2N-4

パーティクルフィルタを用いたリズム量子化 *

箕浦 健統† 林 朗‡
広島市立大学大学院 † 広島市立大学大学院 ‡

1. はじめに

楽譜とは，どのように楽曲を演奏するかを記述したもので，作曲者は楽譜を書くことで楽曲の演奏を人に伝えることができる．音楽を楽譜に記すことを採譜という．人間が採譜をするには，多くの時間を費やして曲を理解し，音の高さを聞き取るといった時間や音楽に関する経験が必要となる．自動音楽採譜はコンピュータによって採譜を行うことである．

現在，商用の自動音楽採譜のソフトウェアは存在するが，演奏に表情を付けるために意図的にテンポやリズムを変動させる通常の演奏では，正規の音符の長さ（以下音価）を表現しきれず非常に見づらい楽譜を出力してしまう．そこで，Cemgil らは [1] で様々な演奏家の演奏スタイルに対応させたテンポモデルを決定し，それを用いてリズム量子化を行った．しかし，１つのテンポモデルを用いて多くのテンポスタイルを持った楽曲のリズム量子化を行う事は難しい．

そこで本稿では，複数のテンポモデルを学習した後，このテンポモデルを併合するクラスタリングによって，ある程度のクラスタ数に減少させたいくつかのテンポモデルから選択された最尤モデルを用いたリズム量子化手法を提案し，また，クラシック楽曲によく現れる装飾音符も１つの音符として，実際に演奏されたクラシック楽曲の単旋律のリズムの量子化を行う．

2. 問題定義

オンライン入力で演奏された MIDI データから抽出した開始時刻情報を使用して，演奏の元となったリズム譜を推定したい．リズム譜とは，リズム情報のみを記した楽譜であり，ここでのリズムとは，離散的な量である音符の並びを示す．テンポの変動軌道 $z_{0:K}$ で演奏者が K 個の音符の並んだリズム符からリズム $\gamma_{1:K}$ を演奏しようと意図して音符の開始時刻 $y_{0:K}$ で演奏したとする．リズムを認識すると言うことは，観測される $y_{0:K}$ から演奏者の意図したテンポの変動軌道 $z_{0:K}$ と $\gamma_{1:K}$ を推定する問題である．これは，観測された $y_{0:K}$ に対して最も尤もらしい $\gamma_{1:K}$ を推定する問題であり，つまり以下の事後確率最大化音符系列を求めることが問題となる．

$$\gamma^*_{1:K} = \arg max_{\gamma_{1:K}} p(\gamma_{1:K}|y_{0:K}) \tag{1}$$

$$p(\gamma_{1:K}|y_{0:K}) = \int dz_{0:K} p(\gamma_{1:K}, z_{0:K}|y_{0:K}) \tag{2}$$

3. 手法

最も尤もらしい音価系列であるリズムを得るには，テンポの変動軌道を推定する必要がある．そのために，テンポの変動のためのモデル（テンポモデル）を与える．連続値であるテンポ変動を隠れ変数とし，音価（離散値）をスイッチ変数としたスイッチングカルマンフィルタを用いてテンポ変動軌道と音価を推定する．しかし，このモデルは正確な確率計算を行うことが困難なスイッチング状態空間モデルであるので，近似法として，パーティクルフィルタを用いる [1].

また，楽曲や演奏者にはそれぞれ特有の「くせ」などがあることから，一つのモデルで様々な楽曲に対応することが困難であるので，k-means クラスタリングを用いて学習データから「くせ」のある観測に対応するモデルを選択する．

3.1 テンポモデル

テンポとは速度であり，k 時刻の基本テンポ $\Delta_{1,k}$ は緩やかに変化するテンポを表し，一時的テンポ $\Delta_{2,k}$ は一時的に変化するテンポを表す．τ_k は音符の理想開始時刻を表し，これらを以下で表す．

$$\Delta_{1,k} = \Delta_{1,k-1} + \xi_{\Delta_{1,k}} \tag{3}$$
$$\xi_{\Delta_{1,k}} \sim N(0, Q_{\Delta_1})$$

$$\Delta_{2,k} = a\Delta_{2,k-1} + \xi_{\Delta_{2,k}} \tag{4}$$
$$\xi_{\Delta_{2,k}} \sim N(0, Q_{\Delta_2})$$

$$\tau_k = \tau_{k-1} + \gamma_k\Delta_{1,k-1} + \gamma_k\Delta_{2,k-1} + \xi_{\tau_k} \tag{5}$$
$$\xi_{\tau_k} \sim N(0, Q_\tau)$$

パラメータ a とノイズ分散 $Q_{\Delta_1}, Q_{\Delta_2}, Q_\tau$ はモデル学習から得られ，楽曲の特有のテンポの変動を表現する．上記の (3),(4),(5) 式を以下の式 (6) で簡潔に表現する．

$$\begin{pmatrix} \tau_k \\ \Delta_{1,k} \\ \Delta_{2,k} \end{pmatrix} = \begin{pmatrix} 1 & \gamma_k & \gamma_k \\ 0 & 1 & 0 \\ 0 & 0 & a \end{pmatrix} \begin{pmatrix} \tau_{k-1} \\ \Delta_{1,k-1} \\ \Delta_{2,k-1} \end{pmatrix} + \boldsymbol{\xi}_k \tag{6}$$

$$\boldsymbol{\xi}_k \sim N(\mathbf{0}, diag(Q_\tau, Q_{\Delta_1}, Q_{\Delta_2}))$$

ここで，$\boldsymbol{\xi}_k = (\xi_{\tau_k}, \xi_{\Delta_{1,k}}, \xi_{\Delta_{2,k}})^T$ とする．また，k 時刻のテンポ z_k を $(\tau_k, \Delta_{1,k}, \Delta_{2,k})^T$ で表す．観測系列 y_k は理想開始時刻 τ_k にノイズを加えた以下の式で表す．

$$y_k = \tau_k + \epsilon_k \quad (\epsilon_k \sim N(0, R)) \tag{7}$$

3.2 音符の事前分布

音価は短くなるほど，起こりにくい．しかし，音価の確率は前に来る音価によっても変わってくると考えられる．例えば，図１のような楽譜は音楽としてあまりない．そこで音価ではなく，k 時刻のとき１時刻から k 時刻までの音価の和を量子化位置とし，量子化位置の小数点以下の桁数を用いて，音符の事前分布を求める．[1],[4]

*Rhythm Quantization Using a Particle Filter
†Takenori Minoura · Hiroshima City University
‡Akira Hayashi · Hiroshima City University

図 1: 楽譜の例

3.3 装飾音符の表現

クラシック音楽の楽曲には，一般的な音符の他に装飾音の一つである装飾音符が頻繁に現れる．装飾音符は実際にはわずかな音価が存在するが，楽譜上では音価は与えられない．ここでも，同様に装飾音符の音価を $0(\gamma = 0)$ とし，１つの音符として扱う．

3.4 クラスタリング

まず，学習データ１曲ごとに１つのテンポモデルのパラメータを EM アルゴリズムで学習する [2]．i 番目の演奏 $y^{(i)}_{0:K_i}$ から学習したテンポモデルを λ_i とする．このとき，モデル間距離 $\mathbf{d}(\lambda_i, \lambda_j)$ を以下の式で定義する．

$$\mathbf{d}(\lambda_i, \lambda_j) = \frac{1}{2}\{P(y^{(i)}_{0:K_i}|\lambda_j) + P(y^{(j)}_{0:K_j}|\lambda_i)\} \tag{8}$$

ここで，$P(y^{(i)}_{0:K_i}|\lambda_j)$ はモデル尤度であり，以下の式で表される．

$$\begin{aligned} P(y^{(i)}_{0:K_i}|\lambda_i) &= \sum_{\gamma^{(i)}_{1:K}} P(y^{(i)}_{0:K_i}, \gamma^{(i)}_{1:K_i}|\lambda_i) && (9) \\ &= \sum_{\gamma^{(i)}_{1:K_i}} P(\gamma^{(i)}_{1:K_i}|\lambda_i) P(y^{(i)}_{0:K_i}|\gamma^{(i)}_{1:K_i}) && (10) \end{aligned}$$

その後，モデル間距離に基づいて，近しいモデルを併合する．モデルが併合されると再度モデルを学習する．このモデル併合をある程度のクラスタ数に減少するまで繰り返す．

なお，(9)，(10) 式のモデル尤度について本来はすべての音符列に関する和をとるが，ここでは，少数の音符列を除いては，音符列が与えられたときの，出力列の尤度は非常に小さいものとして，パーティクルフィルタで最後まで残った音符列（軌跡）についてのみ，和をとるものとする．

4. 実験

50 曲の学習データから得たモデルパラメータから，１曲ごとに１つのモデルを生成し，クラスタ数を５としてモデル併合を行う．その後，新たに与えられた４つの楽曲 (テストデータ) に対し，最も尤度の高いモデルを選択し，リズム量子化を行う提案手法（A）と，１つのモデルを全ての学習データを用いて学習したテンポモデル（B）を用いてリズム量子化した結果の精度を比較する．ここで，それぞれの楽譜を正解の音価列 $\gamma^{true}_{1:K}$ とし，使用するテストデータはアマチュアの演奏家が楽譜上の誤りなく自由なテンポで，(1) ガボット (バッハ作曲)，(2) メヌエット (バッハ作曲)，(3) バラード (ブルグミュラー作曲)，(4) アラベスク (ブルグミュラー作曲) について演奏したデータとする．

4.1 評価尺度

精度を対数尤度差 ((14) 式) とエディット距離 ((15) 式) を使用して評価をする．エディット距離は，リズム量子化を単に誤った音符を数えたものである．

$$\Delta L = \log \frac{p(y_{0:K}|\gamma_{1:K})p(\gamma_{1:K})}{p(y_{0:K}|\gamma^{true}_{1:K})p(\gamma^{true}_{1:K})} \tag{11}$$

$$e(\gamma_{1:K}) = \sum_{k=1}^{K}(1 - \delta(\gamma_k - \gamma_k^{true}) \tag{12}$$

ここで，$\gamma^{true}_{1:K}$ は正解のリズムを表す．

4.2 結果

表１に，２つの手法を用いて評価を行った結果を示す．テストデータごとの総音符数は上から 260,114,209,183 であり，A が提案手法，B が全ての学習データから１つのモデルを生成したものである．

表 1: リズム量子化結果

	対数尤度差		エディット距離	
曲番号	A	B	A	B
1	-7.9755	-18.0	8	23
2	-15.7403	-98.9795	11	19
3	-1.9949	-19.8720	14	17
4	-5.1062	-45.6446	10	16

5. まとめと今後の課題

学習データからの最尤モデル選択を用いた提案手法と全ての学習データから１つのモデルを生成した手法を用いて実際に演奏された４曲の異なるクラシック楽曲を使用してリズム量子化を行い，比較し，提案手法の有効性を確認した．

今後の課題として，クラシックによく出現するフェルマータのような大きくテンポを変化させる場合についても考えていく必要がある．

参考文献

[1] A. T. Cemgil, B. Kappen, ”Monte Carlo Methods for Tempo Tracking and Rhythm Quantization” Journal of Artificial Intelligence Research,18(1),45-81(2003)

[2] Z.Ghahramani, G.E.Hinton,”Parameter estimation for liner dynamical systems”.(crgtr-96-2).Tech,rep., University of Totronto.Dept. of Computer Science.(1996)

[3] A.Doucet, S. Godsill, C.Andrieu. ”On sequential Monte Carlo sampling methods for Bayesian filtering”.Statistics and Computing, 10(3), 197-208.(2000)

[4] A. T. Cemgil, B. Kappen, ”Rhythm quantization for transcription” Computer Music Journal,24:2,60-76.(2000)