ISSN 1882-5370

尚美学園大学芸術情報研究　第 33 号

Journal of Informatics for Arts, Shobi University No.33

研究ノート | Research Notes

時間周波数解析法の精度改善とオーディオ -MIDI 変換ツール開発への応用

Precision Improvement of Time-frequency Analysis and Application to Development of Audio-MIDI Conversion Tool

茂出木　敏雄

MODEGI Toshio

尚美学園大学

情報表現学科　講師

Shobi University

2021 年 1 月

Jan.2021

時間周波数解析法の精度改善とオーディオ -MIDI 変換ツール開発への応用

Precision Improvement of Time-frequency Analysis and Application to Development of Audio-MIDI Conversion Tool

茂出木 敏雄
MODEGI Toshio

［抄　録］

MIDI カラオケなどの自動演奏データを作成する用途や、耳コピーの自動化や自動採譜の用途において、オーディオ信号を MIDI 形式に変換するツールの実現が要望される。前者においては、MIDI のベロシティやピッチベンドを認識して生演奏に近い演奏表現が求められるのに対し、後者においては、演奏上のテンポやピッチのゆらぎを除去して元の譜面に近い音符表現が求められる。いずれの場合においても、与えられたオーディオ信号を MIDI 形式に変換する信号処理系の精度が要求され、それを左右するのが時間周波数解析である。音楽情報は時間軸における音の周波数成分の時系列変化で表現されるが、時間周波数解析により特定される音の周波数成分と時系列変化の精度がトレードオフの関係になる不確定性原理の制約から、実用的な精度を得ることが困難であった。本稿では、一般化調和解析手法に改良を加え、時間周波数解析における周波数分解能と時間分解能の双方を改善させ、オーディオ -MIDI 変換ツールを開発したので、その結果を報告する。

キーワード

MIDI カラオケ , 自動採譜 , オーディオ -MIDI 変換 , 時間周波数解析 , 不確定性原理 , 一般化調和解析

[Abstract]

Realization of Audio-MIDI conversion tools are requested for creation of playback-control data such as for MIDI-Karaoke machines, and for development of an automatic music notation system. Realistic playback with MIDI velocity or pitch-bend parameters recognized are required in the former application, whereas reconstruction of the original music score by removing fluctuations of tempo or pitch is required in the latter application. In both applications, precisions of converted MIDI data from audio signals and time-frequency analysis play an important role. A set of musical information is expressed as frequency components and temporal transitions of sounds. However, it is difficult to analyze precisely both frequency components and temporal transitions at the same time, owing to the known uncertainty principle, which makes difficult to develop a practical audio-MIDI conversion tool. In this paper, we propose an improved generalized harmonic analysis method, which can increase both frequency and temporal resolution in a time-frequency analysis. And we report to apply our improved analysis method, to our developing audio-MIDI conversion tool.

Keywords

MIDI-Karaoke, automatic music notation, audio-MIDI conversion, time-frequency analysis, uncertainty principle, generalized harmonic analysis

１．はじめに

MIDI カラオケ、着信メロディーや BGM 再生機など向けに MIDI 形式の演奏データを作成する用途や、録音楽曲より耳コピーを行って譜面を起こす、耳コピー支援や自動採譜の用途において、オーディオ信号を MIDI 形式に変換するツールの実現が要望される [1)]。前者においては、録音楽曲より MIDI のベロシティ（演奏音の強弱）やピッチベンド（ビブラートなど半音未満の微小な音高の揺れ）に対応する表情パラメータ [2)] を認識して生演奏に近い演奏表現が求められる。これに対し、後者においては、録音楽曲に含まれる倍音や、テンポ、強弱、ピッチ等の演奏表現上のゆらぎを除去して元の譜面に近い音符の表現が求められる。前者に対しては、「オート符」[2)-4)] が、後者に対しては、「採譜の達人」[5)] や「WaveTone」[6)] などオーディオ信号を MIDI 形式および五線譜に変換するフリーウェアのツールが幾つか開発されているが、いずれも業務に対応できる性能には至っていない。

その主な理由として、以下に述べる量子力学の不確定性原理として知られる音響解析上の物理学的な限界があるためである。前述の２つの用途のいずれの場合においても、与えられたオーディオ信号を MIDI 形式に変換する信号処理系の精度が要求されるが、その精度を左右するのが時間周波数解析である。音楽情報は時間軸における音の周波数成分の時系列変化で表現されるため、時間周波数解析により音の周波数成分と時系列変化の双方を高精度に解析する必要がある。ところが、周波数を高精度に解析しようとすると時間分解能を犠牲にする必要があり、時系列変化を高精度に解析しようとすると周波数分解能を犠牲にする必要があった。即ち、時間と周波数の解析精度はトレードオフの関係になり、時間と周波数に関する不確定性原理とよばれる。これは、量子力学における運動量と位置に関するハイゼンベルクの不確定性原理に基づいている。

時間周波数解析の代表的な手法として、短時間フーリエ変換法 [7)] がある。与えられた音響信号に対して、有限長の解析窓で切り取り、ハニング窓で重み付けを行いながら局所的にフーリエ変換を適用するものである。解析窓を時間軸方向にシフトさせれば、時系列なスペクトル（スペクトログラム）を得ることができる。この時､解析窓を大きくすると、周波数の解析精度は向上するが、時間分解能は低下する。逆に解析窓を小さくすると、時間分解能は向上するが､周波数分解能は低下する。この問題を解決する手法として､ウェーブレット変換法 [8)] が提案された。これは、周波数が高い領域では解析窓を小さくし、周波数が低い領域では解析窓を大きくする方法である。この方法を楽音に適用すると、和音や倍音の低音部と高音部で時間分解能が異なり、後述する時間軸方向に解析音素を連結させて音符を認識する際に支障をきたすため、本稿主題のオーディオ -MIDI 変換ツールには向いておらず、前述のフリーウェア [2)-6)] でも採用されていない。

また、短時間フーリエ変換法 [7)] およびウェーブレット変換法 [8)] のいずれの方法においても、解析対象の原音の周波数と周波数解析に使用する調和関数や基底関数の周波数とのミスマッチにより解析スペクトルに疑似ピークが発生し、特に楽音解析において和音や倍音を正確に抽出できない。そこで、前述の「オート符」[2)-4)] では、一般化調和解析 (GHA, Generalized Harmonic Analysis)[9)] が採用されている。これは、短時間フーリエ変換法 [7)] を基本にして、原音信号よりピーク周波数をもつ調和関数成分との差分を算出しながら、短時間フーリエ変換を繰り返し実行して、解析スペクトルの各強度を決定する方法である。本手法をオーディオ -MIDI 変換ツールに適用し､原音としてボーカル信号を与えると､フォ

ルマント成分を MIDI 形式に和音近似させることができ、一般的な MIDI 音源を用いてボーカルをある程度再現できる精度が得られる [10)11)]。しかし、この方法には、調和関数成分との差分を算出する過程で、疑似信号成分が重畳するという問題があった。

本稿では、一般化調和解析 [9)] を基本にして、短時間フーリエ変換における周波数分解能と時間分解能の双方を改善させ、調和関数成分との差分を算出する過程で重畳する疑似信号成分を抑圧する高精度な時間周波数解析法を提案する。併せて、本提案の時間周波数解析法を活用して、オーディオ -MIDI 変換ツールを再設計したので、開発したツールの評価結果についても報告する。

2．既提案の時間周波数解析法の見直しと改良手法の提案

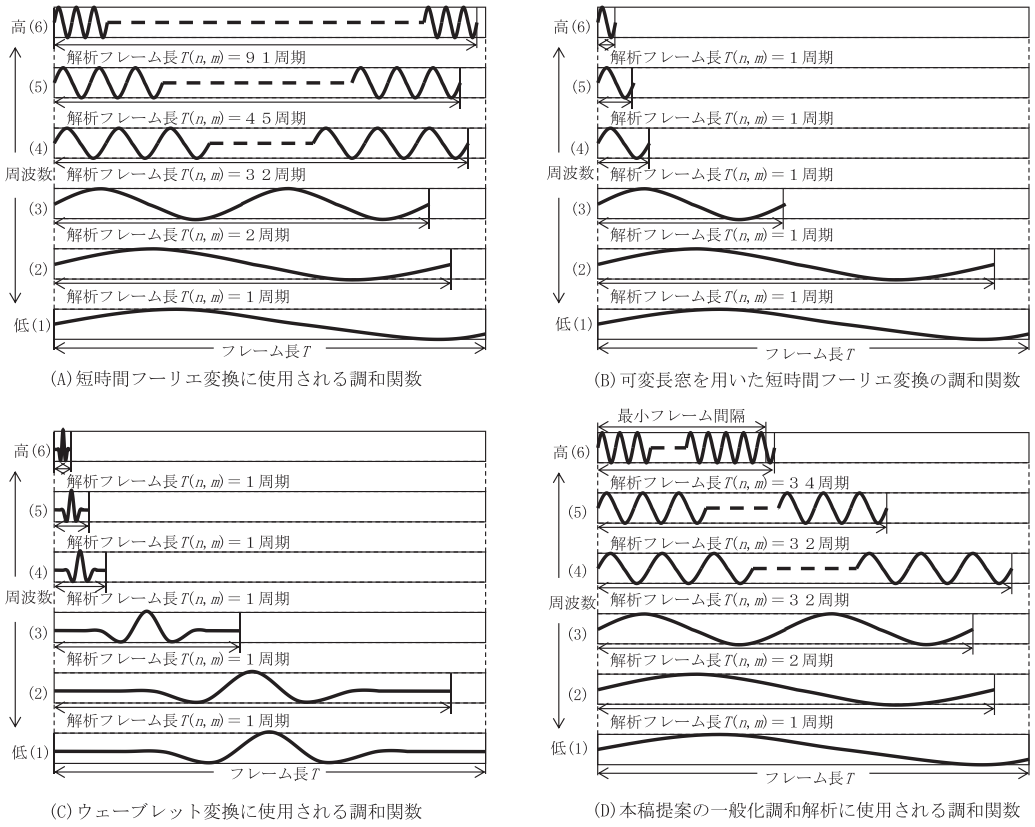


図 1　各種時間周波数解析に使用される調和関数と本稿提案の調和関数

2．1．　既提案の時間周波数解析法に対する時間および周波数解析精度の改善

図 1 -(A) は既提案の短時間フーリエ解析法で使用する調和関数を示す。調和関数としては正弦波および余弦波を使用し、周波数としては基本的に MIDI の半音階のノートナンバー n（$0 \leq n \leq 127$）に対応する 128 種の周波数 f（n）を以下のように与える。

$$f(n) = 440 \cdot 2^{(n-69)/12} \quad [Hz] \qquad (1)$$

ただし、周波数の分解能として半音間隔では不十分で、更に文献 [2)] にあるような微分音・ピッチベンドのパラメータを解析するには最小間隔でセント（1/100 半音）レベルの解

析が求められる。といっても、12800 種の周波数の調和関数を使用して短時間フーリエ解析を行うのは現実的ではない。(1) 式により、ノートナンバー n の周波数 $f(n)$ はノートナンバーが大きくなるほど指数関数的に大きくなるので、ノートナンバー間の周波数の間隔もノートナンバーが大きくなるほど大きくなる。そこで、表 1 に示すように、ノートナンバー n が大きくなるにつれ、隣接ノートナンバー間（半音）で解析周波数の間隔があまり広がらないように、大き目の値をもつ $M(n)$ 個の周波数に分割して調和関数を準備して解析を行うようにする。即ち、分割した周波数をもつ微分音を m として、ノートナンバー n（$0 \leq n \leq 127$）では、以下式で示される $M(n)$ 種の周波数 $f(n,m)$ をもつ調和関数を用いて解析を行う。

表 1　ノートナンバーに対する微分音分解能の設定表

ノートナンバー n	0	・	62												
微分音分解能 $M(n)$	1	・	1												
ノートナンバー n	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77
微分音分解能 $M(n)$	3	3	3	3	3	3	3	3	5	5	5	5	5	5	7
ノートナンバー n	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92
微分音分解能 $M(n)$	7	7	7	9	9	9	9	11	11	11	13	13	15	15	17
ノートナンバー n	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107
微分音分解能 $M(n)$	17	19	19	21	23	25	27	29	31	33	35	37	39	41	43
ノートナンバー n	108	109	・	127											
微分音分解能 $M(n)$	45	45	・	45											

$$f(n,m) = 440 \cdot 2^{(n-69+\frac{m}{M(n)})/12} \quad [Hz] \qquad (2)$$

短時間フーリエ解析法では、解析窓のフレーム長を T とすると、与えられたサンプリング周波数 Fs [Hz] の音響信号より区間 T サンプルだけ切り出して各調和関数と相関計算を行う。その際、相関計算を行う範囲である解析フレーム長 $T(n,m)$ [単位：サンプル] は、$T(n,m) \leq T$ の条件で、以下のように周期 $Fs/f(n,m)$ の整数倍で最大になる Cy 周期分に設定する。

$$T(n,m) = MAX\left[\frac{Cy \cdot Fs}{f(n,m)}\right] \qquad (3)$$

図 1 -(A) の例では、いずれの周波数においても、解析フレーム長 $T(n,m)$ は T に近い値となり、周波数分解能は周波数とともに向上するが、時間分解能は固定である。しかし、最下位 (1) の周波数ではフレーム長 T に 1 周期分の調和関数を収納できていないため、この周波数では精度の良い解析は行えない。フレーム長 T を大きく設定すれば、最下位 (1) の周波数を含め周波数解析精度は向上するが、時間分解能が全体的に低下してしまう。逆に、フレーム長 T を小さく設定すれば、時間分解能は向上するが、最下位 (1) の周波数を含め低い周波数では精度の良い解析は行えなくなる。即ち、時間と周波数の解析精度はトレードオフの関係になり、時間と周波数に関する不確定性原理とよばれる。

これに対して、図 1 -(B) のように、Cy=1 に固定して解析フレーム長 $T(n,m)$ が 1 周期分

Fs/f(*n*,*m*) となる可変長窓にする方法が提案されている [7]。この方法では、周波数が高くなるにつれ時間分解能が向上するが、1 周期分の正弦波や余弦波を用いた相関計算では高域において周波数解析精度が低下してしまう。そこで、図 1 -(C) に示すウェーブレット変換 [8] が提案された。調和関数の代わりに基底関数とよばれる解析対象の信号に適した形状の関数を選択して相関計算を行うことにより、高域における周波数解析精度の低下を防いでいる。しかし、この方法を楽音解析に適用すると、倍音や和音を構成するベース音と高域音の時間分解能が顕著に異なり、一連の音符として認識することが難しくなる。

そこで、本稿では図 1 -(A) の短時間フーリエ変換を基本として、図 1 -(D) のように全域にわたって周波数解析精度を維持しながら、中域から高域における時間分解能を向上させる方法を提案する。図 1 -(D) の周波数 (1) ～ (4) まで、具体的には周期が *Cy*≦32 までの低域周波数では図 1 -(A) と同様に (3) 式に基づいて解析フレーム長 *T*(*n*,*m*) を設定する。*Cy*>32 となる中域周波数では、*Cy*=32 に固定して解析フレーム長 *T*(*n*,*m*) を 32 周期分 32･*Fs*/ *f*(*n*,*m*) とする可変長窓に設定する。これにより、周波数解析精度をあまり低下させずに時間分解能を向上させることができる。ただし、高域周波数では *Cy*=32 のままでは周波数解析精度が低下してしまう。そのため、後述する周波数解析を行う際の最小フレーム間隔を *W* として、解析フレーム長 *T*(*n*,*m*) が W 未満の場合、*T*(*n*,*m*)≧*W* の条件で、以下のように周期 *Fs*/*f*(*n*,*m*) の整数倍で最小になる *Cy* 周期分に設定する。

$$T(n,m) = MIN\left[\frac{Cy \cdot Fs}{f(n,m)}\right] \qquad (4)$$

2．2．既提案の一般化調和解析法における疑似信号成分の抑圧

はじめに、一般化調和解析法 [9] の特徴について述べる。図 2 -(A) は、解析対象の音響信号のスペクトルを表し、周波数解析に使用する調和関数の周波数 f_1 と f_2 の中間の単一周波数をもつ正弦波とする。これに対して、短時間フーリエ変換を実行すると、図 2 -(B) のように調和関数の周波数 f_1 と f_2 の双方に顕著なピークが生じ、楽音信号の場合、周波数 f_1 と f_2 の 2 つの音高の和音と誤認識されやすい。そこで、図 2 -(B) のスペクトルにおいて最大ピークをもつ周波数 f_1 の調和関数を用いて原音信号から差分をとる処理を行う（図 2 -(C)）。この時、解析フレームと周波数 f_1 の調和関数との相関値で調和関数に重み付けして差分演算を行うと、差分信号のスペクトルは図 2 -(D) のようになる。この差分信号に対して再度短時間フーリエ変換を実行すると、図 2 -(E) のように調和関数の周波数 f_1 の成分は殆ど消失し、周波数 f_2 の成分も顕著に減衰する。

この段階で更に、図 2 -(E) において最大ピークをもつ周波数 f_2 の成分を削除した差分信号に対して、短時間フーリエ変換を再度実行して、3 番目以降のピークの周波数の成分を高精度に算出することができる。このように短時間フーリエ変換と差分演算を繰り返しながら周波数成分を順次算出してゆく方法が、一般化調和解析である。図 2 -(F) は一般化調和解析により 2 つの周波数成分を高精度に算出した結果で、図 2 -(B) に示す短時間フーリエ変換を実行して、最大ピークの周波数 f_1 の成分を算出し、周波数 f_1 の成分を削除した差分信号に対して、図 2 -(E) に示す短時間フーリエ変換を再度実行して、2 番目のピークの周波数 f_2 の成分を算出したものである。本方法により、原音信号のスペクトル図 2 -(A)

に近い解析結果が得られ、楽音信号の倍音や和音を高精度に認識できる。

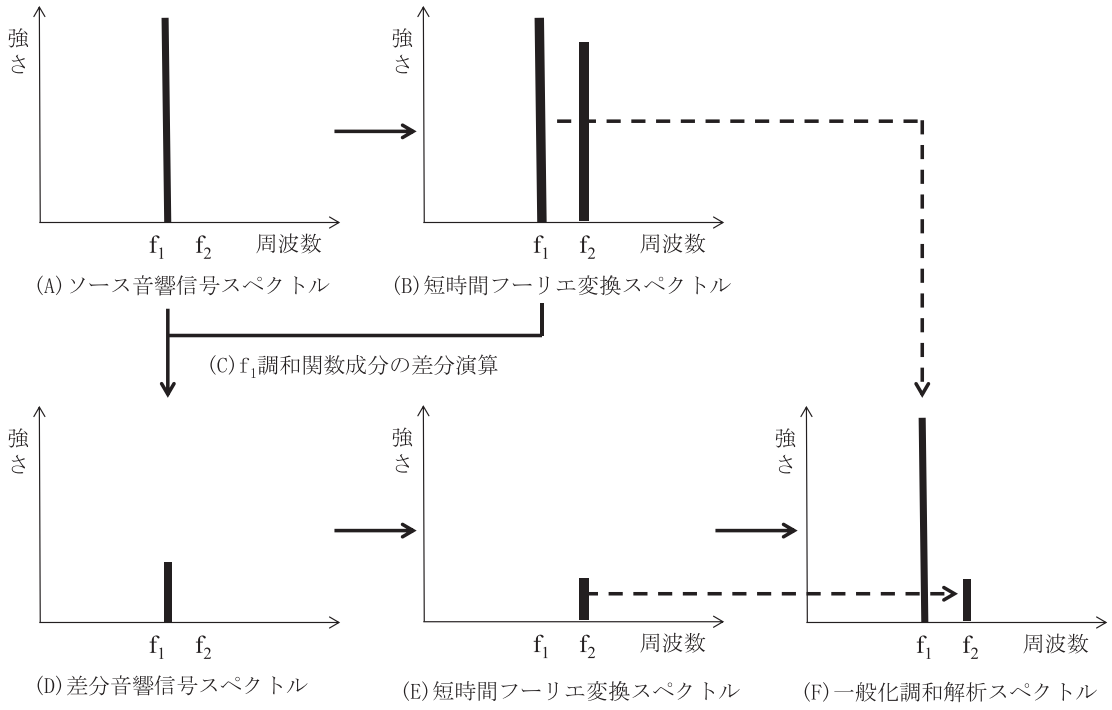


図２　周波数解析法：短時間フーリエ変換と一般化調和解析

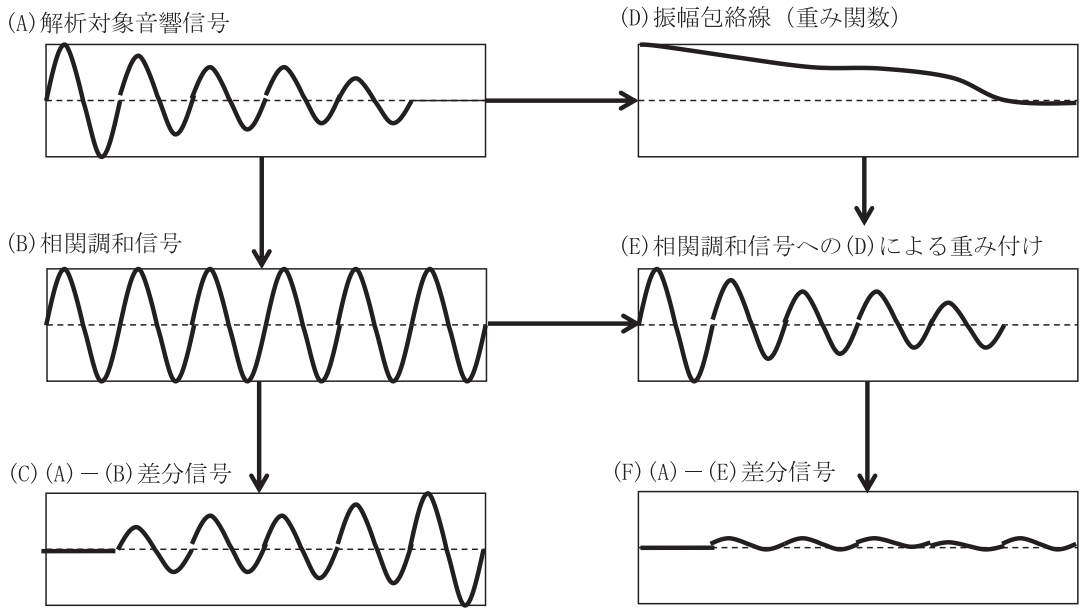


図３　解析フレームより相関成分差分計算時の重みづけ

一般化調和解析では原音信号より調和関数を差分演算する過程で、逆に調和関数の成分が疑似的に重畳され、疑似ピークが生じることがあり、この一例を図３-(A)(B)(C) に示す。図３-(A) の解析対象の音響信号のように解析フレーム内で振幅の変動が大きい場合、図３

-(B) に示す相関が最も高い調和関数と差分をとると、図３-(C) に示すように解析フレーム内に差分をとった調和関数成分が残留してしまう。この差分信号に対して、図３-(B) に近い周波数をもつ調和関数と相関計算を行うと、疑似ピークが生じて精度の良い解析が行えなくなる。

そこで、図３-(A) の解析対象の音響信号に対して振幅包絡線を算出し、解析フレーム内における振幅分布を重み関数として求める。その結果が図３-(D) である。この重み関数を算出する方法としては、図３-(A) の解析対象の音響信号に対して、所定の算出区間（例えば、最小フレーム間隔 W）ごとに最大の振幅絶対値を順次求め、算出区間の間を各区間の最大の振幅絶対値で線形補間することにより平滑化する方法が簡便である。このようにして算出した図３-(D) の重み関数を、図３-(A) の解析対象の音響信号から、図３-(B) に示す相関が最も高い調和関数と差分を算出する際に、調和関数に乗じる。これにより、図３-(F) に示すように、図３-(A) に示す解析フレーム内の音響信号より図３-(E) に示す調和関数成分のみを綺麗に削除することができ、疑似ピークの発生を抑圧できる。

３．提案する時間周波数解析法を用いたオーディオ -MIDI 変換ツールの概要

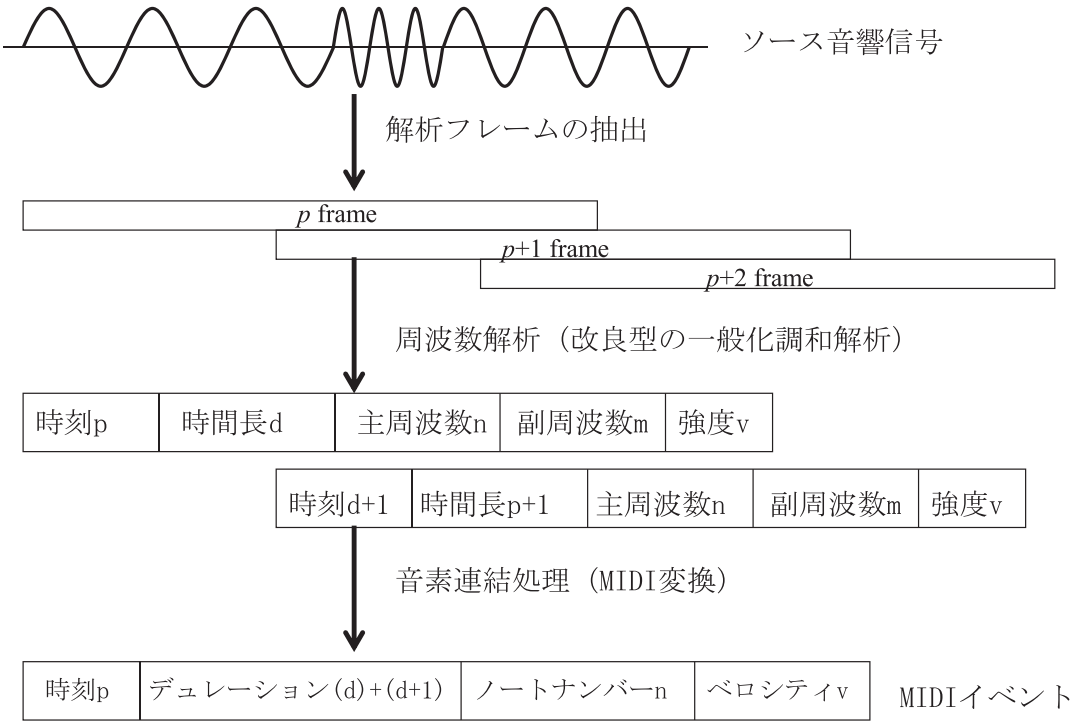


図４　提案するオーディオ -MIDI 変換ツールの処理構成

図４は文献 [10] で提案されているオーディオ -MIDI 変換処理の主要構成を示す。はじめに、与えられたソース音響信号より周波数解析対象の解析フレームを抽出するが、後続する解析フレームとのフレーム間隔はソース音響信号の周波数変動を大まかに検出しながら適応的に可変設定するようにしている。即ち、周波数成分の変化が大きい箇所ではフレーム間隔を狭くし、周波数成分の変化が小さい箇所ではフレーム間隔を大きくする。続いて、前章で述べた改良した一般化調和解析に基づいて周波数解析を行う。ノートナンバーに対

応する周波数ごとに隣接するノートナンバーとの半音間を、表 1 に示した微分音分解能に基づいて微分音（副周波数）に分割して解析を行うようにする。最後に、時間的に隣接する近傍の主周波数をもつ解析成分（解析音素）を連結し音符としてまとめ、MIDI イベント形式で符号化する。図 4 の例では、単一の MIDI イベントで記載しているが、実際にはデュレーション情報の代わりにデュレーションの間隔だけ時刻がずれた 2 つのノートオンとノートオフのイベントで符号化される。また、微分音解析の結果を基に、ノートオンとノートオフのイベントの間にピッチベンドやエクスプレッションなどの表情制御コードを符号化して挿入することもできる。

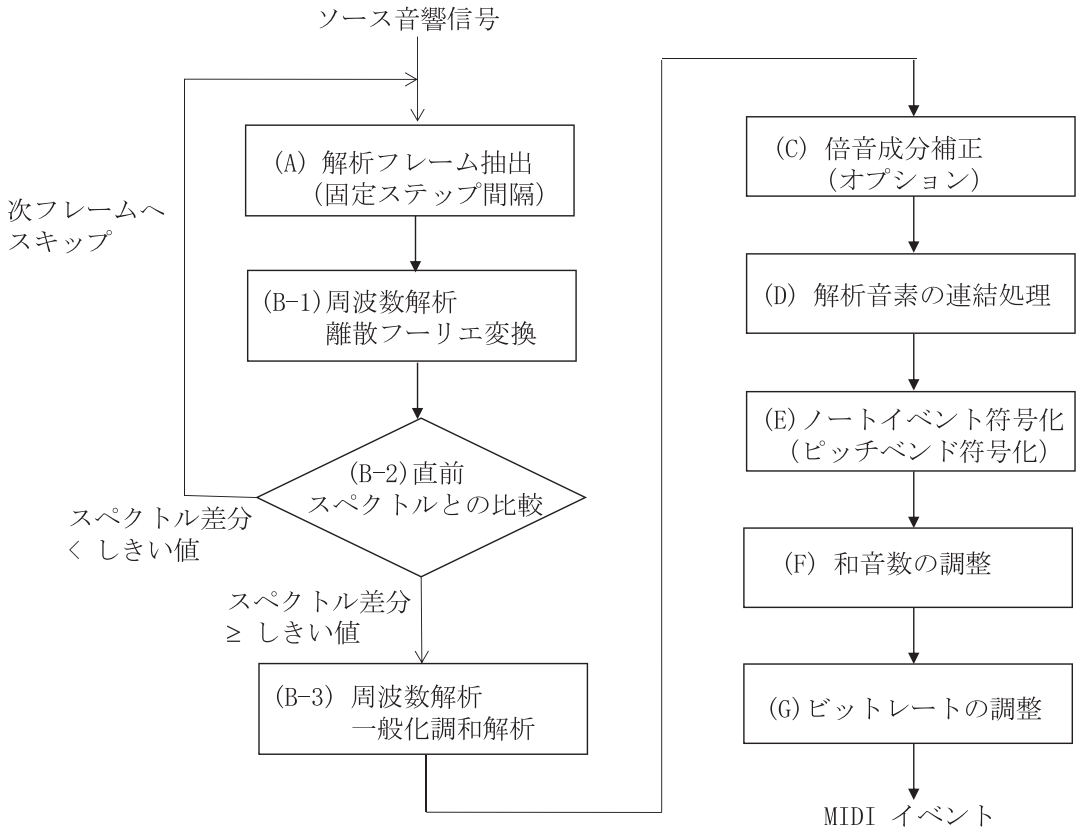


図 5　提案するオーディオ -MIDI 変換ツールの具体的な処理構成

図 5 に、文献 [11] で提案されているオーディオ -MIDI 変換処理の具体的な処理構成を示す。本稿では、処理 (B-1) から処理 (B-3) に示される可変長フレーム間隔の時間周波数解析の処理構成を提案し、処理 (B-3) に前章で提案した一般化調和解析に基づいた周波数解析手法を組み込んでいる。更に、処理 (C) においてオプションで高精度な倍音除去を実現し、処理 (E) のノートイベントの符号化処理においてオプションでピッチベンド符号化を実現できるようにしている。以下、処理 (A) から処理 (G) の各々に対して、7 つの節に分けて説明する。

3.1.　解析フレーム抽出（固定ステップ間隔）(A)

時間周波数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 T とフレーム間隔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 *Fs* が 44.1[kHz] の場合、ピアノ鍵盤の最低音（ノートナンバー :21）まで忠実に解析するためにはフレーム長 *T* として 4096 サンプル以上必要である。解析時の解析フレーム長 *T*(*n*,*m*) はフレーム長 *T* の範囲内で解析周波数ごとに可変に設定するが、フレーム長 *T* は上限値として、例えば 4096 を与える。

一方フレーム間隔は、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（ただし、併せてフレーム長も小さくしないと顕著な時間分解能の向上効果はない）。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間隔を細かくすると、後述する解析音素の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解析音素の連結処理のためにも、フレーム間隔は解析対象フレームごとに変化させ、適応的に設定する方法が望ましい。本稿では文献 [11] の提案に基づき、周波数解析時において一般化調和解析による高精度な周波数解析を行う前に、フレーム間隔の最小値である固定値の最小フレーム間隔 *W*（例えば、*T*=4096 の場合 *W*=16 サンプル）で離散フーリエ変換を行い、周波数変化が顕著な箇所を、高精度な周波数解析を行う箇所として探索する方法をとる。

3．2．　周波数解析・一般化調和解析 (B)

3．2．1．　周波数解析・離散フーリエ変換 (B-1)

前節で述べた方法により、サンプリング周波数 Fs 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≦*i*≦*T* − 1) とする。本周波数解析は、ノートナンバー *n* (0≦*n*≦127) に対して、表 1 に基づいて *M*(*n*) 個の微分音 *m* を定義し、(2) 式に基づく 128 × *M*(*n*) 種の解析周波数 *f*(*n*,*m*) の調和関数を用いて短時間離散フーリエ変換により行う。この微分音を用いた解析は周波数解析精度を向上させることが主目的であるが、後述するオプション処理により、この半音未満の精度で微分音解析された結果を、ピッチベンドなどの表情制御コードへの符号化に使用することもできる。

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*) (0≦*n*≦127) と副周波数配列 S_o(*p*,*n*) および *S*(*p*,*n*) を定義し、0≦*n*≦127 および 0≦*m*≦*M*(*n*) − 1 に対して以下式で相関計算を行う。副周波数配列 S_o(*p*,*n*) および *S*(*p*,*n*) には、相関のあった微分音 *m* の値と、*m* を固定値 M_o（例えば、M_o=25）の微分音分解能に換算した値 *m'*（*m'*=*m* ・ M_o/*M*(*n*) , 0≦*m'*≦M_o − 1）を各々収納する。式 (5) において *T*(*n*,*m*) は解析フレーム長で、前章で述べた通りフレーム長 *T* を超えない範囲で周波数ごとに可変長に設定する。

$$A(p,n,m)=\frac{1}{T(n,m)}\sum_{i=0}^{T(n,m)-1}\left(x(p,i)\cdot\sin\left(\frac{2\pi f(n,m)\cdot(i+pW)}{F_s}\right)\right)$$

$$B(p,n,m)=\frac{1}{T(n,m)}\sum_{i=0}^{T(n,m)-1}\left(x(p,i)\cdot\cos\left(\frac{2\pi f(n,m)\cdot(i+pW)}{F_s}\right)\right)$$

$$E(p,n,m)=A(p,n,m)^2+B(p,n,m)^2 \qquad (5)$$

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、(5) 式の $A(p,n,m)$ と $B(p,n,m)$ を算出する式は (5') 式のように変形でき、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。

$$A(p,n,m) = A(p-1,n,m) - \frac{1}{T(n,m)} \sum_{i=0}^{W-1} \left(x(p-1,i) \cdot \sin\left(\frac{2\pi f(n,m) \cdot (i + (p-1)W)}{F_s} \right) \right)$$

$$+ \frac{1}{T(n,m)} \sum_{i=T(n,m)-W}^{T(n,m)-1} \left(x(p,i) \cdot \sin\left(\frac{2\pi f(n,m) \cdot (i + pW)}{F_s} \right) \right)$$

$$B(p,n,m) = B(p-1,n,m) - \frac{1}{T(n,m)} \sum_{i=0}^{W-1} \left(x(p-1,i) \cdot \cos\left(\frac{2\pi f(n,m) \cdot (i + (p-1)W)}{F_s} \right) \right)$$

$$+ \frac{1}{T(n,m)} \sum_{i=T(n,m)-W}^{T(n,m)-1} \left(x(p,i) \cdot \cos\left(\frac{2\pi f(n,m) \cdot (i + pW)}{F_s} \right) \right)$$

$$E(p,n,m) = A(p,n,m)^2 + B(p,n,m)^2 \qquad (5')$$

続いて、ノートナンバー n ごとに、$0 \le m \le M(n) - 1$ の範囲で相関配列 $E(p,n,m)$ を最大にする $E(p,n,m_{max})$ を求め、$E(p,n)=E(p,n,m_{max})$、$S_o(p,n)=m_{max}$、$S(p,n)=m_{max} \cdot M_o/M(n)$ と定義する。

3.2.2. 直前スペクトルとの比較 (B-2)

（5）式または（5'）式でノートナンバー n ごとに算出された相関配列 $E(p,n)$ と直前解析フレームにおける $E(p-1,n)$ との差分割合の平均値 $dE(p-1,p)$ 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 0.15、ピッチベンド符号化を行う場合は 0.2）未満であれば、**3.1** 節に戻り次の解析フレーム抽出に進み、所定のしきい値以上であれば、次の **3.2.3** 節の周波数解析・一般化調和解析へ進む。

$$dE(p-1,p) = \frac{1}{N} \sum_{n=0}^{N-1} \left\{ \frac{|E(p,n) - E(p-1,n)|}{E(p,n) + E(p-1,n)} \right\} \qquad (6)$$

3.2.3. 周波数解析・一般化調和解析 (B-3)

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強度値 $E_o(q,n)$ $(0 \le n \le 127)$ を定義し、初期値を全て－ 1 とする。

（a）ノートナンバー n に対して $E_o(q,n)<0$ でかつ $E(p,n)$ が最大になる $E(p,n_{max})$ を求め、$m_{max}=S_o(p,n_{max})$ とする。ただし、$p=P(q)$ とする。式 (5) を簡素化した以下式 (7) 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。

$$A(p,n_{max},m_{max})=\frac{1}{T(n_{max},m_{max})}\sum_{i=0}^{T(n_{max},m_{max})-1}\left(x(p,i)\cdot\sin\left(\frac{2\pi f(n_{max},m_{max})\cdot i}{F_s}\right)\right)$$

$$B(p,n_{max},m_{max})=\frac{1}{T(n_{max},m_{max})}\sum_{i=0}^{T(n_{max},m_{max})-1}\left(x(p,i)\cdot\cos\left(\frac{2\pi f(n_{max},m_{max})\cdot i}{F_s}\right)\right)$$

$$E_o(q,n_{max})=E(p,n_{max})=A(p,n_{max},m_{max})^2+B(p,n_{max},m_{max})^2 \qquad (7)$$

（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。このとき、p 番目に抽出された解析フレーム $x(p,i)$ $(0\leq i\leq T\text{-}1)$ に対して、図３-(D) のような振幅包絡線を算出し、解析フレーム内における振幅分布を重み関数 $w(p,i)$（$0\leq w(p,i)\leq 1$）として求め、調和関数に重み付けをする。

$$x(p,i)=x(p,i)-A(p,n_{max},m_{max})\cdot w(p,i)\cdot\sin\left(\frac{2\pi f(n_{max},m_{max})\cdot i}{F_s}\right)$$
$$-B(p,n_{max},m_{max})\cdot w(p,i)\cdot\cos\left(\frac{2\pi f(n_{max},m_{max})\cdot i}{F_s}\right) \qquad (8)$$

(8) 式に基づいてサンプル配列 $x(p,i)$ を更新後、再度（ａ）の処理に戻り、$0\leq n\leq 127$ の全ての強度値 $E_o(q,n)$ の値が０以上の値に決定されるまで（ａ）と（ｂ）の処理を繰り返す。

３．３．　倍音成分補正 (C)

上記算出された $0\leq n\leq 127$ の全ての強度値 $E_o(q,n)$ に対して、2, 3, 4, 5, 6, 7, 8, 9, 10 倍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 $N_o(b)$ $(b=0,\cdots,8)$ 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 $N_o(b)$ の具体例は、$N_o(b)$={ 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 $E'_o(q,n)$ を次式の通り補正する。n=0 から n=127 の順に補正処理を行い、(9) 式の平方根内で参照する、下方にシフトさせたノートナンバー $n-N_o(b)$ の強度値 $E'_o(q,n-N_o(b))$ は、(9) 式で補正した強度値を使用する。また、倍音の次数 b の値に反比例して補正割合を減衰させる。

$$E'_o(q,n)=E_o(q,n)-\sum_{b=0}^{9}\frac{2\gamma}{(b+2)}\sqrt{E_o(q,n)E'_o(q,n-N_o(b))} \qquad (9)$$

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 γ >0.2 に設定し、ボーカルを含む音響信号で、フォルマント成分を倍音として残しておく必要がある場合は γ =0 に設定する。下方にシフトさせたノートナンバー $n-N_o(b)$ が負値または $E'_o(q,n-N_o(b))$ の値が存在しない場合、$E'_o(q,n-N_o(b))$=0 として計算し、補正後の $E'_o(q,n)$ が $E'_o(q,n)$<0 の場合、$E'_o(q,n)$=0 とする。補正された相関配列 $E'_o(q,n)$ を $E_o(q,n)$ として次ステッ

プの解析音素の連結処理 (D) 以降は補正後の値を適用する。

3．4．　解析音素の連結処理 (D)

q 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 の解析音素の成分を［時刻 $Time(q)$，時間長 $Length(q)$，主周波数 n，副周波数 $S(P(q),n)$，強度 $E_o(q,n)$］とし、前方に位置する解析音素との連結可能性パラメータ $Conn$（初期値、$Conn=0$）を設定する。はじめに、直前に隣接する $q-1$ 番目の可変解析フレームが存在する場合、$q-1$ 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 の解析音素の成分を［時刻 $Time(q-1)$, 時間長 $Length(q-1)$, 主周波数 n, 副周波数 $S(P(q-1),n)$, 強度 $E_o(q-1,n)$］とする。$q-1$ 番目の解析音素が存在しない場合、$Conn=0$ とする。時刻 $Time(q)$ および $Time(q-1)$ 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 Fs で除算することで得られる。時間長 $Length(q)$ は $\{Time(q+1)-Time(q)\}\cdot\delta$ で、時間長 $Length(q-1)$ は $\{Time(q)-Time(q-1)\}\cdot\delta$ で与えられる。隣接する可変解析フレームの時刻の差をそのまま時間長に設定すると音の切れが悪くなるため、1 より小さい係数 δ（例えば、$\delta=0.77$）を乗算する。また、$q+1$ 番目の可変解析フレームが存在しない場合、時間長 $Length(q)$ は $T\cdot\delta$ とする。

互いに時間的に隣接する $q-1$ 番目および q 番目の解析フレームの 2 つ解析音素に対して、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、$q-1$ 番目と q 番目の解析フレーム間の周波数の差が所定値 N_{dif} 未満で、双方の強度が各々所定のしきい値 L_{min} 以上でかつ双方の強度の差が所定値 L_{dif} 以下で両者の連続性が認められる場合、即ち、以下 (10-1) 〜 (10-3) の 3 条件のいずれかを満たす場合、連結可能性パラメータ $Conn$ に正の値を設定する。

$|S(P(q),n)-S(P(q-1),n)|<N_{dif}$ かつ $E_o(q-1,n)\geq L_{min}$ かつ $E_o(q,n)\geq L_{min}$
かつ $E_o(q,n)-E_o(q-1,n)\leq L_{dif}$ を満たす場合、$Conn=1$　　(10-1)

$|S(P(q),n-1)-S(P(q-1),n)|<N_{dif}$ かつ $E_o(q-1,n)\geq L_{min}$ かつ $E_o(q,n-1)\geq L_{min}$
かつ $E_o(q,n-1)-E_o(q-1,n)\leq L_{dif}$ を満たす場合、$Conn=2$　　(10-2)

$|S(P(q),n+1)-S(P(q-1),n)|<N_{dif}$ かつ $E_o(q-1,n)\geq L_{min}$ かつ $E_o(q,n+1)\geq L_{min}$
かつ $E_o(q,n+1)-E_o(q-1,n)\leq L_{dif}$ を満たす場合、$Conn=3$　　(10-3)

上記連結条件のしきい値の標準的な設定値は、$N_{dif}=6$ [単位：1 半音を M_o とする微分音]、$L_{min}=1$［単位：ベロシティ］、$L_{dif}=10$［単位：ベロシティ］である。

$Conn>0$ の場合、続いて、既に連結処理が進行している先頭の可変解析フレームを q_o 番目とし、これに対して上記 q 番目の解析音素の連結可能性を判断する。q_o 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 の解析音素の成分を［時刻 $Time(q_o)$，時間長 $Length(q_o)$，主周波数 n，副周波数 $S(P(q_o),n)$，強度 $E_o(q_o,n)$］とする。q_o 番目の解析音素と q 番目の解析音素との時間的なギャップ $Time(q)-(Time(q_o)+Length(q_o))$ が T_{gap} 未満で、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 番目と q 番目の可変解析フレームとの周波数の差が所定値 N_{adif} 未満で、両者の連続性が認められる場合、即ち、以下 (11-1) 〜 (11-3) の 3 条件のいずれかを満たす場合、q 番目の解析音素を q_o 番目の解析音素に連結する処理を実行する。即ち、q_o 番目の解析音素の

時間長 $Length(q_o)$ を $Time(q)+Length(q) - Time(q_o)$ に更新し、q 番目の解析音素を削除する。上記連結条件のしきい値の標準的な設定値は、T_{gap}=1/3 [second]、N_{adif}=8 [単位：1 半音を M_o とする微分音] である。

$Conn$=1 かつ $|S(P(q_o),n) - S(P(q),n)|<N_{adif}$ (11-1)
$Conn$=2 かつ $|S(P(q_o),n) - S(P(q),n-1)|<N_{adif}$ (11-2)
$Conn$=3 かつ $|S(P(q_o),n) - S(P(q),n+1)|<N_{adif}$ (11-3)

連結後の q_o 番目の解析音素の主周波数・副周波数・強度は、
$Conn$=1 かつ $E_o(q,n)>E_o(q_o,n)$ の場合、主周波数 n，副周波数 $S(P(q),n)$，強度 $E_o(q,n)$ に更新し、
$Conn$=2 かつ $E_o(q,n-1)>E_o(q_o,n)$ の場合、主周波数 $n-1$，副周波数 $S(P(q),n-1)$，強度 Eo(q,n − 1) に更新し、
$Conn$=3 かつ $E_o(q,n+1)>E_o(q_o,n)$ の場合、主周波数 $n+1$，副周波数 $S(P(q),n+1)$，強度 $E_o(q,n+1)$ に更新する。

3．5．　ノートイベント符号化（ピッチベンド符号化）(E)

前節で述べた時系列の解析音素の連結処理は、不連続性が認められるまで後続する複数の解析音素に対して繰り返し行い、最終的に統合された［時刻 $Time(q_o)$，時間長 $Length(q_o)$，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、$128・\{E_o(q_o,n)/E_{max}\}^{1/4}$ 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）を整数値で与える必要があり、その時刻の単位は任意に定義でき、例えば、1/1536 [sec] の単位に変換して与える。そして、$Time(q_o)+Length(q_o)$ の時刻で、演奏中のノートナンバー n に対してノートオフ・イベントを発行する。

より表情豊かな MIDI イベントを作成するためには、図 6 に示されるように、前節で行った連結処理を行う前の各解析音素の成分を保存しておき、ピッチベンド・イベント（ノートオン後のピッチを 1/100 半音単位で制御できる）あるいはエクスプレション・イベント（ノートオン後の音量を 128 段階で制御できる）として符号化して、ノートオンおよびノートオフのイベントの間に挿入する。例えば、連結統合された q 番目の解析音素の成分を［時刻 $Time(q)$，時間長 $Length(q)$，主周波数 n，副周波数 $S(P(q),n)$，強度 $E_o(q,n)$］に対して、直前に隣接する $q-1$ 番目の解析音素の成分を［時刻 $Time(q-1)$，時間長 $Length(q-1)$，主周波数 n，副周波数 $S(P(q-1),n)$，強度 $E_o(q-1,n)$］とすると、ピッチベンドの値を $4096・\{S(P(q,n) - S(P(q-1),n)\}/M_o+4096$、エクスプレッションの値を $128・[\{E_o(q,n)/E_{max}\}^{1/4} - \{E_o(q-1,n)/E_{max}\}^{1/4}]+127$ と設定して、ノートオン・イベント発行後のデルタタイム $Time(q) - Time(q-1)$ の時刻にピッチベンド・イベントおよびエクスプレッション・イベントを発行する。この時、ノートオン・イベントとピッチベンド・イベントおよびエクスプレッション・イベントとはチャンネル番号で対応付けを行う。MIDI 規格では最大 16 チャンネルまで使用できるが、第 10 チャンネルは通常はパーカッ

ション系の非音階楽器に割り当てられているため、このチャンネルを除く 15 種類のチャンネルのいずれかを各ノートイベント、ピッチベンド・イベントおよびエクスプレッション・イベントに割り当てる。そのため、ピッチベンド・イベントおよびエクスプレッション・イベントを使用する場合、同時に発音できるノートイベントは 15 和音に制限される。

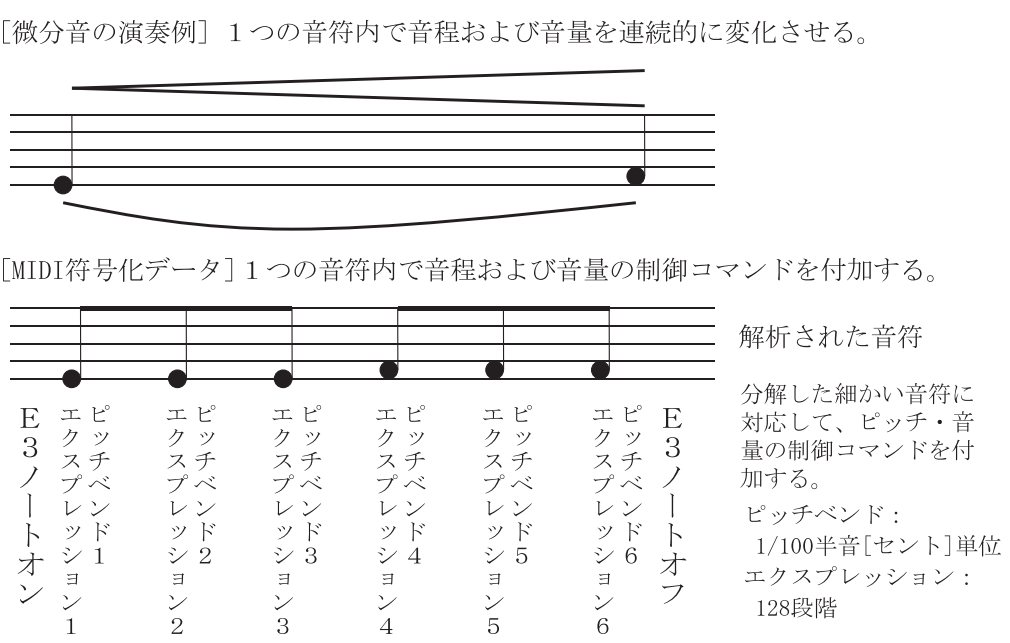


図６　MIDI 規格におけるピッチベンド、エクスプレッションの各イベントの発行方法

３.６.　和音数の調整 (F)

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（前節のピッチベンドを使用している場合は 15 和音）を超えている箇所が見つかった場合は、強制的に優先度の低いノートイベントを削除する処理を行う。基本的には、同時に発音されている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、優先度の低いノートイベントを１対ごと削除する方法をとる。

そうすると、図７-(A) に示されるようにノートイベントが隣接するノートイベントと時間的に重複すると、図７-(B) のように音脈上重要なノートイベントも過剰に削除されてしまう。そこで、本稿では、図７- (C)(D) に示されるように、ノートイベントを分割して部分的に優先度の低いノートイベントの区間を削除する方法を提案する。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 $E_v(i)$ のノートオン時刻を $E_v(i).time$、ベロシティ値を $E_v(i).velocity$ とすると、時刻 t ($>E_v(i).time$) におけるノートイベント $Ev(i)$ の補正ベロシティ値 $V_c(i,t)$ は、

$$V_c(i,t) = E_v(i).velocity \cdot \mathrm{e}^{(t-E_v(i).time)\cdot\tau} \qquad (12)$$

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。）

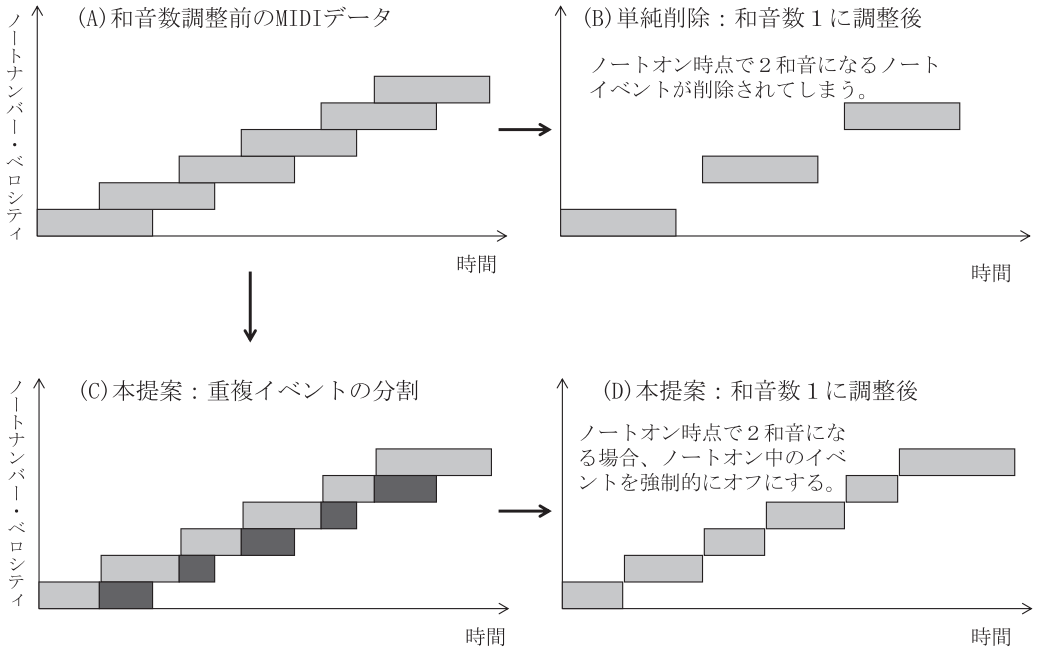


図７　本稿提案のノートイベントの分割による和音数調整機能

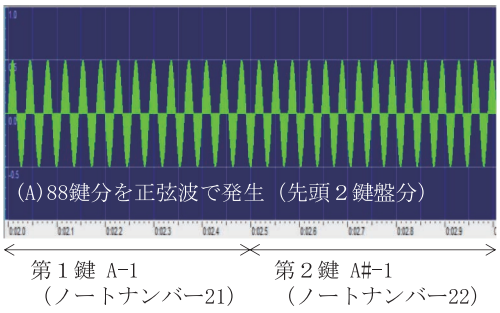
３．７．　ビットレートの調整 (G)

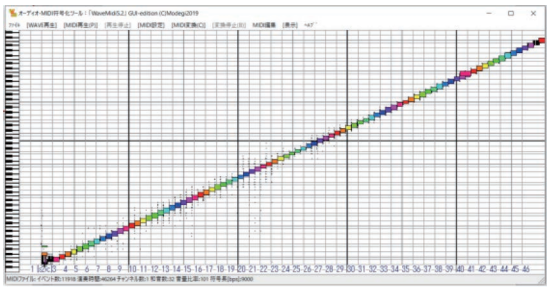
MIDI データ形式に変換する段階で、MIDI 音源で処理可能なビットレートについても考慮する必要がある。時間軸方向に例えば 1 秒間隔にノートオンまたはノートオフのイベントの個数をカウントし、各々の符号長を平均 5 バイト (40bits) とし MIDI 音源で処理可能な最大ビットレートを 9000[bps] とすると（前節のピッチベンドを使用している場合は２倍の 18000[bps] 程度に設定する）、 1 秒間あたりイベント数が 9000/40=225 個を超えている区間が見つかった場合は、その区間に存在するノートオンまたはノートオフのイベントと各々対になるノートオフまたはノートオンの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。

４．おわりに

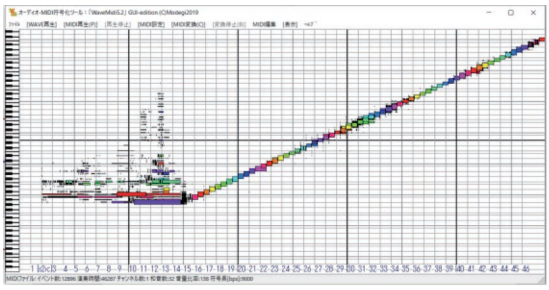
前章までに述べてきた本稿提案の時間周波数解析法を組み込んだオーディオ -MIDI 変換ツールを、32/64bits-WindowsAPI (Win32/64) を用いて 32/64bits-Windows10 デスクトップ・アプリケーションとして C 言語で実装した。

はじめに、半音階の解析精度について評価した結果を図８に示す。図８-(A) に示されているように、ピアノの 88 鍵に対応する半音階を 0.5 秒間隔に正弦波で生成した音響波形（サンプリング周波数 44.1kHz, 量子化 16bits, モノラル）に対し、本稿提案のオーディオ -MIDI 変換ツールにより MIDI 形式に変換した結果を図８-(B)(C)(D) に示す。図８-(B)(C)(D) は変換された MIDI データを独自のピアノロール画面で表示したもので、画面内の着色された小さな矩形が音符（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（音価、デュレーション）を示す。縦軸は音高（ノートナンバー）を示すとともに、縦方向の幅で強さ（ベロシティ）も示している。矩形の色は半音階の階名に基づいて 12 色に色分けしており、オクターブ違いの音は同一の色になる。

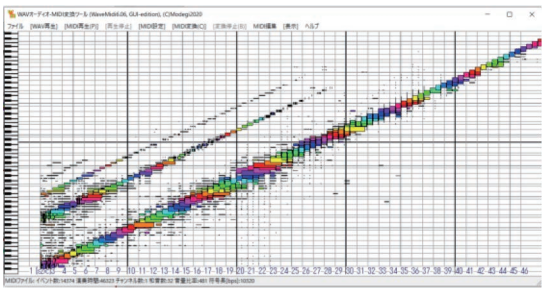




(B)フレーム長4096の固定長窓によるMIDI変換



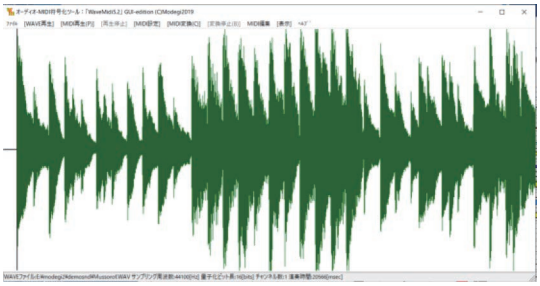
(C)フレーム長1024の固定長窓によるMIDI変換



(D)フレーム長4096以下の可変長窓によるMIDI変換

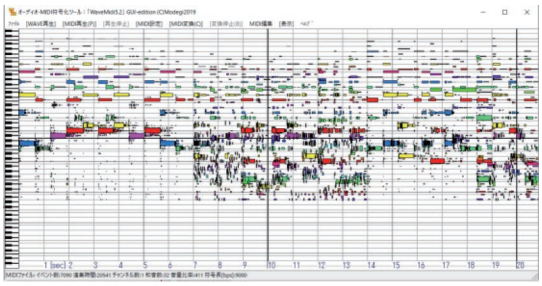
図８　ピアノ８８鍵・半音階スケール（0.5 秒× 88）の MIDI 変換例

図８-(B) は最低音のノートナンバー 21（A-1）を解析できるように、フレーム長を 4096 サンプル固定で解析したものである。概ねの全ての音階で適切な解析精度が得られているが、最低音近辺の解析精度は不正確である。図８-(C) は時間分解能を向上させるように、フレーム長を 1024 サンプルに短縮させて固定フレーム長で解析したものである。低音域の約 22 鍵に対応する音階は、全く解析できていない。図８-(D) は本稿提案の可変長窓を用いたもので、図１-(D) に示すように解析フレーム長を 4096 サンプル以下で可変に設定している。図８-(B) では不正確であった最低音近辺の解析精度が向上し、低音域から中音域にかけて２倍音と３倍音も認識できている。図８-(A) に示すソース音響信号は正弦波を用いて作成しているが、サンプリングおよび 88 音の信号波を連結する過程で歪みが発生していることがわかる。



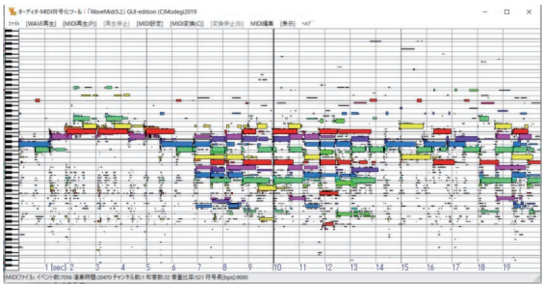
(A) ピアノソロ演奏音の音響波形（冒頭20sec）

(B) フレーム長4096の固定長窓によるMIDI変換

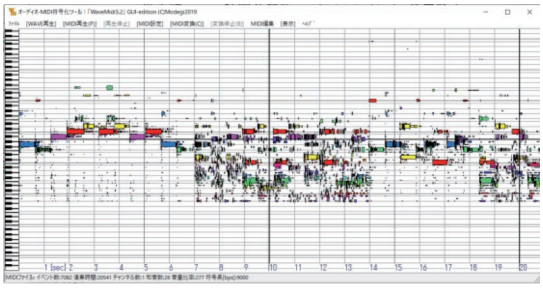


(C) フレーム長1024の固定長窓によるMIDI変換

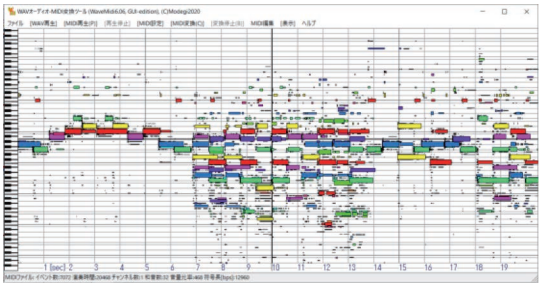
(D) フレーム長4096以下の可変長窓によるMIDI変換



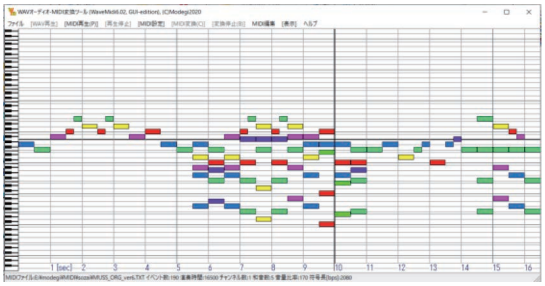
(E) フレーム長4096の固定長窓によるMIDI変換



(F) フレーム長1024の固定長窓によるMIDI変換



(G) フレーム長4096以下の可変長窓によるMIDI変換



(H) 楽曲(A)に対するMIDI打ち込みデータ
（ベロシティ・パラメータ均一）

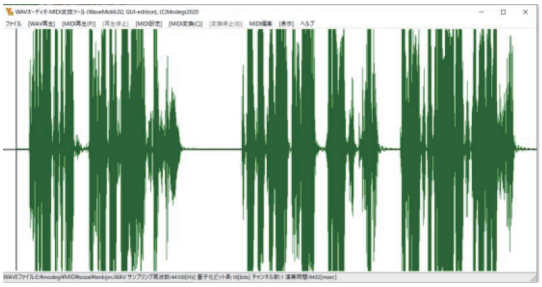
図９　ムソルグスキー「展覧会の絵」ピアノ独奏版の MIDI 変換例

次にピアノ独奏音の採譜精度について評価した結果を図９に示す。図９-(A) はムソルグスキー「展覧会の絵」ピアノ独奏版について、第１プロムナード冒頭 20 秒のピアノ演奏録音の音響波形（サンプリング周波数 44.1kHz, 量子化 16bits, モノラル）である。これに対し、本稿提案のオーディオ -MIDI 変換ツールにより MIDI 形式に変換した結果を図９-(B)(C)(D) に示す。図９-(B) はフレーム長を 4096 サンプル固定で解析したものである。概ねの全ての音が適切に採譜されているが、中低音域から高音域にかけて時間分解能が

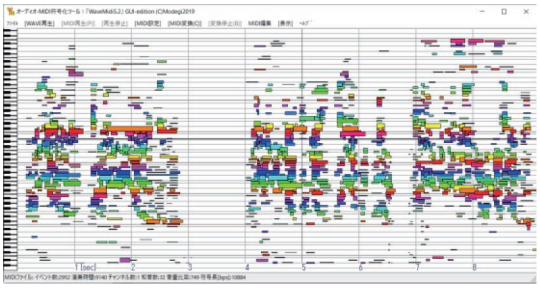
低いため、MIDI 音源による再生音では、発音タイミングや音長がずれているものが目立つ。一方、図９-(C) は時間分解能を向上させるように、フレーム長を 1024 サンプルに短縮させて固定フレーム長で解析したものである。高音域の発音タイミングや音長のずれの問題は解消しているが、低音域から中音域の音が全く採譜できていない。これに対して、図９-(D) は本稿提案の可変長窓を用いたもので、図１-(D) に示すように解析フレーム長を 4096 サンプル以下で可変に設定している。概ねの全ての音が適切に採譜されており、低音域から高音域にかけての発音タイミングや音長のずれの問題も改善されている。

図９-(B)(C)(D) に対して、**３．３**節で述べた倍音成分補正（補正割合：25%）を施した結果を、図９-(E)(F)(G) に示す。図９-(H) は採譜の模範解答として、図９-(A) の譜面を基に手作業で MIDI 打ち込みを行った結果で、ベロシティ・パラメータは 64 に均一にしている。図９-(F) は論外であるが、図９-(G) は図９-(E) に比べ、図９-(H) にかなり近づいていることがわかる。

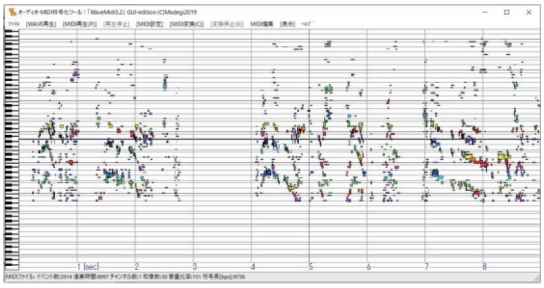
図９に示されるテンポが比較的遅い曲（ムソルグスキー組曲「展覧会の絵」ピアノ独奏版、プロムナードなど）では、図９には存在しない最大６重和音の全てが、演奏者が弾いた通り正確に再現できることを確認した。しかし、本稿では掲載を省略するが、同組曲で第１プロムナードに続く「グノーム（小人）」などテンポが速い楽曲については、倍音除去を行う前段階で正確に拾えない音符があり、周波数解析 (B) における時間分解能の更なる改善と解析音素の連結処理 (D) の高精度化が今後の課題となる。



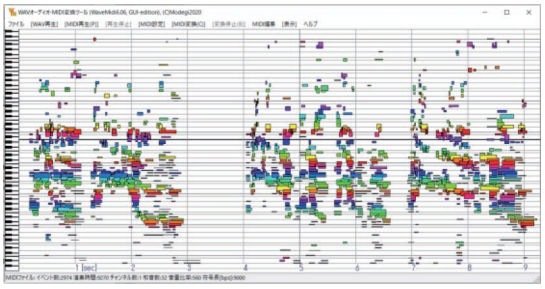
(A)男声アナウンスの音響波形（9.4sec）



(B)フレーム長4096の固定長窓によるMIDI変換



(C)フレーム長1024の固定長窓によるMIDI変換



(D)フレーム長4096以下の可変長窓によるMIDI変換

図１０　男声アナウンスの MIDI 変換例

最後に、ボーカルの解析精度について評価した結果を図１０に示す。図１０-(A) は男声アナウンス約 9.4 秒の音響波形（サンプリング周波数 44.1kHz, 量子化 16bits, モノラル）である。これに対し、本稿提案のオーディオ -MIDI 変換ツールにより MIDI 形式に変換し

た結果を図１０-(B)(C)(D) に示し、GM 標準 MIDI 音源（プログラム No.54, “Voice-Ooh”）を用いてボーカル再生音を評価した。図１０-(B) はフレーム長を 4096 サンプル固定で解析したものである。中低音域から高音域にかけて時間分解能が低いため、音声再生音が不明瞭で聴き取れない。一方、図１０-(C) は時間分解能を向上させるように、フレーム長を 1024 サンプルに短縮させて固定フレーム長で解析したものである。音声再生音は明瞭になり結構聴き取れるようにはなったが、一部の F0 フォルマント成分が欠落して音素の途切れが目立ち不自然である。これに対して、図１０-(D) は本稿提案の可変長窓を用いたもので、図１-(D) に示すように解析フレーム長を 4096 サンプル以下で可変に設定している。音声再生音は比較的明瞭になり、音素の途切れも目立たなくなっている。GM 標準 MIDI 音源（プログラム No.54）を用いて聴取すると、図１０-(D) は図１０-(B)(C)(D) の中で最も品質の良い再生音になっていた。

以上、本稿で紹介したツールおよび、その前身のツールは、過去約２０年間にわたって筆者が担当してきた本学・情報表現学科の演習授業の教材として使用してきたもので、並行して改良開発も進めてきた。本学の学生や教職員の皆様とのインタラクションが本研究開発の推進に多大な影響を与えたものと考えており、改めて本学関係者の皆様に謝意を示す。また、前述の通り、本稿で紹介した最新版の Windows 版ツールについては、筆者が担当している「クロスオーバー学習」（旧名称：マルチフィールド体験演習）等の演習授業で既に活用しているが、個人・法人を問わず学外の方にも、以下サイトにて C 言語ソースコードを含めて Windows 版ソフトウェア一式を公開しているので、教育・研究・音楽業務・商用・その他にご活用ください。

［オーディオ -MIDI 変換ツールの公開サイト (2020.9 現在、ver.6.07 を公開)］

以下サイトにて、基本操作マニュアル [WaveMidi_manual_200922.pdf]、Windows 版ソフトウェア一式 [WaveMidi_kit_200922.zip]（VisualStudio6.0/2017(x86 および x64) プロジェクト形式 C 言語ソースコードと実行ファイル）をダウンロードできます。

筆者のホームページ：http://www.bekkoame.ne.jp/~modegi/

または　https://sites.google.com/view/hptoshiomodegi

筆者の連絡先：modegi@bekkoame.ne.jp

引用文献

1) 茂出木敏雄「聴覚芸術への情報学的アプローチと音楽情報処理ツールの開発事例」『尚美学園大学・芸術情報研究』, Vol.18, Nov. 2010, pp.15-35.

2) 茂出木敏雄「オーディオ -MIDI 符号化ツール「オート符」における表情付け解析機能の実装」『尚美学園大学・芸術情報研究』, Vol.20, Nov. 2011, pp.17-34.

3) 茂出木敏雄「音響信号の MIDI 符号化ツール「オート符」の Windows10 対応に伴う改修」『尚美学園大学紀要「芸術情報研究」』, Vol.26, Mar.2017, pp.85-104.

4) 茂出木敏雄「音響情報の MIDI 符号化ツール「オート符」の開発」『芸術科学会誌 DiVA』, No.2, 夏目書房（株）, December 2001, pp.42-48. （ソフトウェアは一般財団法人デジタルコンテンツ協会（http://www.dcaj.or.jp）より 2010 年頃まで配布、現在は中止）。

5)「採譜の達人」MOONGift, https://www.moongift.jp/2007/03/3507/ (2020 年 8 月アクセス).

6) WaveTone (Softonic Developer Hub),「音楽をピアノロール楽譜に起こしてくれる！耳コピー支援ツール」, https://wavetone.softonic.jp/ (2020 年 8 月アクセス).
7) 永野宏治『信号処理とフーリエ変換』、朝倉書店、Jan.2014.
8) 深山幸穂 , 日野祐志 , 伊藤里美「ウェーブレット変換を用いた採譜システム」『情報処理学会研究報告・音楽情報科学 (MUS) 』(ISSN:09196072),Vol.2008, No.89, September 2008, pp.41-46.
9) Toshio Modegi, "Multi-track MIDI Encoding Algorithm Based on GHA for Synthesizing Vocal Sounds," *Journal of Acoustic Society of Japan*, Vol.20, No.4, April 1999, pp.319-324. (DOI: https://doi.org/10.1250/ast.20.319)
10) 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, October 2003, pp.1768-1775. (DOI: https://doi.org/10.1541/ieejeiss.123.1768)
11) 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, July 2010, pp.1159-1167. (DOI: https://doi.org/10.1541/ieejeiss.130.1159)