HMMによるMIDI演奏の楽譜追跡と自動伴奏

武田　晴登†　　西本　卓也†　　嵯峨山茂樹†

† 東京大学大学院情報理工学系研究科
〒 113–8656 東京都文京区本郷 7–3–1
E-mail: †{takeda,nishi,sagayama}@hil.t.u-tokyo.ac.jp

あらまし　本研究は，楽譜をもとに電子楽器を演奏する演奏者にに合わせて伴奏を再生させる自動伴奏の実現を目的としている．本報告では，自動伴奏の重要な構成要素である，演奏者の演奏位置を実時間で推定する楽譜追跡，及び，楽譜追跡の結果に基づいた適切なテンポで伴奏を再生させて実現される自動伴奏について議論する．人間の実際の演奏では，演奏誤りや弾き直し，和音構成音の発音時刻のずれ等が含まれるので，時間順序通りに楽譜の音と演奏された MIDI 情報を対応させるだけでは楽譜追跡は実現できない．本稿では，楽譜追跡を演奏に対して最も確からしい拍位置を推定する確率的逆問題として扱い, 演奏者の演奏の振舞をモデル化した HMM(Hidden Markov Model, 隠れマルコフモデル) を用いた楽譜追跡を議論する．更に，推定した演奏者の演奏位置の情報を用いて演奏者のテンポ曲線を推定し，演奏者に追従しながら音楽的に自然な伴奏の再生方法についても議論する．楽譜追跡手法の有効性を評価実験で確認し，また，自動伴奏システムを実装し動作を確認した．
キーワード　自動伴奏，楽譜追跡，HMM（隠れマルコフモデル），Viterbi 探索，テンポ推定

Automatic accompaniment system of MIDI performance using HMM-based score followng

Haruto TAKEDA†, Takuya NISHIMOTO†, and Shigeki SAGAYAMA†

† Graduate School of Information Science and Technology, University of Tokyo
7–3–1 Hongo, Bunkyo-ku, Tokyo, 113–8656 Japan
E-mail: †{takeda,nishi,sagayama}@hil.t.u-tokyo.ac.jp

Abstract　This research aims at automatic accompaniment that synchronizes the accompanying parts with the music being performed by human. This paper discusses both a method of score following that estimates performers beat position in music score, and automatic accompaniment system which plays accompaniment parts in the tempo determined by the results of the score following. Since real human performance may include performance error or repetition of the same phrases, score cannot be followed by simple matching performed notes with note in score in time order. To estimate the most probable score position for a given MIDI performance, we formulate the score following as a probabilistic inverse problem using HMM (Hidden Markov Models). This paper also discusses estimation of tempo curve from results of score following and accompaniment system that plays accompaniment parts with the tempo which is musically natural and follows the human performance. Experimental evaluation on score following and implementation of automatic accompaniment system are also reported.
Key words　score following of MIDI performance, automatic accompaniment, beat onset prediction, HMMs (hidden Markov models), Viterbi search algorithm

1. はじめに

本研究は，人間がある楽曲を楽譜に基づいて演奏するとき，計算機に演奏者の演奏に合わせて伴奏者や伴奏オーケストラの演奏をさせ，仮想的にアンサンブル演奏を楽しめるシステムの実現を目指している．本稿では，自動伴奏の構成要素として重要な，演奏者の演奏の楽譜追跡と，楽譜追跡結果に基づく伴奏を再生方法について議論する．

人間の電子楽器による音楽演奏に合わせて伴奏を再生する高性能の自動伴奏は，楽器の練習や演奏会での使用を含む多くの目的に使用できる．特に，今日では MIDI(musical instruments digital interface) 規格の電子楽器が普及し，計算機の性能が向上しているので，このような自動伴奏システムの需要は大きい．

さて，演奏者の演奏に合わせて伴奏を再生するためには，演奏者の演奏と楽譜との対応から演奏位置を求める処理，即ち，「楽譜追跡 (score following)」が必要である．実演奏では，演奏

をされた音を時間順序で楽譜と対応るだけでは正しい演奏と楽譜との正しい対応を得られない．何故ならば，実際の演奏と楽譜との間では，演奏誤りによる音高の不一致，テンポやリズムや和音の打鍵時刻のずれを含む時間情報に関する不一致が存在するためである．更に，伴奏システムを練習に使用する場合は弾き直しや演奏箇所をスキップして可能性も想定した楽譜追跡手法が必要である．

楽譜追跡に関係する従来研究には，人間が音楽演奏の拍を推定する Beat tracking, Tempo tracking に関する研究がある [1]〜[4]．これらの手法を用いれば，演奏曲の楽譜の情報を与えなくても拍の位置を求めることができ，演奏者が弾き直しや演奏を途中で飛ばすこと無く演奏する場合，原理的には拍を数えることで楽譜上の位置を求められる．しかし，推定誤りにより楽譜との対応がずれた場合，そのずれを回復することはできないので，本研究の目的とする自動伴奏には適さない．

一方で，楽譜追跡の従来研究には，音高の情報を用いて楽譜と演奏の音の対応を求めることで拍単位より精密にするアプローチの研究もある．ここでは，主に楽譜と演奏を DP(Dynamic Programming, 動的計画法) マッチングを行なうことで，局所的な誤りを含む可能性のある演奏に対して楽譜追跡を行うことができる [5]．DP マッチングは，一音単位の局所的な誤りを含む演奏に対しても楽譜追跡を行えるが，弾き直しや弾き飛ばしなどの大域的な演奏位置の跳躍を含んだ演奏を扱うことができない．この為，DP と弾き直しの検出を組み合わせた楽譜追跡手法が提案されている [6], [7].

本稿では，このような背景から，弾き直しも含んだ実時間処理が可能な多旋律の楽譜追跡手法を議論する．

また，自動伴奏は，演奏者の楽譜追跡の結果に基づいて伴奏を再生させることで実現できる．MIDI 演奏の楽譜追跡の結果によって与えられるのは，演奏の入力イベントがあった時刻における拍位置であるので，伴奏を再生させるには，演奏者のイベントが無い時刻における拍位置の情報が必要である．本稿では，楽譜追跡結果から，演奏イベントがない間の演奏位置を推定するアルゴリズムについて議論する．

以下，第 2. 章で楽譜追跡を，第 3. 章で自動伴奏について議論する．

2. 演奏者の振舞をモデル化した HMM による楽譜追跡

2.1 MIDI 演奏の楽譜追跡

2.1.1 多声音楽の楽譜と MIDI 演奏

電子ピアノのように MIDI 規格の電子楽器を用いて演奏を行う場合，演奏者の演奏した音に関する情報は MIDI 信号として取得できる．MIDI 信号には，音高情報 (note number) と音強情報（velocity）が含まれており，この信号を受信した時に計算機側で時刻を取得することで，この音の発音時刻が得られる．本稿では，i 番目の発音の MIDI 信号を $x_i = (t_i, n_i)$ と表し，発音時刻 t_i と音高情報 n_i のみを用いて楽譜追跡を議論することにする．x_i は，図 1 に示すようにピアノロールで表された 2 次元平面上で一音を表す線分の右端の位置に対応する．

一方，演奏される曲の楽譜では，各音は冒頭から数えた拍数 (以下,「拍位置」と呼ぶ) と音階名によって指定される音高によって指定される．和音や多旋律の演奏曲では同じ拍位置に複数の音が演奏されることがある．ここでは，同時に演奏されるべき者は一塊として冒頭から数えて j 番目の拍位置を s_j と表すことにする．

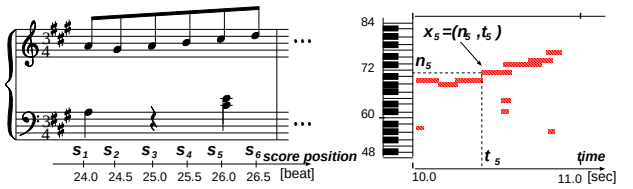

図 1　演奏の MIDI 信号 (右) と演奏曲の楽譜 (左)

本章では，人間の楽器演奏の MIDI 信号を受信した時刻を記録した x_i に対して，この演奏された音に対応する楽譜上の音を求め演奏している拍位置 s_j を求める問題を扱い，本稿ではこの問題を楽譜追跡と呼ぶことにする．

2.1.2 実演奏と楽譜との不一致

楽譜追跡を行なうためには，楽譜をもとに人間が演奏する場合に演奏と楽譜との間にはある以下の不一致となる要素を考慮しなくてはならない．

a） 和音の発音時刻と発音順序

実演奏では，和音の演奏は厳密に同時に演奏されるとは限らず，また，発音順序は一定しない．このため，演奏の MIDI 信号からある拍位置に対応させて同時に発音することを意図された音を群化させることは自明な問題ではない．

b） 演 奏 誤 り

演奏には，楽譜の読み間違い，不注意や技術不足などに起因する演奏誤りが含まれ得る．これらは，楽譜にない音の挿入 (insertion)，楽譜にある音の脱落 (deletion)，楽譜中の音を他の音で演奏してしまう置換 (substitution) に分類できる．

c） テンポと音長の変動

演奏者は演奏に表情を付けるために、通常は音長やテンポに変化を付けて演奏するの．このように時間の変動を考慮すると，演奏すべき音が脱落した演奏は，その箇所でテンポが遅くなった演奏とも解釈できるので，演奏誤りか音長の変動かで演奏と楽譜とのマッチングには曖昧性が存在する．

d） 装飾音の演奏

また，前打音，トリルなどの装飾音は，演奏者により演奏のタイミングが異なり得る．特にトリルは演奏される音の個数も演奏者によって異なることがある．

e） 演奏の弾き直し・スキップ

楽器の学習者が実際の練習で用いる場合などには，特定の部分をくり返して演奏する弾き直しや，ある部分を省略して演奏するスキップなどを含む演奏があり得る．また，楽譜に指定されている繰り返し (リピート記号) の実行の有無による違いもあり得る．

2.1.3 演奏者のモデルと MIDI 演奏の楽譜追跡

このように楽譜と演奏との対応は曖昧ではあるが，しかし，音楽経験者は演奏を聴きながら楽譜を追うことができる．それは，その音楽経験者は演奏している楽曲の構造や演奏者がどのように演奏をするかを知っていて，その知識に照らし合わせて最も確からしいと思う拍位置を瞬時に思い浮かべることができ

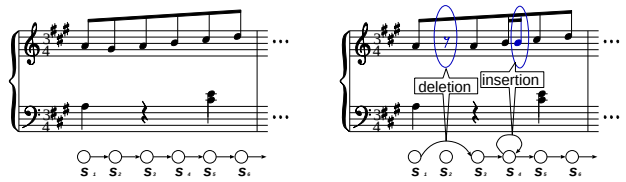

図 2　和音内の状態遷移による和音の演奏のモデル化 (左) と誤りのある演奏の拍位置の遷移 (右)

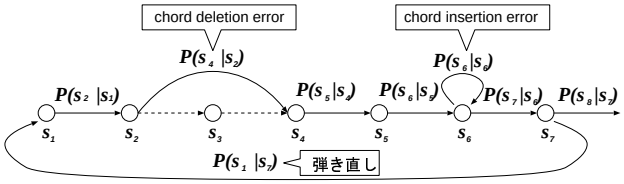

図 3　拍位置のネットワークモデルのマルコフ遷移確率

るからと考えられる．もし，このように経験者にとってある程度予測できる演奏者の振舞を確率的に扱うことができるならば，確率的に最も確からしい拍位置を求めることができる．

2.2　演奏者の振舞を表す確率モデルによる演奏生成モデル

以下，演奏者の演奏のモデル化について議論する．

2.2.1　状態遷移モデルによる和音単位での演奏順序についての確率

図 2 右に示すように，演奏している楽譜上に音が存在する拍位置をひとつの状態としてモデル化すると，楽譜を読み進めていく過程は拍位置の状態遷移と捉えられる．誤りのない演奏は，この拍位置を順に $s_1 \rightarrow s_2 \rightarrow s_3 \cdots$ と遷移する過程に対応する．

しかし，実際の演奏では誤りを含み得るので，これ以外の拍位置の状態遷移系列で演奏される可能性もある．ここでは，ある位置で演奏を誤る確率はそれ以前の演奏には影響を受けないと考え，マルコフモデルでモデル化する．全く誤りがない場合は，$P(s_{j+1}|s_j) = 1$ に対応するが，誤りを含む演奏の可能性を考える為，$P(s_{j+1}|s_j) < 1$ とし，他の状態への遷移の可能性も確率的に扱う．以下の確率は，マルコフモデルの枠組の中で統一的に扱える．

a）　和音の脱落・挿入誤りの確率

和音の挿入誤り，即ち，楽譜のある和音を演奏するために誤って複数の和音を演奏する場合は，同じ拍位置に留まる過程に対応する．この確率は，$p(s_j|s_j)$ で与えられる．また，和音の脱落誤り，即ち，拍位置 s_{j+1} の和音を演奏しなかった場合は，s_j から s_{j+2} への状態遷移に対応する．この場合の確率は $p(s_{j+2}|s_j)$ で与えられる．

b）　演奏の弾き直し・スキップの確率

演奏を拍位置 s_j まで進め，その後にそれより前の拍位置 $s_{j'}$ から弾き直すという場合は，拍位置が $s_{j'}$ から s_j へ遷移した場合と捉えられる．このような弾き直しが起きる確率は，$p(s_{j'}|s_j)$ で与えられる．また，拍位置 s_{j+1} から $s_{j'-1}$(ただし $j' > j+1$) の音をスキップして演奏する場合は，演奏が拍位置が s_j からなる拍位置 $s_{j'}$ に飛ぶ場合に相当し，その確率は確率は $p(s_{j'}|s_j)$ で与えられる．

2.2.2　状態遷移モデルによる拍位置におけるひとつの和音の演奏確率

a）　音高による和音の演奏確率

次に，ある拍位置 s で演奏されるべき和音を，演奏者が K 個の音で $n_1, \cdots, n_K$ で演奏する確率 $P(n_1, \cdots, n_K|s)$ を考える．

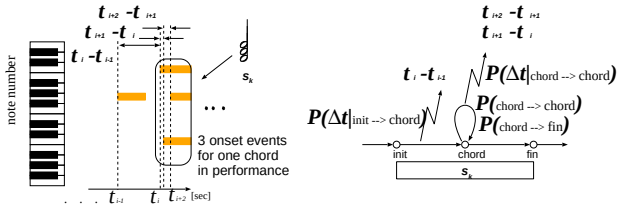

図 4　一つの和音の演奏 (左) の IOI を出力する状態遷移とするモデル化 (右)

一つの和音を構成する全ての音に同時確率を付与すれば見通しの良い確率モデルでモデル化することもできるが [8]，和音を構成する音の組合せを求める為に 2 段 DP 等のマッチングアルゴリズムが必要となり実時間処理には適さない．そこで，ここでは K 個の音を演奏する確率を独立であると仮定して扱う．

$$P(n_1, \cdots, n_K|s) = \prod_{i=1}^{K} p(n_i|s)$$

このように独立に考えることにより，同一の音高の連続に対してもこれをひとつの和音と見なす等，不自然な和音構成にも確率を与えてしまう可能性があるが，これは次の 2.2.2 節で述べる確率と組み合わせることで不自然な和音構成の解釈が選択されることは避けられる．

$p(n_i|s)$ は，拍位置 s で演奏される音がその拍位置で演奏される和音に対して音高が n_i である音が演奏される確率を表す．もし入力される全ての演奏が全く誤りのない演奏を想定できるならば，s で演奏されるべき音が以外の入力に対する確率は 0 となるが，実際にはミスタッチにより楽譜とは異なる音高で演奏される可能性を考慮しなくてはならない．この確率により，例えば初心者は臨時記号等を読み間違え易いなどのミスタッチの傾向を表現できる可能性がある．その確率は，誤りを含む大量の演奏データがあれば統計的に求められるが，$\sum_n p(n|s) = 1$ として正規化されるという確率の定義に注意して，誤り傾向を考慮して経験的に確率を与えることもできる．

同時発音の IOI

和音を構成する音の発音時刻は，実演奏では厳密には一定しない．即ち，和音構成音として入力された連続する 2 つの音の発音時刻 t_{i-1},t_i は，理想的には一致し $t_{i-1} = t_i$ となるが，現実には一致するとは限らない．IOI （inter-onset interval, 発音時刻間隔）$t_i - t_{i-1}$ は理想的には 0 であるが，実際には小さい正値をとる．この特徴を確率的にモデル化するために，IOI $t_i - t_{i-1}$ が前の音と同時に発音された和音である確率密度関数は，正値で積分が 1 に規格化される連続関数 $g(\cdot)$ を用いて

$$p(t_{i-1}, t_i|\text{chord} \rightarrow \text{chord}) = g(t_i - t_{i-1}) \qquad (2)$$

により与えられる．この確率は，十分な量の演奏データとその対応する楽譜があれば統計的な学習で確率値を定めることができる．

和音の挿入誤りと和音構成音の置換誤り

和音の挿入誤りと和音構成音の置換誤りをそれぞれ別に確率的に扱うためには，MIDI 信号の時系列において和音としての区切りを与えるモデル化が必要である．ここでは，図 4 に示す

ように和音内に状態 $u = \{\text{init}, \text{chord}, \text{fin}\}$ を設け，2 つの和音をある拍位置に対応させる和音の挿入誤り (fin から init への遷移) と，正しい音と同時に誤った音を演奏する和音構成音の置換誤り (chord から chord への遷移) とを区別し，それぞれに適切な確率を与えられる．

和音として同時に発音される音の個数の確率

また，ある拍位置で演奏するべき音の個数は楽譜によって指定されているが，実際には演奏誤りにより，楽譜とは異なる個数で演奏される場合もある．その拍位置で演奏されるべき音の個数に対する確率は状態の自己遷移確率 $p(\text{chord} \to \text{chord})$ で与えることができる．ネットワークのトポロジーから確率として $p(\text{chord} \to \text{chord}) + p(\text{chord} \to \text{fin}) = 1$ として正規化されてある必要がある．また，init は chord のみに接続するので，これに対して確率を考える必要はない．また，fin から次の和音の拍位置の init への経路は，2.2.1 節で議論した拍位置のネットワークの経路と同一のものである．

IOI の変動確率

楽譜で演奏すべき拍の長さが $s_j - s_{j'}$(拍) である部分を，演奏者がその音長 (IOI) を $t_i - t_{i-1}$(秒) と演奏される確率は，$p(t_i - t_{i-1}|\text{init} \to \text{chord})$ でモデル化される．IOI は，テンポと音価に依存した音長に対して変動すると考えられるが，テンポは演奏中に変動するものであるので信頼性のある値を求めるのには計算量を要するので，ここでは予測テンポに基づいて厳密な IOI の確率変動を計算はしないことにする．確率としては，演奏を弾き直したりスキップする場合は IOI は演奏時に比べて長い休止がある傾向や，和音の挿入誤りの場合は短い IOI であると傾向を反映させた確率を与えることができる．

IOI と音高により与えられる和音の演奏確率

以上から，ある拍位置 s でひとつの和音が演奏されるときの IOI に基づいて与えられる確率は，以下で与えられる．

$$\begin{aligned}
&p(n_{i+1}, \cdots, n_{i+K}, t_{i+1}, \cdots, t_{i+K}|u_1, \cdots, u_K, s) \\
&= \prod_{k=1}^{K} p(n_{i+k}|s) \\
&\cdot\ p(t_{i+1} - t_i|\text{init} \to \text{chord}) \\
&\cdot \prod_{k=2}^{K} \{p(t_{i+k} - t_{i+k-1}|\text{chord} \to \text{chord}) p(\text{chord} \to \text{chord})\} \\
&\cdot\ p(\text{chord} \to \text{fin}) \qquad (3)
\end{aligned}$$

2.2.3 統合された確率モデルにおける演奏生成確率

以上の確率モデルを統合して，ある楽曲を拍位置を $s_1, \cdots, s_I$ と辿るように MIDI 信号 $x_1, \cdots, x_I$ で演奏する確率は，以下のように書き下すことができる．

$$\begin{aligned}
&P(X|S) \cdot P(S) \\
&= p(n_1|s_1) \cdot p(s_1) \\
&\quad \cdot \prod_{i=2}^{I} p(n_i|s_i) \\
&\qquad \cdot \left\{ \begin{array}{l} p(t_i - t_{i-1}|\text{self}) \cdot p(\text{self}) \\ p(t_i - t_{i-1}|\text{tran}_1) \cdot p(\text{tran}_2) \cdot p(s_i|s_{i-1}) \end{array} \right\} \qquad (4)
\end{aligned}$$

ここで，self は chord → chord，tran_1 は init → chord，tran_2 は chord → find の和音内での各遷移を表すものとする．

表 1　HMM と演奏生成の確率モデルの対応

HMM による表現	演奏生成モデルによる表現
隠れ状態	和音の開始，和音
状態遷移出力	音高，IOI
状態遷移出力確率	和音の一致
HMM	一和音 (拍位置)
HMM 遷移	拍位置の演奏順序

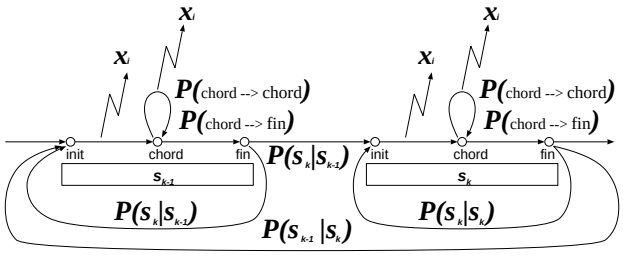

図 5　多重音を含む楽曲の演奏生成をモデル化した HMM

2.3　HMM による演奏過程のモデル

これらの各確率モデルは，表 1 に示すような対応を考えると図 5 に示すように HMM として理解できる．この HMM では，状態と状態遷移の両方からの出力がある．演奏過程は，HMM のネットワーク内を遷移しながら演奏音の音高及び IOI を出力する過程であると見なせるので，演奏と楽譜との対応を求める問題はこの HMM における状態遷移系列を求める問題と等価になる．

2.4　事後確率最大化推定による楽譜追跡

2.4.1　事後確率最大化推定による楽譜追跡の定式化

ここで，事後確率最大化推定として楽譜追跡の定式化を行なう．演奏者の演奏の入力 $X = \{x_i\}_{i=1}^{I}$ に対して，HMM の中で最も尤もらしい拍位置の系列 $S = \{s_i\}_{i=1}^{I}$ を求めたい．Bayes の定理を用いて以下のように定式化される．

$$\hat{S} = \underset{S}{\mathbf{argmax}}\, P(S|X) = \underset{S}{\mathbf{argmax}}\, P(X|S)P(S)$$

全ての可能な S について最大のものを求める．ただし，自動伴奏において楽譜追跡の結果として要求されるのは，直前に入力された x_I に対応する拍位置 $\hat{s}_I$ である．

ここで演奏の MIDI 信号の観測と拍位置の遷移についてマルコフ性を仮定すると，事後確率は

$$\begin{aligned}
&p(s_1, \cdots, s_I|n_1, \cdots, n_I, t_1, \cdots, t_I) \\
&\propto p(n_1, \cdots, n_I, t_1, \cdots, t_I|s_1, \cdots, s_I) \cdot p(s_1, \cdots, s_I) \\
&= p(n_1|s_1) \cdot p(s_1) \\
&\quad \cdot \prod_{i=2}^{I} p(n_i|s_i) p(t_i - t_{t-1}|s_{i-1}, s_i) p(s_i|s_{i-1}) \qquad (6)
\end{aligned}$$

となる．ここで，状態はある拍位置の音を最初に弾いたか否かを区別する為に，ある拍位置 $s_i = s_{i-1}$ で同時発音として連続する 2 音を演奏した場合は，

$$p(t_i - t_{i-1}|s_i, s_i) \cdot p(s_i|s_i) = p(t_i - t_{i-1}|\text{self}) \cdot p(\text{self})$$

であり，それ以外の場合は，

$$p(t_i - t_{i-1}|s_{i-1}, s_i) \cdot p(s_i|s_{i-1}) = p(t_i - t_{i-1}|\text{tran}) \cdot p(\text{tran}) \cdot p(s_i|s_{i-1})$$

に対応させる．このとき式 (6) は式 (4) と同一になり，2.2 節で議論した HMM により事後確率が与えられることが分かる．

HMM では与えられた時系列を出力信号とする最尤状態系列を時間同期 Viterbi 探索によって求められるので，事後確率を最大化する拍位置の系列を求められる．

2.4.2 実時間処理へ向けての計算の簡略化

通常の Viterbi 探索では全ての遷移の可能性を考慮して最適な拍位置を求めることができるが，演奏曲が長くなり拍位置の数が大きくなると計算量も増加する．ここでは，実時間動作を行なうための計算の効率化を目的にした計算の省略方法について議論する．

a） IOI の閾値判定による和音の決定

IOI $\Delta t_i = t_i - t_{i-1}$ が閾値 T 以内であるものを和音と判定することで，状態の同一和音としての自己遷移か異なる和音かを一意に決め，入力された IOI により異なる自己遷移か連接する次の状態への遷移の比較を省略できる．

$$g(\Delta t) = \begin{cases} \text{const.} & (\Delta t \leqq T) \\ 0 & (\Delta t > T) \end{cases} \qquad (9)$$

ただし，この閾値処理が不適切である演奏に対しては，その箇所で誤推定が起きる可能性がある．

b） trellis におけるビーム探索

次に，探索範囲を限定することで計算回数の削減を考える．例えば，大きな尤度差がある仮説はその後最尤拍位置にはならないと仮定して，更新時に計算を行なう遷移の出発点となる trellis のノードを尤度を基準に限定することが考えられる．また，遷移の到達するノードは直前の時刻での最尤拍位置か，事前知識として与えられるフレーズの弾き直しやスキップの対象となりやすい箇所に限定されると仮定して，絞ることができる．ただしこのような仮定が成り立たない演奏に対しては，一時的な誤推定を起こす可能性がある．

2.4.3 Viterbi アルゴリズム

以上の計算省略を行う場合，IOI の閾値判定で和音のまとまりが一意に決定されるので，図 6 に示す 2 種類の更新を IOI に応じて使い分けることでより効率的に Viterbi アルゴリズムを実行できる．即ち，trellis の時間のインデックスを和音単位とし，演奏の k 番目の和音をに対応する累積尤度 $\delta(k, j)$ を次のように更新すれば良い．

（1） $t_i - t_{i-1} \leqq T$ のとき

同一和音内なので trellis の探索は時間方向には進まない．保持しているノードのみ累積尤度を更新する．また，最尤拍位置の更新は行なわない．

$$\delta(k, j) = \delta(k, j) \cdot P(n_i|s_j) \cdot P(\text{self}, s_j)$$

（2） $t_i - t_{i-1} > T$ のとき

異なる和音の開始の音なので，k 番目の和音から $k+1$ 番目の和音へに対応する遷移を求める．b) 節で議論したようにノードを展開探索のビーム J_D 内にあるものに限定する．保持しているノードの集合を J_S とすると，$j \in J_D$ に対して，最適性の原理を用いた次の更新を行なう．

$$\delta(k+1, j) = \max_{j' \in J_S} \delta(k, j') \cdot P(n_i|s_j)$$
$$\cdot P(s_j|s_{j'}) \cdot p(\text{tran}, s_{j'}, s_j) \cdot p(t_i - t_{i-1}|\text{tran}, s_{j'}, s_j)$$

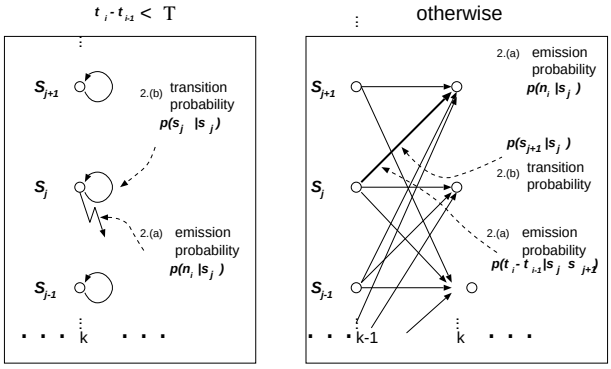

図 6 提案する HMM における Viterbi アルゴリズムの更新方法

表 2 楽譜追跡の性能評価に使用した演奏曲

データ名	誤り	データ数	演奏曲	演奏内容
A	なし	25	BM	各曲をくり返しなしで通し
B	あり	10	DV	冒頭から 57 小節まで
C	あり	5	DV	9〜16, 31〜38,31〜38 小節
D	あり	5	DV	19〜26, 39〜57 小節

BE: Burgmuller: 練習曲 Op.100-25
DV: Dvorak: ピアノ五重奏曲 Op. 81 第 3 楽章のピアノパート

2.5 性能評価実験

a） 実 験 条 件

提案した楽譜追跡手法をを実装し，表 2 に示す多旋律のピアノ曲を MIDI ファイルに録音した実演奏データを用いて評価を行なった．データ A は，ピアノの初心者が学習で演奏する曲である．また，データ B はプロの演奏会でもしばしば取り上げられる典型的な室内楽曲であり，ある程度のピアノの演奏技術が要求される．演奏者はアマチュアのピアノ経験者であり以前にこれらの曲を弾いた経験を持ち，楽譜を見ながらこられの曲を演奏した．演奏するにあたり，演奏者は B〜D は事前に練習を行ない，ほとんど誤りなしで演奏できるレベルに達したところで録音 (MIDI データ収録) を行なった．

IOI の閾値判定は 0.045 秒とした．また，self の自己遷移確率は，和音構成音数が N のときに，$\frac{N-1+\epsilon}{N+\epsilon}$ として単一音の場合に多重音が演奏される確率を与え，今回は $\epsilon = 0.5$ とした．状態遷移出力確率 (以下「IOI の尤度」) は，IOI を自己遷移の可能性が高い 0.08 秒以下，通常の音価に対応する IOI である可能性が高い 0.08 以上 3 秒以下，弾き直しやスキップの可能性が高い 3 秒以上の 3 つの範囲に分けて与えた．評価には IOI の尤度の有無の 2 つの場合で評価を行なった．

b） 評 価 結 果

評価は，演奏に対する正しい楽譜との対応を正解の対応とし，提案手法により逐次入力された演奏に対して演奏位置の推定を行ない演奏の拍位置を記録し，正解の対応に対して DP を用いて削除・挿入・置換誤りを数え，正解率を計算した．この評価法で，表 3 に示す楽譜追跡の正解率を得た．この結果から，提案手法で弾き直しを含む演奏に対して楽譜追跡が行なえることが示された．誤りは，ほとんど同時として発音される装飾音を和音と推定することに起因するものが見られた．

表 3　楽譜追跡の正解率 [%]

条件＼ データ	A	B	C	D
IOI の尤度なし	98.1	96.5	97.0	94.1
IOI の尤度あり	98.3	98.0	97.2	95.2

3. 楽譜追跡の結果に基づく伴奏テンポの設定

3.1 伴奏の再生の定式化

3.1.1 楽譜追跡の結果と伴奏の再生

次に，楽譜追跡の結果に基づいて，演奏者の演奏に合わせて伴奏を再生させる方法について議論する．

楽譜追跡により演奏者の演奏位置が推定されたとき，それに合わせて伴奏を再生させる方法は一意には決まらない．楽譜追跡によって得られる情報は，演奏者の演奏の発音イベントがあった時刻 t_i における演奏位置 $\hat{s}_i$ であり，それ以外の時刻での情報は与えられていない．この為，演奏者の演奏がない場所でも伴奏にイベントがある場合は，そのイベントに対する時刻を決めなければならない．伴奏のイベントとは，発音イベントだけでなく，消音イベントなど MIDI チャネルメッセージに相当するすべてのイベントを含むものとする．本節では，楽譜追跡結果として時刻 t_i に対する演奏者の発音時刻 s_i が与えられた場合に，伴奏の再生位置 s^A を定める問題を扱う．

3.1.2 MIDI ファイルの再生過程の定式化

本稿で最終的に MIDI を用いて伴奏を再生させることを想定している．そのために，まず，伴奏に相当する MIDI の再生する過程を定式化する．

MIDI ファイルには，指定された分解能による整数値で指定された拍位置 s に送信すべき MIDI メッセージが格納されている．MIDI ファイルの再生は，具体的には，MIDI ファイルに格納されているテンポ情報をもとに現実の時刻 t に対応する拍位置 $s(t)$ を求め，その拍位置 $s(t)$ に対応する MIDI イベント送る処理を指す．現実には，計算機の定めた時間分解能によって与えられた離散時刻 $t_1, \cdots, t_n$ に計算処理を実行する．従って，テンポをもとに伴奏を再生するときの拍位置 $r^A(t)$ の変化は，伴奏テンポを $r(t_n)$ (秒/拍) として

$$s^A(t_{n+1}) = s^A(t_n) + r^A(t_n) \cdot (t_{n+1} - t_n) \qquad (11)$$

という更新式で書ける．MIDI ファイルでは，拍位置に対応してテンポが指定されているので，その場合は，$s^A(t_n)$ は正確にはその時刻 t_n の拍位置 $s(t_n)$ に依存したテンポとなる．伴奏の再生は，この時刻に拍位置 $s_n \leqq s < s_{n+1}$ に該当するイベントを送信すればよい．

本稿では，楽譜追跡結果から伴奏を適切に再生するために $r^A(t)$ を求める手法を議論する．

尚，演奏と中で伴奏の再生位置をそれ以前の拍位置に戻す場合は $s_{n+1} < s_n$ であるので，これは式 (11) における $v^A(t_n) < 0$ として扱える．このとき，s_n 以前に発音イベントがあり，s_n より後に消音イベントが存在する音は，発音したままになるので，この音を消音しなくてはならない．また，拍位置 s_n と s_{n+1} が離れている場合は，その間にある多くのイベントを一瞬に全て送ることになり，伴奏として不自然な音を再生する．これは，例えばこの瞬間のみ音は再生しない等で，伴奏再生としての自然さを保てる．

3.1.3 演奏者に追従する伴奏再生

演奏者に合わせた伴奏とは，伴奏が演奏者の演奏した音と同時に演奏すべき音を演奏するだけでは音楽的に不十分である．音楽的な自然さを考慮して伴奏を演奏者のテンポ感などを推定して演奏する．ここでは，様々な伴奏の仕方が可能であり，演奏者によって演奏方法が異なるように，自然な演奏も一通りではない．特に，伴奏が演奏者の演奏をリードする場合は，伴奏者の個性に応じた演奏の表情付けとして様々な演奏が考えられる．また，教師と生徒が合奏する場合は，教師が生徒にテンポ感を生徒に教えるために敢えて生徒のテンポとは異なるテンポで伴奏する場合もあり得る一方，発表会等では生徒のテンポが不自然に大きく揺れる場合もその生徒に一致させて演奏させる場合もあり得る．

我々は，これらの様々な伴奏の仕方が考えられるのに対応し，最終的には伴奏のモデルをパラメトリックに扱って，演奏者の目的に応じた伴奏を提示するシステムの実現を目指している．本稿では，その最初のステップとして，演奏者の演奏にできるだけ一致し，テンポが自然な伴奏の再生方法について議論する．

ただし，演奏者の演奏にぴったり合わせる場合，伴奏が不自然になる可能性は存在する．例えば演奏者が短い音符を局所的に「転んで」演奏した場合のように，演奏者のテンポそのものが不自然である場合，伴奏のテンポも不自然なテンポとなる可能性がある．また，実際に演奏によっては，演奏と伴奏が厳密には一致しないことを意図して演奏する場合もある．例えば，演奏者が一定テンポで「かっちりと」演奏している伴奏に対して僅か発音時刻を速めて演奏する「突込み気味」の演奏や，逆に発音時刻を送らせて旋律を「ゆったり気味」で演奏する場合は，演奏と伴奏のぴったりと一致することを意図しているわけではない．この場合は，演奏としては自然なテンポの揺れも，伴奏が同じテンポで演奏した場合は不自然となる可能性がある．

3.1.4 推定した演奏者の拍位置の合わせる伴奏テンポの設定

演奏者と演奏箇所を一致させるためには，$e(t) = s^A(t) - s^P(t)$ を 0 とするように，式 (11) において伴奏のテンポを変化させればよい．ただし，演奏者の推定拍位置が局所的に戻った時に，伴奏者はその戻った拍位置の音を演奏し直す不自然さを避けるためには，局所的は伴奏の後戻りをなくせばよい．即ち，伴奏のテンポは，演奏者と伴奏者の拍位置の関係から以下のように定める．

$$r^A(t_{n+1}) = \begin{cases} \frac{s^P(t_n) - s^A(t_n)}{t_{n+1} - t_n} & (s^P(t_n) \geqq s^A(t_n)) \\ 0 & (s^P(t_n) < s^A(t_n)) \end{cases} \qquad (12)$$

以上の流れをまとめると，図 7 に示すように，伴奏の拍位置は伴奏テンポ $r^A(t)$ を積分器に入力した出力結果となる．これは一次遅れ系であり，テンポが再生器の時間分解能より細かい周期で変化する場合に，系は不安定となる．しかし，音楽演奏におけるテンポは時間に対して滑らかに変化するものと考えられるのに対し，式 (11) の更新が数ミリ秒程度であるならば，系として安定である．また，人間の聴覚で連続する 2 つの音を識別できるのは少なくとも数十ミリ秒以上であるので，入力に対する応答時間がそりより短ければ，入力があってから伴奏を再

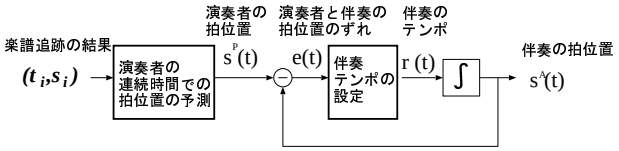

図 7　楽譜追跡の結果から伴奏の演奏拍位置を決定する過程

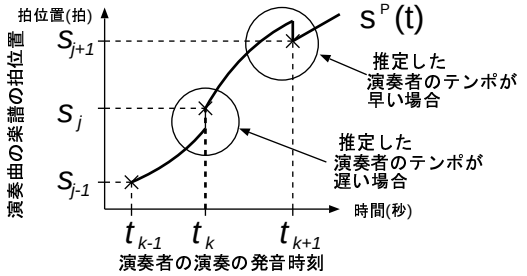

図 8　演奏者した拍位置 $\hat{s}_i$ と過去の演奏者の演奏から推定した $s^P(t)$ との誤差

生したとしても演奏者の演奏した音と伴奏の音とは同時に聴こえる．人間が伴奏するときに演奏者と同時に音を鳴らすには，伴奏側が演奏者の発音タイミングを予測してそれに合わせて演奏を行なうので，この点では，演奏者の演奏の入力があってから伴奏の音を発音させる本システムは伴奏者とは異なる振舞をしている．

3.1.5　演奏者の次拍の発音時刻の予測誤差の最小化

この楽譜追跡の結果である演奏者の拍位置に伴奏の拍位置を一致させられるが，それに加えて，次の拍の時刻を正確に予測し，予測の誤差ができるだけ少なく伴奏することが重要である．何故ならば，予測に誤差が生じる場合は，その誤差に応じた伴奏テンポの伸縮を生じ，本来滑らかであるテンポの特性に反した演奏を行なうことになり，伴奏の演奏として不自然になるからである．

これを具体的に述べると以下のようになる．実際に次の演奏入力のあった時刻 t_{i+1} における拍位置 $s_{i+1}^{\hat{}}$ は，楽譜追跡の結果として求められた時刻 t_i の時点における拍位置 $s^P(t_{i+1}) = \mathrm{argmax}\, P(S|X)$ とは異なる．図 8 に示すように，伴奏が演奏者に対して進んでいるか遅れているかの誤差が存在する．このように演奏者の演奏が演奏の入力のある度に不連続な変化を行う場合，$s^P(t)$ との誤差を最小にするように再生している伴奏も淀みもしくは急激なテンポ変化を含む不自然な演奏に対応している．

そこで，以下，この予測誤差を最小にする伴奏の再生について議論する．

3.2　演奏者の次拍発音時刻の予測誤差の最小化

3.2.1　演奏者の連続時間での拍位置の推定の定式化

伴奏側が認識している演奏者の拍位置を演奏中の任意の時刻 t の連続関数 $s^P(t)$ とする．$s^P(t)$ は，演奏者の演奏の発音イベントが観測された時刻 t_i の楽譜上の拍位置 $s^P(t_i) = s_j$ は正しいとし，これを基点に補間して，t_i 以降の時刻の拍位置を連続曲線を求める方法を考える．時刻 $t > t_i$ では，演奏者の拍位置は時刻 t_i の拍位置 $s(t_i) = s_j$ の付近では

$$s^P(t) = \underbrace{s^P(t_i)}_{=\hat{s}_i=\mathrm{argmax}\, P(S|X)} + \int_{t_i}^{t} \frac{ds^P}{dt}(t')dt'$$

となるので，$s^P(t)$ の推定は，

$$r^P(t) = \frac{ds^P}{dt}(t)$$

を推定する問題となる．$r^P(t)$ は時間に対して連続に変化するテンポを表す量と考えられるので，これを「テンポ曲線」と呼ぶことにする [9]．ここで，演奏の IOI の間ではテンポや演奏位置の極端に大きな変動がないと考えられるので，この区間ではテンポは変動するがテンポのほぼ一定であると仮定し，ここではテンポ曲線を定数 $r^P(t) = a$ とモデル化する．これにより，演奏者の演奏位置も，式 (11) で表される伴奏再生の時間分解能と同期して

$$s^P(t_{n+1}) = s^P(t_n) + a \cdot (t_{n+1} - t_n)$$

と更新できる．

ここでは，演奏者のテンポ $s^P(t_n)$ が正しく推定されていれば予測誤差を最小化できると仮定し，楽譜追跡の結果からテンポ曲線 $r^P(t)$ を求める手法を議論する．

3.2.2　楽譜追跡結果からの演奏者のテンポの推定

演奏者の連続時間の拍位置 $s^P(t)$ は，ある時刻 t での特徴は，その直前の振舞によって特徴づけられと考えられる．例えば，それよりも 1 分間やあるいは冒頭の部分の演奏とは関連は低いと考えられる．ただし，演奏者の大域的な構造に基づくテンポの設定がなされている場合，例えば，冒頭と同じテンポで途中から演奏することを指定される場合は，この仮定が成り立たない．ここでは，このような楽曲構造に基づくテンポのスケジューリングは考慮せずに，局所的な演奏者の演奏情報に基づいて演奏者の拍位置を推定する．

直前の演奏からそれ以前に観測された演奏の発音の時刻 $t_{i-M} \cdots t_{i-1}$ とそれに対応する拍位置 $s^A(t_{i-M}), \cdots, s^A(t_{i-1})$ から最も確からしいテンポ曲線を求める．実演奏の過去のイベントの拍位置 $(t_{i'}, s_{i'})$ $(i' < i)$ はテンポ曲線 $r^P(t)$ を求める．実演奏の過去のイベントの拍位置 $(t_{i'}, s_{i'})$ $(i' < i)$ はテンポ曲線 $r^P(t)$ から変動する．これは，演奏者の発音時刻は，演奏者の無意識なテンポを変動させたり，あるいは，テンポ曲線のモデルの表現力が不十分だからである．ここで，このモデルに対して実際の演奏は確率的に変動すると仮定すると，拍位置を求める問題は，

$$s^P(t) = \underset{s^P(\cdot)}{\mathbf{argmax}}\, P(s(\cdot)|(t_{i-M}, \hat{s}_{i-M}), \cdots, (t_{i-1}, \hat{s}_{i-1}))$$

と定式化できる．

様々な確率モデルが考えられるが，次のモデルでは，実時間処理を行なうために少ない計算で推定を行なえる．即ち，演奏のイベント t_i より以前の演奏について，瞬時テンポ

$$r'_j = \frac{s^P(t_j) - s^P(t_{j-1})}{t_j - t_{j-1}}$$

とモデルのテンポ $r^P(t)$ は正規分布に従うと仮定する．このモデルでは，最尤推定値は $\hat{a} = \frac{1}{M} \sum_{m=1}^{M} r'_{i-m}$ で得られる．

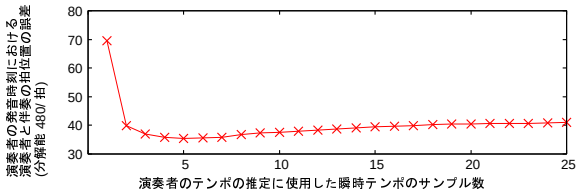

図 9　演奏者の発音時刻における演奏者と伴奏の拍位置の誤差

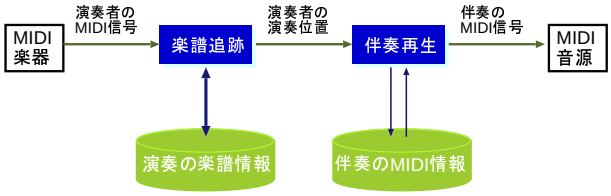

図 10　自動伴奏システムの構成

3.3　演奏者の発音時刻時における伴奏と演奏者の拍位置のずれの評価

実演奏で予測誤差により伴奏に影響を与える拍位置のずれを測定し，テンポ推定に用いるサンプル数との関係を評価した．拍位置の誤差は，楽譜追跡結果に基づいて推定したテンポから $s^P(t)$ を推定し，演奏者の次の入力音があったときの演奏者と伴奏の拍位置との 2 乗誤差 $\sqrt{\sum_i (\hat{s}_i - s^P(t_i))^2}$ を計算した．評価曲は，表の A で楽譜追跡が 100% の精度で行われている 19 曲を用いた．その結果は，図 9 に示すように，拍位置のずれは $M = 4$ で最小値 35.4/480 = 約 0.07(拍) となり，それ以上のサンプル数を用いると逆に予測精度が落ちる傾向が見られた．これから，自動伴奏の伴奏再生のためのテンポ推定には，およそ 4 個程度の過去のサンプル値からテンポを求めればある程度の 2 乗誤差を予測誤差で収まる推定が可能であると考えられる．

3.4　自動伴奏システムの実装

3.4.1　実装したシステム

本稿で議論した楽譜追跡と伴奏テンポの設定手法を直列に接続した自動伴奏システムを実装した．演奏者が伴奏を聴いて伴奏に合わせることの影響は合奏において重要であり，伴奏が演奏者に与える影響についても研究がなされている [10]．本報告では，これらの影響を考慮するのは今後の課題とし，ここでは演奏者が伴奏の影響を受けても受けなくても自由に演奏する場合に，演奏者に合わせて伴奏を再生させることを目指す．その為には，図 10 のように 2 つのモジュールを直列に接続するだけで十分である．

伴奏システムは Machintosh OSX で動作する Cocoa アプリケーションとして実装した．図 10 における「楽譜の演奏情報」と「伴奏の演奏情報」は MIDI ファイルで与えるシステムとした．実行環境として，計算機には Power Book G4 (CPU が Power PC G4 1.25 GHz) を，また，演奏と伴奏再生には MIDI 音源内蔵の電子ピアノ Casio Privia PX-300 を使用した．

3.4.2　動 作 確 認

本システムを表 4 のサンプルを含む複数の楽曲の MIDI ファイルを用意し，人間の演奏に合わせた伴奏を提示することを確認した．本システムは，演奏者が演奏するパートが与えられた楽譜に置いて必ずしもソロパートでない場合以外にも適用でき，演奏を楽しむことができた．

表 4　自動伴奏システムの動作に使用した楽曲のリスト (一部)

曲の種類	演奏曲	演奏	伴奏
ピアノ独奏曲	Chpoin 作曲 幻想即興曲	右手	左手
室内楽曲	Dvorak 作曲 ピアノ 5 重奏曲より第 3 楽章	ピアノ	四弦
管弦楽曲	Bach 作曲 ブランデンブルグ協奏曲 第 1 番より 第 3 楽章	通奏低音	それ以外パート

4.　おわりに

本稿では，HMM を用いた楽譜追跡方法を議論した．更に，この楽譜追跡結果に対して適切な伴奏の再生手法を議論し，これらの提案手法を実装した自動伴奏システムの動作検証について報告した．今後は伴奏システムの性能評価方法を検討し，本システムの評価を行ないたい．

謝　　辞

自動伴奏のソフトウェアは，武田が戦略ソフトウェア創造人材養成 教育コースに受講時に作ったものです．このコースにおいて，平木敬教授，稲葉真里特任教授，鈴木隆文講師には有益な議論をして頂き，ソフトウェア開発に必要な環境を提供して頂きました．また，金子勇元助手，土村展之助手には実装について多くの適切な御指導を頂きました．本ソフトウェア制作を支えて頂いた皆様に深く感謝致します．また，伴奏の再生について東京大学嵯峨山研究室の小野順貴講師と松本恭輔君には制御工学の視点から有益な議論をして頂きました．

文　　献

[1] M. Goto, Y. Muraoka' Real-time Beat Tracking for Drumless AUdio Signals: Chord Change Detection for Musical Decisions, Speech Communication, Vol. 27, No. 3. pp. 311–335, 1999.

[2] S. Dixon: Automatic Extraction of Tempo and Beat from Expressive Performances, Journal of New Music Research, vol. 30, No. 1, pp. 39 – 58, 2001.

[3] C. Raphael: Music Plus One: A System for Flexible and Expressive Musical Accompaniment, Proc. of the ICMC, 2001.

[4] A. Cemgil, B. Kappen, P. Desain, H. Honing, “On tempo tracking: Tempogram Representation and Kalman filtering” Jounal of New Music Research, 2000.

[5] R. B. Dannenberg: An On-line Algorithm for Real-Time Accompaniment, Proc. of ICMC, pp. 193–198, 1984.

[6] 大島，西本，鈴木，“家庭における子どもの練習意欲を高めるピアノ連弾支援システムの提案，” 情報処理学会論文誌, Vol. 46, No. 1, pp. 157-171, 2005.

[7] 尾崎，原尾，平田，“間違いの認識による演奏習得支援システムの構築，” 情報処理学会研究報告, MUS, Vol. 55,No. 1, 2004.

[8] 武田，西本，嵯峨山，“和音の発音順序交替を許容した動的計画法による多声 MIDI 演奏の楽譜追跡，” 日本音響学会 2006 年春季研究発表会 講演論文集, pp. 723-724, 2006.

[9] 武田，西本，嵯峨山，“HMM を用いたリズムとテンポの反復推定による多声 MIDI 演奏のリズム認識，” 日本音響学会 2006 年春季研究発表会 講演論文集, pp. 721-722, 2006.

[10] 堀内，坂本，市川，“合奏における人間の発音時刻制御モデルの推定，” 情報処理学会論文誌, Vol. 43, No. 3, pp. 260-267, 2006.