Vol.2020-NL-246 No.32
Vol.2020-SLP-134 No.32
2020/12/3

中心周波数と極性をランダマイズしたオールパスフィルタの従属接続による時間伸長パルスの生成について

河原英紀[1,a)]　矢田部浩平[2,b)]

概要：本資料では，新しい種類の時間伸長パルス（TSP: Time Stretched Pulse）を提案する．現実に存在する（例えば）音響システムは非線形である．時間伸長パルスはそのようなシステムのインパルス応答を求める方法として広く用いられている．提案する TSP は，特別の付加装置も，習熟した専門家による処理も必要とせずに，線形時不変応答，非線形時不変応答，ランダムおよび時変成分を同時に測定することを可能にする．しかも，この新しい提案は数理的に明確で簡潔な基盤に基づいている．実際，同種の特長を有する TSP として我々が以前に提案した FVN は，この提案の特殊な場合の近似的な実装と位置付けることができる．本資料の後半では，提案する TSP に基づく対話的実時間音響測定ツールを紹介し，音声資料の再利用性を確保するための応用と，オールパスフィルタとしての性質を利用したデータ拡張への応用の可能性を紹介する．

Time stretched pulse generation based on cascaded all-pass filters with randomized center frequencies and phase polarity

1. はじめに

スピーカーやマイクなどの音響機器の性質は（線形時不変である限り）インパルス応答で一意に定まる [1–3]．時間伸長パルス（以降 TSP: Time Stretched Pulse）[1–6] は，1980 年代からこのような音響システムのインパルス応答を求める方法として広く用いられている．しかし，従来の方法では線形時不変応答以外を求める際には，フィルタなどの機器の追加や後処理が必要など，様々な問題があった [2,7–9]．この問題は，最近，高い設計自由度を有する周波数領域ベルベットノイズ (FVN: Frequency domain variant of Velvet Noise) [10,11] を用いることにより，線形時不変応答に加え，非線形時不変応答，ランダムおよび時変成分を同時に測定する方法が発明され解決された [12,13]．

しかし，FVN には幾つもの根本的な問題がある．FVN では，離散時間領域での FIR を前提として，単位インパルスの位相を（たまたま著者らが発明した）6 項の余弦級数 [14] を使って操作することで，利得が周波数に依存せずに 0 dB でありながら，瞬時値がランダムであるように見えてしかもエネルギーが時間的に局在する信号を作成している．このような信号を実現するために，人工的な残響付加のために発明されたベルベットノイズ [15,16] の設計規則を流用し，時間軸と周波数軸を入れ替えるという発見的な方法を用いている．このような経緯で発明されたため，(1) FVN は何を最適化しているのか，(2) 位相操作関数として用いている余弦級数は最適なのか，(3) 連続時間領域ではオールパスフィルタとなるのか，などが不明であり，いわば先が見通せない袋小路の状態にあった．

本資料では，新しい種類の時間伸長パルス（以下 CAPRICEP: Cascaded All-Pass filters with RandomIzed CEnter freqnencies and Phase polality）を提案することで FVN の問題を根本的に解決する [17]．CAPRICEP は，文字通り，中心周波数と位相の極性をランダムに設定したオールパスフィルタを従属接続することで実現される [17]．この新し

[1] 和歌山大学
Wakayama University, Wakayama 640–8510, Japan
[2] 早稲田大学
Waseda University, Tokyo 169–8555, Japan
[a)] kawahara@wakayama-u.ac.jp
[b)] k.yatabe@asagi.waseda.jp

ⓒ 2020 Information Processing Society of Japan

い提案は数理的に明確で簡潔な基盤に基づいている．この観点から振り返ると，FVN は提案する CAPRICEP の特殊な場合の近似的な実装であると位置付けることができる．CAPRICEP は，FVN と同じく時間領域で局在化された（TSP としての）オールパスフィルタであり，FVN に基づいて導かれた同時計測法 [12, 13] は，基盤となる TSP を CAPRICEP に入れ替えてもそのまま成立する．

CAPRICEP は，FVN の単なる置換えではない．さらに機能を拡張することを可能にする．例えば，複雑な調整と試行錯誤が必要だったためにツールには組み込んでいなかった 1) 持続時間への周波数依存性の付加，2) 時間方向に非対称な振幅分布などの機能拡張が，CPARICEP に基盤を置き換えることにより乗算だけで簡単に実装できるようになる．現在，FVN に基づいて実装された対話的実時間音響計測ツールを，基盤部分となる関数を（副作用に注意して）入れ替えることで，CAPRICEP に基づくツールにアップグレードする作業を進めている．

本資料では CAPRICEP の考え方と背景を説明することに重点を置き，具体的な数値例や実測例を用いてイメージを把握できるように工夫する．それらの理解に基づいて，様々な問題に応用して頂くため，ツールや基盤となる関数を MATLAB で実装し，公開している [18].

2. 背景：そもそも何がしたいのか

FVN は，音声合成用の音源の問題を解決するために発明された [10]．摩擦音や破裂音を（昔の）VOCODER で合成する際に，有声音の音源はパルス列，無声音の音源は雑音としていた．STRAIGHT（1999 年の，いわゆる legacy-STRAIGHT）[19] では，有声音のスペクトル包絡から周期性に起因する構造を選択的に抑圧することで，（90 年代の従来の）VOCODER による再合成音声の品質を大きく向上させることに成功したが，最初から有声音以外の合成に本質的な問題をかかえていた[*1]．

白色雑音の長時間平均スペクトルは，周波数に依存しない定数になる．しかし，短い区間を切り出すと，図 1 に示すように統計的変動により強い周波数依存性が現れる[*2]．その結果，合成音声の駆動音源として用いた場合，元の音声には無かった特有の音色が付け加えられてしまう．また，/z/のような有声摩擦音の音源をパルス列と雑音を加算して作ると，合成された音声にパルスの音色と雑音の音色が重なるような印象を与える問題もあった．そのような問題を解決できる音源が必要だった[*3]

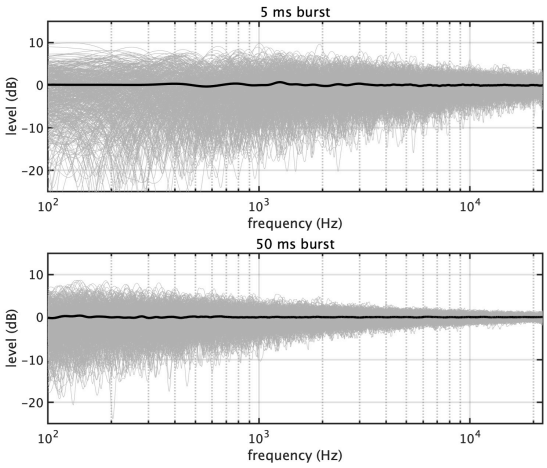

図 1 Power spectral fluctuation by noise truncation. Upper plot shows 5 ms noise burst and lower plot show 50 ms noise burst. Gray lines are 500 samples of bursts. Each power spectrum is a result of frequency smoothing with 2/15 octave. This resolution matches audiroty filter smoothing (calibrated at 1000 Hz).

2.1 ベルベットノイズから FVN へ

ベルベットノイズは，+1 と −1 の値を取るパルスを，二つの乱数を用いた規則に基づいて時間軸上に疎に配置した信号である．信号の大部分は 0 である．このパルスの発生頻度が 1 秒間に数千回を超えると，白色ガウス雑音よりも滑らかな印象のある雑音として知覚される [15, 16]．離散 Fourier 変換の一つの bin の実部あるいは虚部に注目する．すると，それぞれの値は位相をランダムに設定した等振幅の正弦波の和で表される．そのため，実部の値も虚部の値も（加算数の増加に伴って）正規分布に近づく [21]．このベルベットノイズを帯域フィルタに通せば，周波数軸上の実部あるいは虚部は，帯域フィルタの特性が乗算されることで，バースト雑音のように見える．

この解釈が FVN につながった．Fourier 変換は乱暴に言えば座標変換なので，時間軸と周波数軸を入れ替えて，周波数軸上でベルベットノイズを発生させ，それを周波数軸上でフィルタ処理し，Fourier 変換すれば時間軸上のバースト雑音を作ることができる．そのままでは，前の節で説明した雑音の切り出しによる統計的変動の影響を受ける．そこで，単位インパルスの Fourier 変換の位相だけを操作すれば，振幅周波数特性が周波数に依存しない定数であるようなバースト雑音ができる（のではないか）．この粗雑な類推を実装したものが FVN である [10, 11]．結局，FVN で実現したものは，時間的に局在するインパルス応答を持つオールパスフィルタを作るアルゴリズムであると言える．

[*1] これは他の VOCODER も同様.

[*2] 図 1 に，5 ms と 50 ms 長で切り出した白色雑音の例を示す．パワースペクトルは，聴覚での周波数分解能を近似するように 2/25 オクターブ幅で自乗平均を求めている．このような図は，今回初めて計算して求めた．何となく思い込んでいたよりも大きく変動していることに驚いている．

[*3] 複数のパルスを発生させて力づくで合成音声の品質を向上させようとする戦略もある [20]．エンジニアリングとしては妥当であり，最終的には音声符号化の国際標準にもつながっている．しかし，STRAIGHT の発明は，人間の音声知覚を理解したい（聴覚的な同値類の構造を解明したい）という問題意識に基づいている [19]．内容を理解可能な単純な要素を積み上げて，元の音声と区別できない合成音声を作ることをねらっているのであり，力づくの品質向上とは相入れない．

ⓒ 2020 Information Processing Society of Japan

2.2 FVN から CAPRICEP へ

で，2020 年の秋の音響学会の（virtual）休憩室での雑談の後で，共著者から，「オールパスフィルタを作りたいなら，オールパスフィルタをつなげば良いのでは？」という根本的なコメントをもらうことになった．この方向で議論と検討を（Slack などで）続けることで，複雑怪奇で筋が悪かった FVN は，CAPRICEP の特殊な条件の近似的な実装に相当することが明らかになり，完全に置き換えられることになった [17]．以下，こうして整理された CAPRICEP について説明する．

3. 単位 CAPRICEP の構成

以下では，CAPRICEP でアルゴリズムや実装などを含む方法全体を表すこととし，生成されるオールパスフィルタに対応するインパルス応答を単位 CAPRICEP（unit-CAPRICEP）と呼ぶことにする．以下の議論は，離散時間系で行う．

3.1 構成要素：オールパスフィルタ

FVN ではオールパスフィルタを実現するために，単位インパルスの位相のみを操作していた．位相は 2π を周期とするため，位相操作と対応するインパルス応答の対応を線形演算で結びつけることができない．そのため，複数の設計パラメタを含む余弦級数を用いた最適化を論ずることは困難であった．

CAPRICEP では解くべき問題は何かを見直した．オールパスフィルタを実現することが目的であれば，構成要素をオールパスフィルタとして，従属接続により必要な設計の自由度を得れば良い．

CAPRICEP では次式の伝達関数で表されるオールパスフィルタを構成要素とする．

$$H_k(z) = \frac{z^{-1} - z_k^*}{1 - z_k z^{-1}}, \quad (\text{なお: } H_k^*(z)H_k(z) = 1) \quad (1)$$

$$z_k = \exp\left(-\frac{\pi b_k}{f_s} + j\frac{2\pi f_k}{f_s}\right), \quad (2)$$

ここで f_s は標本化周波数を表す．位相特性の形状は帯域幅 b_k で決まり，位置は f_k で一意に決まる．FVN で必要であった位相操作関数の幅の決定や適切な操作量の調整は不要になる．

このオールパスフィルタを操作の単位とすることで，位相操作関数による場合と比べて，操作の意味が明確になる．図 2 に，オールパスフィルタの特性の例を示す．最上段の利得は，点線で表した分母の特性と分子の特性が相殺することで，実線で表した 0 dB の定数となる．ここでは中段の位相よりも，最下段の群遅延に注目する．新しい TSP で設計したいのは，時間方向の波形の振幅の形状である．群遅延はそれぞれの周波数における波形のエネルギー重心を表している．従属接続されたフィルタの群遅延はそれぞれのオールパスフィルタの和になるため，この設計の目的との関連が直接的である[*4]．しかも，オールパスフィルタの応答は指数関数的に減衰するため，形状を設計した区間外の振舞いに，副作用は生じない．

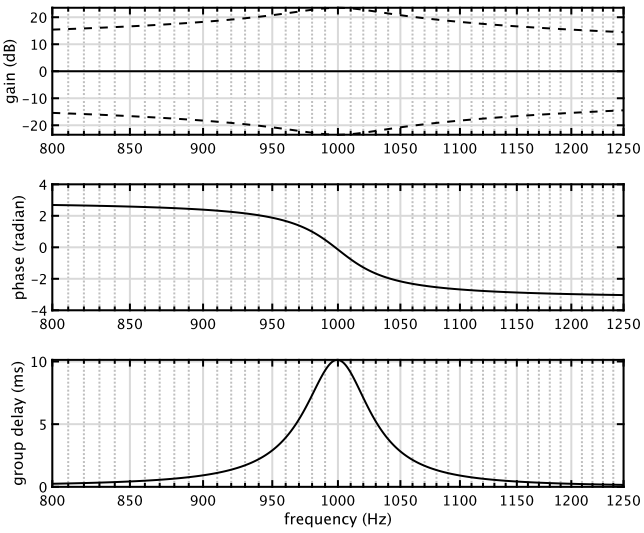

図 2 All-pass filter frequency characteristics. From top to bottom, each panels shows gain, phase, and group delay. ($f_k = 1000$ Hz, $b_k = 20$ Hz)

3.2 構成要素の組み合わせ：二組の乱数

音響機器の測定に用いる場合，試験用の信号はある区間で一定のパワーである方が都合が良い．FVN では，周波数軸の非線形変換で形状を制御する方法を見出した [11]．しかし，設計過程は間接的であり多数のパラメタが最終結果に相互に影響を与えるため，試行錯誤が必要であった．ここでは，オールパスフィルタの配置に新しい自由度を加えることにより，このような絡み合いを除いて独立に調整できるパラメタを用意することを狙う．

オリジナルのベルベットノイズでは，パルスの配置位置をランダマイズすることにより，FVN では，位相調整関数の配置位置をランダマイズすることにより，知覚的印象あるいは生成された波形の瞬時値をランダム化していた [*5]．CAPRICEP でも，中心周波数 f_k をランダムに配置する．実際にはオリジナルのベルベットノイズでも，FVN でも，（実質的に）ランダマイズしているのは，配置間隔である．設定できるのは，FVN の場合，平均配置間隔 F_d だけである．対応する確率密度関数は，図 3 の細線で示すような三角形となり，形状を設計する自由度は無い（$2F_d$ が図 3 の横軸の 1 に対応する）．ここでは，二項分布の一般化であるベータ分布を用いて設計に自由度を加える．さらに，ベータ分布の二つのパラメタ α, β に同一の値 α を設定するこ

[*4] この利点を使おうとして，FVN でも群遅延の方を操作しようと試みた．しかし，様々な副作用が生じたため位相のみを操作することにしたという経緯がある．

[*5] 乱暴な類推であるが，同振幅でランダム位相の正弦波を加えたものが正規分布に近づくことを利用 [21]．

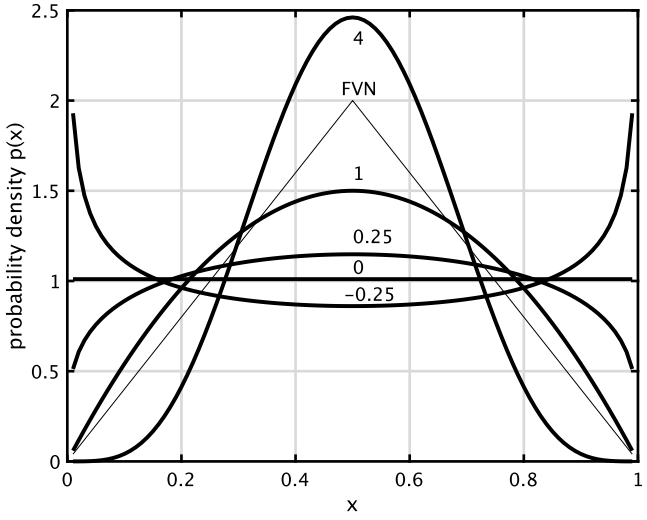

図 3 Probability density function of allocation interval.

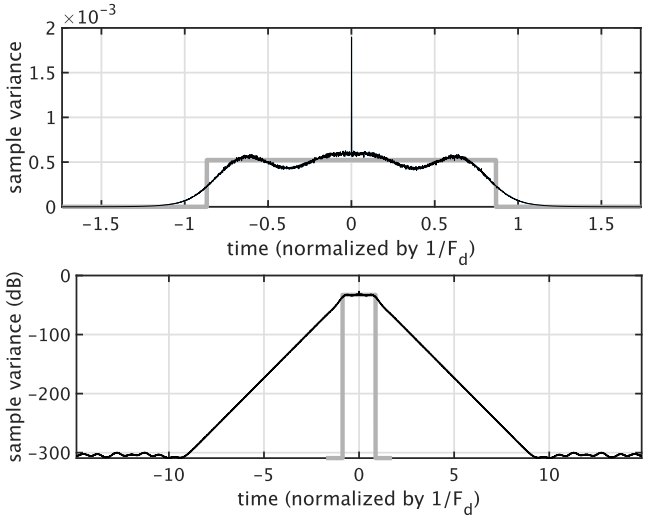

図 4 Optimized shape example

とにした．この α を追加する設計パラメタとした．α の値を変えることにより，図 3 の太線で示すように，関数の形状を，凹型から凸型に，定数関数を経由して連続的に変えることができる．

オールパスフィルタの応答は因果的であるため，従属接続で構成されるオールパスフィルタも因果的になる．ここでも FVN と同様にもう一つの二値（+1 or −1）乱数を導入する．この乱数を用いてオールパスフィルタの伝達関数の位相の符号を反転させることにより，応答の時間軸を逆転させる．このように因果応答と非因果的応答を混在させることにより，形状の設計がより自由になる．

3.3 形状の設計：最適化

ここでは形状の設計を，多数の単位 CAPRICEP から求められる標本分散の形状と目標とする形状との違いを表す距離を定義して，その距離を最小化する最適化問題ととらえる．標本分散の形状を画像とみなした場合，標本分散の曲線の下に含まれる画素は，同じ重みを持っている．確率密度関数の形状も積分すると 1 となるように拘束されているので，同様に解釈することができる．（しかもオールパスフィルタの場合には，標本分散は拘束された範囲外では指数関数的に減少するので，副作用を気にせずに変形できる．）したがって，距離として Wasserstain 距離 [22] を用いることが自然な選択となる．

矩形は，音響計測用関数に応用する際に都合の良い形状である．図 4 に，目標とする形状を矩形として設計した例を示す．図中の灰色の太線が目標とした矩形を表す．上の図では中央に鋭いスパイクが立っている．しかし，このパルスは後で説明する処理により，問題とはならない．下の図では，形状を拘束する矩形の外側で標本分散が指数関数的に直線で減少している．この減少は，数値演算の誤差による制限で生ずる背景雑音レベル −300 dB になるまで続いている．なお，最適化は予備検討により $\alpha = 8$,

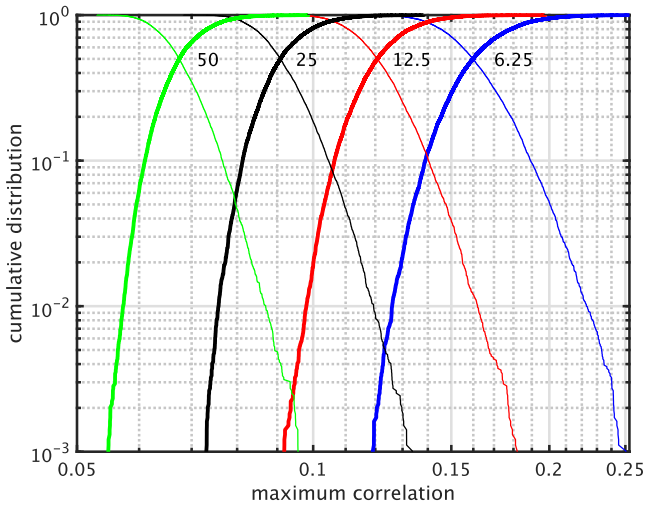

図 5 Cumulative distribution of maximum correlation between different unit-CAPRICEPs. The number placed at each crossing point represents the T_{ERD} represented in ms.

$c_{\mathrm{mag}} = 2^{\frac{1}{4}}$ を求め，矩形の時間長 T_{ERD}（ERD: Equivalent Rectangular Duration）を一次元探索した．その結果に基づけば，$F_d = 1.73/T_{\mathrm{ERD}}$ により，必要な継続時間長に応じて設計パラメタである平均配置間隔 F_d を決めることができる．設計された単位 CAPRICEP には高い設計自由度がある．標本化周波数を $f_s = 44100$ (Hz) とし，$F_d = 40$ (Hz) とした場合，単位 CAPRICEP は，1102 個の周波数と 1102 個の位相の極性により決定される．

3.4 相関と整形

単位 CAPRICEP は，高い設計自由度を持っているため，異なった単位 CAPRICEP の間の相関は低い．図 5 に例を示す．縦軸は累積分布を表し，横軸は単位 CAPRICEP 間の相関の絶対値の最大値を示している．それぞれの色の交点に置かれている数字は，設計に用いた矩形の時間長 T_{ERD}（ここでは ms で表示）を表す．交点は分布の中央値であ

ⓒ 2020 Information Processing Society of Japan

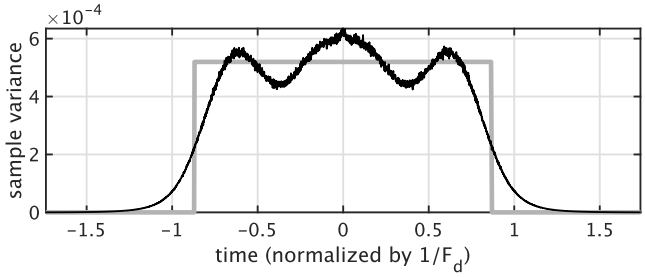

図 6 Pulse removed shape.

り，矩形の時間長の平方根に反比例している．音響計測に用いる際には，0.1 秒以上の時間長が用いられることが多く，相関はさらに低くなる．ただし，この低い相関も音響計測の際には無視できない誤差となるため，同時測定で提案した方法 [13] で消去することが必要になる．

前に触れた中央のパルスは，もう一つの持続時間が短い（0.5 ms）単位 CAPRICEP で処理することにより消去することができる．図 6 に整形された結果を示す．

4. 多属性同時測定：高い設計自由度の利用

ここでは，FVN によって最初に実現された多属性同時測定について考え方と実例を紹介する．ここで紹介する対話的実時間音響計測ツールは，この方法を応用して作られている．音声や音響信号の収録や再生，聴覚実験の際の設定の調整や確認に利用して欲しい [23]．以下では，最初に，ツールの実時間表示に使われている二系列の試験信号を用いた方法を紹介し，次に，多数属性の同時測定を紹介する．以下のの実例では FVN を使っているが，CAPRICEP でも全く同じ実例を構成することができる[*6]．以下では単位 FVN や単位 CAPRICEP を総称して，単位 TSP と呼んでおく．二系列の同時測定については，音響学会の解説でも紹介している [3]．

4.1 二系列の試験信号による同時測定

部屋や，目の前にある PC，ニアフィールドモニタなどで構成されたステレオ再生システムなど，二つの音響経路を同時に測定したいことがある．設計自由度の高い単位 TSP を組み合わせると，次のように簡単に測定することができる．さらに，測定中の背景雑音や，測定系のずれや揺れなどによる測定精度の劣化も実時間でモニタすることができる．その機能は，図 7 に示す FVN を利用して作られた対話的実時間音響測定ツールに組み込まれている．

同時測定の原理を簡単に説明する．まず，二つの異なった単位 TSP を一定間隔 T_0 で周期的に配置して，二つの周期系列を作る．一つ目の系列では，単位 TSP をそのまま繰り返し配置していく．もう一つの系列では，配置する

[*6] 例を示すために FVN を使っているのは，スケジュールの都合による．CAPRICEP を使って一貫して動作するツールとしてまとめ上げデバッグする作業には，今後数週間かかる．

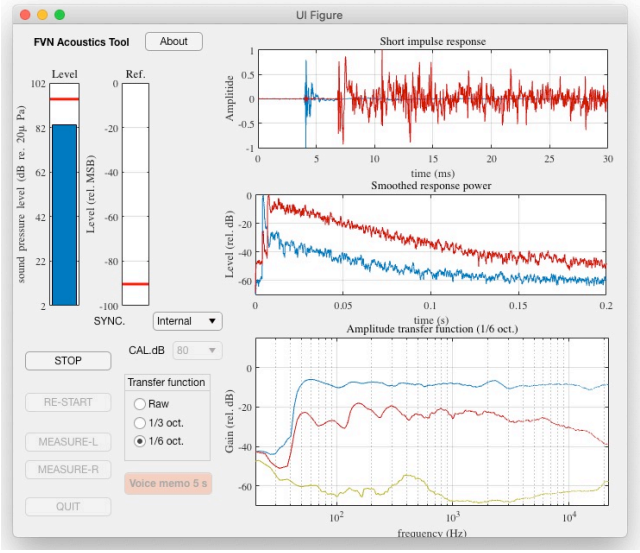

図 7 GUI snap shot of an interactive and real-time acoustic measurement tool.

単位 TSP の極性を毎回反転して，繰り返し配置していく．音響システムの左のスピーカーからは第一の系列，右のスピーカーからは第二の系列を同時に再生し，一つのマイクロフォンで受けて記録する．

こうして記録された信号と，時間反転された第一の単位 TSP を畳み込むと，時間的に拡散されていたパルスが復元される．ただし，第二の単位 TSP とは低い相関があるため，第二の系列からの影響が拡散された雑音のような形で，残る．同様なことは，第二の系列についても生じる．第二の系列では，毎回単位 TSP の符号を反転していたので，復元されたパルスは，正負正負・・・・正負を繰り返す．

ところで，第二の系列から第一の系列への影響も，同様に，毎周期ごとに極性が反転して加わっている．したがって，第一の系列から求められた（相関が残っている）パルス列のコピーを作り，一周期だけ位置を移動させて，元のパルス列との平均を求めると，相関による影響は完全に消去される．

第二の系列から求めたパルス列も，コピーを作り，今度は極性を反転して元のパルス列との平均を求める．すると，第一の系列からの相関による影響は毎回同じ極性なので，この平均操作により，相関による影響は完全に消去される．

パルスだけが残るので，音響系が線形時不変システムであれば，この処理により，同時に二つの音響経路のインパルス応答が求められることになる．図 7 の右側に示したインパルス応答などの表示は，このようにして求められている．これらでは，青線が左チャンネル，赤線が右チャンネルの応答に対応している．この表示は，単位 TSP の配置周期（このツールでは 0.2 秒）毎に更新し続けられる．

ところで右下には，黄土色の線が加えられている[*7]．こ

[*7] このツールでは，表示をみやすくするために 1/3 あるいは 1/6

れは，測定時の雑音や，機器の運動による測定誤差を表している．求め方を説明する．録音された信号と，系列の作成に用いていない別の単位 TSP を時間反転したものを畳み込む．すると，結果には，第一の単位 TSP との相関と，第二の単位 TSP との相関による成分に加えて，暗騒音など，測定信号以外の原因で生じている影響が残る．この処理された信号のコピーを 3 個作る．最初のコピーは一周期分だけ移動させる．二つめのコピーは二周期分だけ移動させて極性を反転させる．三個目のコピーは三周期分だけ移動させて極性を反転させる．こうして作成されたコピーと，元の処理された信号の計四個のコピーの平均を求める．すると，第一の系列からの影響も，第二の系列からの影響も完全に消去されて，背景雑音など，試験信号意外に起因する信号の影響だけが残る．また，測定中に音響機器の配置などが動いてしまう場合には，時間軸のずれによって消去が不完全になり，消去されずに残った成分が現れる．この成分は，背景雑音によるものと同様に，測定の精度を劣化させる．黄土色の線は，この成分を表している．実際に，このツールを使用して測定している最中に声を出して妨害したり，マイクロフォンを動かして測定を妨害すると，黄土色の線が上昇することが分かる．妨害を取り除くと，黄土色の線は，測定環境での定常状態を反映したものに落ち着く．

このツールには，音声研究に必要な音圧較正機能も用意されており，MATLAB によるソースコードや詳しい操作説明書とともに公開されている [24]．また，MATLAB を必要としないコンパイル済みのアプリケーションのインストーラへも操作説明書からリンクされている．

4.2 三系列の試験信号による複数属性の同時測定

非線形性の影響についても同時測定するためには，一つの音響経路に複数の単位 TSP から作った系列を加算して作成した試験信号を用いる．図 7 のツールの左側には，その測定のためのボタンが用意されている．

三つの系列に基づく同時測定を言葉で説明すると煩雑になる．ここでは，考え方を簡単に説明する．具体的な説明は資料 [13] を参照されたい．三個の単位 TSP を用意し，周期配置と配置の際の極性の反転操作を組み合わせると，（最初の系列は常に正の極性とするため）4 通りの極性の組み合わせが生ずる．この組み合わせの極性を反転して平均すると，線形時不変系による応答を消去することができる．また，それぞれの極性の組み合わせ毎に平均を求め，それらの違いを求めると，非線形時不変成分が求まり，試験信号の作成に用いたものとは異なる単位 TSP を用いて，八周期分の信号を利用して線形時不変応答からの相関に基づく影響を消去すると，測定時の暗騒音と，配置の運動などによる時変成分を求めることができる．また，詳細は省くが，測定結果の組み合わせにより，四周期分の長さのインパルス応答も同時に求めることができる [13].

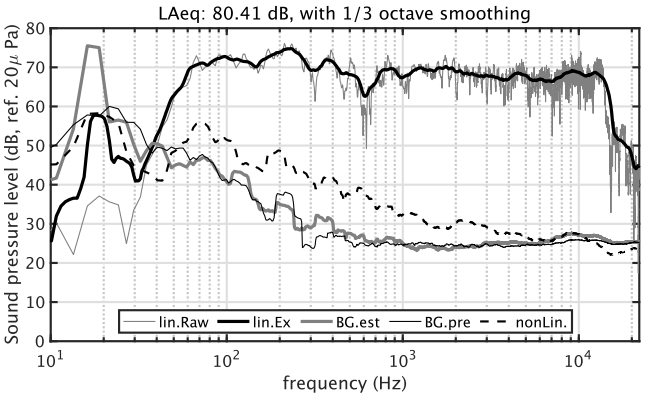

図 8 Example of a simultaneous measurement of an large powered PA system.

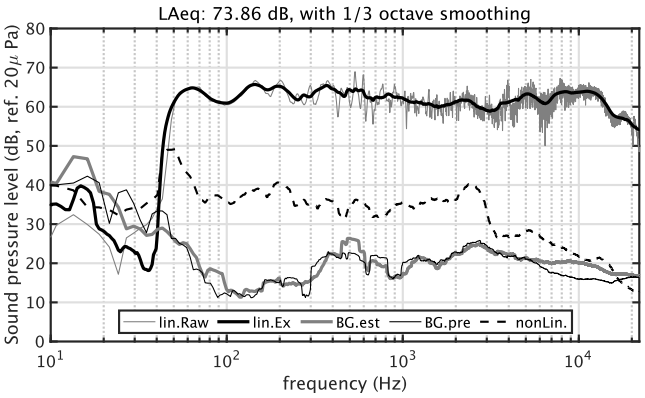

図 9 Example of a simultaneous measurement of an small powered monitor system. This measurement used a large diaphragm omni-directional condenser microhone at 50 cm distance.

図 8 に，38 cm ウーファーのパワードスピーカの測定例を示す．上の二本が線形時不変応答を示し，灰色が平滑化しない特性，黒の太線が 1/3 オクターブ幅で平滑化した特性を示す．下の二本は，太い灰色の線が，同時測定された暗騒音などのランダム時変成分を示し，細線は，試験信号を再生する 1 秒前の無音区間で求められた暗騒音を示す．中間の波線は，非線形時不変応答による．小型のパワードスピーカと比較すると，この成分は非常に少ない．なお測定は和歌山大学総合実験棟の実験室で行った．

図 9 に，小型のパワードスピーカのニアフィールドモニターを測定した例を示す．ここでは，音楽録音などに用いられるラージダイアフラムの可変指向性マイクロフォンを，単一指向性に設定して，50 mc の距離で測定した．小型のシステムのため，非線形時不変成分が多い．暗騒音は，閑静な住宅地で深夜に行ったため，かなり少ない．他にも様々なシステム，環境を測定している．それらについては，研究会で紹介したい．

オクターブ幅でパワースペクトルを平滑化して表示している．

5. まとめ

本資料では，新しい種類の時間伸長パルスである CAPRICEP を提案した．この新しい提案は数理的に明確で簡潔な基盤に基づいている．実際，同種の特長を有する TSP として我々が以前に提案した FVN は，この提案の特殊な場合の近似的な実装と位置付けることができる．時間伸長パルスはそのようなシステムのインパルス応答を求める方法として広く用いられている．提案した CAPRICEP は，FVN と同様に特別の付加装置も，習熟した専門家による処理も必要とせずに，線形時不変応答，非線形時不変応答，ランダムおよび時変成分を同時に測定することを可能にする．しかも，FVN を超える柔軟性と拡張性を有している．本資料の後半では，提案する TSP に基づく対話的実時間音響測定ツールを紹介し，音声資料の再利用性を確保するための応用について説明した．ここでは触れることができなかったが，CAPRICEP には他にも多くの応用可能性がある．例えば，音声への深層学習の応用で重要な役割を果たしている end-to-end アーキテクチャ [25] について，オールパスフィルタとしての性質を利用したデータ拡張への応用の可能性がある [17]．CAPRICEP のコードと，それを応用したツールは，第一著者の GitHub リポジトリで公開している [18].

参考文献

[1] 佐藤史明：はじめてのインパルス応答計測，日本音響学会誌， Vol. 67, No. 4, pp. 155–162 (2011).
[2] 金田　豊：招待講演 音響インパルス応答測定用信号について (信号処理)，電子情報通信学会技術研究報告，Vol. IEICE-115, No. 521(EA), pp. 13–18 (2016).
[3] 河原英紀：インパルス応答の基本概念，日本音響学会誌，Vol. 76, No. 3, pp. 148–155 (2020).
[4] Schroeder, M. R.: Integrated-impulse method measuring sound decay without using impulses, *J. Acoust. Soc. Am.*, Vol. 66, No. 2, pp. 497–500 (1979).
[5] Aoshima, N.: Computer-generated pulse signal applied for sound measurement, *J. Acoust. Soc. Am.*, Vol. 69, No. 5, pp. 1484–1488 (online), DOI: 10.1121/1.385782 (1981).
[6] Müller, S. and Massarani, P.: Transfer-function measurement with sweeps, *J. Audio Eng. Soc.*, Vol. 49, No. 6, pp. 443–471 (2001).
[7] Farina, A.: Simultaneous measurement of impulse response and distortion with a swept-sine technique, *Audio Eng. Soc. Conv. 108*, AES (2000).
[8] Ochiai, H. and Kaneda, Y.: A Recursive Adaptive Method of Impulse Response Measurement with Constant SNR over Target Frequency Band, *J. Audio Eng. Soc*, Vol. 61, No. 9, pp. 647–655 (online), available from ⟨http://www.aes.org/e-lib/browse.cfm?elib=16933⟩ (2013).
[9] Burrascano, P., Laureti, S., Ricci, M., Terenzi, A., Cecchi, S., Spinsante, S. and Piazza, F.: A swept-sine pulse compression procedure for an effective measurement of intermodulation distortion, *IEEE Trans. Instrum. Meas.*, Vol. 69, No. 4, pp. 1708–1719 (2019).
[10] Kawahara, H., Sakakibara, K., Morise, M., Banno, H., Toda, T. and Irino, T.: Frequency domain variants of velvet noise and their application to speech processing and synthesis, *Proc. Interspeech 2018*, Hyderabad, India, pp. 2027–2031 (2018).
[11] 河原英紀：[招待講演]Velvet noise と周波数領域 Velvet noise–偶然から生まれた奇妙な信号が音と聴覚の再考を促す–，電子情報通信学会技術研究報告（応用音響），Vol. IEICE-119, No. 306, pp. 69–74 (2019).
[12] 河原英紀，榊原健一，水町光徳，森勢将雅，坂野秀樹：周波数領域 velvet noise を用いた音響計測の拡張，電子情報通信学会技術研究報告（応用音響），Vol. IEICE-120, No. 118, pp. 35–42 (2020).
[13] Kawahara, H., Sakakibara, K., Mizumachi, M., Morise, M. and Banno, H.: Simultaneous measurement of time-invariant linear and nonlinear, and random and extra responses using frequency domain variant of velvet noise, *arXiv:2008.02439* (2020). (Accepted for APSIPA ASC 2020).
[14] Kawahara, H., Sakakibara, K., Morise, M., Banno, H., Toda, T. and Irino, T.: A new cosine series antialiasing function and its application to aliasing-free glottal source models for speech and singing synthesis, *Proc. Interspeech 2017*, Stocholm, pp. 1358–1362 (2017).
[15] Järveläinen, H. and Karjalainen, M.: Reverberation Modeling Using Velvet Noise, *Proc. AES 30th Int. Conf.*, pp. 15–17 (2007).
[16] Välimäki, V., Lehtonen, H. M. and Takanen, M.: A Perceptual Study on Velvet Noise and Its Variants at Different Pulse Densities, *IEEE Trans. Audio, Speech, Lang. Process.*, Vol. 21, No. 7, pp. 1481–1488 (online), DOI: 10.1109/TASL.2013.2255281 (2013).
[17] Kawahara, H. and Yatabe, K.: Cascaded all-pass filters with randomized center frequencies and phase polarity for acoustic and speech measurement and data augmentation, *arXiv:2010.13185* (2020).
[18] Kawahara, H.: CAPRICEP: Cascaded all-pass filters with randomized center frequencies and phase polarity, an extended TSP, GitHub (online), available from ⟨https://github.com/HidekiKawahara/CAPRICEP⟩ (accessed 30.Oct.2020).
[19] Kawahara, H., Masuda-Katsuse, I. and de Cheveigne, A.: Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction, *Speech Commun.*, Vol. 27, No. 3-4, pp. 187–207 (1999).
[20] Atal, B. and Remde, J.: A new model of LPC excitation for producing natural-sounding speech at low bit rates, *ICASSP '82. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 7, pp. 614–617 (1982).
[21] Lyon, R. H.: Statistics of Combined Sine Waves, *J. Acoust. Soc. Am.*, Vol. 48, No. 1B, pp. 145–149 (1970).
[22] Vallender, S. S.: Calculation of the Wasserstein Distance Between Probability Distributions on the Line, *Theory Probab. Appl.*, Vol. 18, No. 4, pp. 784–786 (online), DOI: 10.1137/1118101 (1974).
[23] 榊原健一，河原英紀，水町光徳：利用価値の高い音声データの録音手順，日本音響学会誌，Vol. 76, No. 6, pp. 343–350 (2020).
[24] Kawahara, H.: FVN: Frequency domain variant of Velvet Noise and resources, GitHub (online), available from ⟨https://github.com/HidekiKawahara/FVN⟩ (accessed 30.Oct.2020).

ⓒ 2020 Information Processing Society of Japan

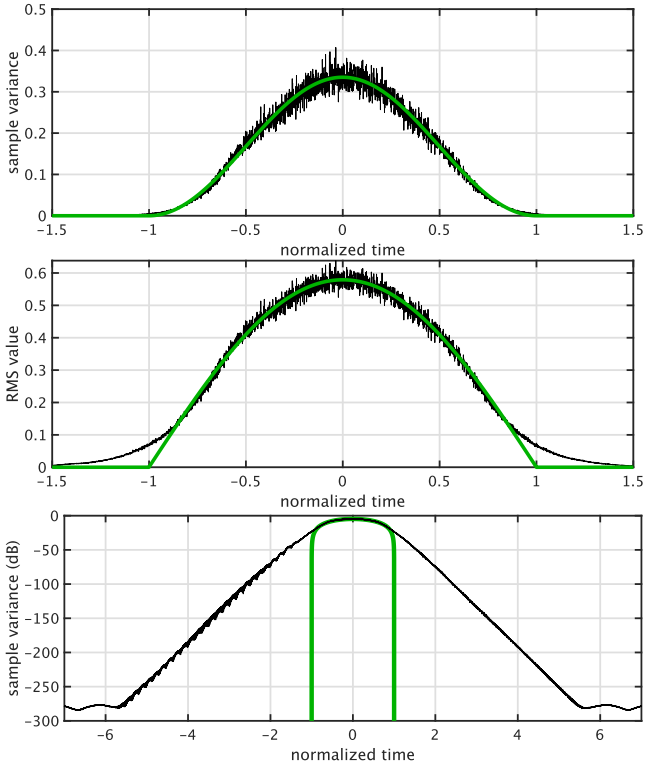

図 A·1 Shape design example: the target is raised cosine shape.

[25] Purwins, H., Li, B., Virtanen, T., Schlüter, J., Chang, S. and Sainath, T.: Deep Learning for Audio Signal Processing, *IEEE J. Sel. Top. Signal Process.*, Vol. 13, No. 2, pp. 206–219 (online), DOI: 10.1109/JSTSP.2019.2908700 (2019).

付　　録

A.1 形状設計例：目標，raised cosine

標本分散の形状の合理的な目標として raised cosine がある．この形状にしておけば，持続時間の 50%の移動幅で周期的に加算することで，標本分散を時間に依存しない定数にすることができる．設計例を図 A·1 に示す．緑色が目標の形状である．上から標本分散，二乗平均値，対数表示を示す．矩形を目標とした場合と同様に，拘束した範囲外は指数関数的に（対数表示では直線で）減衰する．音響計測に用いる際には，設計に用いた持続時間の 3 倍の長さで切り出せば十分である．24bit 量子化のフォーマットをフルに利用したい場合には，持続時間の 4 倍にすれば良い．リアルタイムの応用であれば，持続時間の 2 倍の長さでの切り出しが適している．1.5 倍でも十分かも知れない．

図 A·2 に，この形状設計に用いたコードを示す．外部の関数として呼んでいるのは，単位 CAPRICEP を生成する `generateUnitCAPRICEP` だけである．Wasserstein 距離 [22] は，`wsMeasure` に代入する一本の式で計算されている．

図 A·3 に，この最適化の過程で出力されたコストを示す．作業用なので粗いグリッドサーチで探索している．コストは 500 本の単位 CAPRICEP から求められた標本分散に目標関数を伸縮して最も近づけたときの Wasserstein 距離である．

```
fs = 44100;
npFFT = 15;
avFd = 40;
magList = 2 .^ (0:1/2:3);
abList = 2:10;
iteration = 400;
figure
wsMap = zeros(length(magList), length(abList));
for ii = 1:length(magList)
    mag = magList(ii);
    for kk = 1:length(abList)
        rmsEnvTmp = zeros(2^npFFT, 1);
        for jj = 1:iteration
            outp = generateUnitCAPRICEP(fs, avFd, ...
                mag, abList(kk), npFFT);
            rmsEnvTmp = rmsEnvTmp + outp.xTSP .^ 2;
        end
        durRef = 1.4/avFd/sqrt(mag);
        totalEngy = sum(rmsEnvTmp);
        durList = (round(durRef * fs):round(3 * durRef * fs))';
        ctrIdx = outp.fftl / 2;
        tx = outp.tx;
        wsMeasure = zeros(length(durList), 1);
        for jj = 1:length(durList)
            idx = ctrIdx + (-durList(jj):durList(jj));
            targetWTmp = hanning(round(length(idx)/2));
            targetWTmp = targetWTmp / sum(targetWTmp);
            targetW = idx * 0;
            targetW(round(length(idx)/4) ...
                + (1:length(targetWTmp))) = targetWTmp';
            wsMeasure(jj) = sum(abs(totalEngy * targetW' ...
                - rmsEnvTmp(idx))) ...
                + sum(rmsEnvTmp(tx < tx(idx(1)) ...
                | tx > tx(idx(end))));
        end
        wsMap(ii, kk) = min(wsMeasure);
        loglog(magList, wsMap', 'o-'); grid on;
        set(gca, 'xlim', [magList(1) magList(end)])
        drawnow;
    end
end
```

図 A·2 MATLAB script for shape optimization.

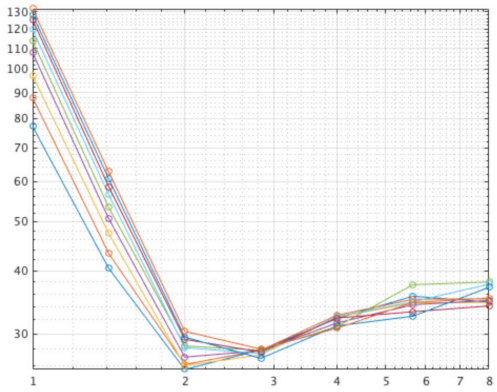

図 A·3 Cost output during optimization process.