

スペクトログラムの長距離滑らかさを考慮した 調波打楽器音分離

橋 秀幸^{1,a)} 亀岡 弘和^{1,2,b)} 小野 順貴^{3,c)} 嵯峨山 茂樹^{1,3,d)}

概要：著者らは従来から音楽信号を調波音と打楽器音とに分離する研究に取り組んでいる．従来のモデルではスペクトログラム上の最近接の時間周波数成分同士の相互作用のみを考慮していたのに対し，本研究では長距離の相互作用を考慮するようにモデルを拡張することの検討を行なった．これにより，従来からのアルゴリズムの若干の高速化に成功した．またこの拡張によって，FitzGerald らによってなされた関連研究との間の類似性がある程度明らかになることについても述べ，さらに今後の展望について述べる．

1. はじめに

我々が普段耳にする機会の多いポピュラー音楽などの多くは，ピアノやギター，歌声のようにピッチを持った定常的な音と，打楽器のようなピッチを持たない瞬間的な音の，大きく分けて2種類の楽器音から成り立っている．このうち，前者によって主に和音やメロディなどが演奏され，一方後者によってビートが刻まれていることが多い．いま，ある楽曲中のコード進行を自動的に推定するような問題を考える．このとき，ピアノやギターの音は本質的に重要である一方，打楽器音はそれほど重要ではなく，それぞれかしばしば「雑音」として和音認識の性能を低下させるといった問題がある．同様に，楽曲のテンポを自動的に推定するような問題においては打楽器音が有用であるのに対して，ピアノの音などの情報も必ずしも必要ではない場合もある．このような，楽曲からコード進行，ジャンル，歌手などの情報を推定するという技術は，楽曲検索のための基礎技術として，特にインターネットで膨大な数の楽曲が共有されるようになった昨今重要性が高まっているが，これらの情報抽出の前処理として，和音認識やテンポ推定など

を行ない易い形に音楽信号を加工する技術が重要となる．すなわち，ピアノやギターの音と打楽器音とを大まかに分離するような技術を確立することには重要な意義がある．またこのような音楽信号分離技術は，新しい種類のエフェクタとしてそれ単体でも利用できる可能性のあるものであり，例えば，従来から一部の音楽愛好家はイコライザにより低音を強調して音楽を聴いていたように，この技術を応用することで，打楽器音を強調して音楽を鑑賞するというようなことも可能になる．

以上に述べたような応用上の重要性などを背景とし，これまでに調波音・打楽器音とを分離する手法がいくつか提案されている．代表的なものを挙げると，スペクトログラムの非負値行列分解 (NMF) とサポートベクターマシンに基づく手法 [5] や，FitzGerald による，スペクトログラム上のメディアンフィルタによる手法 [3] が知られている．また，このような分離技術を利用した音楽アプリケーションとして，Yoshii らによる Drumix [20] がある．

著者らも 2007 年より，音楽信号を分離する手法 “Harmonic/Percussive Sound Separation” (HPSS) の研究に取り組んでいる．HPSS のアイディアは初め文献 [7] において 2次元フィルタという形で提案され，次に文献 [8, 9, 11–13, 15] においてスペクトログラムの異方的な滑らかさコストなどを最小化する最適化問題としてさまざまな形で定式化された．その他，文献 [1, 6] では，NMF を用いる方法やステレオ信号の情報を用いる方法なども検討している．また，HPSS の応用に関しても，和音認識 [18]，リズムパターンに基づく楽曲構造推定 (Rhythm Map) [17]，歌声強調 [13, 16]，自動カラオケ生成 [14] など様々な応用を検討している．なお，上記の FitzGerald [3] の手法は，著者らの研究 [11, 12]

¹ 東京大学大学院情報理工学系研究科
Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1, Hongo, Bunkyo, Tokyo, 113-8656, Japan
² NTT コミュニケーション科学基礎研究所
NTT Communication Science Laboratory, 3-1, Morinosato-Wakamiya Atsugi, Kanagawa, 243-0198, Japan
³ 国立情報学研究所
National Institute of Informatics, 2-1-2, Hitotsubashi, Chiyoda, Tokyo, 101-8430, Japan
a) tachibana@hil.t.u-tokyo.ac.jp
b) kameoka@hil.t.u-tokyo.ac.jp
c) onono@nii.ac.jp
d) sagayama@nii.ac.jp

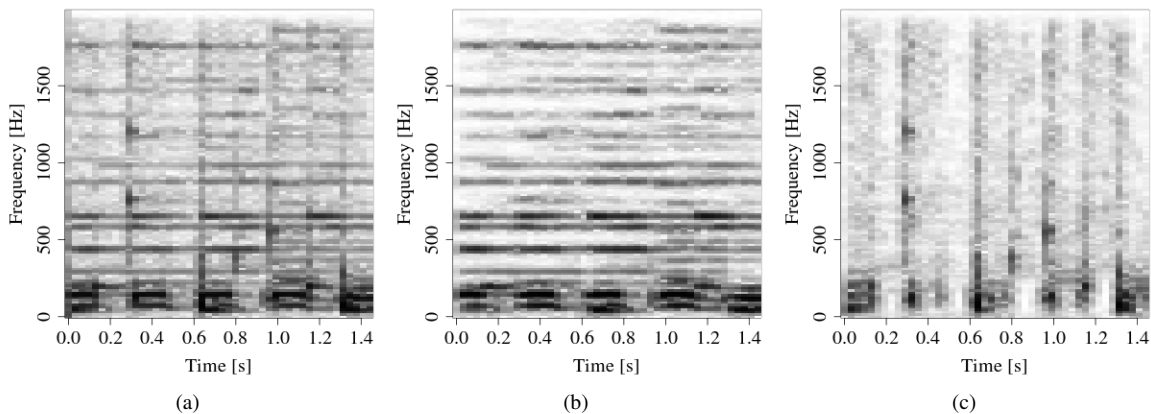


図1 HPSS による音楽信号のスペクトログラムの分離例 [15]. (a) 入力スペクトログラム (RWC 研究用音楽データベース [4] の RWC-MDB-P-2001 No. 14 より), (b) 分離されたスペクトログラム H , (c) 分離されたスペクトログラム P .

を受けて、同様のアイデアを著者らとは別の方法により実現したものである。

本稿では、まずこれまでの HPSS に関する著者らの研究を紹介し、さらに、従来の HPSS ではごく局所的な滑らかさのみを考えていたに対し、より長距離の滑らかさを考慮するように HPSS を拡張することについて述べる。このような拡張を行なうことにより、HPSS を従来よりも高速化することが可能であることを実験的に示し、さらに、当初は具体的な式の上では関連性が乏しいように思われた著者らの HPSS と FitzGerald [3] の方法との間にも実は類似性がある程度明らかになることについても述べる。

2. 調波打楽器音分離 (HPSS)

本節では著者らがこれまでに開発した手法である HPSS について、その概要を紹介する。以下の内容は概ね文献 [8, 11] のものに基づくが、記法などは改めている。

2.1 音楽信号の時間周波数表現

まずモノラル信号 $x(t)$ を時間周波数表現 $x(t, f)$ に変換することを考える。音楽においては、各音の鳴っている時刻および音高の情報が重要なので、時間 t と周波数 f を変数としてとれるような 2 次元平面上に表現するのは自然である。本稿では最もシンプルな時間周波数表現のひとつとして短時間 Fourier 変換を考える。なお、以降の議論は短時間 Fourier 変換のみに限定されるわけではなく、信号を時間と周波数の 2 次元平面上に表現するもので、かつ逆変換が可能ならば、基本的にはどのような表現でも利用できる可能性があり、例えばウェーブレット変換を利用するというアプローチも考えられる。

本稿では、音楽信号 $x(t)$ の短時間 Fourier 変換を $(\tilde{X}_{n,k})_{1 \leq n \leq N, 1 \leq k \leq K} \in \mathbb{C}^{N \times K}$ とし、その要素ごとの絶対値の γ 乗 $X_{n,k} := |\tilde{X}_{n,k}|^\gamma$ をスペクトログラムと呼ぶことにする。添字 $n, k \in \mathbb{N}$ はそれぞれ時間と周波数を示し、 N は曲の長

さ、 K は Nyquist 周波数に対応する。なお絶対値ではなくその γ 乗を考えるのは、以降で定義する最適化問題において、音量の大きな音に解が過剰に適合してしまうことを防止するための処置で、 γ の値はおよそ 0.3 から 0.5 程度の値とする。また、本稿ではスペクトログラムに関する四則演算、指数、および不等式の評価は要素ごとに行なうものとする。すなわち、 $XY = (X_{n,k}Y_{n,k})$, $X^3 = (X_{n,k}^3)$, $X \leq 1 \Leftrightarrow \forall(n, k), X_{n,k} \leq 1$ などである。

2.2 HPSS のアイデア

音楽信号のスペクトログラム Y を図 1(a) に示す。このスペクトログラムは概ね水平な（時間方向に連続的な）成分と鉛直な（周波数方向に連続的な）成分からなる格子状の模様として表現されていることが観察できる。このうち、水平な成分と鉛直な成分はそれぞれ異なる種類の楽器音に由来している。前者は概ねピアノなどの調波楽器音に由来していると考えられる。なぜならば、これらの楽器音は少数の正弦波の和として表現され、かつこれらの各正弦波成分は、スペクトログラムの時間分解能 (10–100[ms] オーダー) と比べても十分に長い時間 (100–1000[ms] オーダー) 持続することが多いからである。一方後者は概ね打楽器に由来していると考えられる。これは、打楽器音は概ねインパルス的な信号であり、瞬間的に広い帯域を占めるという性質があるためである。

このように、調波楽器音と打楽器音は、スペクトログラム上で滑らかさの方向が異なっていて、前者は水平（時間）方向に、後者は鉛直（周波数）方向にそれぞれ滑らかである。HPSS で考えるのはこの逆である。すなわちスペクトログラム (a) を水平な成分 (b) と鉛直な成分 (c) とに分離するとそれぞれが調波楽器音と打楽器音とに概ね相当するはずであると期待し、そのような分離手法を考える。以上が HPSS の基本的なアイデアである。

2.3 HPSS の最適化問題としての具体的定式化

HPSS のアイデアを具体的に実現するには様々な方法がありうるが、Miyamoto [9] 以降の研究ではこの問題は最適化問題として定式化されている。この最適化問題の定式化においては、主に以下の4点が要請される。

- (1) 調波楽器音のスペクトログラム H と打楽器音のスペクトログラム P は、それぞれ時間方向、周波数方向にできるだけ滑らかでなければならず、そのようなものを「良い」解としなければならない。
- (2) 分離されたスペクトログラム H, P を足すとできるだけもとのスペクトログラム Y に戻らねばならず、そのようなものを「良い」解としなければならない。
- (3) スペクトログラムが負の値をとってはならない。
- (4) 入力信号を r 倍したらに出力信号も r 倍にならなければならない。

これらのうち、(1) と (2) は最適化問題における目的関数を、これらの条件を満足するように設計すればよい。要請 (3) は実行可能領域として扱えばよい。要請 (4) を満たすには目的関数を設計する際に工夫して、目的関数が冪次式になるようにすればよい。以下 (1), (2) について具体的に述べる。

まず (1) に関して、著者らはこれまで滑らかさに関するコスト関数として、以下のようなスペクトログラム隣接成分同士の差の2乗和を用いている。

$$S_{\text{time}}(\mathbf{H}) = \sum_{n=1}^N \sum_{k=1}^K (H_{n,k} - H_{n-1,k})^2 \quad (1)$$

$$S_{\text{freq}}(\mathbf{P}) = \sum_{n=1}^N \sum_{k=1}^K (P_{n,k} - P_{n,k-1})^2 \quad (2)$$

スペクトログラム H, P が時間方向、周波数方向にそれぞれ滑らかであれば、 $S_{\text{time}}(\mathbf{H}), S_{\text{freq}}(\mathbf{P})$ は小さい値をとることが確認できる。HPSS では逆に、これらを最小化するような H, P がそれぞれ調波楽器音と打楽器音に概ね相当することを期待している。

次に (2) に関しては、 $H + P$ ともとのスペクトログラム Y との隔たりをコストとする。スペクトログラムの隔たりを測る指標としては様々なものが知られていて、代表的なものとして一般化 Kullback-Leibler ダイバージェンス、Itakura-Saito ダイバージェンスや、それらを一般化した β ダイバージェンスが挙げられるが、文献 [8, 11] では一般化 KL ダイバージェンス

$$D_{\text{KL}}(\mathbf{X} \parallel \mathbf{Z}) := \sum_{n,k} X_{n,k} \log \frac{X_{n,k}}{Z_{n,k}} - X_{n,k} + Z_{n,k} \quad (3)$$

を用いて、 $D_{\text{KL}}(\mathbf{Y}^2 \parallel \mathbf{H}^2 + \mathbf{P}^2)$ をコスト関数としている。なお、ここでスペクトログラムを2乗した値をとっているのは要請 (4) を満たすためである。

以上のコスト関数を用いて目的関数 J は

$$J(\mathbf{H}, \mathbf{P}; \mathbf{Y}) = S_{\text{time}}(\mathbf{H}) + wS_{\text{freq}}(\mathbf{P}) + \mu D_{\text{KL}}(\mathbf{Y}^2 \parallel \mathbf{H}^2 + \mathbf{P}^2)$$

と定義されている。ここに $w, \mu \geq 0$ はそれぞれの項への重みである。この最適化問題の実行可能領域は、要請 (3) より $H \geq 0, P \geq 0$ である。

この最適化により得られたスペクトログラムの各成分を $1/\gamma$ 乗し、元の信号の短時間 Fourier 変換 $\tilde{Y} = |\tilde{Y}|e^{j\phi_{n,k}}$ の偏角 $e^{j\phi_{n,k}}$ を利用して得られる $\tilde{H}_{n,k} = H_{n,k}^{1/\gamma} e^{j\phi_{n,k}}, \tilde{P}_{n,k} = P_{n,k}^{1/\gamma} e^{j\phi_{n,k}}$ を逆短時間 Fourier 変換すると、信号 $h(t), p(t)$ を合成することができる。以上により、音楽信号 $y(t)$ から調波楽器音の推定信号 $h(t)$ と打楽器音の推定信号 $p(t)$ が得られる。

2.4 最適化問題の解法

本節では J を最適化するアルゴリズムを導出する。まず、 $H_{n,k}$ 以外の成分を実行可能領域内の適当な値で固定し、 J を $H_{n,k}$ のみの関数とみなして最適化することを考える。すると、 J は $H_{n,k}$ に関して下に凸だから

$$\partial J(H_{n,k}) / \partial H_{n,k} = 0 \quad (4)$$

であるような $H_{n,k}$ を求めればよいことがわかる。式 (4) は $H_{n,k}$ に関する3次方程式に帰着する。これを解くことは容易だが、計算コストはやや大きい。そこで文献 [8, 11] ではこの問題を2次方程式に帰着させるために補助関数法 [10] と呼ばれる方法をとっている。これは、 $J \leq J^+$ を満たして、かつ微分したときに都合のよい形になる関数 J^+ をひとつ適当に定めて、 J^+ を最小化することにより J も最小化するという方法である。ここでは D_{KL} に関する以下の不等式を利用して J^+ を設計する。なおこの不等式は \log の凸性と Jensen の不等式から容易に示せる*1。等号条件は $\theta = \mathbf{H}^2 / (\mathbf{H}^2 + \mathbf{P}^2)$ である。

$$D_{\text{KL}}(\mathbf{Y}^2 \parallel \mathbf{H}^2 + \mathbf{P}^2) \leq D_{\text{KL}}(\theta \mathbf{Y}^2 \parallel \mathbf{H}^2) + D_{\text{KL}}((1-\theta) \mathbf{Y}^2 \parallel \mathbf{P}^2),$$

where $0 \leq \theta \leq 1$ (5)

J の定義式の D_{KL} をこの不等式の右辺で置き換えたものを J^+ とすると、 $\partial J^+(H_{n,k}) / \partial H_{n,k} = 0$ は2次方程式になることがわかる。具体的には、

$$aH_{n,k} - 2bH_{n,k} - c = 0, \quad (6)$$

ここに各係数は

$$a = 2 + \mu, \quad b = \frac{1}{2}(H_{n+1,k} + H_{n-1,k}), \quad c = \mu \theta_{n,k} Y_{n,k}^2$$

である。この方程式の解は2つあるが、求めたいものは値が非負でかつ目的関数を極小にするものだから、

$$H_{n,k} \leftarrow \frac{b + \sqrt{b^2 + ac}}{a} \quad (7)$$

である。この更新により J^+ は $H_{n,k}$ に関して最小化される。

*1 $\log(px/p + qy/q) \geq p \log(x/p) + q \log(y/q), p + q = 1, p, q > 0$.

$P_{n,k}$ についても同様に J^+ を最小化する更新式を求めることができ、 μ を μ/w に、 b の右辺を $(P_{n,k+1} + P_{n,k-1})/2$ に、 $\theta_{n,k}$ を $1 - \theta_{n,k}$ にそれぞれ置き換えればよい。また、 J^+ は H, P のみならず補助変数 $\theta_{n,k}$ に関する関数にもなっており、 $\theta_{n,k}$ に関する最小化も考えることができるが、これは不等式 $J \leq J^+$ の等号条件に他ならない。すなわち、

$$\theta_{n,k} \leftarrow \frac{H_{n,k}^2}{H_{n,k}^2 + P_{n,k}^2} \quad (8)$$

とすることで、 J^+ は $\theta_{n,k}$ に関して最小化される。

これらの一連の更新において J^+ は増加しない（ほとんどの場合は減少する）から、これらの更新式を全 (n, k) 成分に対して反復的に適用することにより J^+ の値は増加しない（減少する）。また、特に $\theta_{n,k}$ を更新した直後に関しては $J^+ = J$ が成り立つのだから、実はこれらの更新において J も単調非増加である。また、 $(H_{n,k} - H_{n-1,k})^2 \geq 0, D_{\text{KL}} \geq 0$ などより J は下に有界である。したがって、適当な実行可能解（例えば入力スペクトログラムをそのまま使えばよい。すなわち、 $H = P = Y$ ）を初期値とし、十分な回数これらの更新式を反復すると J は収束する。この収束は必ずしも速くはないが、実用上はおおよそ 30 回程度更新すると十分な精度のスペクトログラム H, P を得ることができる。以上が HPSS の解法である。

3. 長距離滑らかさを考慮した HPSS

3.1 モデルの拡張と目的関数

前節で見たように、従来の HPSS ではスペクトログラムの隣接する成分同士の差に基づいてスペクトログラムの滑らかさを調べていた。しかし、スペクトログラムの滑らかさを調べるときに使える情報を最近傍の成分のみに限定する必要は必ずしもなく、むしろ長距離の情報を利用することで、収束が速くなったり、突発的な外れ値に頑健になるなどの可能性があるから、積極的に長距離の情報を利用することが望ましいとも考えられる。本節ではそのような拡張を行なう。

長距離の情報を利用するためのひとつの方法として、式 (1), (2) で定義した滑らかさコスト関数を、次のように拡張することを考えることができる。

$$S_{\text{time}}^{(N')}(\mathbf{H}) = \sum_{n=1}^N \sum_{k=1}^K \left\{ \frac{1}{N'} \sum_{n'=1}^{N'} (H_{n,k} - H_{n-n',k})^2 \right\} \quad (9)$$

$$S_{\text{freq}}^{(K')}(\mathbf{P}) = \sum_{n=1}^N \sum_{k=1}^K \left\{ \frac{1}{K'} \sum_{k'=1}^{K'} (P_{n,k} - P_{n,k-k'})^2 \right\} \quad (10)$$

ここに $N', K' \in \mathbb{N}$ とする。 $N' = K' = 1$ のときは従来法と一致する。この新しい滑らかさコスト関数では、従来のように最近傍 $(n-1, k)$ のみではなく、より遠く離れた成分、すなわち $(n-2, k), (n-3, k), \dots, (n-N', k)$ との 2 乗誤差も加算されている。周波数方向についても同様である。

これらの新しい滑らかさコスト関数を用いて HPSS の最適化問題を定義しなおすことを考え、新しい目的関数 $J_{N',K'}$ を以下で与える。

$$J_{N',K'} = S_{\text{time}}^{(N')}(\mathbf{H}) + w S_{\text{freq}}^{(K')}(\mathbf{P}) + \mu D_{\text{KL}}(\mathbf{Y}^2 \| \mathbf{H}^2 + \mathbf{P}^2)$$

$J_{1,1}$ は従来法の目的関数 J である。 $J_{N',K'}$ の場合も、従来法と同様に D_{KL} を不等式で上から押さえることにより補助関数 $J_{N',K'}^+$ が定義できて、同様の手順により更新式を求めることができる。

J のときと同様の方法で、 $J_{N',K'}$ を最適化するアルゴリズムを導出する。更新式を求めるには、従来法と同様、 $J_{N',K'}^+$ を各変数に関して偏微分して得られた方程式を解けばよい。ここでは計算が煩雑なので項別に調べる。まず簡単な $\mu D_{\text{KL}}(\theta \mathbf{Y}^2 \| \mathbf{H}^2)$ から調べると、この関数の $H_{n,k}$ に関する偏微分は

$$\frac{\partial}{\partial H_{n,k}} (\mu D_{\text{KL}}(\dots)) = 2\mu \left(-\frac{\theta_{n,k} Y_{n,k}^2}{H_{n,k}} + H_{n,k} \right) \quad (11)$$

である。次に $S_{\text{time}}^{(N')}(\mathbf{H})$ の偏微分を計算すると、

$$\begin{aligned} & \frac{\partial}{\partial H_{n,k}} \left\{ \sum_{n=1}^N \sum_{k=1}^K \left(\frac{1}{N'} \sum_{n'=1}^{N'} (H_{n,k} - H_{n-n',k})^2 \right) \right\} \\ &= \frac{\partial}{\partial H_{n,k}} \left\{ \frac{1}{N'} \sum_{n'=1}^{N'} (H_{n,k} - H_{n-n',k})^2 \right. \\ & \quad \left. + \frac{1}{N'} \sum_{n'=1}^{N'} (H_{n+n',k} - H_{n,k})^2 \right\} \\ &= \frac{2}{N'} \sum_{n'=1}^{N'} (H_{n,k} - H_{n-n',k}) - \frac{2}{N'} \sum_{n'=1}^{N'} (H_{n+n',k} - H_{n,k}) \\ &= 4H_{n,k} - \frac{2}{N'} \left(\sum_{n'=1}^{N'} H_{n-n',k} + \sum_{n'=1}^{N'} H_{n+n',k} \right) \\ &= 4 \left(H_{n,k} - H_{n,k}^{(N' \text{ mean})} \right). \end{aligned} \quad (12)$$

ここに

$$H_{n,k}^{(N' \text{ mean})} := \frac{1}{2N'} \left(\sum_{n'=1}^{N'} H_{n-n',k} + \sum_{n'=1}^{N'} H_{n+n',k} \right) \quad (13)$$

である。これは (n, k) の前後 $2N'$ 個の成分 $(n-N', k), \dots, (n-1, k), (n+1, k), \dots, (n+N', k)$ の平均である。以上を用いて $\partial J_{N',K'}^+ / \partial H_{n,k} = 0$ を考えると、 $H_{n,k}$ に関する 2 次方程式

$$(2 + \mu) H_{n,k}^2 - 2H_{n,k} H_{n,k}^{(N' \text{ mean})} - \mu \theta_{n,k} Y_{n,k}^2 = 0 \quad (14)$$

が得られる。この方程式の係数は、 b を除き従来法と一致し、 b を $H_{n,k}^{(N' \text{ mean})}$ で置き換えるだけで更新式 (7), (8) がそのまま使えることがわかる。この最適化アルゴリズムを以降では $\text{HPSS}(N', K')$ と記すことにする。従来法は $\text{HPSS}(1, 1)$ である。

表 1 各 $M(= N' = K')$ に関して, HPSS(M, M) の更新式を 1 回 (全変数を各 1 回ずつ更新) 適用するのに要する時間 (100 回の試行の平均値).

M	1	2	3	5	10	20
計算時間 (単位: ms)	123	125	127	132	153	203
比率 ($M = 1$ を基準)	1	1.02	1.03	1.07	1.24	1.65

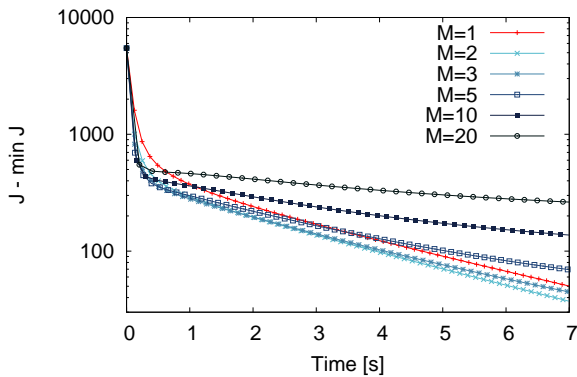


図 2 HPSS(M, M) による目的関数 $J_{1,1}(= J)$ の減少の様子. 縦軸は $J_{1,1} - \min J_{1,1}$ (最適値と現在値の差), 横軸は計算時間である. なおパラメータは $w = 1, \mu = 0.01$ とした.

3.2 計算速度に関する実験

HPSS(N', K') は従来法 HPSS(1, 1) よりも高速かどうかを実験的に検証する. なお以下の実験で用いた音楽信号は, 長さ 14 [s] でサンプリングレートが 16kHz のモノラル信号である. また, 計算時間の計測に使用した計算機は ASUS 社製のネットブック Eee PC で CPU は Intel®Atom™CPU N455 @ 1.66GHz である.

まず, HPSS(M, M) において全変数 $H_{n,k}, P_{n,k}, \theta_{n,k}$ を一回ずつ更新するのに要する時間を $M = 1, 2, 3, 5, 10, 20$ のそれぞれについて比較した結果を表 1 に示す. 一回あたりの更新式の実行時間は単純に M に比例せず, それよりも非常に高速であることが分かる. これは, 更新式のうち M に比例して計算時間が増大するのは $H_{n,k}^{(N' \text{ mean})}$ のみであり, 更新式全体の中で占める割合はごくわずかで, それ以上に平方根などの計算に多くの時間がかかるためと考えられる.

次に, 目的関数 $J_{1,1}$ がどのように減少するかを調べたグラフを図 2 に示す. HPSS(M, M) は本来は $J_{M,M}$ を最適化するアルゴリズムだから, 最終的に $J_{1,1}$ の最適解には当然収束しないが, しかし最初の数回の反復では HPSS(1, 1) よりも速く $J_{1,1}$ を減少させていることが観察できる. 特に HPSS(2, 2), HPSS(3, 3) に関しては, このグラフで示す範囲では HPSS(1, 1) よりも常に速い. なおグラフには示していないが, $J_{3,3}$ や $J_{10,10}$ を減少させる速さについて比較すれば, HPSS(M, M), $M > 1$ は従来法 HPSS(1, 1) よりもさらに有利である.

4. FitzGerald の手法との関連性に関する考察

4.1 HPSS の μ の極限での振る舞いに関する考察

FitzGerald の手法との類似性に関して述べる前に, まず HPSS における μ の極限での振る舞いを簡単に調べておく. 更新式 (7) の極限をとれば容易にわかるように,

$$\lim_{\mu \rightarrow 0} \{ \text{式 (7)} \} = \frac{2H_{n,k}^{(N' \text{ mean})}}{a} = H_{n,k}^{(N' \text{ mean})} \quad (15)$$

$$\lim_{\mu \rightarrow \infty} \{ \text{式 (7)} \} = \sqrt{\theta_{n,k}} Y_{n,k} = \sqrt{\frac{H_{n,k}^2}{H_{n,k}^2 + P_{n,k}^2}} Y_{n,k} \quad (16)$$

である. 前者は (n, k) の前後 $2N'$ 個のスペクトログラムの平均で, 後者は Wiener マスクにそれぞれ相当する. この観点から考えると, HPSS の更新式は, スペクトログラムの移動平均と Wiener マスクを, 式 (7) により補間したものと解釈することもできる.

なお, 重み係数 μ を可変なパラメータとみなし, 例えば小さな μ から始めて徐々に μ を増大させることは, ペナルティ関数法などの観点から正当化しうる.

4.2 FitzGerald によるメディアンフィルタ法

著者らの研究 [11] を受けて, FitzGerald らは著者らのアイデアをよりシンプルに実現するという動機から, スペクトログラムにメディアンフィルタを適用することにより H, P を分離するという方法を提案している [3]. FitzGerald の方法の具体的な手続きは以下の通りである.

- (1) 入力スペクトログラム Y により H, P を初期化.
- (2) $H_{n,k} \leftarrow \text{median}\{H_{n+n',k} | -M \leq n' \leq M\}, M = 8$
- (3) $P_{n,k} \leftarrow \text{median}\{P_{n,k+k'} | -M \leq k' \leq M\}, M = 8$
- (4) Wiener マスク $H \leftarrow \sqrt{H^2 / (H^2 + P^2)} Y$ により H を推定. P についても同様.

この方法は, 当初は著者らと同様のアイデアを全く異なる方法により実現したものだと思われたが, 本稿で既に述べたように, 著者らの HPSS がスペクトログラムの平均値に基づく手法であることを踏まえれば, FitzGerald の手法は著者らの「平均」を「メディアン」に置き換えたものとして, 同様の観点から捉えられると期待できる. 実際, FitzGerald の手法における (1) は HPSS の初期値と同じであるし, (2), (3) は, HPSS の $\mu \rightarrow 0$ の極限 (式 (15)) と類似して, (4) は $\mu \rightarrow \infty$ の極限 (式 (16)) と同じものである.

4.3 FitzGerald の手法の解釈

逆に, FitzGerald の手法によって最適化されるような滑らかさコスト関数はどのような関数であるかを考える. そのような関数は無数にあるが, いま絶対値の和により定義される関数 $f(x) = \sum_i |x_i - x|$ は $x = \text{median}\{x_i\}$ によって最小化されることを考えると, メディアンによって最小化されるような関数の一つとして次のような関数を挙げるこ

ができる*2 .

$$S_{\text{time}}(\mathbf{H}) = \sum_n \sum_k \left\{ \frac{1}{N'} \sum_{n'=1}^{N'} |H_{n,k} - H_{n-n',k}| \right\}$$

この関数の HPSS の滑らかさコスト関数との違いは、HPSS では 2 乗和を考えているのに対し、この関数では絶対値の和を考えているという点である。この滑らかさコスト関数を利用した新しい目的関数

$$\mathcal{J} = S_{\text{time}}(\mathbf{H}^\xi) + w S_{\text{freq}}(\mathbf{P}^\xi) + \mu D_{\text{KL}}(\mathbf{Y}^\xi \| \mathbf{H}^\xi + \mathbf{P}^\xi)$$

は、 $H_{n,k}$ について微分できないから HPSS と全く同様の方法で更新式を導出することはできないが、適当な近似や不等式による評価により、やはり同様の考え方で更新式を導出することは可能で、極限での振る舞いは HPSS と同様、 $\mu \rightarrow 0, \infty$ でそれぞれメディアン、Wiener マスクに漸近するようにすることができる。

4.4 一般の ℓ_p ノルムへの拡張の展望

HPSS は近隣の成分同士の差の 2 乗和 ($\|\bullet\|_2^2$) に基づき、FitzGerald の手法は同じく絶対値の和 ($\|\bullet\|_1$) に基づくということが明らかになったが、これをさらに一般化して、 $\|\bullet\|_p$ の場合へ拡張するとどうなるのかは興味深い問題で、特に $p < 1$ の場合が重要と思われる。そのような研究としては例えば音響信号処理の分野では文献 [19] などがある。また特に $\|\bullet\|_0$ (ℓ_0 「ノルム」) は圧縮センシング [2] などにおいて情報のスパース性を示す指標として用いられているが、HPSS においても、適当な量子化法と組み合わせることで例えば音楽信号符号化などへ応用できる可能性もあり、今後の課題として興味深いと思われる。

5. まとめ

本稿では、音楽信号を調波楽器音と打楽器音とに分離する手法である HPSS について紹介し、さらにその拡張として、長距離の滑らかさを考慮することの検討を行なった。この拡張によって HPSS の計算時間は従来よりも短縮されることが実験により示され、また、FitzGerald による同様のアイデアに基づいた手法との類似性もある程度明らかになった。今後の研究課題としては、目的関数 $J_{N',K'}$ のパラメータ N', K' はどのような値が望ましいのか、具体的なアプリケーションを想定して調べる必要がある。また、一般の ℓ_p 「ノルム」 ($0 \leq p < 1$) を利用した応用の検討なども今後の課題である。

参考文献

[1] Duong, N. Q. K., Tachibana, H., Vincent, E., Ono, N., Grignonval, R. and Sagayama, S.: Multichannel Harmonic and

*2 これを最小化する $H_{n,k}$ は厳密には $\text{median}\{H_{n+n',k} | -M \leq n' \leq M, n' \neq 0\}$ だが、 M がある程度の大きさであればメディアン値は $n' = 0$ を含むかどうかにはそれほど大きく依存しないため同一視してよいと思われる。

- Percussive Component Separation by Joint Modeling of Spatial and Spectral Continuity, *Proc. ICASSP*, pp. 205–208 (2011).
- [2] Eldar, Y. C. and Kutyniok, G.(eds.): *Compressed Sensing, Theory and Applications*, Cambridge University Press (2012).
- [3] FitzGerald, D.: Harmonic/Percussive Separation using Median Filtering, *Proc. DAFX* (2010).
- [4] Goto, M.: Development of the RWC Music Database, *Proceedings of the 18th International Congress on Acoustics (ICA 2004)*, pp. 1–553–556 (2004).
- [5] Helen, M. and Virtanen, T.: Separation of Drum from Polyphonic Music using Non-negative Matrix Factorization and Support Vector Machine, *Proc. EUSIPCO* (2005).
- [6] 北野 佑, 中野允裕, 小野順貴, 嵯峨山茂樹: スパース性と連続性と制約付き非負値行列分解に基づく調波音・打楽器音分離, 音講論(春), pp. 833–834 (2010).
- [7] 宮本賢一, 立園真理, ルルージョナトン, 亀岡弘和, 小野順貴, 嵯峨山茂樹: スペクトログラム 2 次元フィルタによる調波音・打楽器音の分離, 音講論(秋), pp. 825–826 (2007).
- [8] 宮本賢一: 音色モデルに基づいた音楽音響信号処理の研究, 修士論文, 東京大学大学院情報理工学系研究科 (2008).
- [9] 宮本賢一, 亀岡弘和, 小野順貴, 嵯峨山茂樹: スペクトログラムの滑らかさの異方性に基づいた調波音・打楽器音の分離, 音講論(春), pp. 903–904 (2008).
- [10] 小野順貴: 補助関数法による最適化アルゴリズムとその音響信号処理への応用, 日本音響学会誌, Vol. 68, No. 11, pp. 566–571 (2012).
- [11] Ono, N., Miyamoto, K., Kameoka, H. and Sagayama, S.: A Real-time Equalizer of Harmonic and Percussive Components in Music Signals, *Proc. ISMIR*, pp. 139–144 (2008).
- [12] Ono, N., Miyamoto, K., Le Roux, J., Kameoka, H. and Sagayama, S.: Separation of a Monaural Audio Signal into Harmonic/Percussive Components by Complementary Diffusion on Spectrogram, *Proc. EUSIPCO* (2008).
- [13] 橘 秀幸, 小野順貴, 嵯峨山茂樹: スペクトルの時間変化に基づく音楽音響信号からの歌声成分の強調と抑圧, 情処研報, Vol. 2009–MUS–81, No. 12 (2009).
- [14] 橘 秀幸, 水野 優, 小野順貴, 嵯峨山茂樹: 歌声抑圧手法とピッチ変換手法に基づくリアルタイム自動カラオケ生成システム Euterpe, 音講論(秋) (2012).
- [15] Tachibana, H., Kameoka, H., Ono, N. and Sagayama, S.: Comparative Evaluations of Multiple Harmonic/Percussive Sound Separation Techniques based on Anisotropic Smoothness of Spectrogram, *Proc. ICASSP*, pp. 465–468 (2012).
- [16] Tachibana, H., Ono, T., Ono, N. and Sagayama, S.: Melody Line Estimation in Homophonic Music Audio Signals based on Temporal-variability of Melodic Source, *Proc. ICASSP*, pp. 425–428 (2010).
- [17] Tsunoo, E., Ono, N. and Sagayama, S.: Rhythm Map: Extraction of Unit Rhythmic Patterns and Analysis of Rhythmic Structure from Music Acoustic Signals, *Proc. ICASSP*, pp. 185–188 (2009).
- [18] Ueda, Y., Uchiyama, Y., Nishimoto, T., Ono, N. and Sagayama, S.: HMM-based Approach for Automatic Chord Detection using Refined Acoustic Features, *Proc. ICASSP*, pp. 518–521 (2010).
- [19] Vincent, E.: Complex Nonconvex ℓ_p Norm Minimization for Underdetermined Source Separation, *Proc. ICA/BSS*, pp. 430–437 (2007).
- [20] Yoshii, K., Goto, M., Komatani, K., Ogata, T. and Okuno, H. G.: Drumix: An Automatic Player with Real-time Drum-part Rearrangement Functions for Active Music Listening, *IPJS Journal*, Vol. 48, No. 3, pp. 1229–1239 (2007).