

## くし形フィルタによる異楽器音中のピッチ推定と演奏楽器推定

三輪 多恵子<sup>†</sup>, 田所 嘉昭<sup>†</sup>, 斎藤 努<sup>††</sup>

<sup>†</sup>豊橋技術科学大学 電子・情報工学専攻, <sup>††</sup>豊田工業高等専門学校 電気工学科

あらし 本研究では、異楽器音に対する自動採譜システムのためのピッチ推定と楽器推定の一手法として、サンプル値データの差分で表現できるくし形フィルタに基づいた方法を提案する。ある音高に対応したくし形フィルタは、基本周波数(ピッチ周波数)とその倍音を全て除去する性質を持つ。よって、縦続接続したくし形フィルタの零出力を検出することでピッチ推定が可能である。また、くし形フィルタを用いて異楽器和音を単一楽器音に分離し、その分離音のスペクトル(倍音構造)を解析し、保存してある各楽器のテンプレートと比較することで演奏楽器を推定する。本稿では、提案法を説明し、計算機シミュレーションにより、その有効性を確認する。

### Pitch Estimation and Musical Instrument Estimation from Different Musical Instrument Sounds using Comb Filters

Taeko Miwa<sup>†</sup>, Yoshiaki Tadokoro<sup>†</sup>, Tsutomu Saito<sup>††</sup>

<sup>†</sup> Department of Information and Computer Sciences, Toyohashi University of Technology

<sup>††</sup> Electrical of Engineering, Toyota National College of Technology

**abstract** In this paper, we propose a method of the pitch frequency estimation using comb filters and the estimation of playing musical instruments using harmonic frequency structure (spectrum) of a musical tone. We can estimate the pitches of mixed musical instrument sounds by remarking the zero outputs of cascaded comb filters. To estimate the playing musical instruments, we evaluate a subtraction between the spectrum of the separated monophony and the single musical tone prepared for the template matching processing. We could obtain better results than the previous method which used the signal waveform separated from the mixed musical instrument sounds using comb filters.

## 1 まえがき

採譜とは楽曲から楽譜を作成する作業であり、専門的な知識や経験が必要とされるため、計算機を用いた自動採譜の研究が盛んに行なわれている[1]-[10]。また、近年では異楽器による和音のピッチ及び楽器の推定へ、その対象が拡張される傾向にある[5]-[8],[10]。

ピッチの推定には、音楽信号の周波数解析が必要とされる。しかし、平均律音階では、ピッチ間隔が不等間隔(2<sup>1/12</sup>倍)であるため、一般的な周波数解析法であるFFTでは周波数補正等が必要とされる[2]。さらに、演奏楽器の推定では、異楽器和音中の各単音成分を個別に認識し、それらの特徴を判別する必要がある。この特徴判別の方法としては、各楽器の倍音構造を利用したテンプレートマッチング法が一般的である[8]。しかし、異楽器により発せられた各音の倍音成分が複雑に重なり合い、それぞれの音を分離して認識することは非常に困難とされる。

筆者らは、C, C#, ..., B の12音に対応したくし

形フィルタ  $H_{q,p}(z_p) = 1 - z_p^{-q}$  ( $p = 1, 2, \dots, 12, q$ : オクターブに関係) による音楽信号の解析を検討している[9],[10]。時間領域において、 $H_{q,p}(z_p)$  はサンプル点の差分により実現され、演算量の削減及び処理系の単純化が期待できる。また、 $H_{q,p}(z_p)$  は、 $0[\text{Hz}] \sim fs_p(\text{sampling frequency})[\text{Hz}]$  を  $q$  等分する箇所に零点を持つ。このため、問題とされていた倍音成分が除去され、その影響を回避することができるため、異楽器音に対して有効であると考えられる。

この性質を利用し、縦続接続した  $H_{q,p}(z_p)$  の零出力によるピッチ検出法、及び  $H_{q,p}(z_p)$  による異楽器和音の単音への分離方法を提案し、分離音の振幅変化(エンベロープ)による楽器推定法の検討を行なった[10]。しかし、楽器の構造上類似したエンベロープを持つ場合があり、精度の面で問題が生じた。

この問題を解決するために、前述の分離音の倍音構造(スペクトル)に注目した楽器推定法を提案する。本方式では、分離音の倍音構造と各楽器音の倍音構造(テンプレート)との差分を計算することで、演奏

楽器の推定を行なう。また、このテンプレートは、採譜対象である  $C_3$  (オクターブ 3 の  $C$  音) から  $B_5$  の 36 音に対して、各楽器 9 種類程度用意される。

本稿では、始めに、本方式で用いるくし形フィルタの原理を説明した後、くし形フィルタによるピッチ検出方式を説明する。次に、くし形フィルタを用いて異楽器和音を単音に分離する手法及び、分離音の周波数解析方法を示す。最後に、MIDI 音源を使用した計算機シミュレーションにより、本方式の性能を明らかにする。

## 2 くし形フィルタの原理

この章では、本方式で使用するくし形フィルタの構成及び、その原理について説明する。

### 2.1 くし形フィルタ

本方式で使用するくし形フィルタの伝達関数は式 (1) で表現され、式 (2) に示す簡単な減算で実現される。ここで、 $p = 1, 2, \dots, 12$ ,  $q = 2$  (#5:オクターブ 5), 4 (#4), 8 (#3) である。

$$H_{q,p}(z_p) = 1 - z_p^{-q} \quad (1)$$

$$y_p(n) = x_p(n) - x_p(n - q) \quad (2)$$

この  $H_{q,p}(z_p)$  は、図 1 に示すように、 $0[\text{Hz}]$  からサンプリング周波数  $f_{s,p}[\text{Hz}]$  までを  $q$  等分する箇所に零点を持つ。なお、 $f_{i,p}$  はオクターブ番号  $i$  の  $p$  音の基本周波数であり、 $f_{s,p} = 2 \cdot f_{5,p}$  である。

図 1 から、 $H_{8,p}(z_p)$  は  $p$  音の全ての周波数成分を除去できるため、 $y_p(n)_{q=8} = 0$  の検出により入力音  $p$  が判別可能である。逆に  $y_p(n)_{q=8} \neq 0$  の場合は、 $p$  以外の音が入力されたと判断する。

さらに、 $H_{q,p}(z_p)$  によりオクターブ推定が可能である。各  $H_{q,p}(z_p)$  の零点と  $f_{i,p}$  の周波数配置から、8 次のかし形フィルタ  $H_{8,p}(z_p)$  では #3~#5 の全ての周波数成分を除去できるが、 $H_{4,p}(z_p)$  では  $f_{3,p}$  の奇数倍の倍音成分が除去できないことがわかる。この性質を利用して、 $y_p(n)_{q=8} = 0$  であり、かつ  $y_p(n)_{q=4} \neq 0$  の場合、入力は #3 の  $p$  音であることが判別できる。#4, #5 に関しても同様に、 $H_{q,p}(z_p)$  の次数を 8 次から 4 次、4 次から 2 次に変化させ、非零出力を検出することで低いオクターブから順次判別可能である。

### 2.2 Over-sampling 法

各音に対応したくし形フィルタ  $H_{q,p}(z_p)$  は、サンプリング周波数  $f_{s,p}$  が異なるため、直接接続することは不可能である。この問題を解決するために、本方式ではオーバーサンプリング法を用いる。

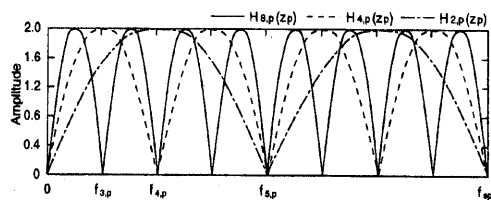


図 1 くし形フィルタの周波数特性

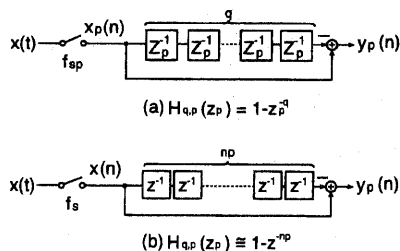


図 2 Over-sampling 法による  $H_{q,p}(z_p)$  の近似

オーバーサンプリング法では、一定の高いサンプリング周波数  $f_s$  により得られるサンプリング値系列  $x(n)$  を使用する。  $x(n)$  のサンプル点間隔は  $1/f_s[\text{s}]$  であり、この場合の単位遅延を  $z^{-1}$  とする。一方、各  $f_{s,p}$  により得られる  $x_p(n)$  の単位遅延を  $z_p^{-1}(1/f_{s,p}[\text{s}])$  で表す。ここで、 $z^{-1}$  を用いて  $z_p^{-q}$  を近似し、各  $H_{q,p}(z_p)$  を近似的に実現する。

各音ごとに式 (3) を満たす整数値  $n_p$  を計算し、式 (1) を以下のように近似する。

$$n_p \cong \frac{q \cdot f_s}{f_{s,p}} \text{ [四捨五入で整数化]} \quad (3)$$

$$H_{8,p}(z_p) \cong 1 - z^{-n_p} \quad (4)$$

式 (4) から、図 2 (a) に示す  $H_{q,p}(z_p)$  は同図 (b) で近似されるため、各  $H_{q,p}(z_p)$  は 1 種類のサンプリング周波数  $f_s[\text{Hz}]$  によって近似的に実現できる。なお、オーバーサンプリング法を用いることで、各  $H_{q,p}(z_p)$  の零点には若干のズレが生じるため、線形補間を行なうことで出力誤差の軽減を図る。

## 3 異楽器音に適したピッチ推定

本章では、各音に対応した 12 種類のかし形フィルタ  $H_{8,p}(z_p)$  を縦続接続し、各段の零出力を検出することでピッチを推定する原理について説明する。

なお、平均律音階では  $2^{1/12}$  倍という周波数間隔により、ある音に対応した  $H_{8,p}(z_p)$  の零点周波数とそれより 7 音離れたピッチ周波数が非常に近い値になる。著者らは、この関係を利用して  $H_{8,p}(z_p)$  の縦続数を 7 個に削減し、ノッチフィルタを併用して重複

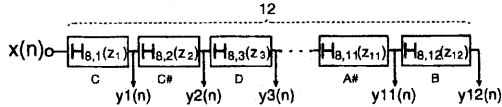


図3  $H_{8,p}(z_p)$  の縦続接続によるシステム構成

音を判別する手法を提案した [9]。提案したシステムでは、テンポ 120(4/4 拍子) の 32 分音符 (64[ms]) までのピッチ検出が可能であるが、本稿では 12 個の  $H_{8,p}(z_p)$  を縦続接続する基本原理の説明にとどめる。

### 3.1 システム構成

本方式では、 $C, C\#, \dots, B$  の 12 音に対応したくし形フィルタ  $H_{8,p}(z_p)$  を図 3 のように縦続接続することでシステムを構成する。このシステムでは、入力  $x(n)$  に対して、 $C$  音に対応した  $H_{8,1}(z_1)$  から  $B$  音に対応した  $H_{8,12}(z_{12})$  を用いて 1 音ずつ順に除去し、 $x(n)$  に含まれる周波数成分を全て除去した段階で初めて零値を出力する。

### 3.2 ピッチ検出アルゴリズム

図 3 のシステムによる入力音判別アルゴリズムを図 4 に示し、以下にその説明を行なう。

#### (a) 単音の推定

図 3 のシステムに単音  $p$  を入力した場合、 $p$  に対応した  $H_{8,p}(z_p)$  で零出力が検出される。この結果、入力  $x(n)$  は少なくとも  $p$  音を含むことが判断できる。次に、他の周波数成分の有無を調べるため、 $H_{8,p}(z_p)$  を縦続の先頭に移動する。この場合の入力は単音  $p$  であるため、 $x(n)$  に含まれる周波数成分は  $H_{8,p}(z_p)$  で全て除去され、先頭段の出力  $y'_p(n)$  は零となる。この結果、 $x(n)$  は単音  $p$  であることが判別される。

#### (b) 和音の推定

図 3 のシステムに和音  $p_1 + p_2 (p_1 < p_2)$  を入力した場合、最初に  $H_{8,p_2}(z_{p_2})$  で零出力が検出される。ここで、単音の場合と同様に  $H_{8,p_2}(z_{p_2})$  を縦続の先頭に移動する。しかし、 $H_{8,p_2}(z_{p_2})$  のみでは  $x(n)$  中の  $p_1$  成分が除去されないため、先頭段の出力  $y'_{p_2}(n)$  は非零値をとる。この結果、 $x(n)$  には  $p_2$  以外の成分が含まれることが判別でき、この時点でのフィルタの並びは  $H_{8,p_2}(z_{p_2}), H_{8,1}(z_1), \dots$  となる。

$x(n)$  に含まれる他の周波数成分を検出するため、再び  $H_{8,1}(z_1)$  から順に同様な処理を行ない、次に零出力を検出するのは  $H_{8,p_1}(z_{p_1})$  である。前述と同様に  $H_{8,p_1}(z_{p_1})$  を  $H_{8,1}(z_1)$  の前段に移動すると、フィルタの並びは  $H_{8,p_2}(z_{p_2}), H_{8,p_1}(z_{p_1}), H_{8,1}(z_1), \dots$  となる。ここで、 $x(n)$  中の周波数成分は、先頭の 2 種類のくし形フィルタ  $H_{8,p_2}(z_{p_2}), H_{8,p_1}(z_{p_1})$  で除去

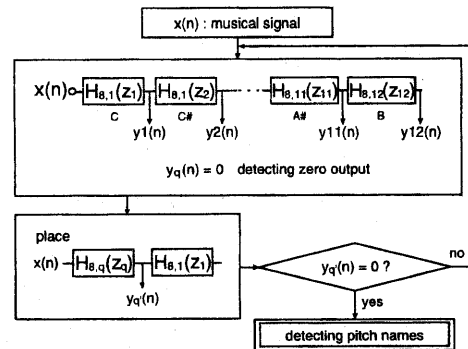


図4 音名判別アルゴリズム

され、出力  $y'_{p_1}(n) = 0$  が検出される。この結果から、入力された和音  $p_1 + p_2$  が判別される。

3 音以上の和音についても、上記の処理を繰り返すことで順次判別可能である。さらに、入力音 ( $C, C\#, \dots, B$ ) の判別の後、2 章で述べたオクターブ推定を行い、ピッチ (基本周波数) を推定する。

本方式で用いた  $H_{8,p}(z_p)$  は、入力音が  $f_{sp}$  に対応する場合、その倍音成分の割合 (楽器の種類) に関わらず、全ての周波数成分を除去可能である。従って本システムでは、異種楽器和音に対して、楽器の種類に依存せずピッチ検出可能であるという利点を持つ。

## 4 異楽器和音の分離

本章では、くし形フィルタを利用した異楽器和音の分離方法と、そのスペクトルの求め方を説明する。

### 4.1 くし形フィルタによる和音分離

異楽器音  $p_1, p_2, p_3$  から成る和音を例にとり、和音分離システムの構成と、その原理を説明する。

図 5 に示すように、入力音  $x(n)$  に対してくし形フィルタ  $H_{8,p_2}(z_{p_2}), H_{8,p_3}(z_{p_3})$  を縦続接続することで、 $x(n)$  から  $p_2, p_3$  音が除去され  $p_1$  成分が抽出できる。この処理は、異楽器和音から注目する楽器音のみを抽出する作業に相当し、 $p_2, p_3$  音についても同様な処理を行なうことで、 $x(n)$  を単一楽器音成分に分離可能である。また、入力が 4 音以上の和音の場合でも、和音に含まれる各音に対して上記の処理を行なうことにより、同様に分離することができる。

但し、和音分離の際に、縦続接続した各  $H_{q,p}(z_p)$  を通過することにより、抽出音の各周波数成分の振幅値に変化が生じる。例として、 $C_3, E_3, G_3$  の和音から  $C_3$  を抽出する際の振幅利得について、図 6 を用いて説明する。この場合、 $E_3, G_3$  に対応した  $H_{8,5}(z_5)$ ,

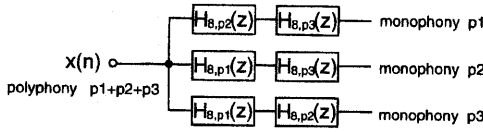


図5 和音分離システム

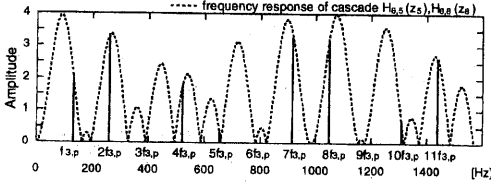


図6 和音分離による振幅利得

$H_{8,p}(z_s)$  が使用され、合成フィルタの周波数特性は図6の点線の形となる。従って、 $C_3$ の各周波数成分  $n \times f_{3,1}$  には、図中の実線で示す  $Gain$  が生じる。

ここで、 $H_{8,p}(z_p)$  の周波数特性は式(5)であるため、合成フィルタの周波数特性は式(6)となる。

$$\begin{aligned} |H_{8,p}(e^{-j\theta})| &= |1 - e^{-j\theta}| \\ &= \sqrt{2 - 2\cos(\theta)} \\ &= \sqrt{2 - 2\cos\left(\frac{8 \cdot 2\pi f}{f_{s,p}}\right)} \quad (5) \end{aligned}$$

$$\begin{aligned} G(e^{-j\theta}) &= \prod_p |H_{8,p}(e^{-j\theta_{8,p}})| \\ &= \prod_p \sqrt{2 - 2\cos\left(\frac{8 \cdot 2\pi f}{f_{s,p}}\right)} \quad (6) \end{aligned}$$

## 4.2 分離音の周波数解析

本方式では、DFTを用いて分離音の周波数解析を行なう。DFTでは、 $N$ 点の入力  $x(nT)$  ( $n = 0, 1, \dots, N-1$ ) に対する周波数スペクトル  $X(e^{j\omega_k T})$  が式(7)で与えられる。ここで、 $\omega_k$ は分析周波数であり、式(8)で計算される。

$$X(e^{j\omega_k T}) = \sum_{n=0}^{N-1} x(nT) e^{-j\frac{2\pi k}{N} n} \quad (7)$$

$$\omega_k = \frac{2\pi k}{NT} = k \cdot \omega_1 \quad (8)$$

式(7)、(8)より、DFTで分析可能な周波数は $\omega_1$ の整数倍に制限されることが分かる。

音楽信号(平均律音階)では、隣り合うピッチ周波数の比が非整数(2<sup>1/12</sup>  $\cong$  1.059...)倍という性質がある。しかし、本方式で扱う信号は、前述の和音分離システムで抽出された単音であるため、含まれる周

表1 各音に対応した  $N_p$  と分析周波数

音名	$N_p$	$f_{3,p}$ [Hz]	$f_s/N_p$ [Hz]
C	413	130.81	130.88
C#	390	138.59	138.60
D	368	146.83	146.89
D#	347	155.56	155.78
E	328	164.81	164.80
F	310	174.61	174.37
F#	292	185.00	185.12
G	276	196.00	195.85
G#	260	207.65	207.90
A	246	220.00	219.73
A#	232	233.08	232.99
B	219	246.94	246.82

波数成分は  $f_{3,p}$  の整数倍に制限される。従って、式(9)により、 $\omega_1/2\pi = f_{3,p}$  を満たすようなサンプル点数  $N_p$  を計算し、 $N_p$  点 DFT を行なうことで、各音の周波数解析が可能となる。

$$N_p = \frac{2\pi f_s}{\omega_1} = \frac{f_s}{f_{3,p}} \quad [\text{四捨五入で整数化}] \quad (9)$$

式(9)において、実際に使用するオーバーサンプリング周波数  $f_s = 54054.054$  [Hz] を用いて計算した各サンプル点数  $N_p$  と、それによる解析周波数  $f_s/N_p$  [Hz] を表1に示す。この表から、若干(最大で  $A_3$  における 0.27 [Hz] 程度)の誤差は生じるものの、各  $f_{3,p}$  とその倍音が解析可能であることが分かる。

しかし、前節で述べたように、縦続接続した各  $H_{8,p}(z)$  の  $Gain$  により、抽出音のスペクトルは入力楽器音とは異なる振幅を持つ。従って、DFTで得られた結果を、式(6)で求めた各  $Gain$  により除算することで、抽出音の元のスペクトルを求める。

## 5 倍音構造を利用した楽器推定

楽器音は、発音、共鳴機構の違いにより固有の倍音構造(スペクトル)を持つ。このため本方式では、テンプレートとして各楽器単音のスペクトルを用い、異楽器和音からの抽出音のスペクトルと比較することで楽器を推定する。しかし、楽器音のスペクトルは音域等の条件により変化するため、各楽器に対して複数のテンプレートを用意する必要がある。

本方式では、採譜対象音域  $C_3 \sim B_5$  の各楽器音のスペクトルを調べることで、比較的類似した倍音構造を持つ近隣の音のグループ化を行ない、同一グループ内のスペクトルを平均化したものを、そのグループのテンプレートとする。これにより、 $C_3 \sim B_5$  の36音に対するテンプレートは9種類程度となる。

なお、処理の簡単化のために、テンプレートと抽出音のスペクトルは、基本周波数成分の振幅値が1.0

となるように規格化する。同一楽器であれば倍音構造が類似していることが予想され、テンプレートと抽出音のスペクトルとの差分を計算し、差分値が最小のものを見つけることで演奏楽器が推定される。

## 6 シミュレーション結果

この章では、本方式を用いて異楽器和音のピッチ推定及び、演奏楽器の推定を行なった結果を報告する。

### 6.1 シミュレーション条件

ここでは、MIDI(KORG:05R/W)音源を用いて異楽器和音を作成し、A/D変換して計算機に取り込むことで計算機シミュレーションを行なった。また、実験で使用したA/D変換器の特性(分解能12bit,最高サンプリング速度 $8\mu s$ )から、オーバーサンプリング周波数 $f_s$ に $54.054[kHz]$ を用いた。

### 6.2 異楽器和音に対するピッチ推定結果

例として、 $pianoC_4, violinE_4, fluteG_4$ の3和音に対して、零出力検出によるピッチ推定を行なった結果を図7に示す。なお、この図では(1)から(2), (3)から(4), (5)から(6)で順に零出力の前後の波形と必要な $H_{8,p}(z_p)$ の並びを右上に示した。

ピアノ音の発音直後には若干の非調和周波数成分が含まれ、バイオリン音はピブラートによる周波数変動を持つ。この影響から、図7の零出力(2),(4),(6)の立ち上がり直後には減衰特性を持つ大きな雑音が生じ、また、零出力部分には共振特性を持つ雑音が生じているが、前段の出力と比較して十分な減衰が得られており、ピッチの推定は可能である。

### 6.3 異楽器和音の分離と周波数解析結果

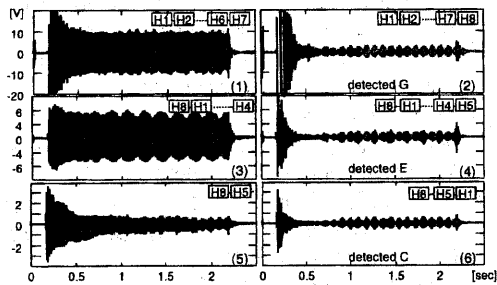


図7 異楽器和音  $pianoC_4, violinE_4, fluteG_4$  の零出力検出結果

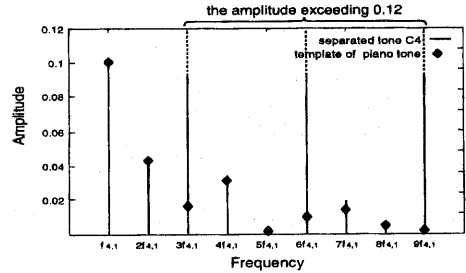


図8 分離波形  $C_4$  のスペクトル (規格化済)

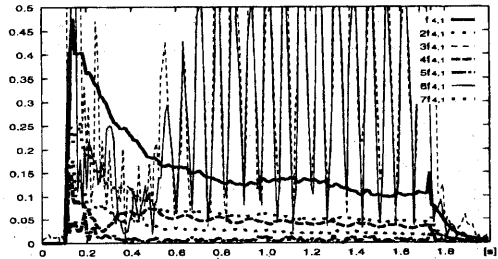


図9 抽出音  $C_4$  のSTFT結果

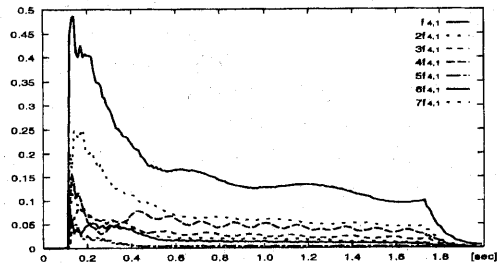


図10 ピアノ音  $C_4$  のSTFT結果

異楽器和音(ピアノ音  $C_4$ , フルート音  $E_4$ , バイオリン音  $G_4$ )について、例として、抽出音  $C_4$  の周波数解析結果を図8に示す。なお、比較としてピアノ音のテンプレートを同図に点で示す。この結果、 $f_{4,1}$  の3,6,9倍音が正しく復元できないことが分かる。

この原因を調べるために、抽出音  $C_4$  のSTFTの結果( $f_{4,1} \sim 7f_{4,1}$ )を図9に、比較として、ピアノ音  $C_4$  のSTFT結果を図10に示す。図9から、図8で復元できなかった倍音(3,6)成分には発振する大きな誤差が生じることが分かる。この原因としては、以下の点が考えられる。

- (1) オーバーサンプリングによる  $H_{q,p}(z_p)$  の近似により、除去できない周波数成分が存在する。
- (2)  $N_p = f_s/f_{3,p}$  (四捨五入) 点DFTの際に、分析周波数にズレが生じる。
- (3) 解析周波数の近傍にゼロ点を持つ  $H_{8,p}(z_p)$  により、合成フィルタのGainが微小な値となり、振幅

表2 和音(a) piano $C_4$ , flute $E_4$ , violin $G_4$ の楽器推定結果(スペクトルの差分の平均値)

	piano	flute	violin	unused
$C_4$	<b>0.015</b>	0.142	0.155	3,6,9 倍音
$E_4$	0.113	<b>0.005</b>	0.274	
$G_4$	0.140	0.133	<b>0.005</b>	2,4,6,8 倍音

表3 和音(b) piano $D_4$ , viola $F_4$ , violin $A_4$ の楽器推定結果(スペクトルの差分の平均値)

	piano	viola	violin	unused
$D_4$	<b>0.003</b>	0.104	0.141	3,6,9 倍音
$F_4$	0.167	<b>0.024</b>	0.171	5 倍音
$A_4$	0.146	0.209	<b>0.004</b>	2,4,6,8 倍音

補正の際の除算で誤差が増大する。

原因(1),(2)は、復元された周波数成分にも等しく生じる誤差であり、それほど大きな影響ではないと考えられる。しかし、(3)を確認した結果、今回使用した合成フィルタ  $H_{8,5}(z_5) \times H_{8,8}(z_8)$  では、 $f_{3,1}$  の3,6,9倍音に生じるGainが非常に小さな値となることが分かった。このため、(1),(2)による若干の誤差が、Gainによる補正で増大したと考えられる。

しかし、このような不都合なGainは、合成フィルタに使用される  $H_{8,p}(z_p)$  の種類が既知であるため、予め計算が可能であり、楽器推定のための差分計算から除外することで推定精度の向上を図る。

2種類の混合音和音(a) piano $C_4$ , flute $E_4$ , violin $G_4$ , (b) piano $D_4$ , viola $F_4$ , violin $A_4$  に対して、スペクトルの差分の平均値による楽器推定を行なった結果を表2, 3に示す。なお、楽器の推定方法としては、前述のように正規化した9倍音までのうち、合成くし形フィルタのGainが0.2以上の値を持つものの差分の平均値を用いた。この結果から、抽出音に対応した楽器との差分値が最も小さく(表中に太字で記入)、合成くし形フィルタのGainが微小な値を持つ倍音(表中のunused)を除いても、演奏楽器が推定できることが分かった。

## 7 むすび

本稿では、各音に対応したくし形フィルタを用いたピッチ推定及び異楽器和音分離を行ない、分離音の倍音構造から演奏楽器を推定することで、以下の点を確認した。

(1) 採譜対象である  $C_3$  から  $B_3$  の36音に対して、演奏楽器の種類と数に関係なくピッチが推定できる。但し、現在のところ、異楽器が異なる音を演奏した場合に限る。

(2) ピッチ推定後、各音に対応したくし形フィルタを用いて、異楽器和音を単一楽器音(倍音の振幅変化あり)に分離できる。また、抽出音の補正後のスペクトルと各楽器のテンプレートとの差分により、piano, flute, 及び、同族楽器である violin, viola を含む異楽器和音の楽器推定が可能であることを確認した。

今後検討すべき課題としては、以下の問題がある。

- (1) 採譜対象音域の拡大: 現在の方式では、#2以下の音域に対してくし形フィルタの次数が16回以上必要であり、ピッチ検出時間がかかりすぎるため短い音のピッチ検出が困難である。また、オーバーサンプリング周波数  $f_s \cong 54k$  [Hz] では、#6以上の音域のピッチ周波数とくし形フィルタの零点との差が大きくなり、線形補間による補正では不十分である。
- (2) 様々な入力に対する性能確認: 現段階では、計算機シミュレーションで使用した楽器や入力音の種類が少なく、十分に性能評価できたとは言えない。異なるオクターブにまたがる和音では、より多くの周波数成分が重なりを持つと考えられるため、本方式の限界を明らかにする必要がある。

## 参考文献

- [1] M.Piszczalski and B.A.Galler, "Automatic music transcription," Computer Music Journal, vol.1, no.4, pp.24-31, 1977
- [2] 原, 井口, "複素スペクトルを用いた周波数同定," 計測自動制御学会論文集, Vol.19, No.9, pp.718-723(1983)
- [3] 高沢, "計算機による採譜," 情報処理学会誌, Vol.29, No.6, pp.593-598(1988)
- [4] W.J.Pielemeier and G.H.Wakefield, "A high-resolution time-frequency representation of musical instrument signals," J.Acoust. Soc. Am. vol.99, no.4, Pt.1, pp.2382-2396, 1996
- [5] 永田, 安部, 城戸, "多数センサによる音源波形の推定," 日本音響学会誌, Vol.47, No.4, pp.268-273, (1995)
- [6] 柏野, 木下, 中臺, 田中, "音楽情景分析の処理モデル OPTIMA における和音の認識," 信学論, Vol.J79-D-II, No.11, pp.1762-1770, (1996)
- [7] 植田, 橋本, "音源分離のためのブラインドデコンポジションアルゴリズム," 情報処理学会誌, Vol.38, No.1, pp.146-156(1997)
- [8] 小野, 斎藤, 小沢, "自動採譜のためのGAを用いた混合音推定," 計測自動制御学会論文集, Vol.33, No.5, pp.417-423, (1997)
- [9] 三輪, 田所, 斎藤, "零出力に注目したくし形フィルタによる音階検出," 電学論, Vol.118-C, No.1, pp.57-64, (1998)
- [10] 三輪, 田所, 斎藤, "くし形フィルタを利用した採譜のための異楽器音中のピッチ推定," 信学論(掲載予定), (1998)