

歪みエフェクタのモデリングにおける 線形ブロックの窓関数の影響

M2016SC017 杉浦至

指導教員：奥村康行

1 はじめに

エレキギターを演奏する際、エフェクタやアンプの増幅回路で、信号を歪ませることがある。エフェクタやアンプは様々な種類があるが、ヴィンテージと呼ばれる、古い個体が存在し、それでしか出せない音がある。それは、多くのギタリスト、アーティストから人気があるが、需要に対しての供給が減り続けている。

そこで、最近では、ヴィンテージのエフェクタやアンプを後世に残すため、デジタルモデリングが行われている。従来の手法としては、リバーブエンジニアリングがある。この手法は、モデリングコストが高く、良質なデジタルモデリングがされた商品は、非常に高価である。歪み回路をデジタルモデリングする方法として、アナログ回路を数学的モデルに変換する方法がある。この方法では、回路とそれを特定する部品のすべてをモデリングしていることより、人の聴覚でオリジナルとデジタルモデリングの音色を聴き分けるのは難しい。それほどに高精度な方法だが、この方法では、モデリングを行うために、回路図が使われている非線形回路要素の特性を知る必要がある。それより、回路ひとつをモデリングするのに必要な時間的なコストが非常に大きいという問題点がある。

歪み回路をモデリングする手段として、モデリングコストを抑えるために、ブロック志向モデルとしてモデリングを行うことが提案されている [1]。この方法は、回路を線形ブロックと非線形ブロックによるものとみなし、具体的な回路の特性を知らなくても、オリジナルに似た出力が得られるように、モデリングを行う。しかし、この方法でもモデルが複雑になるという問題がある。この問題を解決するために、ブロック毎にパラメトリックなモデリングをする方法が提案されている [2]。更にその方法を実装し、結果をまとめた論文では、現在のモデリングの精度、問題点が挙げられている [3]。

先行研究 [3] では、文献 [2] のシステムを実装し、その精度について考察が行われている。本研究では、先行研究 [3] で行われた実験の結果より、線形ブロック部分の高精度化を目標に研究を行う。そのため、次の章では、先行研究 [3] のモデルについて説明する。

線形ブロック部分のシミュレーション、実験においては、

- 信号の外形がモデリング対象とほぼ同じ
- モデリングしたフィルタの出力は、比較すると、振幅が少し大きい

という結果が得られている。振幅が大きくなってしまふのは、モデリングを FIR フィルタによって行っているた

め、高周波成分を十分に抑制できなかったと考えた。ここに、適切な窓関数を乗算することによって、周波数特性を満足させ、かつ、カットオフ周波数付近のリプルを低減させ、高精度化することができるのではないかと考えた。

そこで、本研究では、モデリングした FIR フィルタのインパルス応答に対して、窓関数を乗算した。本紙には、シミュレーション結果と実験結果を記し、考察を行った。

2 モデルの概要

本研究では、文献 [3] のシステムを基に、線形部分の高精度化をすることを目標とする。文献のモデルは、拡張ウィーナーモデルである。これは、線形ブロック、それに続く非線形ブロックから成り立つシステムである。モデルの概要を図 1 に示す。

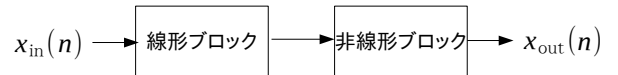


図 1 ウィーナーモデルの概要

このモデルは、歪み回路に特化して制約を与えているため、古典的な意味でのウィーナーモデルではない。線形ブロックは、回路の中で、イコライザの役割をもつ部分を指す。非線形ブロックは、マッピング関数より、入力サンプルが非線形方程式によって処理され、歪んだ信号を出力する。こちらは、回路の中で、波形を歪ませる役割を持つ部分である。

2.1 線形ブロックのモデリング [3]

線形ブロックは、複数の周波数フィルタを通過し、イコライザとしての役割を持つものと考ええる。そこで、周波数特性を測定するため、スウィープ信号をモデリングするエフェクタに入力する。そして、その出力信号を測定することで、線形ブロックのモデリングをする。スウィープ信号は、連続的に周波数を変化させた信号を指す。この信号ひとつで、一度に任意の周波数帯域の特性を測定することができる。変化の開始の周波数を f_{start} 、終わりの周波数を f_{stop} 、サンプリング周波数を f_s 、とした時、

$\omega_1 = 2\pi f_{start}/f_s$, $\omega_2 = 2\pi f_{stop}/f_s$ 、振幅を A 、信号の長さを L 、とすると今回用いるスウィープ信号は、

$$x_{sw} = A \cdot \sin \left(\frac{\omega_1 \cdot (L-1)}{\log(\omega_2/\omega_1)} \cdot \left(e^{\left(\frac{n}{L-1} \log(\omega_2/\omega_1) \right)} - 1 \right) \right) \quad (1)$$

とわすことができる．この時，用意するスイープ信号の振幅を小さくすることで，非線形ブロックの影響を少なくすることができる．

スイープ信号に対して

$$x_{inv}(n) = x_{sw}(L-1-n) \cdot \left(\frac{\omega_1}{\omega_2}\right)^{\frac{-n}{L-1}} \quad (2)$$

という信号を考える．式 (1) と式 (2) の畳み込みは

$$x_{inv}(n) * x_{sw} \approx c \cdot \delta(n - n_0) \quad (3)$$

となり， c は定数である．ここで求めた定数を用いて，式 (4) の値を決定する．式 (1) のスイープ信号を，モデリング対象のエフェクタに入力し，そこで得られた出力を y_{sw} とすると， y_{sw} と式 (2) の信号の畳み込みからインパルス応答を計算することができる．それを式として表すと，

$$h(n) = \frac{1}{c} \cdot x_{inv}(n) * y_{sw} \quad (4)$$

となる． $h(n)$ は，このまま FIR フィルタの係数として利用することができる．この FIR フィルタを作成することで，線形ブロックのモデリングとする．

2.2 窓関数による高精度化 [4]

線形ブロックのモデリングにおいて，インパルス応答 $h(n)$ から，FIR フィルタを作成する．しかし，有限のインパルス応答を用いて，フィルタを作成すると，カットオフ周波数付近の振幅特性にリプルが発生し，仕様を満足しなくなる．そこで，有限インパルス応答に対して窓関数をかけることによって，新たなインパルス応答を求める．それによって，モデリングによって作成したフィルタの精度を向上させようと考えた．

有限のインパルス応答は， n をタップ数と呼び，その長さが大きくなるほど，FIR フィルタの特性は目標に近づく．限られたタップ数の中で，より特性を満足させるために，窓関数を乗算する．窓関数は，タップの係数に重み付けをする．中央から端に行くほど係数を小さくする．この重み付けのパラメータを窓関数と呼ぶ．今回用いた窓関数は，ハミング窓，ブラックマン窓，カイザー窓と呼ばれる窓関数である．以後，シミュレーション，実験の結果それぞれについて述べる．

3 シミュレーション上の実験

本研究では，先行研究 [3] の結果を基に，線形ブロック部分の高精度化を行うことが目標である．そこで，自分の環境でも，同様の結果が得られるのかどうか，検証するため，計算機上で線形ブロックのモデリング実験を行った．

実際に動作を確認するため，既知の線形ブロックを用いて，モデリングを行った．今回実験で用いたスイープ信号は，人間の可聴域を考慮し，20[Hz] から始まり 20[kHz] で終わるものとする．図 3 に，式 (1) において，

表 1 実験環境

OS	Windows7 Professional
ソフトウェア	Matlab R2018a
CPU	AMD PhenomII x6 1090T
メモリ	8GB

$f_s = 44100, A = 1, L = 88200$ に指定した時，88200 サンプル分，2 秒のスイープ信号を示す．

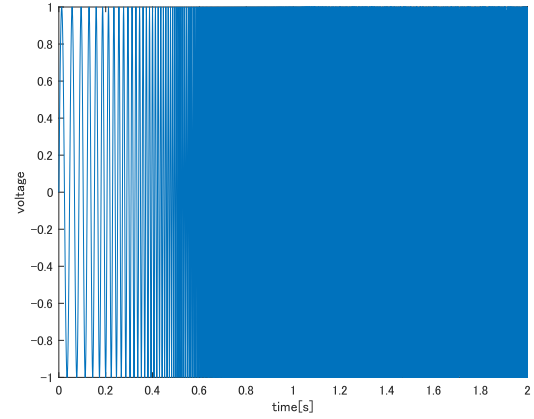


図 2 スweep信号

図 1 の信号を，既知のフィルタに通過させ，出力信号を取り出した．既知のフィルタは，ローパスフィルタとハイパスフィルタを通過させ，を， $F_s/4$ より低い，高い信号成分を減衰させるように設計した．

このフィルタを通過した信号から，式 4 を計算することで，インパルス応答を求め，既知のフィルタのモデリングを行った．そして，求めたインパルス応答からフィルタを作成し，モデリングしたものにも，同じ入力信号を与えた．

ハイパスフィルタを通過した信号を，更にローパスフィルタに通過させ，その出力信号に対して，モデリングを試みた．ハイパスフィルタ，ローパスフィルタ共に，7 次のバダワースフィルタを用いており， $F_s/4$ より低い，高い成分を減衰させるように設計した．

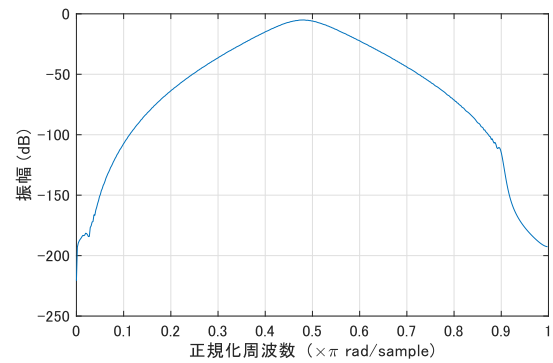


図 3 モデリングした線形ブロックの特性

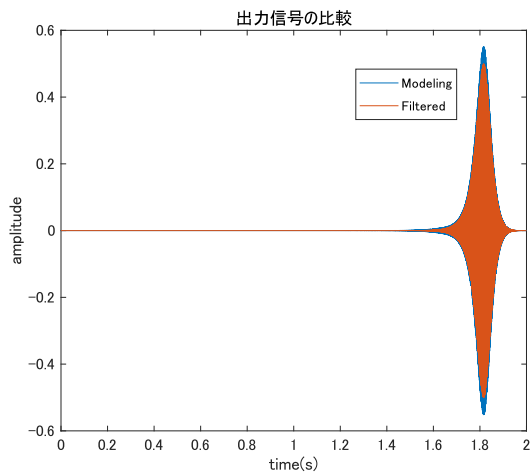


図 4 出力とモデリング出力の比較

図 3 はモデリングした、フィルタの特性を表示した図である。そして、図 4 は出力信号と、モデリングした出力信号の比較である。出力信号を比較したのを見ると、信号の外形をモデリングし、出力できていることがわかる。しかし、モデリングしたフィルタを通過させた信号の振幅が、最大 0.05 ほど大きい。今回は、ハミング窓、ブラックマン窓、カイザー窓の 3 種類の窓関数をそれぞれ、インパルス応答に対して乗算した。

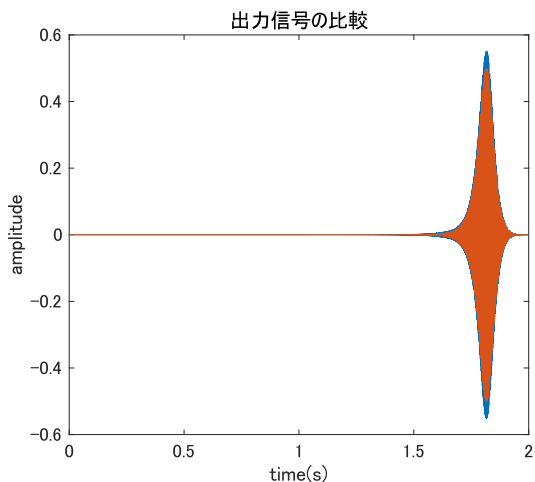


図 5 ブラックマン窓乗算後のフィルタの出力

図 5 ではブラックマン窓を乗算したフィルタの出力と、モデリング対象の出力を比較した。信号から、窓関数乗算後のフィルタの出力も、振幅の差は最大 0.05 と変わらなかった。

4 実機での実験の結果

シミュレーションの結果と合わせて、考察するために、実機を使ったモデリングを行った。実機は、Ibanez の Tube Screamer mini である。実機での実験では、信号の長さを $L = 131072$ として、131072 サンプルのスイープ信号を

用いた。これは、実際にスイープ信号を録音、再生する際に用いたソフトウェアの仕様に合わせるためである。

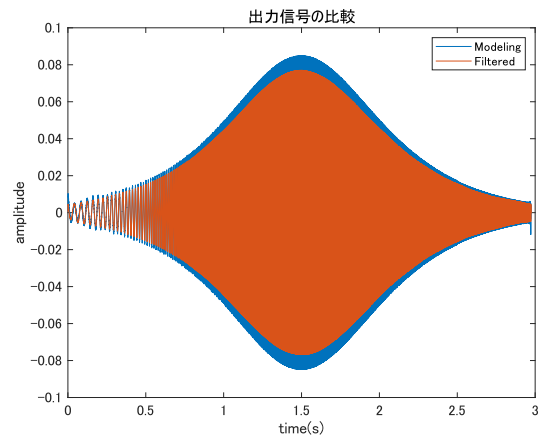


図 6 実機の出力とモデリング結果の比較

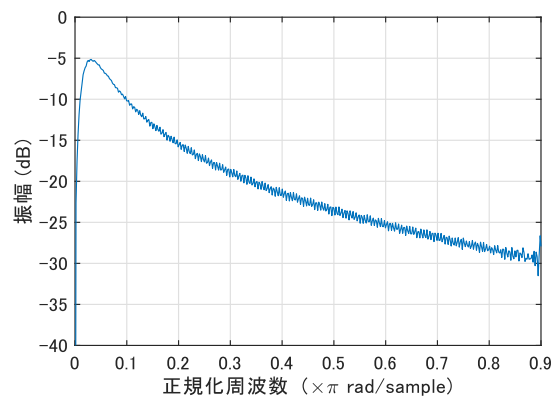


図 7 モデリングした線形ブロックの周波数応答

図 6 は実機からの出力と、モデリングしたフィルタの出力を比較したものである。図 7 はモデリングしたフィルタの周波数応答である。信号を比較すると、概ね外形が一致しているため、モデリングできている。シミュレーションの結果と同様に、信号の振幅が、モデリングしたフィルタの出力は少し大きくなってしまっている。振幅の差は 0.008 程度だった。

ここで、シミュレーションと同様にして、窓関数を乗算した結果についても示す。

図 8 はブラックマン窓乗算後のフィルタと、モデリングしたフィルタの出力を比較したものである。比較すると、振幅が大きくなってしまいう問題は解決できていないように見える。

5 考察

今回、シミュレーション、実験を行った。線形ブロックのモデリングは、今回のシミュレーション、実験の結果より、概ねできている。しかし、モデリングしたフィルタを

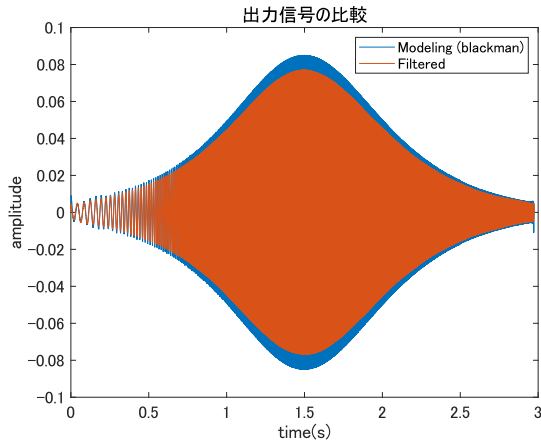


図 8 ブラックマン窓乗算後のフィルタの出力

通過した信号は、元のフィルタを通過した信号より、振幅が少し大きくなってしまふ．そこで、窓関数を乗算することで、モデリングの高精度化を狙った．図 4、図 5、図 6、図 8、の結果を見ると、窓関数を乗算することでは、見て分かるほどの変化が無いことがわかる．

結果を評価するため、次の指標を用いた．実機からの出力と、モデリングしたフィルタの出力を比較して信号とエラーの比を示す．ここでのエラーは入力と出力の差を示す．これを ESR (Error to Signal Rate) と次のように定義する．

$$ESR = \frac{E_{res}}{E_{sys}} = \frac{\sum_{n=-\infty}^n |y_{sys}(n) - y_{mod}(n, \mathbf{p})|^2}{\sum_{n=-\infty}^n |y_{sys}(n)|^2} \quad (5)$$

もうひとつ、入力と出力の違いを明らかにする方法として、相関係数を使用した．相関係数は、2 つの確率変数の線形依存性を表すが、ここでは、それぞれの出力信号の違いを表すために用いた．相関係数は、次のように表すことができる．

$$\rho(A, B) = \frac{cov(A, B)}{\sigma_A \sigma_B} \quad (6)$$

ここで、 $A = y_{sys}(n)$ 、 $B = y_{mod}(n, \mathbf{p})$ は、確率変数だが、ここでは、実機の出力と、モデリングの出力を示している． $cov(A, B)$ は A と B の共分散を表し、 $\sigma_{A, B}$ は確率変数の標準偏差を表している．

この指標を用いて、実機の出力と、モデリングしたフィルタの出力、窓関数を乗算したフィルタの出力を比較した．

実機の出力とモデリングしたフィルタを比較すると、 $ESR = \frac{1.2628}{115.8131} = 1.09\%$ 、 $\rho(A, B) = 0.99$ となり、実機の出力とモデリング後、カイザー窓をかけたフィルタの出力を比較すると $ESR = \frac{1.2603}{115.8131} = 1.09\%$ 、 $\rho(A, B) = 0.99$ となる．ESR は、両者とも同じく 1.09 だが、計算過程の分子の値が 0.0025 だけ小さくなっている．これより、僅かだが、誤差が少なくなっていることがわかる．

これより考えられることは、窓関数を乗算すると、FIR フィルタであることで発生した、リップルを僅かに低減させることはできるが、それ以上に誤差を少なくし、精度を向上させるには及ばないのではないかとのことである．

6 まとめと今後の課題

歪み回路のモデリングを行うために、文献 [3] の線形ブロック部分に関して考察した．この手法はエフェクタを線形ブロックと非線形ブロックに分けて、ブラックボックスとしてモデリングを行うもので、回路や素子の特性を調べる必要がなく、モデリングコストが少なくて済む．

本研究では、線形ブロックのモデリングにおいて、モデリングしたフィルタが実際のもよりも出力が大きくなるという問題を解決しようと考えた．FIR フィルタ作成時に窓関数を乗算することで、この問題を解決できるのではないかと考えた．しかし、シミュレーション、実機を用いた実験を通して、問題を解決できたとは言えない．

これは、窓関数を乗算することでは、カットオフ周波数付近のリップルを低減させることができるが、それ以上に誤差を少なくすることはできなかったのではないかと考えられる．計算の高速化、リアルタイム性が要求される等、FIR フィルタのタップ数が少なくなった場合、リップルが多く発生するため、窓関数を用いることでより大きな効果が期待できるかもしれないが、本研究における問題点を解決することはできないのではないかと考えられる．

先行研究 [3] より、非線形ブロックを含むモデリングでは、より大きな誤差もあり、改善の余地がある．さらに、本来エフェクタには、歪み回路で発生した倍音成分を整えるために、非線形ブロックの後に線形ブロックをおいている．この線形ブロックをモデリングできれば、精度は向上すると考えられる．

参考文献

- [1] A. Novak, L. Simon, P. Lotton, and J. Gilbert, "Chebyshev model and synchronized swept sine method in nonlinear audio effect modeling," Proc. Digital Audio Effects (DAFx-10), Graz, Austria, Sept. 6-10, 2010.
- [2] F. Eichas and U. Zolzer, "Black-box modeling of distortion circuits with block-oriented models," Proc. Digital Audio Effects (DAFx-15), Brno, Czech Republic, Sept. 5-9, 2016.
- [3] 松永悠斗, 青木直史, 土橋宜典, 山本強, "適応信号処理を用いた非線形歪み回路のモデリングに関する一検討," 日本音響学会聴覚研究会資料, Vol. 47, No. 5, pp. 355-360, July 21, 2017.
- [4] 尾知博, "シミュレーションで学ぶデジタル信号処理," CQ 出版社, 東京, July 1, 2001.