4R-10

弾き直し･弾き飛ばしを含む音楽演奏への高速な音響入力楽譜追跡

中村 友彦，中村 栄太，嵯峨山 茂樹
東京大学 大学院情報理工学系研究科

1 はじめに

我々は，楽譜を参照しつつ人間の演奏に自動で同期し伴奏を再生する自動伴奏を目的とし研究を進めてきた [1]．この技術により，合奏曲の個人練習支援や一人でのオーケストラを伴った演奏支援が実現できる．

練習時に演奏者は（脱落･挿入･置換）誤りの修正や繰り返し練習を行うため，弾き直し・弾き飛ばし（以下，ジャンプ）を行いうる．そのため，これらを含む演奏音響信号からの楽譜上の演奏箇所の推定（以下，楽譜追跡）が必要である．従来の手法では，起きうるジャンプを事前に把握できる場合を扱っている [2, 3, 4]．一方で，このジャンプは演奏者により異なりうるため，事前に把握できない場合は任意のジャンプを扱う必要がある．任意のジャンプを扱うと，探索空間が膨大となり計算量が増大し，追従精度が低下する危険がある．そのため本論文では，我々の研究 [1] で議論が十分でない誤り・任意のジャンプを含む演奏への音響入力楽譜追跡アルゴリズムを，計算量削減の観点から議論する．

2 誤りを含む演奏モデル

音符を状態とみなせば演奏過程は状態遷移として表現され，次に演奏される音符が直前に演奏された音符にのみ依存するならば，演奏モデルは隠れマルコフモデル（HMM）により表される [1]．楽譜通りの演奏は，隣接する状態への遷移と解釈できる．同一音高の音響信号でも音色が変化しうるため，音色変化に頑健な特徴量として，半音単位の中心周波数をもつ定 Q フィルタバンク出力を用いた．脱落誤りは 2 つ先への状態への遷移，挿入誤りは自己遷移，置換誤りは楽譜と異なる音高の出力により表される（図 1）．そのため，誤りを含む演奏は left-to-right HMM で表現される．

3 弾き直し･弾き飛ばしのモデル化と計算量

ジャンプは遠方への遷移としてモデル化できるため，演奏モデルは ergodic HMM により表され探索空間は膨大となる．このとき，最尤推定による時刻 t での最尤状態 $\hat{s}_t$ は，時刻 t までの観測系列 $y_{1:t}=\{y_\tau\}_{\tau=1}^{t}$ と状態確率変数 s_t を用いて，$\hat{s}_t = \mathrm{argmax}_{s_t} p(s_t|y_{1:t}) = \mathrm{argmax}_{s_t} p(y_{1:t}, s_t)$ と導かれる．これは前向きアルゴリズムにより効率的に解け，時刻 t 状態 i での前向き変数 $\alpha_t(i)(=p(y_{1:t}, s_t=i))$ は，音符数 M，初期分布 π，時刻 t での特徴量 y_t の i 番目の状態での出力確率 $b_i(y_t)$，遷移確率行列 A を用いて $\alpha_1(i)=b_i(y_1)\pi_i \quad (t=1)$,

$$\alpha_t(i) = b_i(y_t)\sum_{j=1}^{M}\alpha_{t-1}(j)A_{j,i} \quad (t \geq 2) \qquad (1)$$

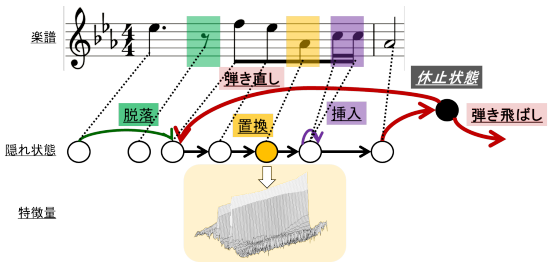

図 1: 休止状態を含む提案法での演奏誤り・弾き直し・弾き飛ばしのモデル化．

と計算でき，(1) の右辺で M 回の計算を M 状態毎に行うため，各時刻での計算量は $O(M^2)$ である．

実際的な楽譜の音符数は数百 ～ 数千程度であるが，(1) による計算では 5 節で示すように数百以上の音符数で実時間処理不可能となるため，一部の実際的な楽譜しか扱えない．したがって，多くの実際的な楽譜を処理するためには，計算量の削減が必要である．

4 計算量削減のためのアルゴリズム

計算量削減のためジャンプを表す遷移確率に制約を設けつつ，多様な演奏を表現可能なモデルが必要である．以下，2 通りの演奏モデルを提案し計算量を削減した推定アルゴリズムを議論する．

1 つ目のモデルでは，ジャンプの確率が直前の音符に関して独立であると仮定する．このとき A は，

$$A_{i,j} = B_{i,j} + C_i D_j \quad (C_i = 1 - \sum_{j=1}^{M} B_{i,j}, \sum_{j=1}^{M} D_j = 1) \quad (2)$$

と表される．ここで，B は当該状態から 2 個先へまでの遷移を許す誤りを含む演奏を表す行列，C_i は状態 i でのジャンプ前の演奏休止の確率，D_j は状態 j でのジャンプ後の演奏再開の確率を表す．(2) を (1) に代入し，

$$\alpha_t(i) = b_i(y_t)\left[\sum_{j=i}^{i+2}\alpha_{t-1}(j)B_{j,i} + \Big(\sum_{j=1}^{M}\alpha_{t-1}(j)C_j\Big)D_i\right] \qquad (3)$$

を得る．右辺第 2 項の括弧内が i によらないため，計算量を $O(M)$ に削減できる．C_i, D_j が i と j に依存して設定できるため，各音符での演奏休止・再開のしやすさを表現できる．また，$C_i D_j$ を i, j によらず一定とすれば以前我々が開発した高速推定法と一致する [1, 5]．

ジャンプ時の演奏再開前の演奏休止区間に着目することにより，同様のモデルが構築できる．この休止を表す状態（以下，休止状態）を演奏モデルに付与すると（図 1），遷移確率行列 $\tilde{A}_{i,j}$ は休止状態を含めた状態数 $N(=M+1)$ を用いて，$i, j \in [1, M]$ に関し $\tilde{A}_{i,j} = B_{i,j}$, $\tilde{A}_{i,N} = C_i$, $\tilde{A}_{N,j} = (1 - \tilde{A}_{N,N})D_j$ と表される．休止状態以外は隣接する状態のみ計算すればよく，計算量を

Fast Score Following for Acoustic Signal of Musical Performance with Repeats and Skips
Tomohiko Nakamura, Eita Nakamura, and Shigeki Sagayama
Graduate School of Information Science and Technology, The University of Tokyo

表 1: 休止状態あり（w/ pause）, 休止状態なしの提案法（w/o pause）, 弾き直し・弾き飛ばしをモデル化しない手法（w/o modeling）での，弾き直し・弾き飛ばしの検出率と，$\Delta = 300, 500, 1000$ ms での平均追従時間（秒単位，音符単位）と標準誤差．

評価尺度	Δ [ms]	w/ pause	w/o pause	w/o modeling
検出率	300	32/43	29/43	8/43
	500	32/43	29/43	8/43
	1000	32/43	29/43	8/43
平均追従時間 [s]	300	3.9 ± 0.8	4.9 ± 1.0	11 ± 3
	500	3.8 ± 0.8	4.8 ± 1.0	11 ± 3
	1000	3.8 ± 0.8	4.8 ± 1.0	11 ± 3
平均追従時間 [音符]	300	8 ± 1	10 ± 1	17 ± 8
	500	8 ± 1	10 ± 1	17 ± 8
	1000	8 ± 1	10 ± 1	17 ± 8

$O(N) \simeq O(M)$ $(M \gg 1)$ に削減できる．

実演奏ではジャンプ中に休止区間が期待されるため，2 つのモデルのうち休止状態のある提案法の追従性能がより高いと考えられる．上記の議論は Viterbi アルゴリズム，Mealy 型の出力分布を持つ HMM についても同様の議論が成り立つ．

5 計算量と楽譜追跡性能の実験による評価

5.1 実験条件

計算量と楽譜追跡性能の評価のため 3 つの実験を行った．16 kHz にダウンサンプリングした音響信号を窓長 128 ms・シフト長 20 ms で特徴量に変換し，事前に RWC 楽器音データベースのクラリネット演奏で出力分布を学習した [6]．パラメータは，$\boldsymbol{\pi}=[1, 0, 0, \cdots, 0]^\top$, $C_i=e^{-1000}$, $D_i=1/M$ $(i \in [1, M])$, $\tilde{A}_{N,N}=0.98$ とした．

最初の実験では，Intel Core 2 Duo P9400 2.40 GHz の CPU，2 GB の RAM の計算機により，シフト長に対する処理時間の割合を表すリアルタイムファクタ（Real Time Factor, RTF）を用いて，様々な音符数で計算量を評価した．第 2 の実験では，クラリネットの実演奏（計 1404 s）に対し，ジャンプから追従するまでの時間（追従時間，秒・音符単位）を用いて追従性能を評価した．ジャンプ後，演奏のオンセット時刻から Δ ms 以内の伴奏が再生された場合を追従したとみなした．最後の実験では，ジャンプを含まない RWC ポピュラー・著作権切れデータベース中 112 曲の旋律パートの MIDI を音響信号に変換し [6]，各曲毎の音符検出率の平均（Piecewise Precision Rate, PPR）と全曲中の音符検出率（Overall Precision Rate, OPR）によりジャンプのモデル化による追従精度低下を評価した [7]．

5.2 結果と考察

計算量について，図 2 に示す RTF の結果を得た．2 つの提案法では同様の結果が得られたため休止状態のある場合を示した．従来法では約数百個，提案法では約 10000 個の音符数まで実時間処理できた．すなわち，提案法では数百から数千音符の実際的な楽譜を実時間処理可能である．

第 2 の実験では，表 1 の結果を得た．ジャンプをモデル化しないアルゴリズムよりも，モデル化する提案法が約 5 割検出率が高く，平均追従時間も 7 s（8 音符）以上速かった．そのため，モデル化によりジャンプに対する追従性能が優位に向上すると言える．また，休止状態のある提案法が休止状態のない提案法より検出率で 0.06 程度高く，平均追従時間も 1 s（2 音符）程度速かった．これは，ジャンプ時に休止区間が存在しやすいためと考えられる．また，提案法で検出出来なかった 11 個のジャンプはポピュラー音楽でのサビの繰り返しや，ジャンプ後に数音符しか演奏されなかったことが

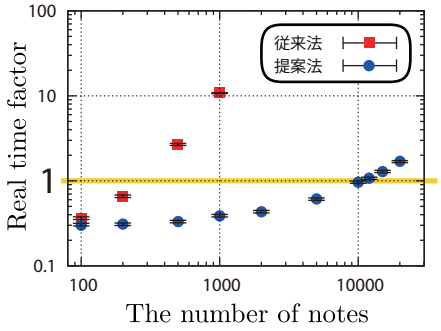

図 2: 様々な音符数での平均リアルタイムファクタ．

原因と考えられる．これらは，人間の伴奏者でも追従困難であるように提案法でも困難である．人間もジャンプへの追従に数秒・数音符はかかると考えられるため，休止状態のある提案法の平均追従時間 3.8 ± 0.8 s (8 ± 1 音符) は実用に耐えうる程度であると考えられ，提案法は有効であると言える．

最後の実験で，PPR は全てのアルゴリズムで 0.839 ± 0.009 であり，OPR は休止状態のない提案法で 30070/36051，その他のアルゴリズムは 30073/36051 であった．よって，ジャンプのモデル化により追従性能は優位に低下しなかった．

6 結論

本論文では，誤り・任意のジャンプを含む演奏に対する楽譜追跡アルゴリズムを 2 つ提案し比較した．ジャンプの確率が直前の演奏した音符に関して独立とするモデルにより，計算量を $O(M^2)$ から $O(M)$ へと削減できた．また，ジャンプ中の休止区間の存在に着目し，この休止をモデル化することにより同様の計算量削減を達成した．実験により，ジャンプのモデル化による追従性能の優位な低下は見られず，実際的な楽譜を実時間処理できることを示した．また，追従結果より提案法がジャンプへの追従に有効であると確認した．ジャンプおよびその際の休止区間のモデル化によりジャンプに対する追従性能が向上することも示した．

今後は，ピアノやギターのような多重音楽器による演奏への拡張が過大である．

謝辞 クラリネット演奏により評価に協力して下さった伊東直哉氏に感謝する．本研究の一部は，文部科学省/学術振興会科学研究補助費 課題番号（23240021）から補助を受けて行われた．

参考文献

[1] 中村他, “音楽演奏の誤りや反復に頑健な音響入力自動伴奏,” **日本音響学会秋季研究発表講演集**, pp. 931–934, Sep 2012.

[2] M. Tekin *et al.*, “Towards an intelligent score following system: Handling of mistakes and jumps encountered during piano practicing,” in *Proc. CMMR*, pp. 211–219, 2004.

[3] B. Pardo *et al.*, “Modeling form for on-line following of musical performances,” in *Proc. AAAI*, vol. 20, p. 1018, 2005.

[4] N. Montecchio *et al.*, “A unified approach to real time audio-to-score and audio-to-audio alignment using sequential montecarlo inference techniques,” in *Proc. ICASSP*, pp. 193–196, 2011.

[5] 中村他, “任意箇所への弾き直し・弾き飛ばしを含む演奏に追従可能な楽譜追跡と自動伴奏,” **情報処理学会論文誌**, 2013. to apper.

[6] M. Goto, “Development of the RWC music database,” in *Proc. ICA*, pp. l–553–556, 2004.

[7] A. Cont *et al.*, “Evaluation of real-time audio-to-score alignment,” in *Proc. ISMIR*, 2007.