

計算機処理に基づく純正律音楽表現の実現

北川 尚希* 白井 治彦** 黒岩 丈介*** 小高 知宏* 小倉 久和***

The Real Time Expert System to Support the Pure Temperament Music Play

Naoki KITAGAWA* , Haruhiko SHIRAI** , Josuke KUROIWA***
Tomohiro ODAKA* and Hisakazu OGURA***

(Received February 10 , 2011)

In this study, we developed the Pure Temperament Performance System (PTPS), that can play pure temperament music. The PTPS is a kind of expert system to coordinate the frequency interval of the chord based on the knowledge processing technology. The PTPS plays melody in equal temperament and a chord in pure temperament to avoid the conflict. The conflict is that we can not play better consonance by only equal temperament and pure temperament.

The PTPS has a knowledge base, it contains a value to convert equal temperament into pure temperament. We have created the knowledge base.

The PTPS can play the pure temperament by the MIDI data by using MIDI sound source such as GM(general MIDI) sound source that is installed in default on Windows. The PTPS uses a value to call a pitchbend. Pitchbend is the value used to adjust the frequency in the MIDI data.

Key words : MIDI, Audio, Frequency of Sound, Desktop Music, Computer Processing

1. はじめに

本研究では鍵盤楽器を使用して、純正律演奏を行うためのソフトウェアシステムの開発を行った。

音楽は人間の歴史とともに様々な発展をしてきた。現代における一般的な音楽の定義は「材料に音を用いる」「音の性質を利用して組み合わせる」「時間の流れの中で音を組み合わせる」というものである。また、クラシックや教会音楽が中心の西洋音楽では音楽

の3要素であるリズム(律動)、メロディー(旋律)、ハーモニー(和声)を持つものが音楽とされている。音楽の3要素の形式や形態の研究は積極的に行われ、それぞれの時代の音楽形態にあったものが開発され使われてきた。そして、音楽とともに発展してきたものが楽器である。

同じ旋律や和音であっても、演奏される楽器によってその印象は大きく変化する。そして、1人で奏法の異なる複数の楽器を習熟することは非常に難しく、習熟できたとしても同時に演奏することができる楽器は少ない。だからこそ人は、バンドやオーケストラなどの形態をとり、曲の演奏を行う。しかし、PCの性能が上がり様々なソフトウェアの開発が行われていくうえで、1人で複数の楽器を操ることも不可能ではなくなってきた。それがDTM(デスクトップミュージック)と呼ばれるジャンルの音楽である。

昨今の音楽業界ではこのDTMと呼ばれる、シンセサイザを中心とした電子楽器やPC上で動作するソフ

*大学院工学研究科 原子力・エネルギー安全工学専攻

**技術部

*** 大学院工学研究科 知能システム工学専攻

*Nuclear Power and Energy Safety Engineering Course, Graduate School of Engineering

**Dept. of Technology

***Human and Artificial Intelligent Systems Course, Graduate School of Engineering

に

DTMは作曲だけでなく、音の加工や編集などを個人レベルでより簡単により高いレベルで行えるようにしており、実際の楽器を使用して行う演奏では再現不可能な様々な発音、表現が可能である。実際の楽器を使用しての演奏が困難な音楽表現の1つに純正律というものがある。この純正律は、音響心理学的に最も協和して聞こえるように調整して作られている音律である。

本研究ではその純正律の再現を、計算機処理を用いて行う。純正律の再現には電子楽器の世界的な規格であるMIDIを利用した。Windows上での動作を想定し、ソフトウェアシステムの開発はC言語によって行った。

本稿では、第2章に協和音程と純正律、平均律についての説明と、本研究で使用した音律手法の具体的な方法を述べる。第3章には、本研究で作成した純正律演奏システムについての構成とその概要、そしてMIDIについての詳細について述べる。そして第4章では前述したシステムを使用時と不使用時の出力を解析、比較する。第5章ではシステムの動作実験を行い、考察と問題点の解決方法について述べる。第6章には本研究を通して判明したことや今後の課題をまとめとして述べる。

2. 協和音程と音律

一般的には一定の空気の振動のことを音とよんでいる。また、特定の周波数に名前をつけることで音の音程と音名を明確に定義し、万人に通じる概念としていている。2音以上の音が同時に演奏された際、協和状態と不協和状態のどちらかの状態が発生する。しかし、この2つの状態に明確な線引きはなく、2音間の協和度が重要な要素となってくる。人間の耳は音響心理学的に、重音を構成する2音の振動数の比が単純であるほど協和して聞こえ、複雑な比になっているほど不協和音に聞こえるという性質がある。

2.1 協和音程

協和音程には完全1度(ユニゾン)、完全8度(オクターブ)、完全4度、完全5度といったものがある。

中でも、完全1度と完全8度は周波数比が1:1と1:2というとても単純な整数比であるため、どの音に適用した場合でも周波数のズレが発生することが無い。(図1) また、完全1度は完全同音、完全8度はオクターブ違いの同音である。

完全4度と完全5度は2つの異なる音で構成され

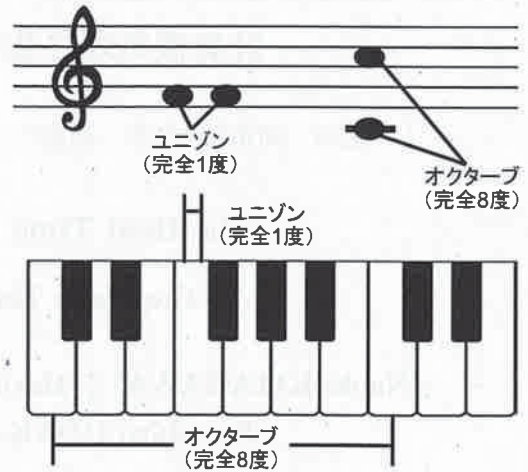


図1: ユニゾン (完全1度) / オクターブ (完全8度)

た和音であるが、その音は常に安定しており、1度、8度と共に完全和音と呼ばれている。また完全1度と完全8度はその中でもさらに絶対協和音程と呼ばれる。(図2)

長3度と短3度、長6度と短6度は、その和音を聞き慣れてない場合は必ずしも協和しているとは感じられないため、昔は不協和音とされていた。しかし、音楽のなかで多用されるようになり、バロック時代には協和音として認められるようになった。そのような時代背景もあり、これらは和音は不完全協和音と呼ばれる。(図3・図4)

以上で述べた以外の和音の組み合わせ、例えば長2度と短2度や長7度と短7度といった和音は音響学的には、全て不協和音と呼ばれる。(図5)

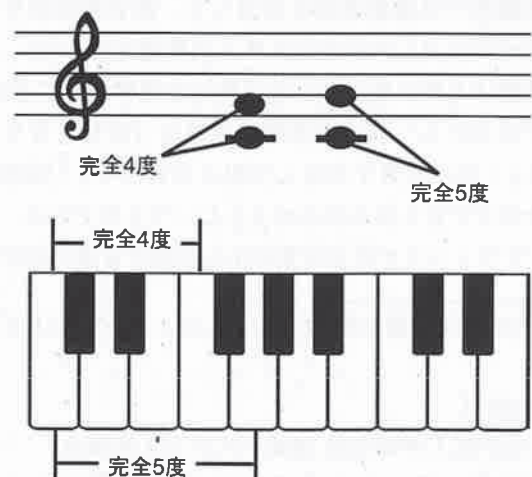


図2: 完全4度 / 完全5度

以上の協和音程と不協和音程を表1としてまとめ

た、表中の振動数比は和音を純正に取った場合の比となっている。最後の列の音名は、C (ド) を基音とした場合にその比率で和音となる音である。

和音は2音以上の場合に発生するが、同時に発音する音の数が増えても、人が感じる協和・不協和の音程条件は変わることが無い。そのため、音の数が増えれば増えるほど完全に協和するように音程を調整することは難しくなる。本研究では3音の和音を基本として想定している。これは、主旋律の1音に対して2音を和音として付加するためである。また、音楽的に3和音の種類は、長3和音、短3和音、増3和音、減3和音の4種類となっている。(図6)

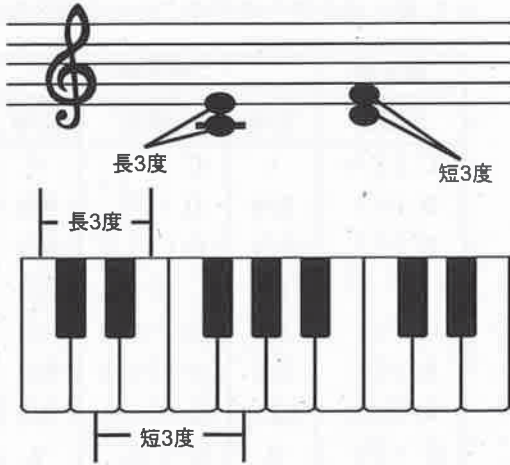


図3: 長3度/短3度

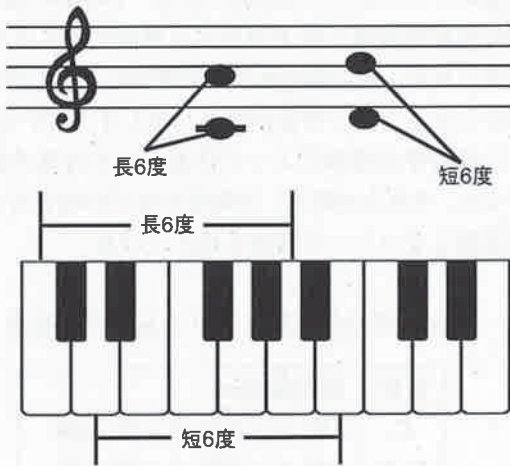


図4: 長6度/短6度

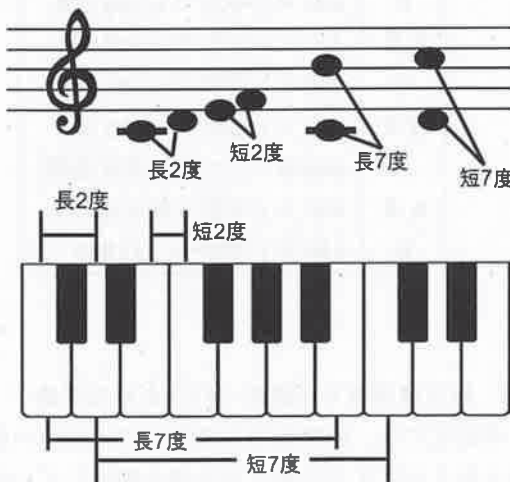


図5: 不協和音程

表1: 協和音と不協和音のまとめ

大分類	小分類	名称	振動数比	音名
協和音程	絶対協和音程	完全1度	1:1	C
		完全8度	1:2	C
	完全協和音程	完全5度	2:3	F
		完全4度	3:4	G
	不完全協和音程	長3度	4:5	E
		短3度	5:6	D #
長6度		3:5	A	
不協和音程		短6度	5:8	G #
		長2度	8:9 他	D
	転回音程	短2度	16:17 他	C #
		長7度	8;15	B
		短7度	5:9	A #
	それ以外			

長3和音		短3和音	
第5音 → 8	} 完全5度	第5音 → 8	} 完全5度
第3音 → 8		第3音 → 8	
根音 → 8		根音 → 8	
増3和音		減3和音	
第5音 → 8	} 増5度	第5音 → 8	} 減5度
第3音 → 8		第3音 → 8	
根音 → 8		根音 → 8	

図6: 3和音の種類と音程

2.2 音律

音律とは、各音の周波数比を一定のルールに従って定義したものである。この音律によって作られた音の連なりを音階と呼んでいる。

この音律は明確な周波数によっては決まっておらず、多くは1音ごとの周波数の比によって表される。

そのため、同じ音律であっても基準の音をどの周波数の音にするかによってその内容は大きく変わる。例えば、現在は A (ラ) = 440Hz を基準音とすると 1939 年の国際会議で決められている^[3]が、17~18 世紀のオルガンは A = 395Hz で調整されていた。他にも、モーツァルトやベートーベン、シューベルトなどが活躍した古典派~ロマン派の時代ではオーケストラのピッチに 420~430Hz 前後がよく使われたとされている。このような事実が存在するため、作曲された当時の演奏と現代の演奏とではその印象自体大きく異なるはずである。この変化は楽器の製造技術や演奏方法の進歩によるものであり、現代においてもその研究が行われている。

音律自体も昔からの研究、研鑽によって様々なものが生み出されてきた。最も古い音律は、ピタゴラス音律とよばれるものであり、これは純正 5 度 (周波数比 2:3・表 1) を 12 回重ねることで作られた音律である。そのピタゴラス音律を改善したものが純正律と呼ばれる音律である。そして、現在ではより音楽的な改良が加えられた、平均律と呼ばれる音律が主流となっている。この平均律は現代の音楽に良く合い、幅広く使用されている。

2.2.1 純正律

本研究では純正律と平均律の両方を利用している。そのため、それぞれの音律の特徴を簡潔にまとめる。

純正律は前記したピタゴラス音律の発展系であり、周波数比が長 3 度を 4:5、短 3 度を 5:6、完全 5 度を 2:3 となるように調整した音律である。長音階の 7 音と短音階の 7 音を基準音からの周波数比率であらわす。各表の音階名は基準音を C (ド) とした場合に対応する音である。(表 2) その特徴としては、基準音に関する特定の和音では協和性の高い演奏を行うことができるが、それ以外の和音では著しい不協和になるという点がある。そのため、実際の楽器演奏での使用は非常に困難である。また、純正律は周波数比によって決定されるので、基準音の数だけ作ることができる。

2.2.2 平均律

一般に平均律と呼ばれているのは、1 オクターブの周波数を 12 等分した十二平均律である。平均律は 1 オクターブ (周波数比 1:2) の区間を隣り合う音の周波数比が等しくなるように 12 等分した音律なので、任意の隣り合った 2 音間の比は $1: \sqrt[12]{2}$ となる。平均

表 2: 純正律音階の基準音に対する周波数比

長音階		短音階	
音階名	比率	音階名	比率
C (ド)	1	C (ド)	1
D (レ)	9/8	D (レ)	9/8
E (ミ)	5/4	E (ミ) \flat	6/5
F (ファ)	4/3	F (ファ)	4/3
G (ソ)	3/2	G (ソ)	3/2
A (ラ)	5/3	A (ラ) \flat	8/5
B (シ)	15/8	B (シ) \flat	9/5
C' (ド)	2	C' (ド)	2

律は基準音を定めることによってそれぞれの音の周波数を単一に決定することができる。表 3 は A = 440Hz を基準とした場合の周波数である。平均律の利点は楽器の音程を調整し直す事無くに移調することが可能な点である。純正律の場合は基準音によって協和する和音とそうでない和音が決定してしまっていたので、この平均律の登場によって作曲される音楽の幅は広がった。それと同時に、平均律ではどの和音も完全には協和しないという問題も抱えている。

表 3: 平均律音階における各音の周波数

音程	周波数 (Hz)
C	$440 \times (\sqrt[12]{2})^{-9} = 261.626$
C #	$440 \times (\sqrt[12]{2})^{-8} = 277.183$
D	$440 \times (\sqrt[12]{2})^{-7} = 293.665$
D #	$440 \times (\sqrt[12]{2})^{-6} = 311.127$
E	$440 \times (\sqrt[12]{2})^{-5} = 329.628$
F	$440 \times (\sqrt[12]{2})^{-4} = 349.228$
F #	$440 \times (\sqrt[12]{2})^{-3} = 369.994$
G	$440 \times (\sqrt[12]{2})^{-2} = 391.995$
G #	$440 \times (\sqrt[12]{2})^{-1} = 415.305$
A	440.0(ドイツ国際高度基準)
A #	$440 \times (\sqrt[12]{2}) = 466.164$
B	$440 \times (\sqrt[12]{2})^2 = 493.883$

2.3 純正律演奏を矛盾なく行うための手法

本研究では、鍵盤楽器を使用して純正律の演奏を行えるようにするために平均律を利用している。具体的には、和音の基音を平均律でとり、その音に対して純正律の和音をとるという方法である。平均律の利点は平均律の基準音を設定することによって、各

音を単一の周波数で決定することができる点である。この利点を利用することで、純正律の欠点である、どの周波数の音でも基準にできてしまうために無数の純正律が作れてしまう、という点を解決できる。

より具体的な例として、C-E-Gの長3和音を使用し説明を行う(図7)。この和音の場合は3和音のうち、最も低い音であるC(ド)が基音となっている。ほとんどの和音は基本的に一番低い音が基音となるが、転回和音と呼ばれる和音ではその限りではない。その基音を平均律でとり、その平均律の周波数を基にして純正律の和音をとる。まず、1つ上のE(ミ)は基音との関係が長3度となっているので、周波数比を4:5でとる。もうひとつ上のG(ソ)は基音との関係が完全5度であるため、周波数比2:3でとる。このように和音をとることで、この長3和音の周波数比はC-E-Gの順に4:5:6となり、協和した和音を作ることができている。

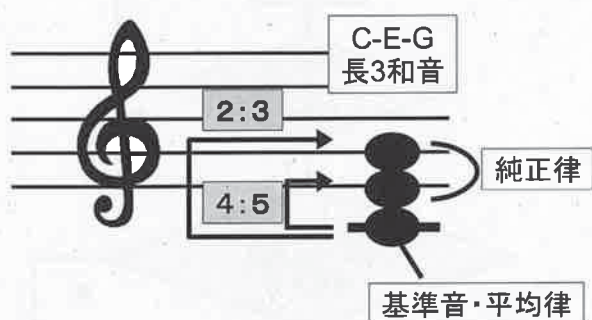


図7: 本研究での手法例

表4は平均律、純正律、本研究での手法のそれぞれでC-E-G(Cメジャー)の和音を構成する各音の周波数がどの程度異なっているかをHzで表したものである。平均律と純正律はA=440Hzを基準音としてハ長調の音階を作った場合の周波数である。純正律は表2のAの部分に440を代入して計算している。本研究での手法は図7と同じ場合のもので、C音は平均律でとったものなので同じ値となっている。この3パターンの周波数で和音中のうなりの数を算出して比較する。まず、長3和音の場合、3音が協和する周波数比は前述したとおり4:5:6となる。そのため、最小公倍数は60となり、最初に3音で重なる倍音はそれぞれCを15倍、Eを12倍、Gを10倍した音となる。計算の結果を表5に示す。音響学的には計算結果は同じ周波数にならないといけないが、平均律で作った和音に関しては他2つと比べて大きな差が見られる。純正律と本研究の手法では、計算結果に差がほ

んど見られず、基の和音の協和性の高さがわかる。

以上のことから本研究での手法は協和性を高める上で平均律よりも有効であるといえる。次に、この手法を使用しているシステムについて説明していく。

表4: 周波数の差(Cメジャー)

	平均律	純正律	本研究での手法
G	391.99	396.00	392.45
E	329.63	330.00	327.04
C	261.63	264.00	261.63

表5: うなりの回数の差

	平均律	純正律	本研究での手法
G*10	3919.9	3960.0	3924.5
E*12	3955.56	3960.0	3924.48
C*15	3924.45	3960.0	3924.45
最大最小の差	35.66	0.0	0.05

3. 実時間エキスパートシステムについて

本研究ではシステムの開発をC言語によって行った。Windows上での動作を想定し、システム内で扱う音楽データはWindowsに標準でインストールされているMIDI形式とした。

3.1 システム構成

システム構成図を図8に示す。今回作成したシステムには大きく分けて4つの処理部分がある。入力部ではMIDIキーボードからの入力の監視を行っている。キーボードからの入力が入力メッセージとして受け取れた場合、そのメッセージの解析、プログラム内での変数への代入を行う。

判別部では入力された音の判別を行っている。主に何音同時に入力されているか、入力された音はどの音なのか、入力された音の組み合わせは何の和音を構成しているのかを判別する。その判別を行う際は、和音テーブルを参照している。和音テーブルには、入力された音の組み合わせによってどの和音が構成されているかのデータが格納されている。

音程調整部では平均律の音程を純正律の音程へと上書きするという処理を行っている。音程の上書き

は和音の種類ごとに異なる値を適用する必要があり、どの和音のときにどの音に対してどういう値を上書きするかというルールはシステム内でリアルタイムに計算しておらず、ピッチ調整ルールに格納された値を参照して行っている。

最後に、出力部ではその名のとおり MIDI データの出力を行っている。出力されるデータは、入力データとピッチバンドに関する部分に上書きが行われただけの MIDI データとなる。MIDI の性質上、出力されるのは音そのものである波形データではなく、MIDI メッセージ形式のデータである。出力部からは PC 内の GM 音源へとデータが渡され、そこで発音されることとなる。

本研究で作成したシステムには、楽器的な演奏を行うためのインターフェースを含んでいない。実際にこのシステムを使用する際には、MIDI メッセージを行うことのできるピアノ鍵盤型キーボードをケーブルによって接続して使用することとなる。

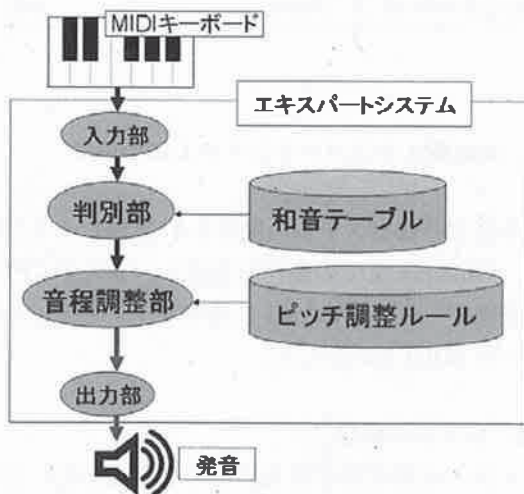


図 8: システム構成図

3.2 システム内の処理手順

システム内での処理手順を図 9 に示す。まず、システム内での処理は入力部への入力待ち状態となる。MIDI キーボードからのデータの入力を常に監視し、入力があるまで待機状態となる。キーボードからの入力が行われると、処理は判別部へと移る。判別部で入力された MIDI データの解析を行い、その結果を和音テーブルと照らし合わせる処理を行う。その際、入力された音が和音テーブル内に存在するかの確認を行い、テーブル内にデータが無かった場合は入力時

の MIDI メッセージに上書きを行わずにそのまま出力部へとデータを渡す。テーブル内にデータが存在した場合は、次の処理部である音程調整部へとデータが渡される。音程調整部では MIDI メッセージ内のピッチバンドの値に対して上書きを行う。この上書きが行われた段階で平均律から純正律への変換が行われたこととなる。純正律へと調整された上書きするためのピッチバンドの値はピッチ調整ルール内に格納されている。ピッチバンドの上書きを行った後、処理は出力部へと移る。出力部では、入力時の MIDI メッセージと照らし合わせることでエラーのチェックを行い、問題が見つからなければ MIDI メッセージとして出力する。本システムから出力された MIDI メッセージは PC 内の GM 音源によって波形化され、その後発音されることとなる。

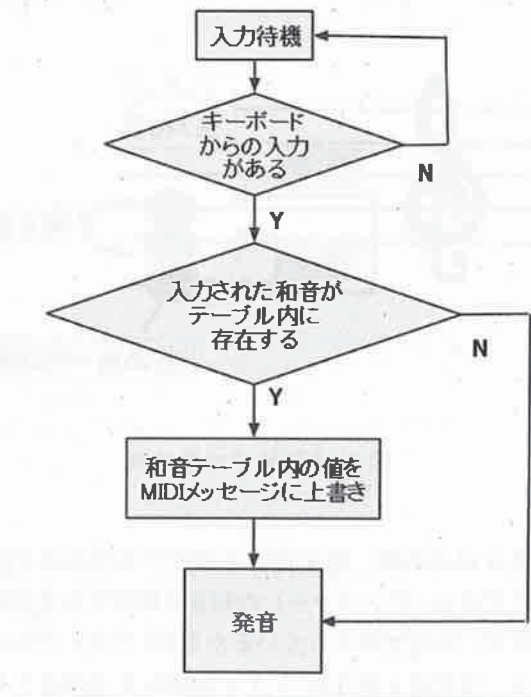


図 9: システム内の処理手順

3.3 MIDI を使用しての再現

本システムでは MIDI メッセージを利用して純正律の再現を行っている。MIDI は Musical Instrument Digital Interface の略称で、日本の MIDI 規格協議会と国際団体の MIDI Manufacturers Association (MMA) によって作られた、電子楽器間での世界共通の規格である。WAVE などの波形データとは異なり、音符の並びや使われる楽器、音色などを扱える楽譜データであり、Windows API によるプログラミングで扱うこと

ができる。また、MIDI 音源には Windows に標準搭載されている GM (General MIDI) 音源を使用する。

MIDI メッセージ内にはさまざまな情報が記されている。大きく分けて、個々のトラックや音を管理するチャンネルメッセージとシステム全体をコントロールするためのシステムメッセージが存在する。本研究では、チャンネルメッセージ内のピッチベンド (Pitch Bend) を利用して各音の音程を変化させる。ピッチベンドは MIDI 内で音程を操作するための固有の値で、ビブラートやスラー、ギターのコッキングの再現に使用されることが多い。-8192~0~+8191 (初期値は 0) の値を設定でき、その範囲で実際にどの程度音程が変化するかはピッチベンドセンシティブリティ (ベンドレンジ) という別の値によって決定される。ピッチベンドセンシティブリティは半音ごとに設定でき、初期値の 2 では上下どちらとも全音分変化することになる。本実験のシステム内では初期値の 2 を前提としている。

MIDI はデフォルトの状態では平均律になるように作られている。そのため、純正律での演奏を行う場合、平均律と純正律の周波数の差を求めるとピッチベンドに変換→MIDI 内の値の上書き、という手順を踏むこととなる。計算式は以下ようになる。

$$\log\left(\frac{F_p}{F_t}\right) \div \frac{\log 2}{6} * 8191 = 49146 \frac{\log\left(\frac{F_p}{F_t}\right)}{\log 2} \quad (1)$$

平均律の周波数を F_t 、純正律の周波数を F_p とおく。 $\log(F_p/F_t)$ を平均律の全音である $(\log 2)/6$ で割った値が補正值となり、それに対して全音分変化させるピッチベンドの値である 8191 (変化幅は -8192 からであるが、上下どちらも 8191 が全音分の変化に相当すると仮定する) を掛けた値が入力すべきピッチベンドの値となる。

本システムでは以上の計算をあらかじめ行い、実際にシステム内で上書きするピッチベンドの値を数値で格納している。

4. 実験

4.1 実験環境

Windows 環境下で実験を行った。今回の実験では、GM 音源として Windows に標準でインストールされている、Microsoft Wavetable GS SW Synth を使用した。MIDI メッセージを送信できるソフトウェア MIDI キーボードとして K.Nukui さんが web 上で公開しているフリーソフトウェアの「midiKey」^[6]を使用した。

また、ソフトウェア MIDI キーボード、本研究での開発ソフトウェア、MIDI 音源をつなぐための仮想 MIDI ケーブルとして、「MIDI Yoke」^[7]を使用するためあらかじめインストールを行った。

4.2 実験結果

実験の評価は、本研究で作成したソフトウェアの使用時と不使用時の MIDI データを比較することで行う。音ではなく目に見えるデータとして評価するために、今回は波形データを使用することとする。そのために以下の手順を行った。まず、MIDI データを PC 内で録音し、録音したデータを WAVE ファイルに変換する。その際本研究で作成したエキスパートシステムを使用時と不使用時の 2 パターン録音する。演奏の内容は同じである必要があるため、特定の和音を演奏することとする。その後 WAVE データを波形データへと変換するソフトウェアを使用し、波形データの作成を行う。

以上の手順で作成した波形データを図 10 に示す。なお、今回の実験では E メジャーの和音を利用した。

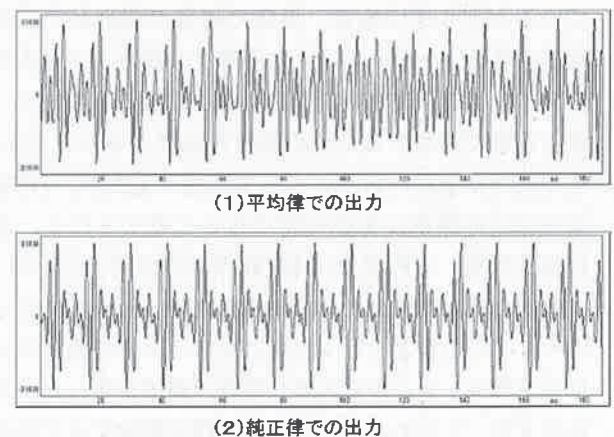


図 10: 波形の比較

(1) は本システムを使用しない平均律の場合の出力、(2) は本システムを使用した純正律の場合の出力となる。それぞれの波長を比較すると、平均律は一定の周波数ではなく、大きな周期で波打っていることが分かる。それに対して純正律は、一定の周波数で出力されていることが分かる。この実験結果から、本研究で作成したシステムによって周波数の変換が行われていることがわかる。

5. 考察

周波数は 1Hz 以下で測定可能であり、演奏される楽器や肉声の振動数は楽器や体のコンディション、演奏する環境、気温、聴衆との距離などの非常に多くの要素によって左右されるため、簡単に揺らぎが発生する。そのため、厳密な周波数を保つての演奏というのは、実際問題非常に難しい。

そのため、完全な純正律で曲の演奏を行うためには 2 つの大きな前提条件が必要となると考えられる。まず 1 つは、演奏空間の調整である。この場合の演奏空間は、音が演奏者から発せられて聴衆に届くまでの空間とする。演奏者から発せられた音が減衰、干渉、ノイズの混入が起こることなく聴衆まで届く空間が必要となるが、実際にそのような演奏空間は存在し得ない。音は演奏者から発せられた瞬間に空間からの干渉を受け、微小な変化をおこすこととなる。この変化は演奏者の演奏技術での軽減が難しく、あらかじめ完璧な演奏空間を用意しておくことが必要となる。2 つ目は演奏者から音が発せられた音の状態である。つまり、音が発せられた段階で和音が純正となっているかどうかである。単体で和音の演奏ができる楽器の場合、チューニングの段階での調整がひとつようである。また、演奏中に音をぶれさせることなく、一定の音程で演奏するための技術も必要となってくる。その技術は楽器の習熟によって上達可能だが、DTM ならばより簡単に発音可能であると考えられる。その点も含め、本研究では DTM の利点と考えている。

今回作成したシステムでは MIDI メッセージのピッチベンドの上書きを行うためにシステム外に和音テーブルを作り、そこから内容の検索・適用を行った。これにより、システム内での計算処理を軽減することができたが、それにより和音テーブル内にあらかじめ用意してある和音にしか対応ができないものとなった。今後の改良点として、周波数をテーブルで管理するのではなく和音パターンごとの計算式を作成することで対応させる必要がある。また、楽器として扱えるようにするための GUI インタフェースが必要である。より扱いやすいソフトウェアとするために、どのようなインタフェースが適しているか、作成と実験が必要である。

6. まとめ

本研究では純正律を使用しての演奏が可能なシステムの開発を行った。Windows 上での操作を前提とし、C 言語を使用して Windows アプリケーションと

して作成した。システム内部で扱う音楽データの形式には、windows に音源が標準搭載されており、音程の操作ができる MIDI データを使用した。システムを使用時と不使用時で出力された音をそれぞれ波形解析することで、本システムによって平均律から純正律への変換が行われていることが確認できた。今後、本研究で作成したシステムを一種の楽器として操作できるようにするために、GUI インタフェースの追加や対応できる和音数の増加などを行う必要がある。

参考文献

- [1] 小島知成, 黒岩文介, 小高知宏, 小倉久和, 白井治彦: 平均律楽譜情報を用いる純正律演奏システム, 平成 16 年度電気関係学会北陸支部連合大会講演論文集
- [2] 北川尚希, 黒岩文介, 小高知宏, 小倉久和, 白井治彦: 純正律による音楽表現を支援するエキスパートシステム, 平成 20 年度卒業研究, (2008)
- [3] 最相葉月: 絶対音感, 小学館, (1998).
- [4] 日本シンセサイザー・プログラマー協会 (JSPA): MIDI 検定 3 級公式ガイドブック, ミュージックトレード, (2002).
- [5] 日本シンセサイザー・プログラマー協会 (JSPA): MIDI 検定 2 級公式ガイドブック, ミュージックトレード, (2000).
- [6] midiKey : <http://hp.vector.co.jp/authors/VA018952/>
- [7] MIDI Yoke : <http://www.midiox.com/>