

Pythonで作るエフェクタ

青木 直史

リスト1 WAVEファイルの読み込み、書き込み、波形のプロットを行うプログラム (test.py)

s0: 入力の音データ, s1: 出力の音データ, n: 時刻, fs: 標準化周波数

```
import numpy as np
from scipy.io import wavfile
import matplotlib.pyplot as plt

def wave_read_16bit_mono(file_name):
    fs, s = wavfile.read(file_name)
    s = s.astype(np.float)
    length_of_s = len(s)
    np.random.seed(0)
    for n in range(length_of_s):
        s[n] = s[n] + 32768 * (np.random.rand() - 0.5)
        s[n] = s[n] / 65536 * 2 - 1
    return fs, s

def wave_write_16bit_mono(fs, s, file_name):
    length_of_s = len(s)
    for n in range(length_of_s):
        s[n] = (s[n] + 1) / 2 * 65536
        s[n] = int(s[n] + 0.5)
        if s[n] > 65535:
            s[n] = 65535
        elif s[n] < 0:
            s[n] = 0
        s[n] -= 32768
    wavfile.write(file_name, fs, s.astype(np.int16))

fs, s0 = wave_read_16bit_mono('test(input).wav')
s1 = s0
wave_write_16bit_mono(fs, s1.copy(), 'test(output).wav')

plt.figure()
plt.plot(s0)

plt.figure()
plt.plot(s1)

plt.show()
```

● エフェクタはデジタル信号処理の塊

エフェクタとは音色を加工する音響機器のことです。通常、マイクやギターなどで生成された信号は電気回路で増幅された後、スピーカを通して音に変換されます。その際、何らかの処理を施し、原音を魅力的なものに加工するのがエフェクタの役割です。

かつてはアナログ回路のみで実現されていたエフェクタも、プロセッサやメモリの性能が向上した現在、デジタル信号処理によって実現されることが一般的になってきています。

エフェクタの仕組みを理解するうえで不可欠なのがデジタル・フィルタです。デジタル・フィルタを適用すると、音の波形や周波数特性を変化させることで音色をさまざまに加工できます。

本稿では、こうしたデジタル・フィルタの効果を実感してもらうことを目的として、Python言語を使って実際にエフェクタのプログラミングに挑戦してみます。

エフェクタのアルゴリズムの実装は、デジタル・フィルタの有用性を理解する上で格好の題材です。初心者にはブラックボックスに思えるエフェクタのアル

ゴリズムも、実際に自分で手を動かして音を作ってみると、仕組みがよく分かること請け合いです。こうした体験はデジタル信号処理に対する理解を深める上でも役立つでしょう。

なお、エフェクタの効果は、言葉で表現するよりも実際に聞いてみた方が理解しやすいことも事実です。そこで本稿では、具体例として分かりやすい曲をいくつか挙げてみました。YouTubeなどの動画サイトをご自身で検索の上、ぜひ一度視聴されることをお勧めします。

基本構造…WAVEファイルの読み書きと波形プロットのプログラム

手始めとして、WAVEファイルから音データを読み込み、何もしないでそのまま書き出すプログラムを動作させてみます。以降のプログラムは、このプログラムを骨格としています。

● 環境

使用する言語はPython、筆者の開発環境はAnaconda