

Kagurame Phase-I —事例ベースの演奏表情生成システム—

鈴木泰山, 宮本朋範, 西田深志, 徳永健伸, 田中穂積

東京工業大学 情報理工学研究科 計算工学専攻

{taizan, tomonori, nishida, take, tanaka}@cs.titech.ac.jp

我々は事例ベースの手法を用いた演奏表情の自動生成の研究を行なっている。この手法では、対象曲と与えたい演奏表情の特徴を入力すると、あらかじめ用意した演奏データ集の中から対象曲に類似した曲を検索し、それらに見られる演奏表情を対象曲に適用することで対象曲の演奏データを生成する。この手法を用いて演奏表情の自動生成を行なうシステムが“Kagurame”である。現在は、機能を限定した構成の“Kagurame Phase-I”の実装を行なうと共に、このシステムを利用して事例ベースの演奏表情生成手法の有効性を確認するための評価実験を行なっている。本稿では事例ベースの演奏表情生成手法を簡単に説明した上で、“Kagurame Phase-I”の実装及び実験について報告する。

Kagurame Phase-I — An Example-Based Expressive Performance Generation System —

Taizan Suzuki, Tomonori Miyamoto, Fukashi Nishida, Takenobu

Tokunaga, Hozumi Tanaka

Department of Computer Science

Graduate School of Information Science and Engineering

Tokyo Institute of Technology

We have been developing an example-based approach for the production of expressive musical performance. In this approach, the input is the subject piece and performance condition information. First of all, subject piece is divided into musical fragments. Then, for each musical fragment, similar fragments are selected from a sample performance data set. Finally, the system process extracts a common musical expression pattern from the examples, and applies this pattern to subject fragment. By generating a musical expression based on a example selection, this approach can generate varying expressive performances.

“Kagurame” is our implementation of this example-based approach. We are currently developing the first phase of this system under UNIX, that is called “Kagurame Phase-I”. In this paper, we briefly describe the algorithm of our example-based approach, and report implementation of “Kagurame Phase-I” and the evaluation of the resulting performance of this system.

1 はじめに

我々が普段耳にしている音楽演奏は、必ずしも楽譜通りに正確に演奏されているわけではない。実際の人間の演奏ではテンポを微妙に揺らしたり強調したい部分の音を強くするなど様々な演出が行なわれており、これにより聴者は音楽に様々な表情を知覚するのである。楽曲の演奏中にみられるこれらのテンポや音の強さの揺らぎのことを演奏表情と呼ぶ。演奏表情には、テンポの揺らぎや音の強弱以外にも、和音のずれや音の出ている時間の長さ（スタッカート、レガート）、個々の楽器に特有な奏法など様々な要素がある。

演奏表情は自然な音楽演奏に必要な不可欠な要素であり、時によって聴衆に深い感銘を与えることもある。同じ曲でも素人が演奏したものと著名な演奏家が演奏したのでは全く異なる印象になるのは演奏表情の与え方が違うからである。演奏表情を極力与えず楽譜通りに演奏した曲は聞いていても面白みに欠けたものになってしまい、曲にもよるが演奏表情の与えられていない演奏はあまり良い演奏とはいえない。

こういった演奏表情の与え方は楽譜には殆んど記載されておらず、その大部分は演奏者が判断しなければならない。他の人の演奏を模倣する場合もあるが、通常は演奏者は知識や経験、趣向などに基づいてどういう風な演奏表情を与えるかを考える。自然な音楽演奏における人間の思考活動は直観や経験によるところが多く、またその多くは無意識に行なわれているので、生成の過程や根拠が不明確である。

近年コンピュータの普及とともに計算機による音楽演奏が普及するようになった。しかし、計算機の場合は人間のような知識や経験がないため、演奏表情を自ら生成し演奏することは困難である。通常は人間が楽譜情報を入力する際に演奏表情を同時に入力する。一方でこういった演奏表情には無意識に行なっている部分も多く、そのような要素も考慮して演奏データを作成するのは非常に根気のいる作業である。

我々は計算機による自然な演奏の自動生成を目指しており、そのための手法として事例ベースで演奏表情を自動生成する手法を提案した [11]。現在、この手法を用いて演奏表情を計算機で自動生成し自然

な音楽演奏を行なうシステム、“Kagurame Phase-I”の構築を行なっている。本論文では、まず2章で我々の提案した事例ベースの演奏表情生成手法について簡単に説明する。3章で“Kagurame Phase-I”の実装を解説し、4章では“Kagurame Phase-I”を用いた演奏表情の生成実験の結果を報告する。

2 事例ベースの演奏表情生成

2.1 演奏表情の自動生成

計算機に演奏表情を自動生成させる試みは古くから行なわれており、様々な手法が提案されている。以前は人間が与えた演奏表情のルールを利用して演奏表情を生成するという手法 [5, 6, 7, 9] が中心であったが、最近では人間の演奏データから演奏表情の生成ルールを学習し、それに基づいて未知曲に対する表情付けを行なう手法 [8, 1, 2, 3] が主流である。いずれにせよ、これらの手法は基本的にみなルールベースの手法であるということが出来る (図1)。

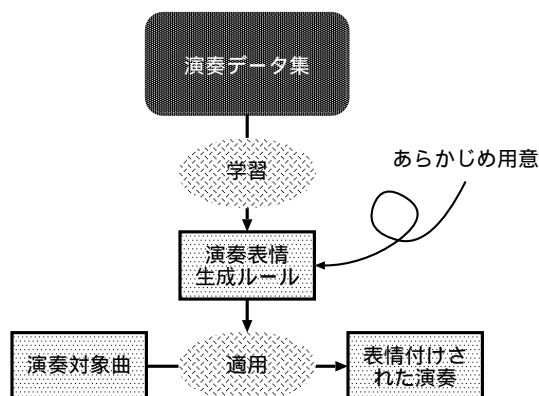


図1: ルールベースの演奏表情生成

さて、一般に演奏表情は非常に自由度が高く許容される範囲が広いという特徴がある。演奏家や時代、スタイルなどに応じて様々な演奏表情の与え方があり、同じ曲でも様々な異なった演奏表情を与えた演奏を行なうことが可能であり、むしろ演奏表情の与え方が全く同じになることの方が稀である。演奏家やスタイルなど演奏表情に影響を与える要素を演奏状況と呼ぶことにする。演奏状況が演奏表情に与える影響を学習、分析することが困難なため、殆どのルールベースによる手法では、演奏表情の生成規則を作成する際には、曲の特徴と演奏表情との関連付けしか行なっておらず、演奏状況は全く考慮されて

いない。したがって、これらの手法を用いたシステムでは一つの曲に対して一種類の演奏表情の生成しか行なえないという問題がある。

我々はこの問題を解決するために演奏表情の生成に事例ベースの手法を提案した。我々の手法では、まず人間の演奏したデータを参考データ集として用意する。個々の参考データは演奏データだけではなく、楽譜情報や演奏状況などの情報も付加しておく。入力として対象曲と演奏状況の設定が与えられると、対象曲に類似した曲の演奏データを参考データ集から検索し、見つかった事例を分析して演奏表情のパターンを作成する(図2)。このパターンを対象曲に適用することで対象曲に演奏表情の生成するという手法である。参考データを検索する際には、対象曲と参考データとの曲の類似性だけではなく、入力された演奏状況と参考データの演奏状況の近さも考慮するので、演奏状況の与え方によって検索される参考事例が変化し、その結果生成される演奏表情も異なったものになる。

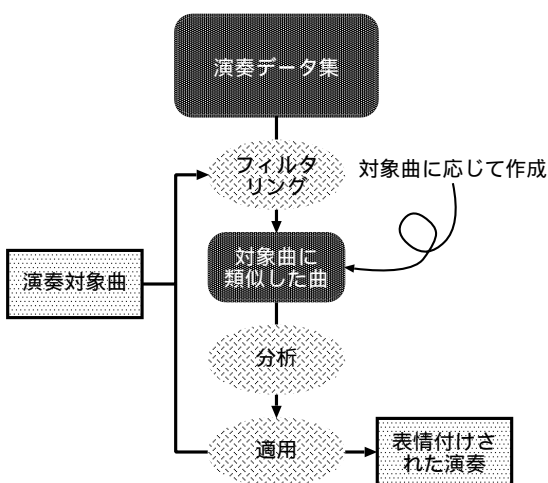


図2: 事例ベースの演奏表情生成

2.2 事例ベースの生成

さて、一般に事例ベース手法で問題になる点としてデータスパースネスがあげられる。事例ベースの手法では、サンプルのデータ集から参考になりそうな事例を検索し、その事例を利用して様々なことを行なうわけであるが、サンプルのデータの量が不十分だと適切な参考事例を見つけれない可能性が非常に高くなる。演奏表情生成の場合でも同様の問題が発生する可能性は十分にある。我々の手法に於

いて最も理想的な参考事例は対象曲そのものの演奏データであるが、サンプルの演奏データ集に対象曲の演奏データが含まれることはまず期待できないし、類似した曲を検索するにしても曲全体が似ているような演奏データが参考事例集に含まれることは稀であろう。

そこで、利用可能な参考事例を有効に利用する方法を考える必要がある。本手法では対象曲を曲のまとまりごとに分割し、その部分ごとに類似したものを演奏データ集から検索することでこの問題を解決する。分割された曲の各部を構成要素と呼ぶことにする。曲全体が類似している演奏データが見つかる可能性は稀であるが、曲の構成要素ごとに類似しているデータを検索すれば、類似している構成要素が演奏データから見つかる可能性は構成要素の長さの指数オーダーに反比例して高くなる。

最終的な演奏表情は、検索された構成要素の事例を組み合わせて生成する。しかし、単純に分割して得られた演奏表情を繋ぎ合わせただけでは、大局的に見た時の演奏表情のつじつまが合わなくなってしまう。演奏表情には局所的に現れるものもあるが、曲の長い範囲に影響を及ぼすものもある。よくある例としては主題は最初に出てきた時より二度目の方を強く演奏するなどがある。構成要素の分割が主題より細かいと、このような演奏表情を生成するのが困難になってしまう。

そこで、我々の手法では粒度の違う分割を幾つか用意し、それぞれの分割ごとに構成要素の参考事例を検索する。最終的な演奏表情は、粒度のことなる構成要素ごとの演奏表情を組合せて生成を行なう。最も粒度の荒い分割は曲全体を一つの構成要素とする分割なので、曲全体に影響するような演奏表情も再現することが可能となる。ここでは便宜上粒度のことなる分割を階層と呼ぶことにする。より粒度の荒い分割は階層が高いものとする。

分割の境界は曲の構造によって決める。殆んどの曲には繰り返しや主題、小節、拍など、様々な大きさのまとまりを定義することができる(図3)。こういった曲のまとまりを楽曲構造と呼ぶ。本研究ではこの楽曲構造を利用して曲を分割をすることにする。最下層分割は一拍を一つの構成要素とするものである。これを2ないしは3つまとめたものがそれより一段階上の階層の分割になる。曲にもよるがそのさらに1ないしは2段上が小節になる。

階層の上部になるとサンプルの量が少くなるが、細部の類似性はより下層の構成要素で考慮されるため、細部の類似性はあまり考慮する必要がなくなる。よって、下の階層に比べ細部の類似性を考慮しない分、類似度の判断の制約が緩くなるため、類似する部分の検索に失敗する可能性はさほど下がらないと考えられる。

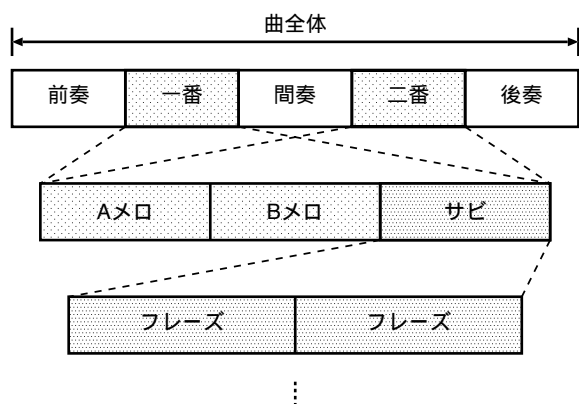


図 3: 楽曲構造の例

2.3 事例ベースの演奏表情生成のアルゴリズム

事例ベースの演奏表情生成のアルゴリズムは図 4 に示す通りである。

演奏表情生成を行なうためには、対象曲の楽譜情報と演奏状況設定を入力として与える(図 4:入力データ)。演奏対象曲の楽譜情報が必要なのは言うまでもない。演奏状況設定は生成される演奏表情の傾向を指定するもので、この設定を変更することによって多様な演奏表情の生成が可能となる。この値は生成の過程では参考事例の重要度の評価の際に利用される。

演奏データ集は、人間の演奏データを解析し収集したものである。個々の演奏データは演奏の記録と演奏曲の楽譜情報及び楽曲構造情報からなる。

システムは、まず最初に入力された楽譜情報から楽曲構造を解析する(図 4:楽曲構造解析)。この作業ではフレーズのまとまりや繰り返しを考慮して対象曲を構成要素に分割する。

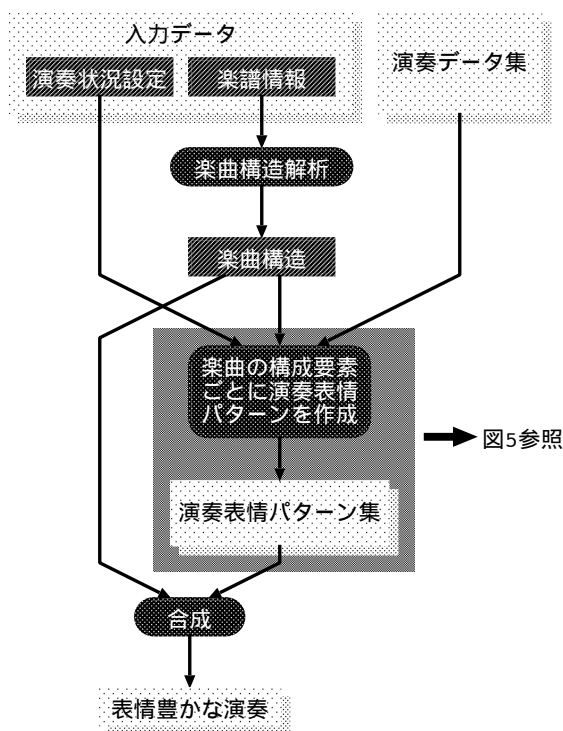


図 4: 事例ベースの演奏表情生成システム: 全体像

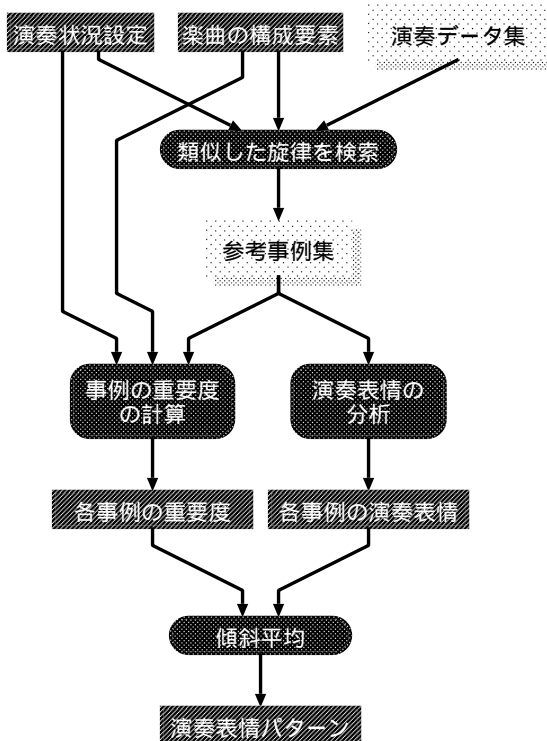


図 5: 事例ベースの演奏表情生成システム: 演奏表情パターンの獲得

次に楽曲構造から対象曲中の構成要素をひとつずつ取り出し、各構成要素について演奏データ集中から類似した旋律の演奏事例を検索する。演奏データ中の構成要素と対象曲の構成要素との類似性は、旋律そのものの類似性から評価する(図5:類似した旋律を検索)。これにより、対象曲の演奏表情の参考になりそうな演奏事例(図5:参考事例集)が収集される。

参考事例を検索したら、それぞれの演奏事例ごとにその重要度を評価する(図5:事例の重要度の計算)。事例の重要度は、最初に入力された演奏状況設定と事例の演奏状況との近さと、演奏対象曲の構成要素の旋律と事例の旋律の類似度から求める。対象曲に適用するのにふさわしい参考事例の方が重要度が高くなる。また、同時に参考事例ごとに演奏表情の分析も行なう(図5:演奏表情の分析)。

そして、参考事例の演奏表情をその事例の重要度に基づいて傾斜平均をとり(図5:傾斜平均)、これを注目している対象曲の構成要素の演奏表情パターンとする(図5:演奏表情パターン)。

対象曲の全ての構成要素について演奏表情パターンを獲得したら(図4:演奏表情パターン集)、最後にそれらを合成し(図4:合成)対象曲に適用することで、演奏表情が付与された演奏データを作成する。

3 Kagurame Phase-I

3.1 Kagurame Phase-I の概要

我々はこの事例ベースの演奏表情生成手法を用いた演奏表情生成システム“Kagurame”の構築を行なっている。Kagurameの第一段階であるKagurame Phase-Iでは、事例ベースの演奏表情生成手法の有効性を確認することを目的としているため、システムの構成は非常にシンプルにした。

Kagurame Phase-Iの主な特徴と機能は以下の通りである。

- 対象曲は単旋律のみ
- 楽曲の構造はあらかじめ人間が入力
- 演奏状況は演奏スタイルのみ

単旋律とは和音を含まないという意味である。複旋律にすると類似性を評価する際に考慮しなければ

ならない要素が増えるため、本システムでは単旋律の曲のみを対象とする。対象曲や参考データ中の曲の構造情報(構成要素の分割)は自動的に取得するのが非常に困難なので、本システムは楽譜情報と同様に人間が入力して与えることとした。演奏状況情報は演奏データのスタイルのみである。

3.2 取り扱う演奏表情

Kagurame Phase-Iでは以下の3種類の演奏表情を取り扱う。

- 局所テンポ
- 音ごとの音長
- 音ごとのヴェロシティ

音符*i*での局所テンポ T_i は音ごとのテンポのことである。これは次の式で求められる。

$$T_i = \frac{b_{i+1} - b_i}{t_{i+1}^{on} - t_i^{on}}$$

b_i (拍)は*i*番目の音符の拍位置である。 t_i^{on} (秒)は*i*番目の音符が演奏された時刻、つまり演奏者が鍵盤を叩いた時刻である。拍位置および演奏時刻は曲の先頭からの絶対位置である。Phase-Iでは単旋律の曲を対象としているので、複数の音符が同じ時刻に演奏されることはない。なお、通常テンポは分を基準としているため、上の式を60倍した値を用いるが、本研究では秒単位で計算を行なうためそのままの数値を利用している。

音ごとの音長 D_i は以下の式で求める。

$$D_i = \frac{t_i^{off} - t_i^{on}}{l_i} \cdot T_i$$

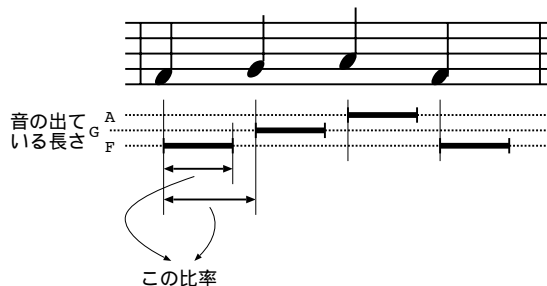


図6: 音ごとの音長

l_i (拍) は音符 i の楽譜上での長さを拍数で表したものである。4 分の 4 拍子の場合、4 分音符が 1, 8 分音符が 0.5 となる。 t_i^{off} (秒) は音符 i の離鍵の時刻である。 D_i は楽譜上に実際に書かれている音符の長さとして実際に演奏者が演奏した時間の比率を表す。この数値が短いとスタッカートになり長くなるとレガートになる。

音ごとのヴェロシティは演奏時の打鍵の強さを表す数値である。ヴェロシティは 0 から 127 までの数値で表される。打鍵が強いほどヴェロシティの数値は大きくなる。ヴェロシティ値は電子楽器から直接獲得することができる。

3.3 Kagurame Phase-I の構成

Kagurame Phase-I のシステム構成は図 7 に示す通りである。

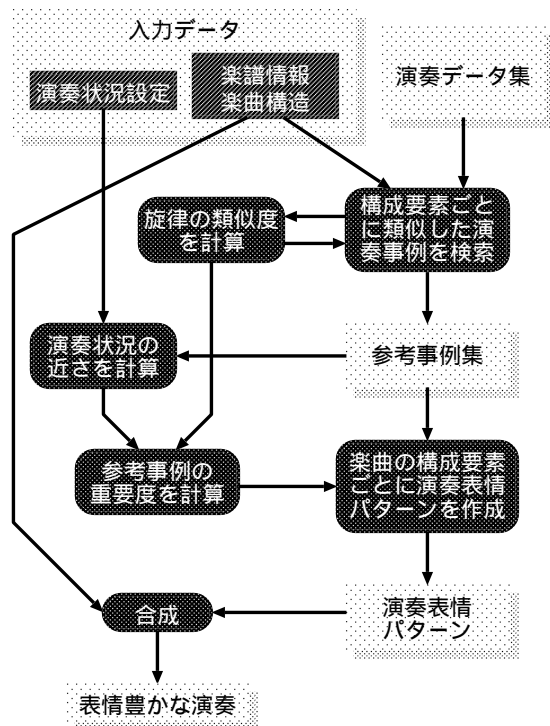


図 7: Kagurame Phase-I のシステム構成

入力データ 人間の演奏データには標準 MIDI ファイル (SMF) 形式 [10] で記録したシーケンスデータを利用する。SMF は鍵盤の打鍵と離鍵の時刻及び打鍵した音のキーと強さを時間順に記録したものである。このフォーマットのデータは電子楽器と計算機の専用ソフトを利用することで容易に収集が可能

であり、演奏情報の記録形式として一般に広く普及している。

また、楽譜情報及び楽曲構造の記述には独自フォーマットのテキストファイルを利用している。

参考事例の検索 参考事例の検索では、対象曲の構成要素ごとに参考事例中に含まれる全ての構成要素との比較を行ない類似度が高かった構成要素を参考事例として利用している。構成要素の類似性の評価には以下の情報を利用している。

- 構成要素の拍数
- 平均音高の差
- 前後の構成要素との関係
- 音度の分布

あまりにも拍数がことなる旋律は、仮に似ていたとしても参考事例には適さないで類似度は低くなるようにしている。平均音高の差は、構成要素の前半と後半の音高にどの程度の差があるかを表している。これは上昇系か下降系かといった旋律の外形的な要素も含まれている。これについては、本来なら音符ごとの変化から求めるべきであるが、構成要素の長さが長くなると旋律の全体の傾向を見るのが困難なので、構成要素内の前半と後半での音高で比較している。前後の構成要素との関係は隣接する構成要素との間の音高の変化である。音度の分布は構成要素に含まれる音階ごとの音符の分布である。これにより、構成要素のコードの傾向がわかる。分布に近いものは音階も近くなる。

事例の重要度の計算 参考事例の重要度は旋律の類似度と演奏状況の近さから計算する。旋律の類似度は参考事例の検索の際に用いた旋律の類似度と同じ値を利用している。演奏状況の近さは、対象曲の生成時に指定した演奏スタイルと、演奏データの演奏スタイルを比較し、同じであれば重要度を高くしている。

演奏表情パターンの作成 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。まず、参考事例の演奏データとを比較して参考事例と楽譜とのずれを計算する。次に、ずれの数値から 3.2 節の方法でそれぞれの演奏表情を取り出す。さ

らにそれぞれの演奏表情の値と、一つ上の階層の構成要素の平均値との比をとり、参考事例の演奏表情パターンとする。参考事例ごとの演奏表情パターンを重要度で重み付けをして傾斜平均をとり、その結果を構成要素の演奏表情パターンとする。

演奏表情生成部 各階層の構成要素の演奏表情パターンを全て対象曲に適用することで対象曲の演奏表情の生成が行なわれる。演奏表情パターンは比率で表されているので、掛け合わせることが可能である。従って音符ごとに、その音符が含まれている構成要素の演奏表情パターンを全部掛け合わせ、最後に曲全体のテンポやヴェロシティの平均値を掛け合することで対象曲の演奏表情が生成される。曲全体のテンポやヴェロシティは最上層の参考事例の平均から求める。

4 評価実験

Kagurame Phase-I に生成させた演奏と人間の演奏を比較し、Kagurame Phase-I の生成した演奏について評価を行なった。

参考データとして Czerny の “8 小節の練習曲” から 4 分の 4 拍子及び 4 分の 2 拍子の曲を 6 曲選択し、それぞれについて A(ロマン派風) と B(古

典風) の二種類のスタイルの演奏を行なった MIDI データ、計 12 本を利用した。このうちから 1 曲を対象曲、残りの 5 曲 10 本の演奏データをサンプルデータとしてオープンテストを行なった。まず Kagurame Phase-I を用いて対象曲にスタイル A、スタイル B の二種類の演奏表情を与えたデータを生成し、生成された演奏表情と対象曲の演奏データの演奏表情とを比較した。クロスバリデーションにより対象曲とサンプルデータセットを順次変更し 6 曲全てについて生成実験を行なった。

実験結果は図 8 の通りである。人間が演奏したデータとの比は、音符ごとに Kagurame Phase-I のデータと人間のデータの比率を計算し、曲全体で平均をとったものである (式 1)。

$$result = \frac{1}{n} \cdot \sum_{i <= n} abs \left(\frac{Exp_i^{KaguramePh-I}}{Exp_i^{humanperf.}} - 1 \right) \quad (1)$$

$Exp_i^{KaguramePh-I}$ は Kagurame Phase-I が生成した演奏データの i 番目の音符の演奏表情の値 (テンポ、音長、ヴェロシティのいずれか) であり、 $Exp_i^{humanperf.}$ は人間の演奏での演奏表情の値である。また、 n は曲中に含まれる音符の総数である。 $result$ が小さいほど、人間の演奏と Kagurame Phase-I の出力が近いことを表す。二つの演奏が全く同じであれば $result$ は 0 になる。

曲の番号	生成したデータの演奏スタイル	人間が演奏したスタイル A の演奏データとの比			人間が演奏したスタイル B の演奏データとの比			生成に要した時間
		テンポ	音長	ヴェロシティ	テンポ	音長	ヴェロシティ	
1	A	0.134	0.132	0.185	0.115	0.140	0.128	65.2
12	A	0.122	0.143	0.100	0.108	0.148	0.157	62.4
15	A	0.213	0.128	0.114	0.107	0.175	0.157	61.6
24	A	0.155	0.292	0.114	0.117	0.430	0.127	61.8
37	A	0.131	0.135	0.108	0.109	0.143	0.133	52.6
65	A	0.124	0.086	0.086	0.117	0.167	0.132	82.0
1	B	0.159	0.122	0.197	0.069	0.136	0.132	64.5
12	B	0.129	0.156	0.100	0.072	0.145	0.130	62.0
15	B	0.237	0.130	0.124	0.074	0.153	0.131	60.9
24	B	0.187	0.296	0.120	0.082	0.438	0.125	61.7
37	B	0.158	0.130	0.108	0.071	0.124	0.106	52.7
65	B	0.219	0.128	0.142	0.052	0.104	0.078	82.1

図 8: Kagurame Phase-I による演奏表情の生成実験

計算時間は Alphaserver 1200 を利用して生成を行なった際の CPU 時間を秒単位で表したものである。

この結果を見ると、一部の演奏データでは、スタイルが違う演奏データより同じ演奏データの方がより 0 に近い値になっている部分が多く、入力として与えた演奏スタイルに近い演奏表情が生成できたといえる。しかしながら、あまり違いの見られないデータもあり、これらは有効な事例が発見できなかったのではないと思われる。

5 おわりに

本論文では事例ベースによる演奏表情の自動生成手法について概説し、事例ベースの演奏表情生成システム Kagurame Phase-I の実装および実験について報告した。Kagurame Phase-I の生成した演奏データを分析することにより、事例ベースの演奏表情手法が表情豊かな演奏の生成に有効であることを示した。また、スタイルの設定を変更することで同じ曲に対し多種多様な演奏表情を生成することが可能であることを確認した。

この手法では、参考になる旋律の選択方法や重要度の算出方法によって、生成する演奏表情が変化する。Kagurame Phase-I では、参考とする演奏事例の検索や個々の参考事例の重要度の計算に単純なアルゴリズムを用いており、参考事例の選定が効果的に行なわれているとはいえない。より適切な演奏表情の生成を行なうためには、旋律の類似性や重要度の客観的な評価方法について検討する必要がある。

今回の実験では数値的な比較により生成された演奏の評価を行なったが、演奏の評価は必ずしも数値的だけで評価できるものではない。心理実験など別の角度からの評価を行なう必要がある。

Kagurame Phase-I では、類似旋律の検索や重要度の評価、演奏表情の分析などを容易にするため、対象を単旋律の曲に限定している。一方で通常の楽曲は殆どが和音を含んでいるため現在のシステムを利用して、一般の楽曲を演奏させ楽しむことは不可能である。今後、和音を含む旋律の類似性の評価方法や演奏表情の分析手法などについて検討し、Kagurame の次の Phase では一般の曲に対して適用できるようにする予定である。

また、現在は楽曲構造情報の入力を人手で行なっているが、データの書式が特殊であるため作成に手間がかかるという問題点がある。したがって演奏データとして用意できる楽曲の量が限られてしまう。本研究で利用している楽曲構造情報は構成要素の階層的な分割に限られており、これらは楽譜上の繰り返しなどからある程度類推することが可能であると考えられる。今後は楽曲構造の解析手法を検討し自動化することでデータの収集を容易にすることが望まれる。

参考文献

- [1] 青野裕司, 片寄晴弘, 井口征士, “重回帰分析を用いた演奏表現法の抽出”, 情報処理学会論文誌, vol. 38, No. 7, pp.1473-1481, 1997.
- [2] 白川健一, 小田安彦, 熊谷俊行, 梶川嘉延, 野村康雄, “演奏者情報を加味したピアノの自動演奏システムの構築: ~音符属性に着目した個人的解釈ルールの生成~”, 情報処理学会研究報告, 95-MUS-10, pp. 1-6, 1995.
- [3] 小田安彦, 白川健一, 村上豊, 梶川嘉延, 野村康雄, “演奏者情報を加味したピアノの自動演奏システムの構築: ~ニューラルネットワークによる局所部分に於ける演奏特徴の抽出~”, 情報処理学会研究報告, 95-MUS-10, pp. 7-12, 1995.
- [4] 山内敏彰, 野池賢二, 野瀬隆, 乾伸雄, 小谷善行, 西村怒彦, “Ga を用いたピアノ演奏の局所テンポ解析”, 情報処理学会研究報告, 96-MUS-14, pp. 1-6, 1996.
- [5] 田口友康, “自動演奏と演奏モデル:演奏の芸術性を目指して”, bit 別冊, コンピュータと音楽, 共立出版, pp.42-53, 1987.
- [6] 野池賢二, 西岡大祐, 高田正之, 小谷善行, “曲の構造情報から表情付けを行なう自動演奏”, 情報処理学会第 44 年全国大会, 6U-3, 1992.
- [7] 彌富あかね, 五十嵐滋, “音楽構造分析を用いたピアノ演奏の表情付け”, 情報処理学会第 50 年全国大会, 7S-6, 1995.
- [8] Gerhard Widmer, “Modeling the rational basis of musical expression”, Computer Music Journal, Vol. 19, No. 2, pp. 76-96, 1995.
- [9] L. Fryden and J. Sundberg, “Performance rules for melodies, origin, functions, purposes”, Proc. of the 1984 ICMC, pp. 221-224, 1984.
- [10] D. Oppenheim, “Standard MIDI Files 0.06”, 1988.
- [11] 鈴木泰山, 徳永健伸, 田中穂積, “事例ベースの演奏表情生成手法に関する研究”, 情報処理学会研究報告, 96-MUS-21, pp. 7-12, 1997.