

LMS 型フーリエアナライザを用いた神楽音の分析 II

Analysis of real played musical signals using an LMS-typed Fourier Analyzer II

○渡辺 裕貴 工藤 憲昌† 田所 嘉昭†

○Hiroataka Watanabe Norimasa Kudoh† Yoshiaki Tadokoro††

八戸高専専攻科 †八戸高専 ††豊橋技科大

Hachinohe National College of Tech. Advanced Engineering Course

†Hachinohe National College of Tech.

††Toyohashi University of Technology

キーワード : LMS アルゴリズム (LMS algorithm), 採譜(musical transcription)
MIDI(Musical Instrument Digital Interface)

連絡先 : 〒039-1192 八戸市田面木上野平 16 番 1 号 八戸高専 電気情報工学科
tel:0178-27-7281, e-mail:kudohk-e@hachinohe-ct.ac.jp

1. はじめに

神楽とは神社等で演奏される民俗芸能であり, その演奏は笛・太鼓・鉦などから構成され, 代々弾き語りや口伝えによって伝承されてきた. しかし, 時代の変移とともに後継者が減少しつつあり, やがては神楽が途絶えてしまう恐れがある. そのため, 民俗芸能の伝承という点から楽譜として残すことは有意義である.

一般に, 楽音は「振幅・周波数・音の長さ」の 3 つの要素から構成され, 楽器は各音高に固有の周波数を持つ. そこで, 本研究では各音高(周波数)における振幅・時間指標を推定することで演奏音を同定し, また, 併せて演奏音の分析結果を基に MIDI ファイルの作成および楽譜化を検討した. なお, 分析対象としては, 分析が最も難しいとされる「笛の演奏」とする.

以下に本稿の構成を示す. 2. では神楽音の分析の前処理である周波数推定について概説する. 3. は振幅推定についての概説であり, 3.1. では定常状態における振幅推定, 3.2. では隣接音高からの信号の干渉に対する提案法について述べる. 4. は MIDI ファイル化について概説であり, 4.1. では MIDI のファイル構成, 4.2. ではデータ構成について述べる. 5. は数値例, 6. はまとめである.

2. 周波数推定

Fig.1 に示す周波数適応フィルタを用いて, 笛の音階の周波数推定を行う. 周波数適応フィルタは, ノッチフ

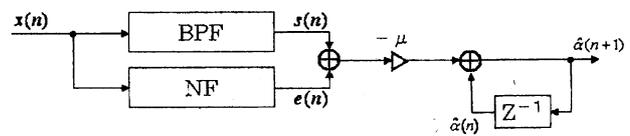


Fig.1 周波数適応フィルタの構成(単一周波数)

ィルタ(NF)とバンドパスフィルタ(BPF)の組み合わせによって構成され, それぞれの伝達関数は $H_N(Z)$, $H_S(Z)$ で与えられる. ここで, α はフィルタ係数, γ は極半径である.

$$H_N(Z) = \frac{1 - \alpha Z^{-1} + Z^{-2}}{1 - \gamma \alpha Z^{-1} + \gamma^2 Z^{-2}} \quad (1)$$

$$H_S(Z) = (1 - \gamma) Z^{-1} \frac{-1 + \gamma Z^{-2}}{1 - \gamma \alpha Z^{-1} + \gamma^2 Z^{-2}} \quad (2)$$

式(3)のように入力信号 $x(n)$ を単一周波数としたとき, フィルタ係数の真値およびその更新式はそれぞれ式(4),(5)となる. ここで, n は時間指標, $\phi(n)$ はガウス性白色雑音, μ はステップサイズ, f_s はサンプリング周波数, $\hat{\alpha}$ はフィルタ係数の推定値, $e(n)$, $s(n)$ は各フィルタの通過信号である.

$$x(n) = a \cos \omega n + b \sin \omega n + \phi(n) \quad (3)$$

$$\alpha = 2 \cos \left(\frac{2\pi f}{f_s} \right) \quad (4)$$

$$\hat{\alpha}(n+1) = \hat{\alpha}(n) - \mu \cdot e(n) \cdot s(n) \quad (5)$$

式(5)において、NFによって除去される周波数が入力信号の周波数に一致することで、 $e(n)$ はほぼ0となる。よって、 α が収束し、式(6)から周波数が算出できる。

$$f = \frac{f_s}{2\pi} \cos^{-1} \left(\frac{\alpha}{2} \right) \quad (6)$$

このとき、複数の周波数の場合に対しては、NFをトリ一状に縦続接続することで容易に拡張できる。

3. 振幅推定

3.1. 定常状態における振幅推定^[1]

LMS(Least Mean Square)型フーリエアナライザとは、対象の周波数における振幅を逐次推定するアルゴリズムである(以下、LMS法と呼ぶ)。LMS法では、入力信号 $x(n)$ とその推定信号 $\hat{x}(n)$ との推定誤差 $e(n)$ の最小化を行うことで振幅を推定するとともに、時間指標 n から音の長さを推定することができる。入力信号を式(7)としたとき、推定信号および振幅の更新式はそれぞれ式(8)、(9)となる。このとき、時間指標 n における時刻 t は式(10)で与えられる。ここで、 ω_i は既知の角周波数、 γ はLMS法における周波数特性の帯域幅を制御するパラメータである。

$$\hat{x}(n) = \sum_{i=1}^p \{ a_i \cos \omega_i n + b_i \sin \omega_i n \} + \phi(n) \quad (7)$$

$$\hat{x}(n) = \sum_{i=1}^p \{ \hat{a}_i \cos \omega_i n + \hat{b}_i \sin \omega_i n \} \quad (8)$$

$$\hat{a}_i(n+1) = (1+\gamma)\hat{a}_i(n) - \gamma \cdot \hat{a}_i(n-1) + \mu \cdot e(n) \cos \omega_i n \quad (9-a)$$

$$\hat{b}_i(n+1) = (1+\gamma)\hat{b}_i(n) - \gamma \cdot \hat{b}_i(n-1) + \mu \cdot e(n) \sin \omega_i n \quad (9-b)$$

$$t = \frac{n}{f_s} \text{ [sec]} \quad (10)$$

LMS法のブロック図および周波数特性をそれぞれFig.2, Fig.3に示す。Fig.3のように定常状態におけるLMS法の周波数特性は、推定対象の周波数 f_i を中心周波数とするBPF特性を持つ。しかし、他の周波数に対して振幅がヌル特性でないため、隣接音高からの信号の干渉が発生する。この信号を洩れ込み信号と呼ぶ。

3.2. 提案法^[2]

隣接音高からの洩れ込み信号を排除するアルゴリズムを提案する。例として、推定対象の音高 f_1 と隣接音高 f_2 があり、隣接音高からの洩れ込み信号の排除を行う場合について述べる。まず、推定対象の音高に加えて隣接音高の振幅も推定する。そして、推定対象の音高の振幅推定値のみを採用し、隣接音高の振幅推定値は破棄する。

提案法のブロック図をFig.4に示す。 $x(n)$ から $\hat{x}(n)$ の

経路の伝達関数は式(11)となり、隣接音高の振幅推定値を破棄した伝達関数は式(12)となる。ここで、 $G_1(Z), G_2(Z)$ はFig.2における $e(n)$ から $\hat{x}(n)$ の経路の伝達関数であり、各音高におけるLMS法のアルゴリズムを示す。

$$\text{伝達関数: } G = \frac{\hat{X}(Z)}{X(Z)} = \frac{G_1 + G_2}{1 + G_1 + G_2} \quad (11)$$

$$G = \frac{\hat{X}(Z)}{X(Z)} = \frac{G_1}{1 + G_1 + G_2} \quad (12)$$

一例として、 $f_1=2000$ [Hz], $f_2=2343.75$ [Hz]における式(11)、(12)の周波数特性をFig.5に示す。Fig.5より、提案法である式(12)では、隣接音高の周波数成分を排除できることがわかる。また、従来法と提案法の周波数特性を比較したものをFig.6に示す。提案法では、従来法とほぼ同一の帯域幅を維持したまま、洩れ込み信号のみを排除することが確認できる。

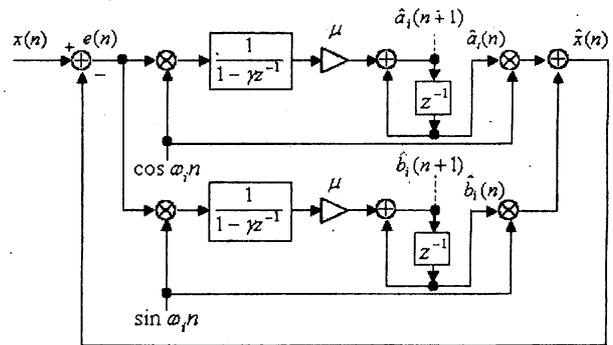


Fig.2 LMS法のブロック図

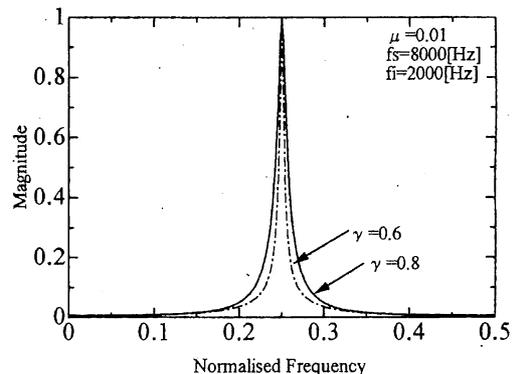


Fig.3 LMS法の周波数特性

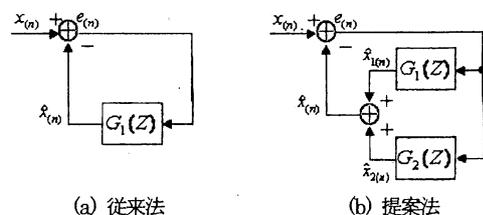


Fig.4 LMS法のブロック図

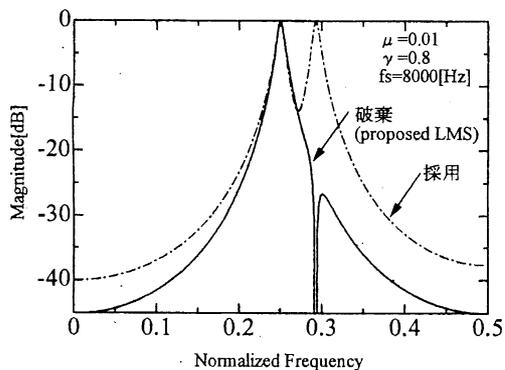


Fig.5 提案法の周波数特性

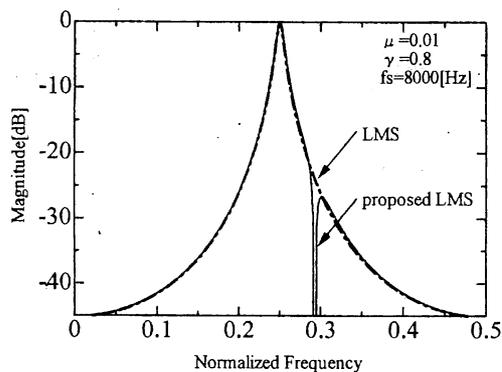


Fig.6 周波数特性の比較

4. MIDI ファイル化

MIDI ファイルとは、音声ファイルの一種であり、専用のソフトウェアによって、容易に楽譜を作成することができる。そこで、神楽音の分析結果を基に MIDI ファイルを作成し、楽譜化を検討する。

4.1. ファイル構成

MIDI ファイルは、ヘッダチャンクと複数のトラックチャンクから構成される。ヘッダチャンクでは MIDI ファイルの書式設定などを行い、トラックチャンクには音の情報であるデータ本体が記憶されている。

4.2. データ構成

トラックチャンクでの 1 つのイベント(音を鳴らすなど)は、「デルタタイム+イベント」という形式で記述される。ここで、イベントは固定長のデータであり、記述する Byte 数が定められている。一方、デルタタイムは前のイベントとの時間差であり、可変長のデータとなる。一般に、MIDI ファイルは 1Byte 単位でデータを読み取るため、Byte 数が可変長のデータであるデルタタイムは特殊な表現が必要となる。これを可変長表現と呼ぶ。

可変長表現では 1Byte = 8bit の内、8bit 目を可変長フラグとし、残りの 7bit(16 進数で 0~7F の値)がデータ表現の使用領域となる。ここで、8bit 目が "1" ならば可変長表現とする。例として、2Byte のデータを可変長表現に変換するアルゴリズムを Fig.7 に示す。このとき、2Byte

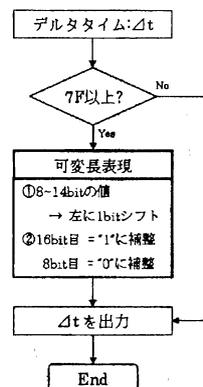


Fig.7 可変長表現のアルゴリズム

表.1 笛の音階の周波数

音高	規定周波数[Hz]
No.1	687.5600
No.2	848.3856
No.3	991.1740
No.4	1057.0540
No.5	1246.1890
No.6	1373.6860
No.7	1688.5900
No.8	1968.0630

の内、各 Byte の 1bit ずつを可変長フラグとして使用するため、データ表現の使用領域は 0~14bit(16 進数で 0~3FFF の値)となる。

5. 数値例

5.1. 笛の音階の周波数推定

笛の音階の周波数推定結果を表.1 に示す。これは規定の音高の演奏音を分析したもので規定周波数と呼ぶ。表.1 より、音高 No.1~No.3 と音高 No.6~No.8 はそれぞれ倍音関係にあることがわかる。

5.2. 洩れ込み信号の排除

笛の音高には倍音が存在するため、隣接音高以外の音高からの洩れ込み信号の発生も考えられる。そこで、推定対象以外の全ての音高からの洩れ込み信号の排除を行う。このとき、提案法の伝達関数は式(13)となる。ここで、 G_i は推定対象の音高における LMS 法のアルゴリズムである。

$$G = \frac{\hat{X}(Z)}{X(Z)} = \frac{G_i}{1 + \sum_{m=1}^8 G_m} \quad (13)$$

笛の音階の振幅推定結果を Fig.8 に示す。Fig.8 より、音高 No.6~No.8 において倍音を検出していることがわかる。これは周波数間隔が極めて狭いため、倍音からの洩れ込み信号を検出してしまったのだと考えられる。このとき、Fig.9 に音高 No.3, No.4 の振幅推定結果を示す。Fig.9 より、従来法では隣接音高以外からの洩れ込み信号も検出されており、これに対して提案法ではそれらの洩れ込み信号を軽減できていることがわかる。

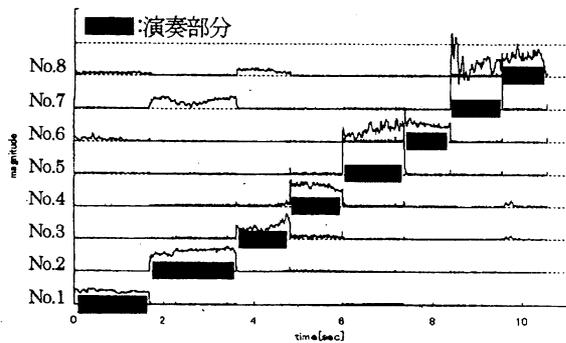
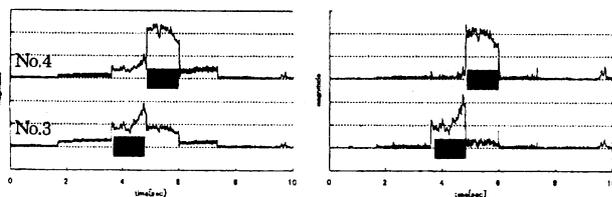


Fig.8 笛の音階の振幅推定結果



(a) 従来法 (b) 提案法
Fig.9 音高 No.3, No.4 の振幅推定結果

5.3. 神楽音の振幅推定

神楽音の振幅推定を行う。このとき、提案法による振幅推定の結果に対して以下の処理を加える。

①雑音の除去

微小な振幅は、雑音として除去する。

②倍音の除去

Fig.8 より、倍音が検出された場合、実際に演奏されている音は周波数の低い音高であることから、周波数の高い音高は演奏されていないものとする。

③同時刻音の排除

一般に楽器の構造上、同時刻に2つ以上の音が演奏されることはない。しかし、ヒトの演奏は人為的誤差(笛の穴の押さえ方が不十分など)を伴うため、規定周波数との誤差が生じる。このとき、提案法では規定周波数以外の周波数に対する洩れ込み信号の排除は行っていないため、洩れ込み信号が発生する。その結果、同時刻に複数の振幅が検出される。そこで、任意の時刻において最大振幅の音高を「演奏者が最も演奏したい音」とし、それ以外の音高は演奏されていないものとする。

④閾値による判定

演奏された1つの音に対して、その期間に推定された振幅の平均振幅の60%を閾値とし、それ以下の振幅の場合は演奏されていないものとする。

音源である神楽音の入力波形を Fig.10、神楽音の振幅推定結果を Fig.11 に示す。推定結果より、神楽音を同定できていることが確認できる。

5.4. MIDI ファイルおよび楽譜の作成

5.3.の結果から、MIDI ファイルを作成し、楽譜化を行った結果を Fig.12 に示す。このとき、楽譜化にはフリーソフトである Finale を使用した。Finale を用いて MIDI

ファイルを開くことで、容易に楽譜が作成できることが確認できた。

6. まとめ

本報告では、推定対象以外の音高からの洩れ込み信号を排除するアルゴリズムを提案した。その結果、提案法では洩れ込み信号を大きく軽減することで、推定性能の向上が確認できた。また、併せて MIDI ファイルおよび楽譜の作成を試みた。今後は、閾値による判定に AGC(Automatic Gain Control)の機能を追加すること、そして、実用性のある楽譜の作成に向けて、より高精度の MIDI ファイルの作成について検討していきたい。

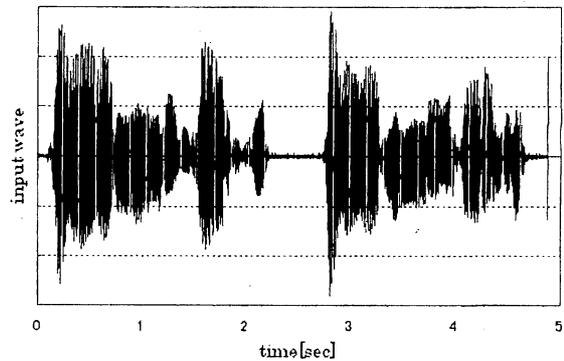


Fig.10 神楽音の入力波形

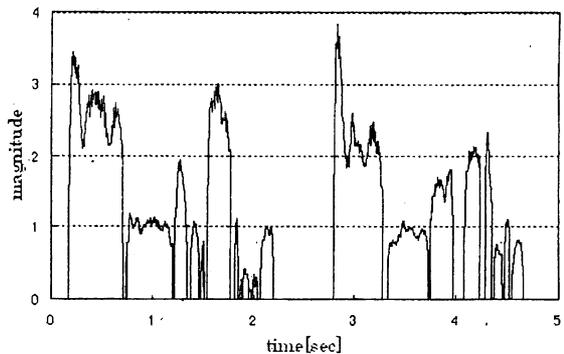


Fig.11 神楽音の振幅推定結果

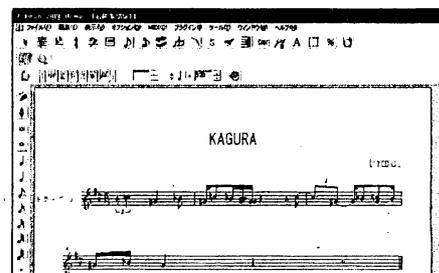


Fig.12 楽譜の作成結果

参考文献

- [1] 工藤, 田所 “ノッチフィルタと適応アルゴリズムによる雑音中の信号に対するフーリエ係数推定法” 信学論 A J83 pp.379-386, Apr. 2000.
- [2] 渡辺, 工藤, 釜谷 “LMS 型フーリエアナライザを用いた演奏音の分析” 電気関係学会 p.153, Aug.2009.