

オーディオ-MIDI 符号化ツール「オート符」 における倍音除去機能の実装

茂出木敏雄[†]

既開発の音響信号から MIDI 符号に自動変換するツール「オート符」は、一般化調和解析に基づく周波数解析を採用することにより倍音を含む和音を高精度に解析できるという特徴があり、音声信号を与えると、標準的な MIDI 音源で近似的に音声を再現可能な MIDI データを生成できる。しかし、本ツールを楽曲の MIDI 打ち込み支援の用途に活用する場合、生成される MIDI データより倍音に対応する音符だけを高精度に除去する機能が要望される。一般的な楽曲では倍音は複数の基音に由来して発生することが多く、倍音と基音を兼ねる音符も少ないため識別は容易ではなかった。本稿では、周波数解析における時間分解能を向上させることにより、倍音と基音の識別精度を大幅に向上させることが実現できたので、その結果を報告する。

Implementation of Harmonic Tone Removal Functions for Audio-MIDI Encoder Tool “Auto-F”

Toshio Modegi[†]

Our previously developed audio to MIDI code converter tool “Auto-F” has a feature of high-precision harmonic tone analysis functions based on the Generalized Harmonic Analysis algorithm. Applying this tool, from given vocal acoustic signals we can create MIDI data, which enable to playback the voice-like signals with a standard MIDI synthesizer. However, for general MIDI editing purposes, encoded harmonic tone note-events should be identified and removed. It has been difficult to distinguish harmonic tone note-events from fundamental ones, because a harmonic tone may be based on multiple fundamental tones and some harmonic tone may be also a fundamental one in general music works. In this paper, we propose an improved frequency analysis method with high-precision temporal analysis, which can distinguish harmonic tone components from fundamental ones more precisely.

1. はじめに

筆者らは与えられた音響信号に対して一般化調和解析を用いて平均律音階のスケールで高精度な周波数解析を行い、MIDI データ形式に自動変換する技術の開発を進めてきた[1][2]。本技術は「オート符@SA」という名称で汎用的な音響解析ツールとしてまとめ、2001年より財団法人デジタルコンテンツ協会のホームページより無償配布を進めており、主として採譜業務の支援等に活用いただいている[6]。

本解析ツールは、特に和音解析精度が高く、音声信号に適用すると解析されたフォルマント成分が MIDI 形式に和音近似され、一般的な MIDI 音源を用いてボーカルが再現できるという特徴をもつ。そこで、より明瞭なボーカルが再現できるよう周波数解析機能に改良を加え、SMF ファイルに著作権情報などを MIDI 形式に符号化したボーカルデータを可聴または非可聴の形態で埋込んだり[3]、楽器音を用いた新規な音声合成システムへの応用を試みている[4]。また、電子透かし埋め込みなどの音響信号処理を施した前後の各音響信号に対して一般的な MIDI 音源で再生可能な MIDI データに符号化変換を行い、更に2つの MIDI データの時間軸方向の排他的論理和に対応する差分 MIDI データを作成することにより、音響信号処理による信号劣化成分をピアノロール等により可視化したり、標準 MIDI 音源により可聴化する手法も提案した[5]。

しかし、本ツールを楽曲の MIDI 打ち込み支援の用途に活用する場合、高精度に解析される倍音は返って邪魔であり、生成される MIDI データより倍音に対応する音符だけを高精度に除去する機能が要望される。倍音除去機能については、既開発のツール[6]にも実装していたが、一般的な楽曲では倍音は複数の基音に由来して発生することが多く、倍音と基音を兼ねる音符も少ないため識別は容易ではなかった。そのため、基音を過剰に削除することが多く、実用に供しない面があった。本稿では、解析フレームの配置方法を見直し、周波数解析における時間分解能を向上させることにより、倍音と基音の識別精度を大幅に向上させ、低音部の音符から倍音成分を逐次削除する手法をとることにより、倍音を高精度に除去することが実現できたので、その結果を報告する。

2. 既提案の音響信号の MIDI 符号化ツールの概要と改良手法

図1において(1)~(5)の処理は、筆者らが先に開発した MIDI 符号化処理の基本構成を示し[2]、(6)~(9)の処理は文献[4]で提案した音声信号向けの追加機能を示す。本稿では、処理(10)の機能を新たに追加しており、処理(1)~(4)に対して改良を加えており、処理(2)については、(2-1)~(2-4)という構成に拡張している。

[†] 大日本印刷株式会社 情報コミュニケーション研究開発センター
Media Technology Research Center, Dai Nippon Printing Co., Ltd. (E-mail: Modegi-T@mail.dnp.co.jp)

既提案の機能を含む図1の(1)~(10)の各処理機能に対応して以下10節で個別に説明する。

(1) 解析フレームの抽出

音響解析では解析フレームを時間軸方向に移動させながら、信号全体の解析を行うが、この際のフレーム長とシフト幅の設定方法について以下述べる。

周波数分解能はフレーム長により変化し、経験上ソース音響信号のサンプリング周波数が44.1[kHz]の場合、低域部まで忠実に解析するためには4096サンプル以上必要である。解析時の解析フレーム長は解析周波数ごとに可変に設定するが、抽出するフレーム長は上限値として、例えば $T=4096$ を与える。

一方フレームシフト幅 W は、小さくするほど時間分解能が向上するが計算時間も増大する。そして、解析対象信号が単調である箇所に対して、必要以上にフレームシ

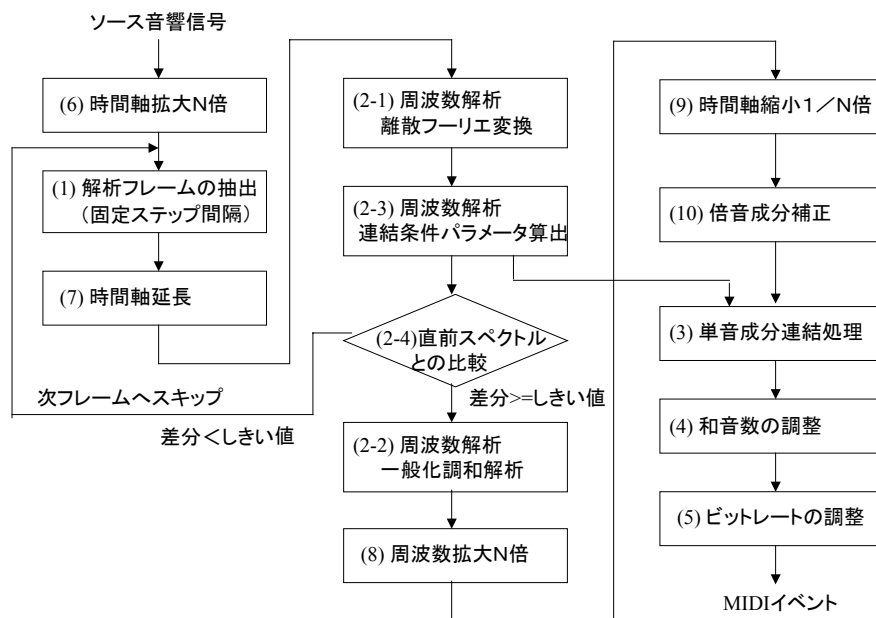


図1 既提案のMIDI符号化処理の構成と追加改良構成

フトを細かくすると、後述する単音成分連結処理で支障をきたす。そこで、効率的な計算および高精度な単音成分連結処理のためにも、フレームシフト幅 W は解析対象フレームごとに変化させ、最適な値を設定する方法が望ましい。先提案ではゼロ交差解析を行って、最適なフレームシフト幅を設定していたが[4]、本稿では後述する周波数解析時において離散フーリエ変換を行う段階で決定するようにしたため、ここでは最小値(例えば、 $T=4096$ に対して $W=64$ サンプル)を固定で与えればよい。

(2) 周波数解析

(2-1) 周波数解析・離散フーリエ変換 前節で述べた方法により、サンプリング周波数 f_s の原音響信号より p 番目に抽出された解析フレームのサンプル配列を $x(p,i)$ ($0 \leq i \leq T-1$)とする。本周波数解析は、 n ($0 \leq n \leq 127$)をMIDIのノートナンバーとして128種の解析周波数 $f(n)=440 \cdot 2^{(n-69)/12}$ の調和関数を基本にした離散フーリエ変換により行う。ただし、後述する(6)時間軸拡大 N 倍を実行した場合は、 $a=12 \cdot \log_2 N$ (例えば、 $N=2$ の場合 $a=24$)として、解析するノートナンバーの上限は $127-a$ となる。周波数が高くなるにつれ、ノートナンバー間の周波数間隔が広がるため、特に $n > 60$ では解析精度が低下してしまう。そこで、ノートナンバー間を以下のように M 個の微分音に分割した $128M$ 種の調和関数を用いて解析を行う[4]。

$$f(n,m)=440 \cdot 2^{(n-69+m/M)/12} \quad (1)$$

p 番目の解析フレームに対して、ノートナンバー分の相関配列 $E(p,n)$ ($-a \leq n \leq 127-a$)と副周波数配列 $S(p,n)$ を定義し、 $-a \leq n \leq 127-a$ および $0 \leq m \leq M-1$ に対して以下式で相関計算を行う。式(2)において $T(n,m)$ は解析フレーム長で、調和関数の1周期がフレーム長 T 以下の場合、フレーム長 T を超えない範囲で調和関数の周期の最大の整数倍になるよう設定し、 k を1以上の適当な整数値として、 $T(n,m)=k/f(n,m)$ で与える。調和関数の1周期がフレーム長 T より大きい場合、 $T(n,m)=1/f(n,m)$ で与え $T < i \leq T(n,m)$ における $x(p,i)$ の値については、後述する(7)時間軸延長処理に基づき延長する処理を行う。

$$\begin{aligned} A(p,n,m) &= 1/T(n,m) \cdot \sum_{i=0, T(n,m)-1} x(p,i) \sin(2\pi f(n,m)(i+pW)/f_s) \\ B(p,n,m) &= 1/T(n,m) \cdot \sum_{i=0, T(n,m)-1} x(p,i) \cos(2\pi f(n,m)(i+pW)/f_s) \\ E(p,n,m) &= A(p,n,m)^2 + B(p,n,m)^2 \end{aligned} \quad (2)$$

ここで、 $p > 0$ で $A(p-1,n,m)$ と $B(p-1,n,m)$ の値が既知の場合、(2)式の $A(p,n,m)$ と $B(p,n,m)$ を算出する式は以下のように変形でき、直前解析フレーム $p-1$ における相関計算結果を用いて、計算範囲を縮小でき高速に算出できる。

$$\begin{aligned} A(p,n,m) &= A(p-1,n,m) \\ &\quad - 1/T(n,m) \cdot \sum_{i=0, W-1} x(p-1,i) \sin(2\pi f(n,m)(i+(p-1)W)/f_s) \end{aligned}$$

$$\begin{aligned}
 & +1/T(n,m) \bullet \sum_{i=T(n,m)-W, T(n,m)-1} x(p,i) \sin(2\pi f(n,m)(i+pW)/f_s) \\
 B(p,n,m) & = B(p-1,n,m) \\
 & -1/T(n,m) \bullet \sum_{i=0, W-1} x(p-1,i) \cos(2\pi f(n,m)(i+(p-1)W)/f_s) \\
 & +1/T(n,m) \bullet \sum_{i=T(n,m)-W, T(n,m)-1} x(p,i) \cos(2\pi f(n,m)(i+pW)/f_s) \\
 E(p,n,m) & = A(p,n,m)^2 + B(p,n,m)^2 \quad (2')
 \end{aligned}$$

続いて、ノートナンバー n ごとに、 $0 \leq m \leq M-1$ の範囲で $E(p,n,m)$ を最大にする $E(p,n,m_{max})$ を求め、 $E(p,n) = E(p,n,m_{max})$ 、 $S(p,n) = m_{max}$ と定義する。本節で算出された $E(p,n)$ と $S(p,n)$ を保存し、(3) 単音成分連結処理にて使用する。

(2-3) 周波数解析・連結条件パラメータ設定 (2-1) で算出された $E(p,n)$ と $S(p,n)$ を保存し、(3) 単音成分連結処理にて使用する。

(2-4) 周波数解析・直前スペクトルとの比較 (2) 式で算出された $E(p,n)$ と直前解析フレームにおける $E(p-1,n)$ との差分 $dE(p-1,p)$ を以下のように算出し、 $dE(p-1,p)$ が所定のしきい値 (例えば、40) 未満であれば、(1) で次の解析フレームに進み、所定のしきい値以上であれば、次の (2-2) 周波数解析・一般化調和解析へ進む。

$$dE(p-1,p) = 100/N \bullet \sum_{n=0, N-1} |E(p,n) - E(p-1,n)| / \{E(p,n) + E(p-1,n)\} \quad (3)$$

(2-2) 周波数解析・一般化調和解析 解析フレーム p は q 番目に一般化調和解析を行う可変解析フレームであるとし、解析フレーム ID 配列を $P(q)$ とすると、 $P(q) = p$ と設定し、可変解析フレーム q において、ノートナンバー分の相関配列 $E_o(q,n)$ ($-a \leq n \leq 127-a$) を定義し、初期値を全て -1 とする。

(a) $E_o(q,n) < 0$ でかつ $E(p,n)$ が最大になる $E(p,n_{max})$ を求め、 $m_{max} = S(p,n_{max})$ とする。式(2)を簡素化した以下式(4)を用いて $A(p,n_{max},m_{max})$ および $B(p,n_{max},m_{max})$ を再計算する。

$$\begin{aligned}
 A(p,n_{max},m_{max}) & = 1/T(n_{max},m_{max}) \bullet \sum_{i=0, T(n_{max},m_{max})-1} x(p,i) \sin(2\pi f(n_{max},m_{max})i/f_s) \\
 B(p,n_{max},m_{max}) & = 1/T(n_{max},m_{max}) \bullet \sum_{i=0, T(n_{max},m_{max})-1} x(p,i) \cos(2\pi f(n_{max},m_{max})i/f_s) \\
 E_o(q,n_{max},m_{max}) & = A(p,n_{max},m_{max})^2 + B(p,n_{max},m_{max})^2 \quad (4)
 \end{aligned}$$

(b) 上記決定した $A(p,n_{max},m_{max})$ および $B(p,n_{max},m_{max})$ を用いて、以下式でサンプル配列 $x(p,i)$ の全ての要素 ($0 \leq i \leq T(n_{max},m_{max})-1$) を更新する。

$$\begin{aligned}
 x(p,i) & = x(p,i) - A(p,n_{max},m_{max}) \bullet \sin(2\pi f(n_{max},m_{max})i/f_s) \\
 & \quad - B(p,n_{max},m_{max}) \bullet \cos(2\pi f(n_{max},m_{max})i/f_s) \quad (5)
 \end{aligned}$$

(c) 再度(a)の処理に戻り、 $0 \leq n \leq 127$ の全ての $E_o(q,n)$ の値が 0 以上の値に決定されるまで(a)から(c)までの処理を繰り返す。

(3) 単音成分連結処理

q 番目と $q+1$ 番目の可変解析フレームにより周波数解析されたノートナンバー n の単音成分を [時刻(q), 時間長(q), 主周波数 n , 副周波数 $S(P(q),n)$, 強度 $E_o(q,n)$] および [時刻($q+1$), 時間長($q+1$), 主周波数 n , 副周波数 $S(P(q+1),n)$, 強度 $E_o(q+1,n)$] とする。時刻(q)および時刻($q+1$)は各々 $P(q)$ 番目および $P(q+1)$ 番目の解析フレームの第 1 サンプルの原音響信号上の絶対サンプルアドレスをサンプリング周波数で除算することで得られる。時間長(q)は時刻($q+1$) - 時刻(q)で、時間長($q+1$)は時刻($q+2$) - 時刻($q+1$)で与えられる。また、連結する最初の変解析フレームを q_o 番目とし、同様に、ノートナンバー n の単音成分を [時刻(q_o), 時間長(q_o), 主周波数 n , 副周波数 $S(P(q_o),n)$, 強度 $E_o(q_o,n)$] と定義する。

連結条件は、文献[2]手法に比べ高精度に算出でき、 $q+1$ 番目の可変解析フレームの直前に位置する解析フレーム $P(q+1)-1$ と $q+1$ 番目の可変解析フレームの先頭に位置する解析フレーム $P(q+1)$ の解析結果、および連結する最初の q_o 番目の可変解析フレームの解析結果により決定される。即ち、 $P(q+1)-1$ 番目、 $P(q+1)$ 番目および $P(q_o)$ 番目の解析フレームにおけるノートナンバー n の単音成分である [副周波数 $S(P(q+1)-1,n)$, 強度 $E(P(q+1)-1,n)$] と [副周波数 $S(P(q+1),n)$, 強度 $E(P(q+1),n)$] および [時刻(q_o), 時間長(q_o), 主周波数 n , 副周波数 $S(P(q_o),n)$, 強度 $E_o(q_o,n)$] とが連結判定に使用される。

連結する最初の $P(q_o)$ 番目の解析フレームの単音成分と、後続する互いに時間的に隣接する解析フレームの 2 つ単音成分に対して、ノートナンバー n において上下 ± 1 の変移を考慮し、副周波数を考慮した、 q_o 番目の可変解析フレームとの周波数の差が所定値 N_{dif} 未満で、隣接する解析フレーム間の周波数の差が所定値 N_{dif} 未満で、双方の強度が所定のしきい値 L_{min} 以上でかつ双方の強度の差 L_{dif} が所定値以下で両者の連続性が認められる場合、即ち、以下(6-1)~(6-3)の 3 条件のいずれかを満たす場合、後続単音成分を前方単音成分に連結統合する。

$$\begin{aligned}
 |S(P(q_o),n) - S(P(q+1),n)| & < N_{dif} \text{ かつ} \\
 |S(P(q+1)-1,n) - S(P(q+1),n)| & < N_{dif} \text{ かつ} \\
 E(P(q+1)-1,n) & > L_{min} \text{ かつ } E(P(q+1),n) > L_{min} \text{ かつ} \\
 E(P(q+1),n) - E(P(q+1)-1,n) & < L_{dif} \quad (6-1)
 \end{aligned}$$

$$\begin{aligned}
 |S(P(q_o),n) - S(P(q+1),n-1) - M| & < N_{dif} \text{ かつ} \\
 |S(P(q+1)-1,n) - S(P(q+1),n-1) - M| & < N_{dif} \text{ かつ} \\
 E(P(q+1)-1,n) & > L_{min} \text{ かつ } E(P(q+1),n-1) > L_{min} \text{ かつ} \\
 E(P(q+1),n-1) - E(P(q+1)-1,n) & < L_{dif} \quad (6-2)
 \end{aligned}$$

$$\begin{aligned} |S(P(q_o),n) - S(P(q+1),n+1)+M| < N_{dif} \text{ かつ} \\ |S(P(q+1)-1,n) - S(P(q+1),n+1)+M| < N_{dif} \text{ かつ} \\ E(P(q+1)-1,n) > L_{min} \text{ かつ } E(P(q+1),n+1) > L_{min} \text{ かつ} \\ E(P(q+1),n+1) - E(P(q+1)-1,n) < L_{dif} \quad (6-3) \end{aligned}$$

ただし、連結後の主周波数・副周波数・強度は q_o 番目の可変解析フレームの単音成分の各値を採用し、時間長は双方の和、即ち時刻 $(q+2)$ - 時刻 (q_o) で与える。

本時系列の連結処理は、不連続性が認められるまで後続する複数の単音成分に対して繰り返し行い、最終的に統合された [時刻 (q_o) , 時間長 (q_o) , 主周波数 n , 副周波数 $S(P(q_o),n)$, 強度 $E_o(q_o,n)$] に対して、2つのMIDIノートイベントに変換する。時刻 (q_o) で、ノートナンバー n のノートオン・イベントを発行し、ベロシティ値は $E_o(q_o,n)$ の最大値を E_{max} として、 $128 \cdot \{E_o(q_o,n) / E_{max}\}^{1/4}$ で与える。時刻については、Standard MIDI File では、直前イベントとの相対時刻 (デルタタイム) で与える必要があり、その時刻単位は任意の整数値で定義でき、例えば、1/1536 [sec] の単位に変換して与える。そして、時刻 (q_o) + 時間長 $(q_o) \cdot C$ で、ノートナンバー n のノートオフ・イベントを発行する。C は $0 \leq C \leq 1$ の実数で、使用するMIDI音源の音色に依存するが、MIDI音源の余韻を考慮して早めにノートオフ指示をするためである。C=1 にしてもMIDI音源の処理上は問題ないが、後続音と部分的に重なる場合がある。

上記連結条件のしきい値の具体的な設定値は、通常、 $N_{dif}=8/25$ [単位: ノートナンバー]、 $N_{dif}=4/25$ [単位: ノートナンバー]、 $L_{min}=1$ [単位: ベロシティ]、 $L_{dif}=10$ [単位: ベロシティ] である。

(4) 和音数の調整

MIDI符号に変換する段階で、MIDI音源で処理可能な同時発音数についても考慮する必要がある。時間軸方向に発音期間中 (ノートオン状態) のノートイベントの個数を連続的にカウントし、例えば32和音を超えている箇所が見つかった場合は、ノートオン時のベロシティ値に対してノートオン時刻からの経過時間で補正した補正ベロシティ値を算出し、補正ベロシティ値で優先度を評価し、指定和音数以下になるよう優先度の低いノートイベント対を強制的にノートオフさせる補正処理を行う。この際、ベロシティ値またはデュレーション値のいずれかが所定の下限値より低い場合、優先度に関係無く削除する処理も加える。

i 番目のノートイベント $Ev(i)$ のノートオン時刻を $Ev(i).time$ 、ベロシティ値を $Ev(i).velocity$ とすると、時刻 $t (> Ev(i).time)$ におけるノートイベント $Ev(i)$ の補正ベロシティ値 $Vc(i,t)$ は、

$$Vc(i,t) = Ev(i).velocity \cdot \exp\{(t - Ev(i).time) \cdot \tau\}. \quad (7)$$

で定義する。 τ は減衰係数で例えば -1/1536 を与える。(時刻の単位を1秒あたり1536とすると、1秒後に1/2.7に減衰する。)

(5) ビットレートの調整

MIDI符号に変換する段階で、MIDI音源で処理可能なビットレートについても考慮する必要がある。時間軸方向に例えば1秒間隔にノートオンまたはノートオフイベントの個数をカウントし、各々の符号長を平均5バイト(40bits)としMIDI音源で処理可能な最大ビットレートを9000[bps]とすると、1秒間あたりイベント数が9000/40=225個を超えている区間が見つかった場合は、その区間に存在するノートオンまたはノートオフイベントと各々対になるノートオフまたはノートオンイベントを近傍区間内で探索し、各ノートイベント対のベロシティ値とデュレーション値 (ノートオフ時刻 - ノートオン時刻) の積 (エネルギー値) で優先度を評価し、指定イベント個数 (225) になるよう優先度の低いノートイベント対を局所的に削除する処理を行う。この際、ベロシティ値またはデュレーション値のいずれかが所定の下限値より低い場合、優先度に関係無く削除する処理も加える。

(6) 時間軸拡大N倍

上記(1)を実行する前に、与えられた音響波形信号に対して、時間軸方向にN倍 (例えば、N=4) だけ線形補間により拡大する処理を行う。サンプル i に対して原音信号を $x(i)$ 、拡大される信号 $x'(i)$ とすると、

$$x'(i \cdot N + k) = (1-k) \cdot x(i) + k \cdot x(i+1). \quad (0 \leq k \leq N-1) \quad (8)$$

(7) 時間軸延長処理

上記(6)を実行した場合、必要に応じて、上記(2-1)および(2-2)内において本処理を加える。解析フレーム長の上限值 T より、解析するノートナンバーに対応する調和関数の1周期が長い場合、解析フレーム長を擬似的に1周期分まで延長する処理を行う。この処理はノートナンバーごとに異なるため、(2-1)および(2-2)周波数解析において相関演算を実施する箇所で行う。

延長方法としては、解析に使用する調和関数の1周期分である相関計算区間 $T(n,m)$ を90度単位に4分割し、角度が低い方からQ1、Q2、Q3、Q4とする。相関計算区間が正弦波であると仮定すると、Q1とQ2は90度を軸に、Q3とQ4は270度を軸に時間軸方向に左右対称であり、Q2とQ3は180度を軸に時間軸および振幅軸方向に180度回転させた対称形である特徴を利用して、欠落部に左右反転または振幅反転を加えながら複写して補間する。

(8) 周波数拡大N倍

上記(6)を実行した場合、上記(2-2)を終了後に本処理を加える。全てのノートイベント(ノートオンまたはノートオフイベント)のノートナンバー値に $12 \cdot \log_2 N$ だけ加算する。例えば、 $N=4$ の場合 24 半音 (2 オクターブ) だけ全体的に音高を上げる。

(9) 時間軸縮小 1/N 倍

上記(6)を実行した場合、上記(8)を終了後に本処理を加える。全てのノートイベントの時刻(ノートオン時刻またはノートオフ時刻)に $1/N$ を乗算して縮小する。この処理を行うと、時間あたりのノートイベント数が N 倍に増大するため、後続処理(5)のビットレートの調整に対応する。

(10) 倍音成分補正

上記算出された $0 \leq n \leq 127$ の全ての $Eo(q,n)$ の値に対して、2,3,4,5,6,7,8,9,10 分の 1 の周波数に対応する 9 個のノートナンバー・オフセットテーブル $No(b)$ ($b=0, \dots, 8$) を定義して、次の通り補正を行う。ノートナンバー・オフセットテーブル $No(b)$ の具体例は、 $No(b) = \{12, 19, 24, 28, 31, 34, 36, 38, 40\}$ である。そして、ノートナンバー n に対応する強度値 $Eo(q,n)$ を次式の通り補正する。

$$Eo'(q,n) = Eo(q,n) - \sum_{b=0, \dots, 9} \{Eo(q,n) \cdot Eo(q,n - Nr(b))\}^{1/2} \cdot \gamma \quad (9)$$

γ は、 $0 \leq \gamma \leq 1$ の実数値で倍音補正強度を与える。通常の楽曲では $\gamma=1$ に設定し、ボーカルを含む音響信号で、ボーカルを生成する必要がある場合は $\gamma=0$ に設定する。下方にシフトさせたノートナンバー $n - Nr(b)$ が、 $n - Nr(b) < 0$ か $Eo(q,n - Nr(b))$ の値が存在しない場合、 $Eo(q,n - Nr(b)) = 0$ として計算し、補正後の $Eo'(q,n)$ が $Eo'(q,n) < 0$ の場合、 $Eo'(q,n) = 0$ とする。補正された相関配列 $Eo'(q,n)$ を $Eo(q,n)$ として次ステップの(3)単音成分連結処理以降は補正後の値を適用する。

3. おわりに

公開サイト[6]で提供している「オート符 version2.6」と前節で述べた改良方法を実装したツールを用いて、単旋律のピアノ独奏曲(モーツァルト「きらきら星変奏曲」、冒頭約 15 秒、44.1kHz/16bits/1-ch)に対して、最大和音数 3 2 に設定して符号化を行った結果を図 2・3 に示す。図中の着色された小さな矩形は音符(ノートイベント)を示し、横軸は時間で、横幅はノートオンからノートオフ区間を示す。縦軸は音高(ノートナンバー)を示すとともに、縦方向の幅でベロシティも示している。

図 2 で示される最大和音数 3 2 の MIDI データに対して最大和音数 1 に削減するように設定すると、倍音の代わりに基音が除去されたり、直前の音符がオーバーラップしている基音が一部欠けたり、して、演奏者が弾いた音符を正しく再現できなかった。

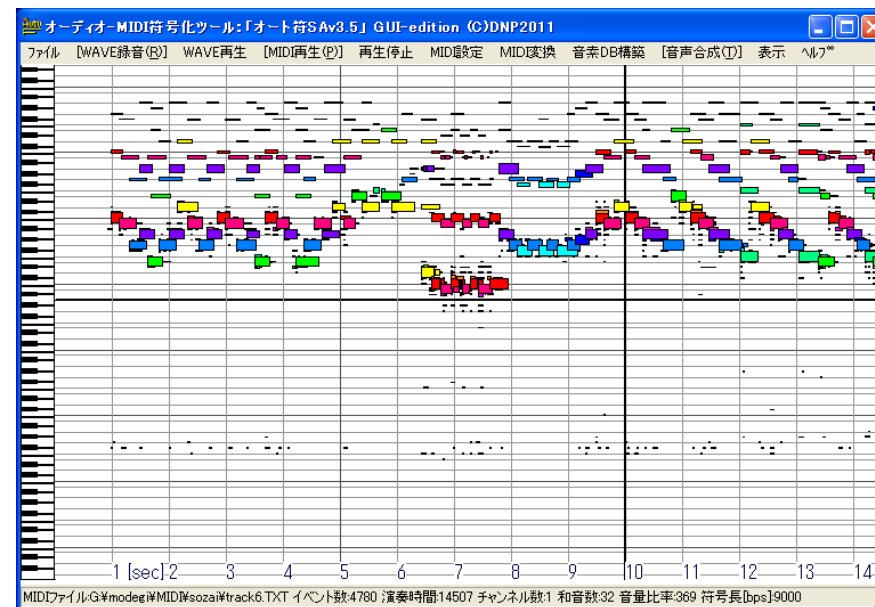


図 2 既提案の手法による MIDI 符号化結果 (3 2 和音)

一方、図 3 で示される最大和音数 3 2 の MIDI データに対して最大和音数 1 に削減するように設定して、五線譜に変換した結果を図 4 に示す。図 4 の結果は演奏者が弾いた音符を完全に反映できている。

紙面制約の都合で掲載を省略するが、和音を含む一般的なピアノ独奏曲で試したところ、テンポが比較的遅い曲(ムソルグスキー「展覧会の絵」プロムナードなど)では、図 2・図 3 には存在しない最大 6 重和音の全てが、演奏者が弾いた通り正確に再現できることを確認した。しかし、ショパンの幻想即興曲のようにテンポが速い楽曲については、倍音除去を行う前段階で正確に拾えない音符があり、(2)周波数解析における時間分解能の更なる改善と(3)単音成分連結処理の高精度化が今後の課題となる。

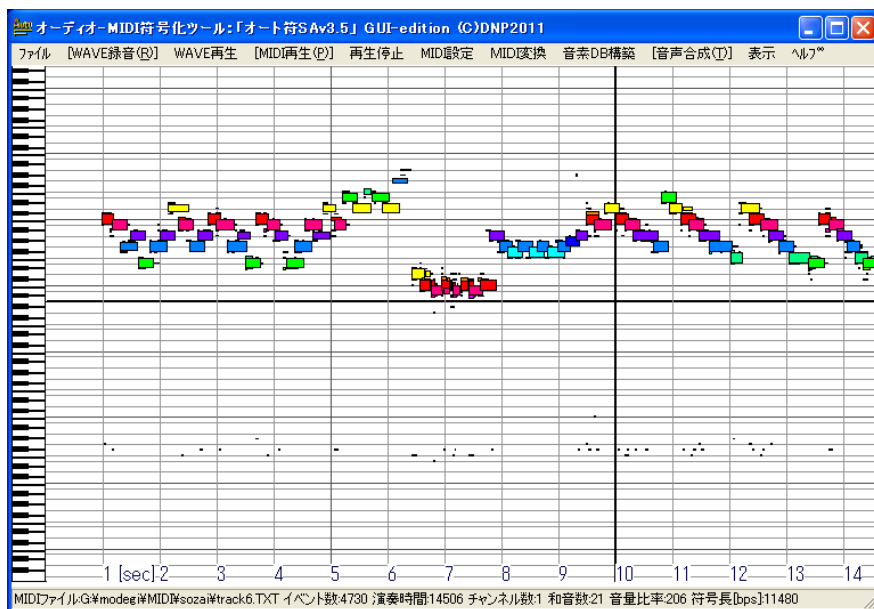


図3 本稿で提案する改良手法による MIDI 符号化結果 (3 2 和音)



図4 符号化された MIDI データの五線譜変換結果 (1 和音、五線譜エディタ・イーフロンティア PrintMusic2008 使用)

参考文献

- 1) Toshio Modegi: "Very low bit-rate audio coding technique using MIDI representation," Proceedings of the ACM 11th international workshop on Network and operating systems support for digital audio and video, pp.167-176, New York, USA, (June 2001).
- 2) 茂出木敏雄, "音響信号の平均律音階に基づく汎用解析ツール「オート符」の開発," 電気学会・電子情報システム部門誌, Vol.123-C, No.10, pp.1768-1775, (October 2003).
- 3) 茂出木敏雄: "音声 MIDI コードを用いた SMF ファイルへの情報埋め込み手法," 2009 電子情報通信学会・総合大会, 情報システム講演論文集 2, DS-3-3, pp.S35-S36 (March 2009).
- 4) 茂出木敏雄, "MIDI 符号化ツール「オート符」を用いた音素 MIDI コードの設計と楽器音による音声合成機能の実現," 電気学会・電子情報システム部門誌, Vol.130-C, No.7, pp.1159-1167, (July 2010).
- 5) 茂出木敏雄, "MIDI 符号化ツール「オート符」を用いた各種音響信号処理における品質劣

化の評価手法," 電気学会・2010 年電子情報システム部門大会・論文集, GS5-2, pp.1318-1325, (September 2010).

- 6) 財団法人デジタルコンテンツ協会 d-CON Support, <http://www.dcaj.org/d-con/frame09.html> (「オート符@SA」の無償配布元).