

演奏表情付けに関する一検討

- 打ち込みとルールベースによる表情付けの比較

橋田光代 *1*2, 野池賢二 *1, 片寄晴弘 *1*3

*1 PRESTO/JST, *2 和歌山大学システム工学研究科,

*3 関西学院大学理工学部情報科学科

{hashida, noike, katayose}@ksc.kwansei.ac.jp

本稿では、演奏ルールを与えて音楽に表情を付けた場合と、人間がマニュアルで表情付けを行った場合における各パラメータの与え方の差異について述べる。比較のために、あらかじめ手入力（打ち込み）で作成した演奏事例を用意し、次にその演奏を目標としたルールを考案し、音高、発音時刻、音価のみ記述した Standard MIDI File（以下、SMF）に適用した。今回はショパンの「別れの曲」に記述されている記号情報、楽語について検討を行ったところ、その多くは一般化できるのではないだろうかという手ごたえがあった。

An Analysis of Music Performance Rendering: - Comparing a Rule-based rendering with a manual rendering

Mitsuyo HASHIDA *1*2, Kenji NOIKE *1, Haruhiro KATAYOSE *1*3

*1 PRESTO/JST, *2 Wakayama University

*3 Kwansei Gakuin University

{hashida, noike, katayose}@ksc.kwansei.ac.jp

This paper describes an analysis of music performance rendering of differences between manual rendering and rule-based rendering. We rendered two types of MIDI sequences; we first made a MIDI sequence by manual with complete expressions, and then made another one by rule-based processing for targeting the manual data. This time the information clearly including a score such as slur, dynamics, and articulations are considered. We could expect some of the rules created for Chopin's Etude Op.10-3 would generalize for other pieces...

1. はじめに

数年前から、演奏の表情付けシステムの評価の一環として、Rencon(Performance Rendering Contest)の準備が始められてきた。2002 年は、5 月の ICAD-RENCON¹、9 月の FIT-Rencon²が開催されるなど、Rencon にとっては、記念すべき年となった[平賀 2001]。

演奏の表情付けに関する研究は、音楽に関する基礎的な興味、計算機科学、また、応用面においても魅力的なテーマであり、日本、ヨーロッパなどで積極的な研究が行われている。

表情付けに計算機を利用した研究は 1980 年代の Sundberg らの演奏ルールに関する研究[Sundberg 1984]、Clynes らのパルスの階層的な表現による拍節表現にさかのぼることができる。1990 年からは、演奏ルールの学習（抽出）[Katayose 1990][Widmer 1996]、事例ベース推論[Suzuki 1999][Hirata 2002]。など、機械学習応用に焦点をおいた研究が積極的に行われるようになってきている。

1 <http://shouchan.ei.tuat.ac.jp/~rencon/ICAD2002/>

2 <http://shouchan.ei.tuat.ac.jp/~rencon/FIT2002/>

このような中、ルールによる演奏生成が十分に機能する範囲と限界、どのような対象に対して、事例ベースが適するか、等については十分議論されてこなかった。我々は、ルールによる演奏システム "Yutaka" を用い、アナリシスパイシンセシスの手法で、この問題に取り組むことにした。以下、研究全体のアプローチ(第2章)、Yutaka の概要(第3章)、データの解析と検討について述べる(第4章)。

2. 処理の概要

まず、打ち込みによる MIDI データを作成し、ルールで生成する表情付き演奏に対する自分の目標を立てる。次に、ルールによるレンダリングエンジン "Yutaka" を用いて、基本的な音高、ヴェロシティ、音価のみを入力したシンプルな SMF に演奏ルールを適用し、その結果を打ち込みデータにどこまで近づけられるか検証する。

本研究で採用した処理手順を図 1 に示す。

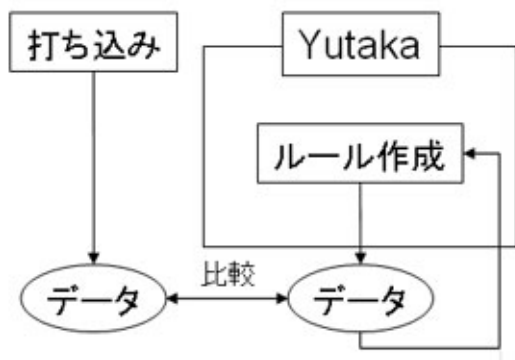


図 1

この図に示すように、まず、

1. 打ち込みによる MIDI データの作成を行い、
2. レンダリングエンジン "Yutaka" を用いたルールと、
3. 演奏データを作成し、
4. 聴取の上、音楽的に許容できない場合、3) 4) を繰り返す。

生成と評価を繰り返して解析する手法は、アナリシスパイシンセシスと呼ばれる。パフォーマンスレンダリングにアナリシスパイシンセシスを用いる手

法は 1980 年代に Sundberg らが採用した古典的な手法である。Sundberg らは Flydon (演奏者) によるナイーブな判断を元にルールとして与える制御パラメータの設定を行っている。これに対し、本研究では、まず、目標事例 (ショパン / 別れの曲冒頭 21 小節) を完全に作り上げてから、それに近づけるデータ生成を行っている。具体的なりファレンスを利用することで、より客観的な視点で、ルールベースシステムがカバーできる範囲と限界を提示することを目的とする。なお 4 章では、Sundberg らが提示している原則ルールの問題点についても論じる。

3. 演奏生成エンジン "Yutaka"

Yutaka は、ルールベースによる表情付け演奏生成エンジンである。

ユーザはスコアファイルとルールファイルを用意する。学習機構はなく、ユーザが自分で思うとおりのルールを作成する (図 2)。これにより、演奏意図をルールでどこまで記述できるかという限界を知ることができる。

Yutaka によって制御できる演奏パラメータは、現在以下の 5 つがある。

- 各音符の打鍵強度 (1 ~ 127, MIDI ヴェロシティ値そのもの)
- 各音符の打鍵時刻ずらし値 (楽譜上の時刻に対して ± 値を指定)
- 各音符の打鍵離鍵時間比率 (レガート演奏を 1.0 とした比率)
- 演奏速度 (bpm で指定)
- ダンパーペダル踏度 (0 ~ 127)

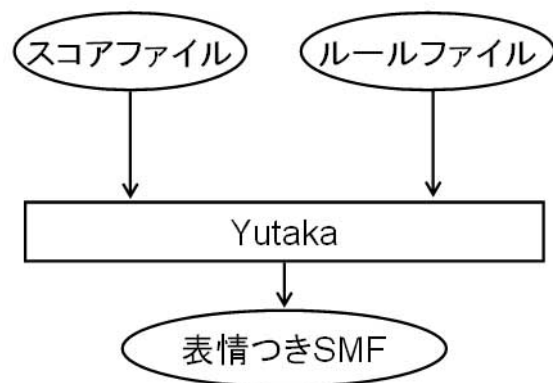


図 2 "Yutaka" システム図

3.1 スコアファイル

スコアファイルでは、以下の 4 つの情報を記述する。それぞれの楽譜情報については、ユーザが任意に命名することができる。

- 楽曲基本情報（クレジット、分解能）
- ノート基本情報（音高、発音時刻、音価）
- 各ノートに付随する楽譜情報（声部、強弱、演奏記号 など）
- 楽曲全体に関する楽譜情報（ペダル、楽想、ダイナミクス など）

スコアファイルは以下のように記述される。

```
composer F.Chopin
songname Chanson_de_l'adieu_op64_no2
key E-Dur
Tempo quarter 2.00000
==<1>== 24 * 2
;-----
; Note, Time, Duration (rules, ...)
;-----
B3 37 12 (g0, vel 57, g1, m1, _legato_start, slur->)
;-----
!! 49 (pedal off, t-start)
E2 49 24 (g0, vel 48, g2, b1)
E4 49 12 (g0, vel 59, g1, m1)
G#3 49 6 (g0, vel 37, g1, i1, cres->)
!! 52 (pedal on)
B2 55 12 (g0, vel 42, g2, b1, ac)
B3 55 6 (g0, vel 43, i1)
...
```

3.2 ルールファイル

ルールファイルでは、スコアファイルで命名した各演奏表現情報について、どのような場合にどのようなパラメータを変化させるかという演奏ルールを記述する。演奏ルールは、

ルール条件： 値代入 or 式

という形式で記述し、ある時刻の一音、またはある範囲に対して、定数または一次式を与える。一次式は、式を直接記述することも適用終了時までの増減値を与えて補間させることもできる。ある 1 音に複数のルールが適用された場合は、ルールの記述順に、それぞれ得られた増減値が加算されていく。したがって、ルールの優先順位もユーザの任意に決定することができる。

以下に、演奏ルールの例を示す。

- ・クレッシェンドの始めから終わりまで、Velocity を徐々

に 20 増やす：

cres-start, cres-end: VEL -> +20

- ・テヌートのとき、Velocity をもとの 1.3 倍、音価を 2 bpm

延ばす： ten: VEL = * 1.3, DUR = + 2

4. データの解析と検討

4.1 データの生成

今回の生成では、F. ショパン「別れの曲」の冒頭 24 小節を採用した。生成される MIDI データの基本情報として、1 小節あたりの分解能は 480、基本再生速度は 1 分間当たり四分音符 30 回としている。



図 3 冒頭部分



図 4 後半部分

打ち込みでの生成にあたって、その演奏表現は、楽譜に明記された指示にできるだけ忠実に従った。その上で、「自分だったらこのように弾きたい」という主観的な理想分を加味している。

ルール生成では、この打ち込みに近いデータを作成するために、下記のようなことを行った。

- もっとも単純なグルーピング構造として、右手 3 声（旋律、時折表れる副旋律、内声）と左手 1 声にわけた
- アクセントなど、音符に固有の演奏記号から

ルールを適用した。

- のペダル入力については、仕様上、直接 ON/OFF を時刻で指定した。

なお、打ち込みデータの制作期間は約 1 週間、"Yutaka"によるルール生成データは 4 週間かけた。

4.2 データの解析

今回用いた"Yutaka"では、前述の通り、現時点ではルールに対して定数を代入するか一次曲線を与えるかのどちらかである。そこで、一般に楽譜記号に対して定数や一次曲線を与えただけでどこまでの表現が可能かを考察すると、次のようなことが挙げられる。

まず、音量(Velocity)、音価(Duration)などに直接関わる記号については、以下のルールを適用した。

種類	今回適用したルール
強弱記号(p, fなど)	_p_ : VEL = - 10 _ff_ : VEL = + 25
強弱変化(クレッシェンド, ディミヌエンド)	cres->, <-cres : VEL -> +10 dim->, <-dim : VEL -> -10
演奏記号(スタッカート, テヌートなど)	st : DUR = * 0.5 ten : VEL = * 1.2, DUR = + 1 _legato_ : DUR = + 1

強弱変化は、その指定期間に関わらず到達量を一律にした。その結果、変化期間が 1 拍のみでも 1 小節あっても不満なく効果を感じることができた。楽譜指定が特別だったのは後半のクレッシェンドからフォルティシモに至るまでであるが、ここではクレッシェンドが 2 度指定されているため、これらが合わさることによって、基本の 2 倍の増加量を得ることができた。また、このようにルールを重複させることによって、Velocity の増加線を一次曲線から二次曲線的にすることができる。

テンポに関しては、以下のルールを適用した。

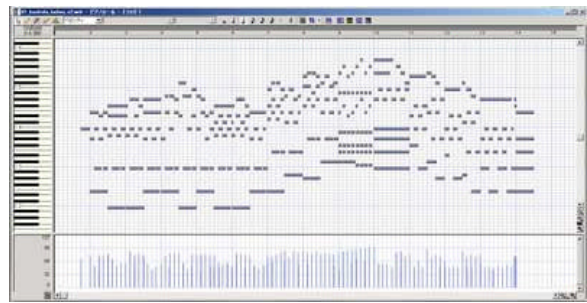


図 5 Velocity 変化 (打ち込み)

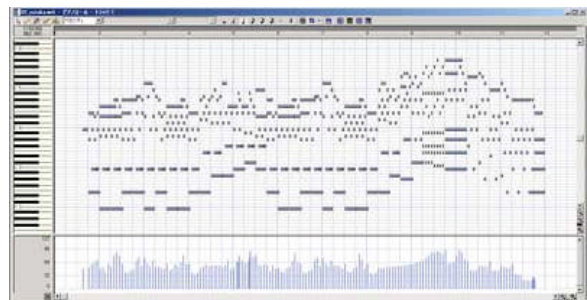


図 6 Velocity 変化 (Yutaka)

テンポ記号	今回適用したルール
stretto (だんだん速く)	t-start, t-end : TEMPO -> +3 stretto->, <-stretto : TEMPO -> +10
ritenuto (ただちに遅く)	riten->, <-riten : TEMPO -> -5
rall. (だんだん遅く)	rall->, <-rall : TEMPO -> -15
フェルマータ	ferm : TEMPO = * 0.5
a tempo (元の速さで)	atempo : TEMPO = 30

今回の Yutaka では一次曲線を利用しているが、打ち込みではテンポ変化に一次曲線を与えずとも十分な効果を得ることができた。図 7 に示すように、ある強拍の始めと直前の十六分音符で、それまでの速度の約 1 割程度テンポ値を下げているが、これらの多くはフレーズ(多くは実質的にスラーの始めから終わりまで)の冒頭部分に相当する。accel. や今回の stretto のように、だんだんテンポを上げていく場合などは、1 拍目が開始点なら 2 拍目までには到達点の約 8 割に達していないと「テンポ変化のかけり

が遅い」と感じた。一方、テンポが遅くなる場合は一次曲線でも違和感はそれほどなかった。打ち込みでは定数を代入しているが、少し視点を広げれば、ステップ進行的な一次曲線であるということもできる。



図 7 テンポ (打ち込み)

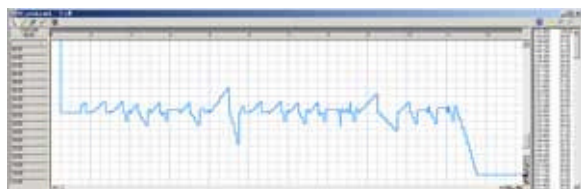


図 8 テンポ (Yutaka)

今回、ルール適用において最も重点を置いたのはスラー表記である。通常考えられているスラーの表現とは、主にレガート(legato)のことであり、制御パラメータについても音価のみ取り上げられることが多い。しかし、特にピアノ楽器の場合、スラーの括り方はそのままフレージングと直結することがしばしばである。そこで、次のような仮説を立ててみた。仮にスラー部分が、そのままあるフレーズであった場合、フレーズ内で最も表現が深くなる部分とは、

	今回適用したルール
音量	slur-start, slur-peak: VEL -> +8 slur-peak, slur-end: VEL -> -8
音価	slur-start, slur-end: DUR = + 2 slur-end: DUR = - 1
テンポ	slur-start, slur-peak: TEMPO -> +3 slur-peak, slur-end: TEMPO -> -4

スラーが描く弧の頂点に当たり、演奏者は無意識にそこがフレーズの頂点であると理解しているのではないだろうか。Sundberg らの研究では、スラー表記

については触れていないが、フレーズに対しては立ち上がりや終わりの部分に向かってテンポや音量の変化が現れている。そこで、ルール生成にあたっては、スラーの開始点、終了点のほかに、楽譜を見ながら頂点となる位置を設定した。そうして上記のパラメータを与えた結果、かなりの割合で演奏の「自然らしさ」が向上し、打ち込みデータのように揺れのある演奏をある程度出すことができた。

4.3 音楽面からの検証

今回は、楽譜で明確に記述された楽譜情報についてのルール考案に重点を置いた。実際に影響を受けたパラメータは主に Velocity と Duration, Tempo である。この点に関しては、予想以上にルール適用が効率良く働き、おかしな部分も多少あるが比較的自然な表情のついた演奏が生成できた。

しかし、右手部分 3 声と左手 1 声以外に読み取れる複雑なグルーピング構造や、伴奏部分での定型的な音列パターン、旋律の推移に伴う表情変化などに対するルール適用は行わなかった。そのため、その声部またはグループ全体に適用されるルール以外の情報が反映されなかったノートも多く、打ち込みデータと比べて、全体的にはまだ大きく不足していると感じる部分があった。

スラーとして表記されたフレーズであればルールも適用しやすいが、そうでない場合は、任意にフレージングする必要がある。それらについては、「Phrase」という隠し表情ルールを設けることでスラー表記と別に考慮することができるだろう。

なお、FIT-RECON(片寄 2002)に、打ち込みデータとルールによる演奏の両者をエントリーしたところ、一般参加者による評価では、ルールによる演奏の方が人気を得た³。その要因としては、音源の差を考えている⁴。

5. おわりに

ここでは、ルールベースでの表情付けについて、について述べ、比較的単純な定数や一次式でも可能なルールとそうでないものとを区別した。今回適用

³ 専門審査員は、フレージング等の視点で、打ち込みデータが優れると講評があった

⁴ 作り込みにおいては MIDI ピアノでも演奏を確認していた。

させたルールはショパン「別れの曲」に表記されたものに対応しているが，楽譜に明記される情報であるということから，おそらく一般化できるのではないだろうか．今後，さらに多くのデータに対して検証を進めていきたい．

謝辞

本研究は，科学技術振興事業団さきがけ研究 2 1 「協調と制御」領域の研究テーマとして実施されました．

6. 参考文献

1. [Sundberg 1984]L. Fryden and J.Sundberg:
Performance Rules for Melodies. Origin, Functions,
Purposes, Proc. Intl. Computer Music Conf.,pp.
221-225 (1984)
2. "The Science of Music Performance",
http://www.speech.kth.se/music/performance/performance_rules.html
3. [平賀 2001] 平賀瑠美，平田圭二，片寄晴弘：蓮根：目指せ世界一のピアニスト，情報処理，Vol.43, No.2,
pp.136-141 (2001)
4. [片寄 2002] 片寄，平賀，平田，野池，橋田：
ICAD-RENCON - 報告と課題 - ，SIGMUS-48(本報告資料集) 2002