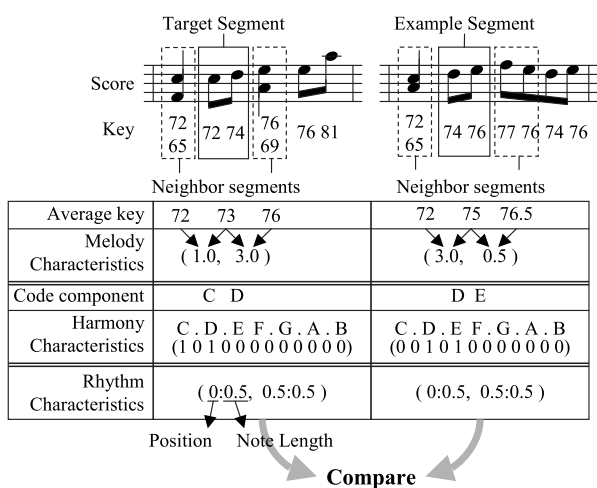


図 5 素性による旋律類似性評価

LengthEvaluator は旋律断片の長さである。HarmonyEvaluator は旋律断片に含まれている音の音階の分布である。個々の音の音高を楽曲の調で標準化し、さらにオクターブの相違を取り除くことで、それぞれの階名を決定する。そして、旋律断片に含まれる音の数をドからシまでの階名ごとに集計した 12 要素のベクトルである。RhythmEvaluator は旋律断片内での発音位置の分布を表したものである。短い旋律断片であれば拍子に近い情報が得られ、また長い旋律断片では穏やかな部分と激しい部分の情報が得られることが期待される。旋律断片を 16 個の等間隔の区間に分割し、それぞれの区間での発音数のベクトルを素性値として用いている。

複旋律の楽曲に対して効果的な素性情報評価ロジックとして新たに 4) KeyRateEvaluator と 5) MelodyEvaluator を追加した。

KeyRateEvaluator は、注目している旋律断片と近傍の旋律断片との音高の関係を表している。この値が大きい場合は、

周囲に比べて音高が高く盛り上がっている部分であることを表す。逆に低い場合は周囲に比べて沈滞した部分であることを表す。KeyRateEvaluator の出力値は、対象の旋律断片を包含する一回り大きい旋律断片の平均音高と、対象の断片自身の平均音高の比である。

MelodyEvaluator は旋律断片の主旋律の特徴を現す素性情報である。旋律断片を 16 の区間に分割し、それぞれの区間の中で一番音高が高い音を取得する。そして 16 の全ての区間の最高音高値を求めたら、その平均値を算出し平均値との差を求めて素性情報とする。このベクトルは主旋律の外形を現しており、この素性を比較することで主旋律の推移の近さを判断することが可能である。

3.1.3 類似性の算定方法の改善

個々の素性の類似性から旋律の類似性のスコアを算出する方法を改善した。Phase-I では個々の素性から得られる情報を等価とし、分布が均質になるようにそろえた上で単純に加算していた。一方で、Phase-I による実験の結果個々の素性情報ごとにその有効性に差があることが明らかになっていた。そこで、Phase-II では個々の素性情報に対してそれぞれ固有の係数をかけたものの和とした。

$$\text{類似性} = \sum_{f \in \text{素性情報}} A_f D_f$$

A_f が素性ごとの係数、 D_f が素性の類似性を表す。これにより、個々の素性の効果に応じて A_f を変化させ、よりの確な類似性の値を得ることが可能となる。

3.2 演奏表情の分析と合成

本節では、複旋律楽曲への対応のうち、演奏表情の分析や合成にかかわる部分について説明する。演奏表情に関連した変更点は、大別すると 1) 新たな演奏表情パラメータの導入、2) 演奏表情の分析アルゴリズムの変更、3) 演奏表情の合成アルゴリズムの変更の 3 点である。

3.2.1 新たな演奏表情パラメータの導入

単旋律楽曲の演奏表情と複旋律楽曲の演奏表情の最大の相違点は同じ拍位置にある音符の間の演奏表情値の相違にある。和音においては和音を構成する各音の強さの相違によって和音の聴感に変化する。また、グリッサンドなどのように同拍の和音をずらして演奏するといった演奏技法も演奏表情の要素として重要である。同拍の和音がずれて演奏されるということは、つまり同じ拍位置に対して複数の時刻が存在することを表す。

Phase-I では単旋律を対象としていたため、拍位置に対応する時刻は 1 つだけで、和音の非同時性については一切考慮していない。そこで、和音での拍の非同時性を表す演奏表情パラメータとして「和音のずれ」という値を新たに導入した。和音のずれとは、同じ拍位置にある和音の個々の音符の発音位置の非同時性を表す演奏表情値である。

つまり、和音の構成音の各音が、その拍位置の標準的な演奏時刻にたいして、どの程度ずれて演奏されたかを拍数に換算して表した数値である。この数値が大きいほど、和音のほかの音からずれて演奏されたことを表す。また負の値は前のりで演奏されたことを表し、正の値は遅れて演奏されたことを表す。

このほかに、Phase-I から対象となっていた演奏表情として 1) 局所テンポ、2) 音の強さ、3) 発音長 (デュレーション) の 3 種類がある。局所テンポは個々の旋律断片のテンポ値である。ある旋律断片の局所テンポは、その旋律断片の区間の拍数をその区間を演奏する際に要した時間で割った値である。この値が大きければその旋律断片の区間の演奏はすばやく、小さければゆっくりと行われる。

音の強さは演奏時の打鍵の強さを表す数値である。SMF ファイル中に記録されている velocity 値を利用する。velocity 値は 0 から 127 までの数値で表され、数値が大きいほど打鍵が強いことを表す。

音ごとの音長は、個々の音符の楽譜上の拍数と実際に演奏した拍数との比率である。演奏した拍数は打鍵から離鍵までの時間の差に局所テンポをかけたものである。1 の場合は楽譜上の音符の長さと同じ長さだけ演奏したことを表す。0.5 の場合は楽譜上の長さの半分で離鍵したことを表す。この数値が小さいとスタッカートになり大きくなるとレガートになる。

3.2.2 分析

本項では、生成手順中の分析プロセスで行っている参考事例の演奏表情値の分析処理について説明する。

複雑旋律楽曲での演奏表情の分析でもっとも複雑な点は、和音を含む部分での各音の発音時刻からの局所テンポ値の算出である。局所テンポは個々の旋律断片のテンポ値である。楽曲のある旋律断片の局所テンポは、その旋律断片の区間の拍数をその区間を演奏する際に要した時間で割った値である。さらに区間の演奏時間を計算するためには、その区間の開始拍と終了拍の演奏時刻が特定できなければならない。

単旋律楽曲の場合は全ての音が単音で演奏されるため、個々の音の発音時刻をその音の拍の時刻として利用できる。一方で複雑旋律楽曲の場合は和音の場所では同一拍に複数の音が演奏されるため、和音を構成する各音の発音時刻からその拍の時刻を算出しなければならない。

和音での拍時刻の特定方法としては、和音の各音の発音時刻の平均値を用いる方法や、和音構成中の最初に演奏された音の時刻を用いる方法、和音中の最低音あるいは最高音の時刻を用いる方法、さらには書く音の音の強さなどから加重平均する方法などが考えられる。Kagurame Phase-II では和音構成音の発音時刻の平均値を用いることとした。それ以外の方法についても評価を行う必要があると考える。

局所テンポの算出の際には、この平均発音時刻を利用している。また、前項で説明した和音のずれは、和音構成音の平均発音時刻と各音の実際の発音時刻の差から求めている。

一方で、発音長の計算の際には平均発音時刻は用いず、各音の実際の発音時刻と離鍵時刻との差から算出している。

なお、音の強さについては従来どおり SMF ファイル中の各音の velocity 値をそのまま利用している。

演奏表情の分析処理全般に関しては、このほかに楽曲と楽譜とのマッチング部分の頑健性を高めた。特に和音部分では和音を構成する各音は任意の順序で演奏されるため、和音部分の演奏音と楽譜上の音符との対応が適切に取れるように改善した。

3.2.3 合成

上記の変更にあわせて、合成部分も同様の変更を行った。複雑旋律楽曲に対する演奏表情の合成での主要な課題は、分析部分と同様に局所テンポや和音のずれの値から各音の発音時刻を特定する部分である。各音の発音時刻の算出は下記の手順で行う。

- (1) 拍位置ごとの局所テンポを算出する
- (2) 局所テンポを積算することで拍位置ごとの時刻を算出し平均発音時刻とする
- (3) 和音については和音のずれの値と局所テンポから平均発音時刻との時間差を計算する
- (4) 2 の平均発音時刻に 3 の時間差を加算して実際の発音時刻を算出する

離鍵時刻については、局所テンポと音符の長さの拍数と発音長を掛け合わせ、これを上記の 4 で求めた発音時刻に加算することで算出している。

velocity については特に変更は行っていない。

4. 和音対応の課題分析

本章では、Kagurame Phase-II 生成した演奏表情の分析から、3 章で説明した各手法の効果について分析する。

4.1 旋律断片の類似性評価ロジック

本節では複雑旋律楽曲への対応手法として 3.1.1 で紹介したもののうち、追加した素性情報の有効性と類似性評価法の改善についての分析と評価について報告する。

4.1.1 素性情報の追加

新たに導入した KeyRateEvaluator と MelodyEvaluator の素性情報の効果の評価のために、これらの素性情報を利用した場合と利用していない場合の演奏表情の違いを分析した。生成される演奏表情の分析には相違度という数値を利用した。この数値は、生成された演奏と同じ楽曲の人間の演奏との異なりを表す数値で、値が小さいほど、生成された演奏が人間の演奏に近いことを表す。従って、相違度が小さいものの方がよりの確な演奏表情の生成ができているとみなすことができる。

この実験の結果を表 1 に示す。なお、実験の際には Chopin の小曲集から 3 曲を利用し、演奏データ集とは異なる Chopin の楽曲の演奏表情を生成させた。また、相違度の算出には音の強さを利用した。

KeyRateEvaluator については、KeyRateEvaluator が無い場合に比べると KeyRateEvaluator の導入によって相違度が 0.04 程度低下している。従って、KeyRateEvaluator は演奏表情の生成に対して効果的であったと考えることができる。一方で MelodyEvaluator については、MelodyEvaluator の有無によって相違度は殆んど変化していない。従って、演奏表情の決定に対してあまり有効に機能していないと考えられる。

MelodyEvaluator については、人間がメロディと判断する

表 1 追加した素性情報の効果

生成時に利用した素性	相違度 (velocity)
従来の 3 種類のみ	0.2044
3 種類 + KeyRateEvaluator	0.1660
3 種類 + MelodyEvaluator	0.2042
5 種類全て	0.1659

旋律が的確に抽出できているかどうか疑問があることから、この点を中心にさらなる分析を行う必要がある。

4.1.2 類似性の算定方法の改善

類似性の算定方法の改善に関しては、結合式の定数を変化させることによって生成される演奏表情がどう変化するかについて調査を行った。ここでは、個々の素性情報を単体で利用した場合の、演奏表情の相違度を評価し、相違度が最低になる A を求めた。その結果を表 2 に示す。なお、4.1.1 項で示したとおり MelodyEvaluator はあまりよい効果が得られていないことから、実験では MelodyEvaluator 以外の 4 つの Evaluator について評価を行った。なお、実験で用いた演奏データ集及び対象曲は 4.1.1 項の実験とは異なるため、相違度の値も異なっている。

この実験から、 A の値を変化させることによって生成される演奏表情が変化することが確認できた。また、一定の条件下での最適値を求めることができた。

一方で、演奏データ集と対象曲を他のものに変えると、最適値が変化するという問題も発見された。現状は素性ごとの類似性の一次結合を利用しているが、将来的にはより複雑な式にすることについて検討する必要がある。さらに、生成された演奏の評価を還元することで、類似性評価ロジックを学習させるような手法についても検討する必要がある。

なお、この実験の詳細については文献 4) で報告している。

4.2 演奏表情の分析と合成

本節では演奏表情の分析と合成に関する変更のうち、新たに導入した和音のずれの効果について報告する。

4.2.1 和音のずれの効果

和音のずれの有効性の検証として、人間による演奏の和音のずれとシステムの演奏での和音のずれの平均値をそれぞれ調査した。その結果を表 3 に示す。

この表から明らかなように、人間の演奏に比べてシステムが生成した演奏は、和音のずれの総量がほぼ半分程度になってしまっている。実際に、聴取を行ってみても人間の演奏に比べるとシステムの演奏はグリッサンドなどの再現が十分には行われていない。

これは参考事例に含まれる単旋律からなる旋律断片の影響が大きいものと考えられる。和音を含む旋律断片の参考事例の中には、単旋律の旋律断片も含まれていることが想定される。しかしながら、Phase-II では和音を含まない旋律断片の和音のずれの値は 0、つまりずれがないものとして扱っている。従って、和音を含む旋律断片の参考事例中の中に重要度が高く和音を含まない旋律断片が多数含まれていた場合、

表 2 A による演奏表情の変化

生成時に利用した素性	A の最適値	相違度 (velocity)
LengthEvaluator	16	0.218
KeyRateEvaluator	128	0.185
RhythmEvaluator	128	0.186
HarmonyEvaluator	32	0.205
4 種類全て		0.140

表 3 和音のずれの平均値

演奏	和音のずれの平均値
人間の演奏	7.09 ms
Kagurame Phase-II が生成した演奏	3.55 ms

和音のずれの値が小さくなってしまふ。和音のずれの値を算出する際には、和音の事例を優先するなどの処理の検討を行う必要がある。

5. ま と め

本論文では事例に基づく演奏表情生成手法を用いた演奏表情生成システム Kagurame の概要を示した上で、Phase-II システムで複旋律楽曲への対応のために導入した各種の手法について説明した。また、Phase-II システムで生成した演奏データの分析から、本論文で報告した複旋律楽曲への対応手法の課題を明らかにした。現状では、複旋律楽曲の演奏表情生成に対応できているものの、まだ複旋律楽曲に十分対応できてはならず、旋律類似性評価アルゴリズムのさらなる改良と、和音の同時性のずれの再現に改善の余地がある。

参 考 文 献

- 1) Bresin, R.: Importance of note-level control in automatic music performance, *Proceedings of the ICAD 2002 Rencon Workshop*, pp. 1–6 (2002).
- 2) Hiraga, R., Bresin, R., Hirata, K. and Katayose, H.: Rencon 2004: Turing Test for Musical Expression, *Proceedings of the NIME 2004*, pp. 120–123 (2004).
- 3) 平田圭二, 平賀瑠美: ハーヒーフン: 2 段階演奏表情付け法によるインクリメンタルな演奏生成システム, 情報処理学会研究報告 2001–MUS–39, Vol. 2001, No. 16, pp. 19–26 (2001).
- 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表情生成式の最適化, 情報処理学会研究報告 2004–MUS–58, Vol. 2004, No. 127, pp. 27–32 (2004).
- 5) 片寄晴弘, 平賀瑠美, 平田圭二, 野池賢二, 橋田光代: ICAD–RENCON —報告と課題—, 情報処理学会研究報告 2002–MUS–47, Vol. 2002, No. 100, pp. 79–83 (2002).
- 6) 片寄晴弘, 平田圭二, 平賀瑠美: IJCAI–RENCON の報告と課題, 情報処理学会研究報告 2003–MUS–52, Vol. 2003, No. 111, pp. 149–152 (2003).
- 7) 野池賢二, 豊田健一, 片寄晴弘: コーパスベース表情付けシステム COPER の基礎機能の実装とその評価, 情報処理学会研究報告 2001–MUS–52, Vol. 2005 (2005).
- 8) 鈴木泰山, 徳永健伸, 田中穂積: 事例に基づく演奏表情の生成, 情報処理学会論文誌, Vol. 42, No. 4, pp. 1134–1145 (2000).
- 9) Suzuki, T.: Kagurame Phase-I, musical expression generation system with case-based method, *Proceedings of the ICAD 2002 Rencon Workshop*, pp. 13–20 (2002).
- 10) Suzuki, T.: The second phase development of case based performance rendering system “Kagurame”, *Proceedings of the IJCAI–03 Rencon Workshop* (2003).
- 11) Widmer, G.: Inductive learning of general and robust local expression principles, *Proceedings of the 2001 International Computer Music Conference*, International Computer Music Association, pp. 322–329 (2001).