

2017 年度桂田研究室卒業レポート

ギターの歪みサウンドのシミュレーションと解析

指導教員 桂田 祐史

2017 年 2 月 14 日

河田 洋人

総合数理学部 現象数理学科

4 年 2 組 10 番

1.はじめに

エレキギターにおいて歪みとは、エレキギター特有の歪んだ音のことを指す。歴史的に歪みは真空管アンプの音割れが起源であるが、今日ではギターの歪みはブルースやロックのジャンルでは欠かせないサウンドとなっている。本研究ではその歪みをコンピュータでシミュレーション、解析をすることを目的とする。

2.エレキギターについて

エレキギターは 1920 年代から 30 年代にかけて登場した。エレキギターの木材でできたボディにはピックアップという弦の振動を弱い電気信号に変える装置が取り付けられている。ピックアップによって変換された弱い電気信号はシールドというギターに取り付ける配線によってアンプに送られ、アンプによってその信号を増幅し、音が生じる仕組みとなっている。エレキギター初期には歪みは音楽的に用いられることはなく、エレキギターそのものの音(クリーンと呼ばれる)で演奏することが一般的であった。しかし、真空管アンプで大音量をギターで鳴らす際に生じる音割れのような歪みサウンドをブルースやロックのジャンルで活用するミュージシャンが現れ始め、今では歪みサウンドはエレキギターと切り離せない重要なサウンドとなっている。また、このような歪みは現代ではエフェクターを用いて作られることが一般的である。エフェクターとはエレキギターのサウンドに音の効果を付け加えるための装置で、歪み、リバーブ(残響感)、エコーなどさまざまなエフェクトをかけることができる。本研究ではアンプに内蔵されたエフェクターを用いて歪みを作った。

3.,録音再生環境など

本研究では音声データを扱うため、それに必要となったライブラリをまとめる。Python を利用することを軸に環境を整えていった。具体的なプログラムはまとめて 8.サンプルコードに載せたので必要であれば参照してほしい。

音の録音、再生には PyAudio と呼ばれるライブラリを利用した。PyAudio は音の録音、再生を行うための Python 用のライブラリで PortAudio という C 言

語で書かれた録音再生用のライブラリのラッパーライブラリである。Python の文法で手軽に再生録音ができ、リアルタイム入力や外部マイクの利用なども可能である。

データの計算には Numpy というライブラリを利用した。これは Python 上で高速にベクトル計算を行うためのライブラリで FFT もこのライブラリ上で計算できる。

可視化には Matplotlib を用いた。これはグラフ作成用のライブラリである。PyAudio で録音再生、Numpy で数値計算、Matplotlib で可視化という流れで研究を進めた。

4.Sin 波を用いたシミュレーション

実際のエレキギターの音を用いて解析をする前に Sin 波を用いた歪みのシミュレーションを行う。歪みをコンピュータで再現するのにクリッピングという処理が用いられる。これは閾値を用意して（今回は 1.0 と-1.0）エフェクトをかけた際にその閾値を超えた場合は閾値に値を固定するという処理である。今回は振幅 1.0 の純粋な Sin 波に対して、閾値を 1.0 と-1.0 に設定し、振幅を 2 倍に増幅してクリッピングを行った。周波数 440Hz、サンプリングレート 44100Hz のサイン波を 500 サンプル分プロットすると以下のようなになる。

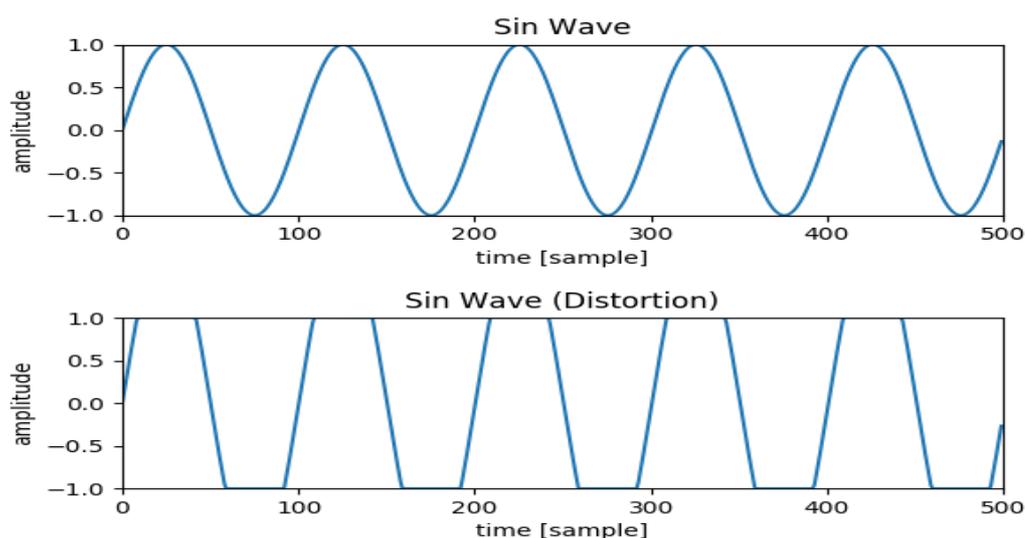


図 1. Sin 波と歪ませた Sin 波

また、これらの周波数スペクトルは以下のようになった。

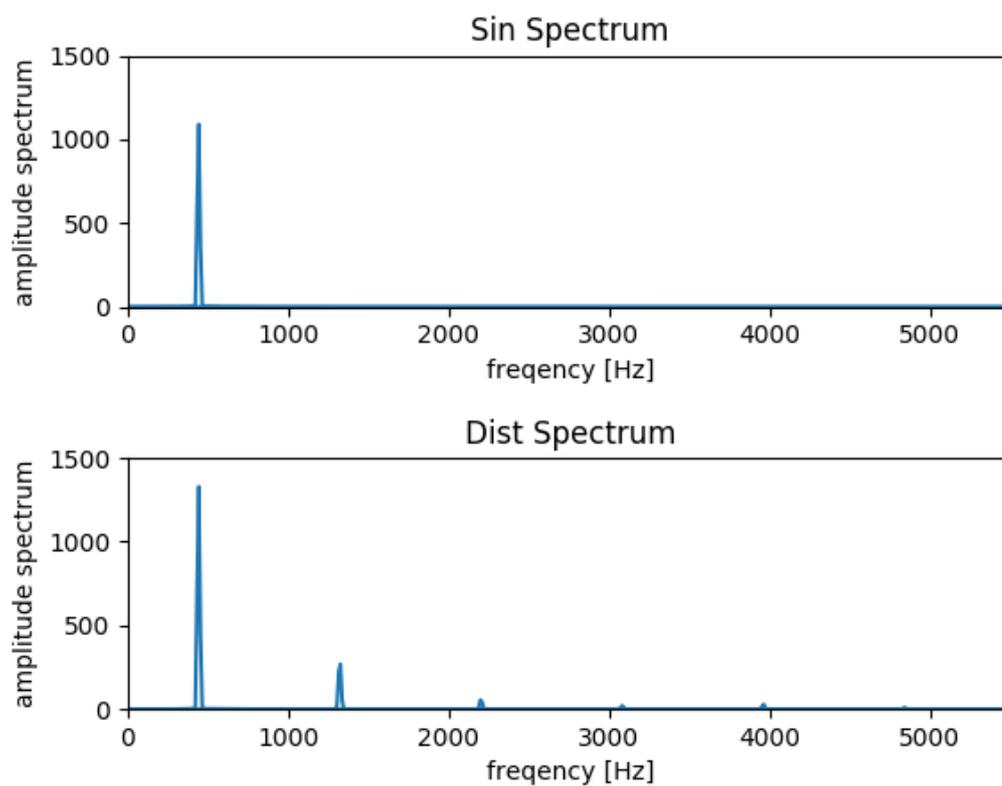


図 2. Sin 波と歪ませた Sin 波の周波数帯

5. エレキギターの音の解析

次にエレキギターの音をクリーンと歪みありで2種類用意してそれらを解析する。歪みはアンプに内蔵されたエフェクターを用いて作り、iPhoneでwav形式で保存したものをPythonで展開して解析した。以下はクリーン、歪みともに220HzのAの音を録音したものを1000サンプルプロットしたものである。

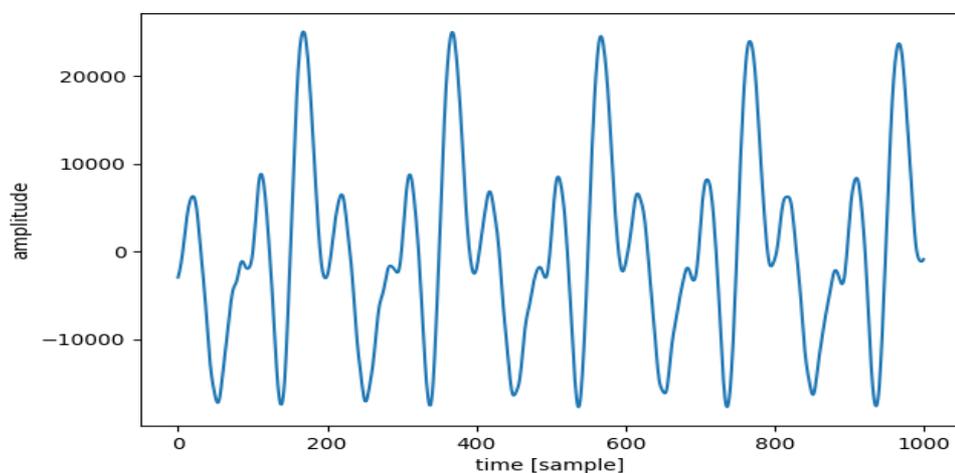


図3. エレキギター、クリーン

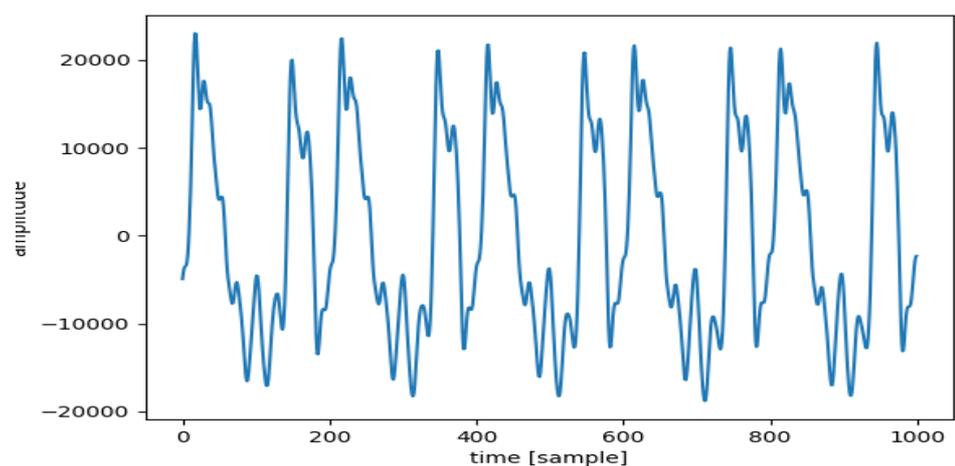


図4. エレキギター、歪み

また、以下はこれらの周波数スペクトルである。フーリエ変換に用いたサンプル数は44100である。

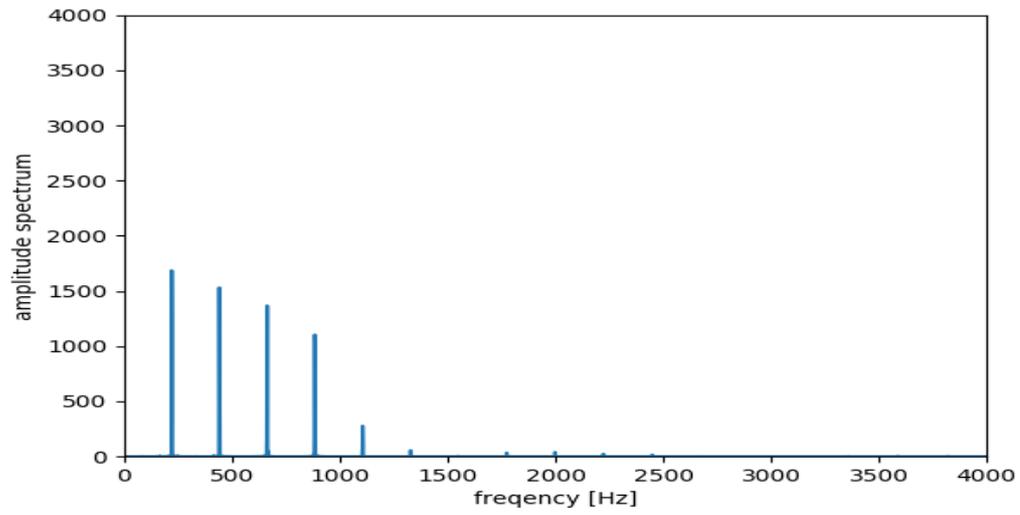


図 5. エレキギター、クリーン 周波数帯

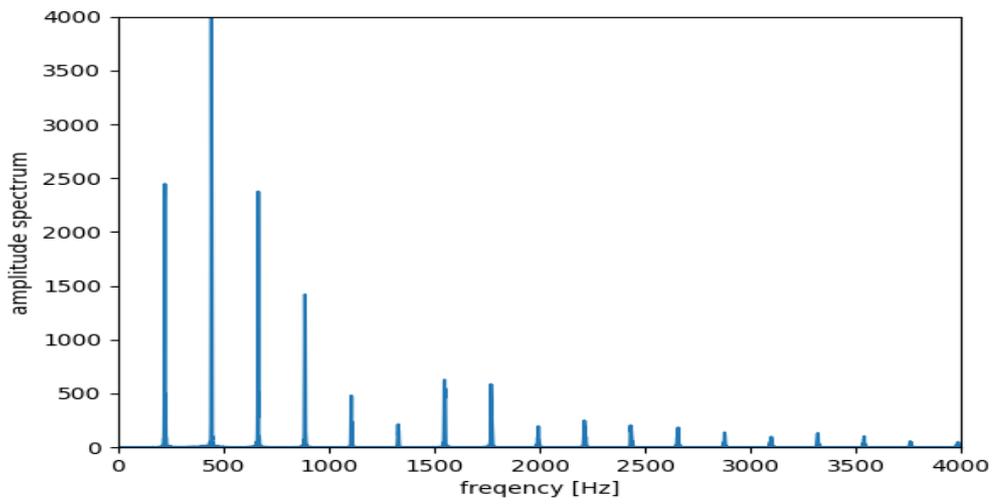


図 6. エレキギター、歪み 周波数帯

6. エレキギターの歪みのシミュレーション

次に先述したエレキギターのクリーンサウンドをコンピュータで歪ませるというものを考える。音源の波形を $f(t)$ として、以下のアルゴリズムで歪ませる。

1. $-1.0, 1.0$ を超える場合はクリッピングする。

$$y = \begin{cases} 1.0 & (\text{if } \text{gain} * f(t) > 1.0) \\ -1.0 & (\text{if } \text{gain} * f(t) < -1.0) \\ \text{gain} * f(t) & (\text{else}) \end{cases}$$

2. level ($0 \sim 1.0$ の実数値) をかけて音量を調整する。

このようなアルゴリズムで gain を 20、level を 0.9 にして、クリーンサウンドを歪ませた。以下はその結果を 1000 サンプルプロットしたものである。

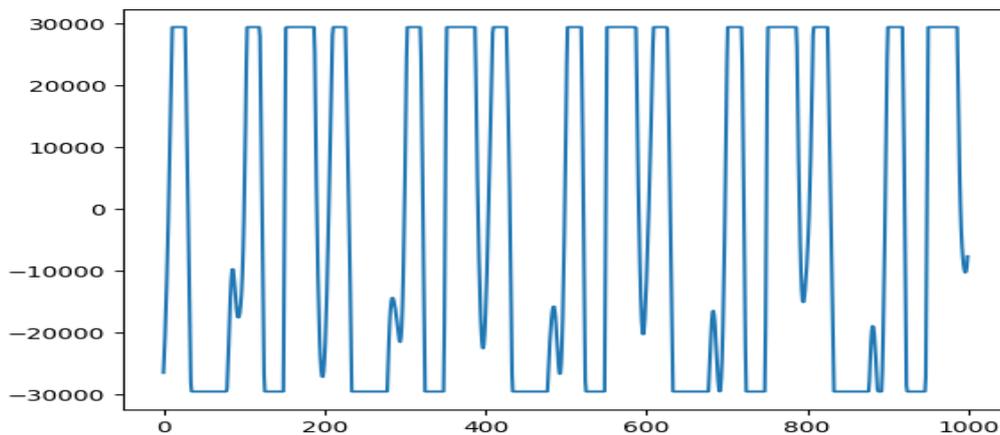


図7 クリーンサウンドをコンピュータで歪ませたもの

またこの周波数帯は以下ようになった。

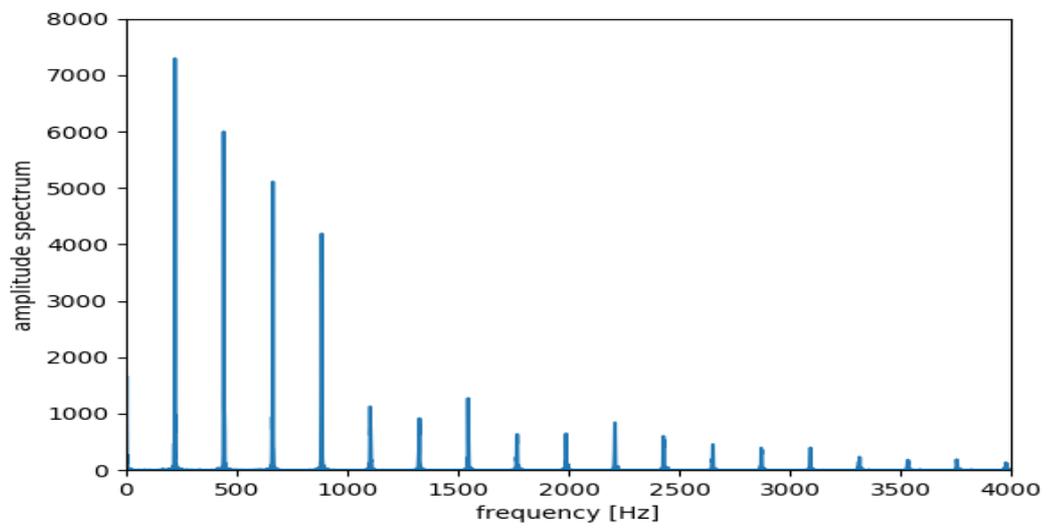


図8 クリーンサウンドをコンピュータで歪ませたもの 周波数スペクトル

7. 考察

7.1 サイン波のシミュレーションの考察

図 2.を見ると 440Hz で作成した Sin 波に対して 440Hz にピークをもつ周波数帯が予想通り確認できる。また、Sin 波をプログラムで歪ませると基本周波数である 440Hz に加えて、1320Hz、2200Hz の奇数倍音にピークが立つことが確認できる。倍音の違いが音の音色の違いに作用するため音の三要素のうち音の高さが変わらず音色が変化したことが、図から確認できる。また、一般的には音色が変わる際に偶数次倍音も生じることがあるのだが今回は奇数倍音のみが生じた。この理由は矩形波のフーリエ変換によって説明できる。厳密には歪ませた Sin 波は矩形波ではないのだが矩形波であると仮定する。

7.2 エレキギターのクリーン、歪みの考察

図 5, 6 を比べるとエレキギターを歪ませた結果、倍音成分が増加することが確認できる。

7.3 エレキギターの歪みのシミュレーションの考察

6.では原始的に増幅、クリッピングによって音を歪ませたが図 4, 7 で波形を見比べると視覚的には同じようには見えない結果になった。ただ、音を聴き比べると歪んでいる音に変化している印象はある。周波数帯はクリッピング後に図 6, 8 を比べると倍音成分が増加していることがわかる。

8. サンプルコードについて

soundutil.py はよく利用する手続きをまとめたものである。また pyaudio を用いた録音再生は play.py と record.py を参照してほしい。

9.参考文献

[1] 青木直史

C言語ではじめる音のプログラミング—サウンドエフェクトの信号処理
(オーム社) 2008

[2] フレッチャー、 ロッシング

楽器の物理学 (丸善出版株式会社) 2012

[3] 藤城裕樹

丸ごとエレキギターの本 (青弓社) 2017