

“ピアニスト”を演奏するインタフェース: *punin*

片寄 晴弘^{†‡} 奥平 啓太[†] 野池 賢二[‡]

[†]関西学院大学理工学部 [‡]科技団さきがけ研究 21

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

[†]http://ist.ksc.kwansei.ac.jp/katayose/

1. はじめに

音楽演奏を行うには、まずは、演奏技術の修得が必要である。「ピアニスト誰その真似をしてみたい」というような夢を持ったとしても、手が動かないので、できないということも少なくない。本稿では、そのような夢の実現を支援する演奏インタフェースについて紹介する。

我々は、1991年に指1本の打鍵動作で、テンポ・音量など指揮的な演奏表現感覚を楽しむためのトイシステム(One Finger Piano)を作った。指一本の打鍵操作では、テンポを変化させたいのか“間”を挿入をしたいのかの区別がつかなかったため、翌年、2本指の演奏動作で明示的にその区別を行うTFPを作り、教育分野での活用をはかってきた[1]。

今回、TFPの設計を一から見直し、「ピアニスト誰その演奏」を下敷として、指一本で自分が好きなように演奏表現を行う演奏インタフェース: *punin* を設計した。以下、システムの概要、インプリメントについて紹介する。

2. *punin* の演奏インタフェース

図1にシステムの概要を示す。この図に示すように、*punin* では、演奏の元となる演奏deviationデータ¹(図2)を読み込み、ユーザの打鍵にしたがって演奏を実施する。基本的な動作、操作方法を以下に示す。

- 1) 指一本の打鍵でのテンポ・“間”、音量の制御を行う(計算機キーボードを利用する際には、キーあるいはマウス位置によって、音量を弾き分ける。)
- 2) 既存の演奏例から抽出した演奏deviationデータを用いることにより、名演奏者を指揮するような表現感覚を楽しむ。

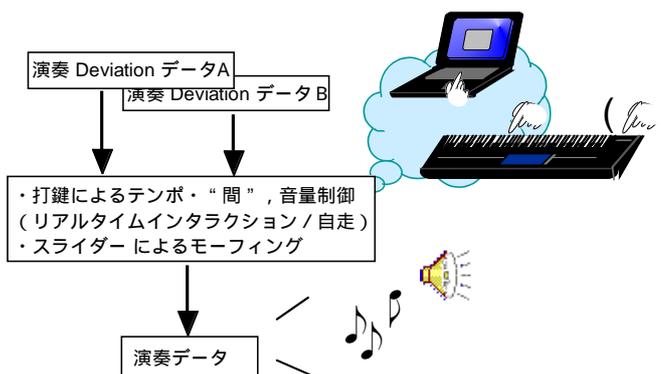


図1 システムの概要

.....
2.00 BPM 126 4
2.00 (E3 m 78 3.00 0.1)
=2
1.00 BPM 128 4
1.00 (C#4 m 76 0.75 0.1) (E1 a 60 1.00 0.1)
1.75 (+0.1 D4 m 77 0.25 -0.1)
1.00 BPM 130 4
2.00 (B3 m 75 1.00 0.1) (+0.1 G#3 a 56 1.00 0.1)
3.00 BPM 127 4
3.00 (0.04 B3 m 72 1.00 -0.04) (G#3 a 56 1.00 -0.04) (D3 a 57 1.00 -0.04)
=3
1.00 BPM 127 4
1.00 (B3 m 77 2.00 -0.04) (G#3 a 57 2.00 -0.04) (0.04 D3 a 57 2.00 -0.04)
3.00 BPM 129 4
3.00 (F#4 m 75 1.00 -0.04) (0.04 D4 a 54 1.00 -0.03)
=4
9.00 BPM 127
9.00 (D#4 73 0.75 -0.03) (+0.04 C4 65 0.75 -0.03)

図2 演奏deviationデータ

拍毎のテンポ、各音のオンセット時間、長さの規準拍長からのズレ、音量(Velocity値)、オプションで、メロディや伴奏データを指し示すフラグなどが記載される。

- 3)演奏deviationパラメータ²の内、どのパラメータをユーザがコントロールするかを指定することにより、さまざまな形での演奏表現を楽しむ³。
- 4) 二つの演奏deviationデータを用い、スライダー

¹必ずしも、演奏表情を含んだデータである必要はない。

²拍毎のテンポ、拍内の各音のオンセットと音長のズレ、音量

³拍打をオートにした場合、ピアニストAとピアニストBの演奏をモジュレーション・フォイルドでモーフィングするといったことも可能である。

(ベンダー)で割合を変えることで,リアルタイム演奏モーフィングを行う⁴.

- 5) PID 制御[2]に基いた操作感の設定が可能である.
- 6) 適当なデータを用意することで,メロディ入力に対し,伴奏をつけることができる⁵.

3. 演奏制御方式と“間”の表現

One Finger Pianoや指揮システムなど,拍単位で演奏データのスケジューリングを行うシステムにおいては,当該拍(とその拍に含まれるより短い音)の音符の発消音時刻は,前拍からその拍までにかけての時間(IOI: Inter Onset Interval)に基づき配置がなされる.実際の演奏において IOI は,テンポに関係する時間とその拍における“間”が合算された時間として計測されるため,“間”が加えられた演奏には,うまく対処できないという問題があった.

punin では,PID制御と演奏 deviation ファイルの利用により,指一本の単純な打鍵操作に基づいてテンポの“間”の弾き分けが可能となっている.

3.1 PID 制御による演奏制御

PID 制御は,目標値と現状の値の偏差に相当する P (比例) 値,偏差の積分値を反映する I (積分) 値,偏差の変化量を反映する D (微分) 値に適切な重みを配することで,制御対象の収束時間の改善を目指す古典的な制御手法の一つである.

punin の最も基本的な使用法(テンポ,音量のバランスをユーザが与える場合)において,次の拍打までのテンポは,演奏 deviation ファイルに記載されたテンポ(P 値),ユーザが与えるものとして,現在の拍から(さかのぼって)指定した拍数の平均テンポ(I 値),現在のテンポと直前テンポの差分(D 値)のそれぞれに対し,指定した重みを掛け合わせることによって計算がなされる⁶.重みを変えることによってさまざまな追従特性が得られる.例えば,P 値を重視すれば,真似をしたい演奏に引っ張られがちな演奏が生成される.また,D 値を重視すれば,全体の演奏傾向より局所的な拍打表現の影響を受け

やすい演奏が生成される.拍打に関しては,指定した半分の間隔で刻むことも可能である.さらに,そこでの演奏表現は,システムの演奏スケジューリングに反映される.

3.2 “間”の表現

punin では,大きく分けて三つの“間”の表現方法がある.

まずは,“間”の入った演奏データから取得した演奏 deviation ファイルと,表情のついていないデータと適宜,モーフィングを行うという方法である.比較的,効果の得やすい方法であるが,用意した演奏データの表現の制約を大きく受けるといった欠点がある.

二つ目の方法は,半分の拍打を適宜利用することである.I 値,D 値をうまく設定してやることで,指揮に似た感覚で,テンポ,“間”の切り分けを行うことができる.

もう一つの方法は,鍵盤を離すタイミングに対し積極的な意味付けを与える方法である.鍵盤を押している時間が現状のテンポ間隔より長い場合には,それを明示的な“間”と判断し,次の拍打までスケジューリングを中断し,テンポに関しては,直前のものを使うことで,非常に自然な演奏感が得られるようになった.

4. おわりに

本稿では,「ピアニスト誰その演奏」を下敷として,指一本で自分が好きなように演奏表現を行う演奏インタフェース: **punin** を紹介した.ここでは,“間”の表現のためのインタフェースを中心に,設計における考慮点について述べてきた.紙面上で表現することは難しいが,名ピアニストの表現を自分なりに“演奏する”ことは非常に痛快である.本シンポジウムに来られた方は,是非体験していただきたい.実施例については,<http://ist.ksc.kwansei.ac.jp/katayose/punin/> に所収されているビデオを参照されたい.

参考文献

- [1] 片寄晴弘, 竹内好宏, 上符裕一, 井口征士: TFP の改良と教育利用における評価, 情報処理学会音楽情報科学研究会研究報告 96-MUS-16, 21-25 (1996)
- [2] 須田信英, PID 制御 - システム制御情報ライブラリ -, 朝倉書店 (1992)

⁴ 適当な範囲で外挿も可能である.

⁵ 指定拍の半分の刻みを行う楽曲(次章参照)に対して可能である.自動伴奏と呼ばれる研究分野が存在する.そこでは,ソリストがどの部分を弾いているのかの監視し,それに合わせた伴奏データのスケジューリングがなされている.

⁶ 厳密な定義での PID 制御とは異なるが,比例量,積分量,微分量の重みで制御を実施すること,直感的にもわかりやすいという点で同質のものである.