

# 楽器構成に着目した楽曲サムネイルとプレイリスト生成機能つき音楽プレイヤー

戸谷直之<sup>†</sup> 北原鉄朗<sup>†,††</sup> 片寄晴弘<sup>†,††</sup>

## Music Player with Music Thumbnailing and Playlist Generation Functions based on Instrumentation

NAOYUKI TOTANI,<sup>†</sup> TETSURO KITAHARA<sup>†,††</sup> and HARUHIRO KATAYOSE<sup>†,††</sup>

### 1. はじめに

近年、定額制音楽配信サービスが普及しつつある。定額制音楽配信サービスでは、様々な楽曲を無制限に聴けるので、音楽を「流し聴き」するスタイルがますます増加すると考えられる。流し聴きを支援する機能として多くの音楽プレイヤーにはシャッフル再生機能が搭載されているが、シャッフル再生機能でランダムに選ばれる楽曲群には、その雰囲気統一感がなく、必ずしも有用とは言い難い。自分の置かれた状況や雰囲気にマッチした楽曲を流し聴きするのに有用な方法は、自分で条件に合った楽曲を何曲か選び出すインターフェイスとその選曲に雰囲気の近い楽曲を芋づる式に検索する機能を提供することである。これらの機能は、既存の音楽再生インターフェイスやプレイリスト生成の研究<sup>1)~3)</sup>でもすでに実現されているが、楽曲の雰囲気をどういった特徴量で表現すべきかという問題に対しては十分な解は得られていない。

我々は、楽曲の雰囲気を記述する音楽要素として、楽器構成(どのような楽器で演奏されているか)が重要な役割を担うと考えている。同一楽曲であっても異なる楽器で演奏すると、聴取時の印象全く異なるものになる場合がある。こういったことから楽器構成と楽曲の雰囲気との関連が示唆される。このような考えの下、我々はこれまで、楽器構成を多重奏の音響信号から自動的に推定する技術開発に取り組んできた<sup>4)</sup>。しかし、この技術を応用した音楽再生インターフェイスの設計には至っていなかった。

本稿では、この楽器構成推定技術を用いて開発したプレイリスト生成機能つき音楽プレイヤーについて述べる。このプレイヤーを立ち上げると、サーバ上に存在する楽曲リストが、楽器構成を表す楽器アイコン付きで表示される。ユーザはそのアイコンを参考にしな

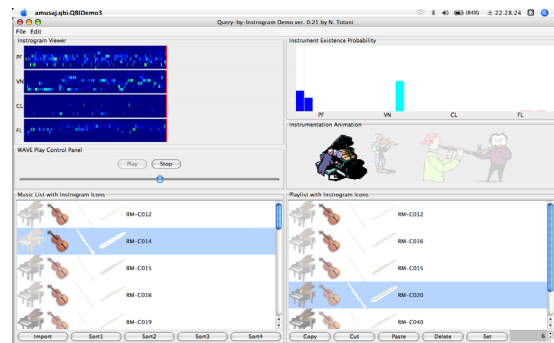


図1 本研究で開発した音楽プレイヤー。左下のリストにサーバ上の楽器一覧が表示され、右下がプレイリストとなっている。

がら、聴きたい楽曲を何曲か選ぶ。そうすると、その楽曲に楽器構成の観点から類似する楽曲を検索し、プレイリストを生成する。

### 2. 楽曲サムネイルとプレイリスト生成機能つき音楽プレイヤー

#### 2.1 システムの概要

本プレイヤーは、定額制音楽配信サービス提供者がユーザに配布するソフトウェアを想定して開発している。そのため、音楽ファイルやメタ情報などの情報はすべてサーバ上に存在するものとする。こうすることで、時間のかかるメタデータの生成はあらかじめ終わらせておくことができる。後述の Instrogram の生成は計算量が多いため、サーバに楽曲を登録の際に行っておくものとする。なお、実際の運用では本プレイヤー利用時にユーザ認証を行うことが想定されるが、現在のプロトタイプではユーザ認証機能は搭載していない。

本プレイヤーの画面を図1に示す。本プレイヤーを起動後、[Import] ボタンを押すと、サーバから楽曲リストを取得し、ウィンドウの左下に一覧表示する。曲名の横には各楽器のアイコンが表示され、アイコンの色の濃さがその楽器の優勢度を表す。[Sort] ボタンを用いることで、指定した楽器の優勢度順に楽曲リスト

<sup>†</sup> 関西学院大学 / Kwansei Gakuin University

<sup>††</sup> 科学技術振興機構 戦略的創造研究推進事業 CrestMuse プロジェクト / CrestMuse Project, CREST, JST

を並び替えることもできる。ウィンドウの右下のリストがプレイリストになっている。右下のテキストボックスでプレイリストの曲数を指定し、プレイリスト内の好きな項目に、聴きたい楽曲を左の一覧リストからドラッグアンドドロップする。たとえば、最初のうちは弦楽曲を聴いて、最後はピアノ曲で締めたいと思えば、プレイリストの一番上に弦楽曲を、一番下にピアノ曲を配置する。このように、自分の聴きたい楽曲の雰囲気（ここでは楽器構成に限定）のおおまかな移り変わりを指定した上で、[Set] ボタンを押すと、プレイヤーは自動的に、プレイリスト内の未配置の項目に、楽曲の雰囲気（楽器構成）ができるだけ連続的に変化するように楽曲を探索し配置する。上の例では、楽器構成が徐々に弦楽系からピアノに変わるように楽曲を選び出す。このようにしてプレイリストが生成されたら、楽曲リストをダブルクリックすることで楽曲の再生がスタートする。再生中は、楽器構成の時間的な移り変わりを棒グラフやアニメーションで可視化する。

## 2.2 楽曲サムネイルの生成方法

楽曲サムネイルは、あらかじめ計算しておいた各楽器の優勢度を楽器アイコンの透明度に反映させる（優勢度が高いほど透明度を低くする）ことで実現する。以下、各楽器の優勢度を Instrogram<sup>4)</sup> に基づいて計算する方法を述べる。

まず、各楽曲に対して Instrogram を求める。Instrogram とは、楽器存在確率の時間・周波数表現である。いま、対象楽器を  $\Omega = \{\omega_1, \dots, \omega_m\}$  とし、楽器  $\omega_i$  の楽器存在確率を  $p(\omega_i; t, f)$  とすると、Instrogram は  $\{p(\omega_i; t, f) \mid 0 \leq t \leq T, f_L \leq f \leq f_H, \omega_i \in \Omega\}$  で表される（ $T$ ：当該楽曲の長さ、 $f_L, f_H$ ：解析対象周波数の上限と下限）。

次に、各時刻・各周波数において最も優勢な楽器  $\hat{\omega}(t, f)$  を次式で求める：

$$\hat{\omega}(t, f) = \operatorname{argmax}_{\omega_i \in \Omega} p(\omega_i, t, f).$$

その後、各楽器が  $\{\hat{\omega}(t, f)\}$  全体の中で占める割合を求め、これを当該楽器の優勢度とする。

## 2.3 プレイリストの生成方法

プレイリスト生成の中心的な処理は、未配置の項目に楽曲を配置することである。プレイリストの最初と最後には必ず楽曲を配置することとすると、この処理は始点と終点の楽曲が与えられ、その間に決められた曲数  $n$  の楽曲を補間する問題となる。楽曲リスト  $\mathcal{M} = \{M_1, \dots, M_K\}$  中の任意の楽曲間に距離  $d(M_j, M_k)$  が定義されているとすると、必ず  $n$  曲経由するという条件下で始点の楽曲  $M_S$  から終点の楽曲  $M_E$  までの最短経路を探索する問題と解釈できる。こ

の問題は、最短経路問題の解法であるダイクストラ法を利用して、以下の手順で解くことができる。

1.  $d(M_S, M_E) = \infty$  とおく。これは通常の最短経路問題とは異なり任意の楽曲に距離が定義されているため、ダイクストラ法で単純に最短経路を探索すると、必ず  $M_S \rightarrow M_E$  が最短になるからである。
2. ダイクストラ法を用いて最短経路を探索する。
3. 2. で得られた結果  $M_S \rightarrow \dots \rightarrow M_k \rightarrow M_E$  において、 $M_S$  と  $M_E$  の間に  $n$  曲未満しかなかったら、 $d(M_k, M_E) = \infty$  とおく。
4.  $M_S$  と  $M_E$  の間に  $n$  曲挿入されるまで、2.~3. を繰り返す。

距離尺度には、前節で求めた各楽器の優勢度からなる  $m$  次元空間内でのユークリッド距離を用いる。

## 3. 実装・試用実験

上記で述べたプレイヤーとサーバプログラムを Java を用いて実装し、試用実験を行った。試用実験には、RWC 研究用音楽データベース（クラシック、ジャズ）を用いた。試用の結果、指定した楽曲に対して、正しく楽器構成が連続的に変化するプレイリストが生成できていることが確認できた。

## 4. おわりに

本稿では、楽器構成に基づいて楽曲をサムネイル化し、楽器構成が徐々に変化するプレイリストを生成する音楽プレイヤーについて述べた。今後は、より大規模な楽曲を用意し、本プレイヤーを広く配布することで有効性の検証を行うとともに、楽器構成の他にリズムやハーモニーの特徴量も付加することで、より自由度の高いプレイリスト生成機能を実現していきたい。

謝辞 本研究の基となった Instrogram は、奥乃博教授（京都大学）、後藤 真孝博士（産業技術総合研究所）との共同研究から生まれたものです。両人に感謝致します。

## 参考文献

- 1) 後藤他：Musicream：楽曲を流してくっつけて並べて思い出すことのできる音楽再生インターフェイス、インタラクシオン 2005 (2005)。
- 2) R. Gulik *et al.*: Visual Playlist Generation on the Artist Map, *Proc. ISMIR 2005* (2005)。
- 3) R. Neumayer *et al.*: PlaySOM and Pocket-SOMPlayer, Alternative Interfaces to Large Music Collections, *Proc. ISMIR 2005* (2005)。
- 4) T. Kitahara *et al.*: Instrogram: Instrogram: Probabilistic Representation of Instrumental Existence for Polyphonic Music, *IPSJ Journal*, 48, 1, pp.214-226 (2007)。