% step 1. tedious file-importation

% 檔案命名規則:從左數到右的3個數字--

% 1st number:音高 0:跟中央C在一起的；1:比中央的再高8度

% 2nd number:當時錄製的順序

% 3rd number:檔案類型 0 : .aac；1 : .flac

% I find that Fs of .aac = 48000, Fs of .flac = 96000 = Fs1[[1]](#footnote-1)

% Also, I find that, .flac file, after imported into matlab, has 2 identical columns (test by function "isequal")[[2]](#footnote-2)

[c000,Fs] = audioread('c0-0.aac'); C000 = fft(c000); [c001,Fs1] = audioread('c0-0.flac'); c001 = c001( : , 1); C001 = fft(c001);

[c010,Fs] = audioread('c0-1.aac'); C010 = fft(c010); [c011,Fs1] = audioread('c0-1.flac'); c011 = c011( : , 1); C011 = fft(c011);

[c020,Fs] = audioread('c0-2.aac'); C020 = fft(c020); [c021,Fs1] = audioread('c0-2.flac'); c021 = c021( : , 1); C021 = fft(c021);

[d000,Fs] = audioread('d0-0.aac'); D000 = fft(d000); [d001,Fs1] = audioread('d0-0.flac'); d001 = d001( : , 1); D001 = fft(d001);

[d010,Fs] = audioread('d0-1.aac'); D010 = fft(d010); [d011,Fs1] = audioread('d0-1.flac'); d011 = d011( : , 1); D011 = fft(d011);

[d020,Fs] = audioread('d0-2.aac'); D020 = fft(d020); [d021,Fs1] = audioread('d0-2.flac'); d021 = d021( :, 1); D021 = fft(d021);

[e000,Fs] = audioread('e0-0.aac'); E000 = fft(e000); [e001,Fs1] = audioread('e0-0.flac'); e001 = e001( : , 1); E001 = fft(e001);

[e010,Fs] = audioread('e0-1.aac'); E010 = fft(e010); [e011,Fs1] = audioread('e0-1.flac'); e011 = e011( : , 1); E011 = fft(e011);

[e020,Fs] = audioread('e0-2.aac'); E020 = fft(e020); [e021,Fs1] = audioread('e0-2.flac'); e021 = e021( : , 1); E021 = fft(e021);

[f000,Fs] = audioread('f0-0.aac'); F000 = fft(f000); [f001,Fs1] = audioread('f0-0.flac'); f001 = f001( : , 1); F001 = fft(f001);

[f010,Fs] = audioread('f0-1.aac'); F010 = fft(f010); [f011,Fs1] = audioread('f0-1.flac'); f011 = f011( : , 1); F011 = fft(f011);

[f020,Fs] = audioread('f0-2.aac'); F020 = fft(f020); [f021,Fs1] = audioread('f0-2.flac'); f021 = f021( : , 1); F021 = fft(f021);

[g000,Fs] = audioread('g0-0.aac'); G000 = fft(g000); [g001,Fs1] = audioread('g0-0.flac'); g001 = g001( : , 1); G001 = fft(g001);

[g010,Fs] = audioread('g0-1.aac'); G010 = fft(g010); [g011,Fs1] = audioread('g0-1.flac'); g011 = g011( : , 1); G011 = fft(g011);

[g020,Fs] = audioread('g0-2.aac'); G020 = fft(g020); [g021,Fs1] = audioread('g0-2.flac'); g021 = g021( : , 1); G021 = fft(g021);

[a000,Fs] = audioread('a0-0.aac'); A000 = fft(a000); [a001,Fs1] = audioread('a0-0.flac'); a001 = a001( : , 1); A001 = fft(a001);

[a010,Fs] = audioread('a0-1.aac'); A010 = fft(a010); [a011,Fs1] = audioread('a0-1.flac'); a011 = a011( : , 1); A011 = fft(a011);

[a020,Fs] = audioread('a0-2.aac'); A020 = fft(a020); [a021,Fs1] = audioread('a0-2.flac'); a021 = a021( : , 1); A021 = fft(a021);

[b000,Fs] = audioread('b0-0.aac'); B000 = fft(b000); [b001,Fs1] = audioread('b0-0.flac'); b001 = b001( : , 1); B001 = fft(b001);

[b010,Fs] = audioread('b0-1.aac'); B010 = fft(b010); [b011,Fs1] = audioread('b0-1.flac'); b011 = b011( :, 1); B011 = fft(b011);

[b020,Fs] = audioread('b0-2.aac'); B020 = fft(b020); [b021,Fs1] = audioread('b0-2.flac'); b021 = b021( :, 1); B021 = fft(b021);

[c100,Fs] = audioread('c1-0.aac'); C100 = fft(c100); [c101,Fs1] = audioread('c1-0.flac'); c101 = c101( : , 1); C101 = fft(c101);

[c110,Fs] = audioread('c1-1.aac'); C110 = fft(c110); [c111,Fs1] = audioread('c1-1.flac'); c111 = c111( : , 1); C111 = fft(c111);

[c120,Fs] = audioread('c1-2.aac'); C120 = fft(c120); [c121,Fs1] = audioread('c1-2.flac'); c121 = c121( : , 1); C121 = fft(c121);

[d100,Fs] = audioread('d1-0.aac'); D100 = fft(d100); [d101,Fs1] = audioread('d1-0.flac'); d101 = d101( : , 1); D101 = fft(d101);

[d110,Fs] = audioread('d1-1.aac'); D110 = fft(d110); [d111,Fs1] = audioread('d1-1.flac'); d111 = d111( : , 1); D111 = fft(d111);

[d120,Fs] = audioread('d1-2.aac'); D120 = fft(d120); [d121,Fs1] = audioread('d1-2.flac'); d121 = d121( : , 1); D121 = fft(d121);

[e100,Fs] = audioread('e1-0.aac'); E100 = fft(e100); [e101,Fs1] = audioread('e1-0.flac'); e101 = e101( :, 1); E101 = fft(e101);

[e110,Fs] = audioread('e1-1.aac'); E110 = fft(e110); [e111,Fs1] = audioread('e1-1.flac'); e111 = e111( : , 1); E111 = fft(e111);

[e120,Fs] = audioread('e1-2.aac'); E120 = fft(e120); [e121,Fs1] = audioread('e1-2.flac'); e121 = d121( : , 1); E121 = fft(e121);

[f100,Fs] = audioread('f1-0.aac'); F100 = fft(f100); [f101,Fs1] = audioread('f1-0.flac'); f101 = f101( : , 1); F101 = fft(f101);

[f110,Fs] = audioread('f1-1.aac'); F110 = fft(f110); [f111,Fs1] = audioread('f1-1.flac'); f111 = f111( : , 1); F111 = fft(f111);

[f120,Fs] = audioread('f1-2.aac'); F120 = fft(f120); [f121,Fs1] = audioread('f1-2.flac'); f121 = f121( : , 1); F121 = fft(f121);

[g101,Fs1] = audioread('g1-0.flac'); g101 = g101( : , 1); G101 = fft(g101); % I forgot to record .aac for G1

[g111,Fs1] = audioread('g1-1.flac'); g111 = g111( : , 1); G111 = fft(g111);

[g121,Fs1] = audioread('g1-2.flac'); g121 = g121( : , 1); G121 = fft(g121);

[a100,Fs] = audioread('a1-0.aac'); A100 = fft(a100); [a101,Fs1] = audioread('a1-0.flac'); a101 = a101( : , 1); A101 = fft(a101);

[a110,Fs] = audioread('a1-1.aac'); A110 = fft(a110); [a111,Fs1] = audioread('a1-1.flac'); a111 = a111( : , 1); A111 = fft(a111);

[a120,Fs] = audioread('a1-2.aac'); A120 = fft(a120); [a121,Fs1] = audioread('a1-2.flac'); a121 = a121( : , 1); A121 = fft(a121);

[b100,Fs] = audioread('b1-0.aac'); B100 = fft(b100); [b101,Fs1] = audioread('b1-0.flac'); b101 = b101( : , 1); B101 = fft(b101);

[b110,Fs] = audioread('b1-1.aac'); B110 = fft(b110); [b111,Fs1] = audioread('b1-1.flac'); b111 = b111( : , 1); B111 = fft(b111);

[b120,Fs] = audioread('b1-2.aac'); B120 = fft(b120); [b121,Fs1] = audioread('b1-2.flac'); b121 = b121( : , 1); B121 = fft(b121);

Step 2.

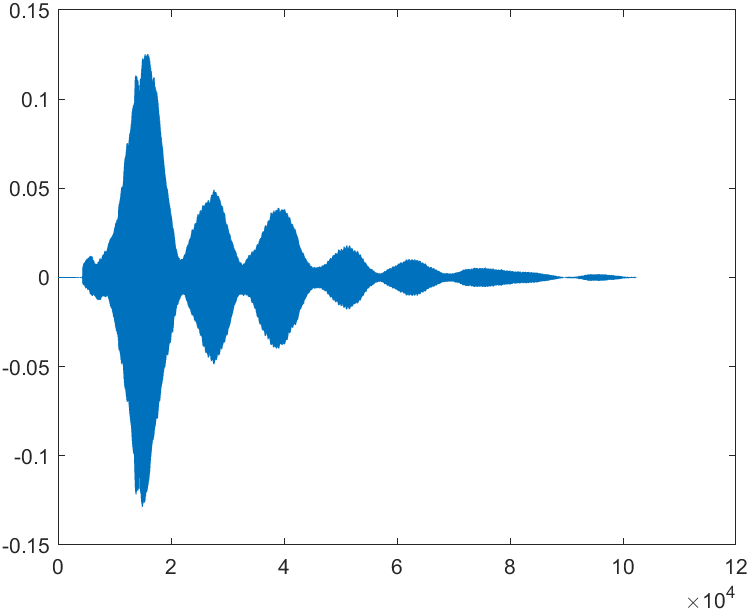
I have to choose which of these recordings should I use.

By the "sum" function of matlab, I find that sum of absolute value of array of .flac are generally larger than those of .aac (ex: sum(a111) =26.2665, sum(a110) = -0.4004)

Ideally a sinusoidal wave should have its sum = 0. But I think it's too early to jump into conclusion that .aac is more suitable than .flac

I found that b120 has a really beautiful form.

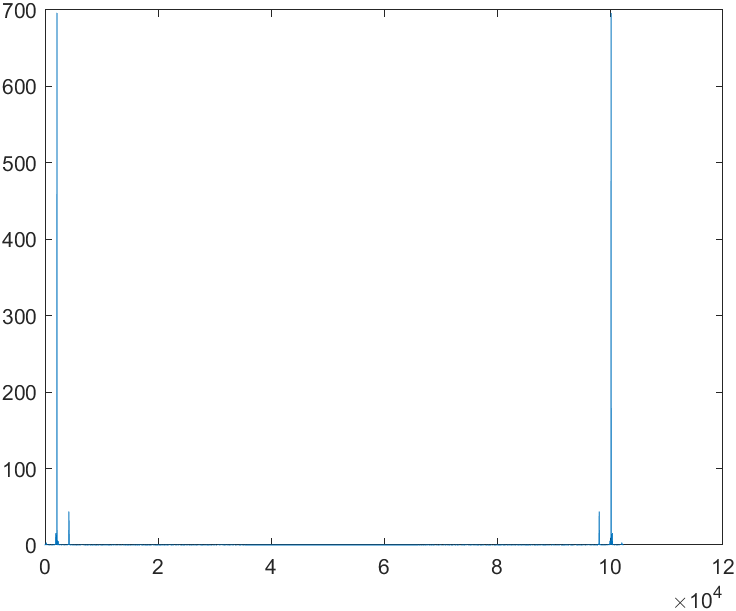
plot(b120);



This is like a beating with decaying amplitude.

Not surprisingly, it's k-domain looks like:

plot(abs(B120));



Which is approximately 4 pulses (since the signal is real, its absolute value of FFT is circular-even) corresponding to the 2 frequency of the beating (of course there are other nonzero values, but those are small and not so obvious, compared to the 4 pulses).

Not only this, but many other recordings have shape like this, which I call "amplitude-decaying beating".

為何會有脈波?

What I see:

|  |  |
| --- | --- |
|  |  |
|  |  |

What I know: 樂器演奏時，會產生泛音，而泛音的頻率是基音的整數倍。

參照上面的4張圖，可以發現各根主要的pulse的k軸座標，大概成整數倍關係。

推得:脈波是泛音導致的結果。

發現: 跟其他音相比，C1(比中央C高8度的那個C)的泛音很明顯(至少試了6筆錄音都是如此)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

But before we can discuss anything about this issue, there is one thing that has to be dealt with: the strange spectra from .flac file.

For every k-domain plot from .flac I've seen, there are 2 points:

(1) the saddle-like bump in the middle of the plot

(2) a pair (due to symmetry) of pulses respectively lie at the beginning and the end of the "saddle"

(3) an extremely-low-frequency (not sure if it is dc) pulse, which by the property of overtone:「Overtones are at integer multiples of the fundamental frequency」 can I tell that it does not belong to the harmonic series.

I would guess it is some kind of noise. But if it is noise, then why it only shows up in .flac files?

What I am going to do: % knowing the shape of |C101|, size(C101) = 151552

C101n = C101;

for n = 1: 75776

if (n > 200) && (n < 29020)

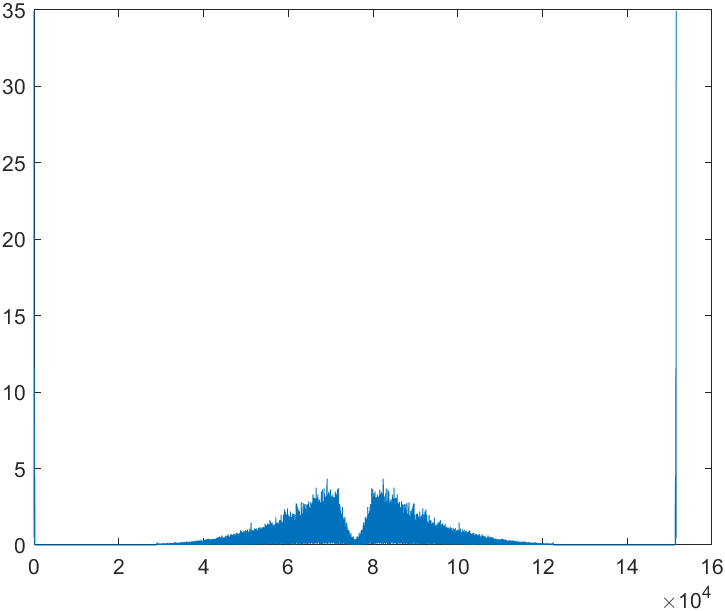
C101n(n) = 0;

C101n(151552 - n) = 0; % circular-even

end

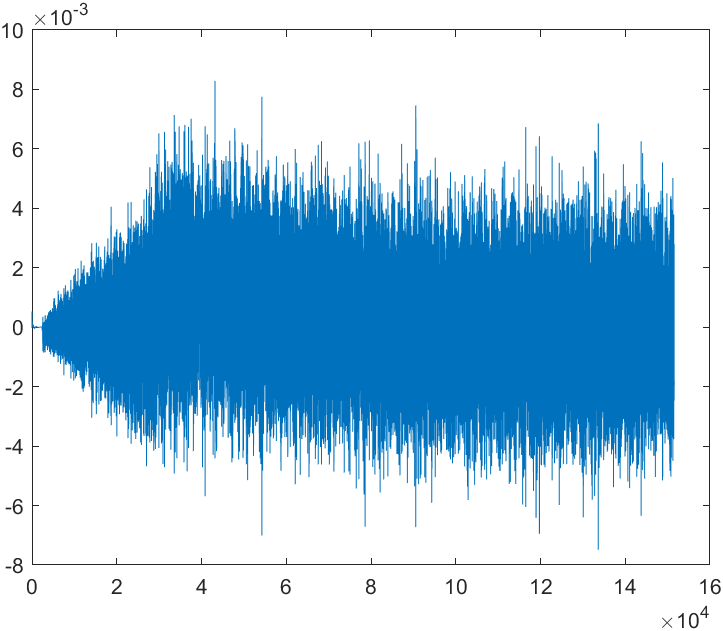
end

plot(abs(C101n));



c101n = ifft(C101n);

plot(real(c101n)); % maybe due to my imprecise process, it has imag part. Just ignore it.



Conclusion: Yeah, I would say this is noise.

Further: What if I also remove those 2 extremely-low-freq pulses?

C101m = C101;

for n = 1: 75776

if (n < 29020)

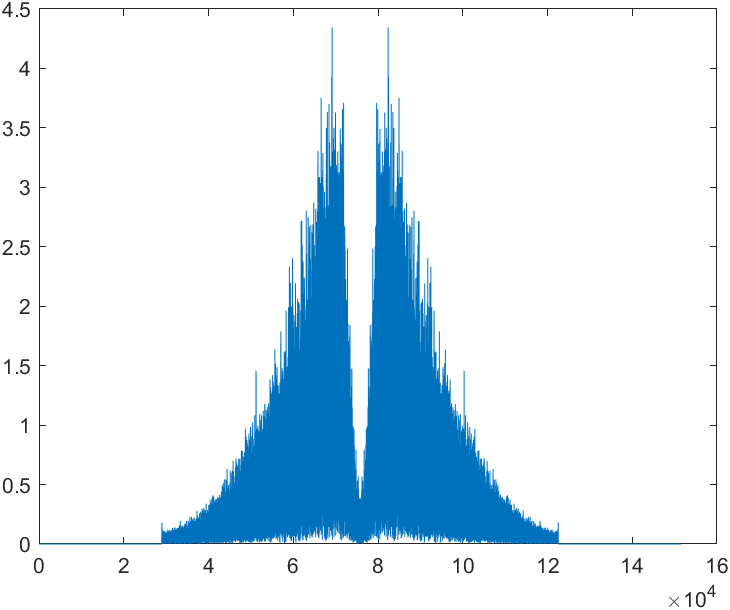
C101m(n) = 0;

C101m(151552 - n) = 0; % circular-even

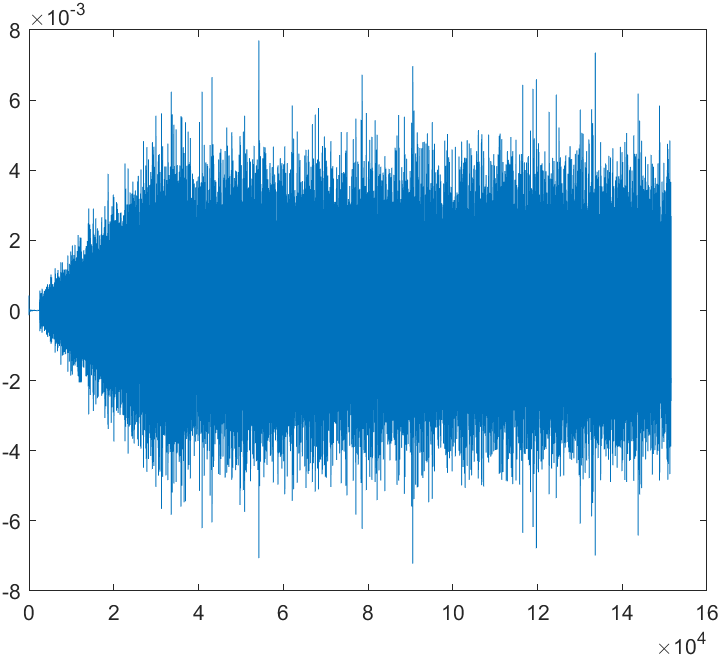
end

end

C101m(151552) = 0; plot(abs(C101m));



c101m = ifft(C101m); plot(real(c101m));



Still noise.

c101m = (real(c101m));

c101n = (real(c101n));

sum(c101m) % ans = -3.5475e-16

sum(c101n) % ans = 30.3257

% Now I am almost sure the 2 pulses are from the DC part of the noise.

1. [↑](#footnote-ref-1)
2. 關於這兩點，其實我只試了前幾個。 [↑](#footnote-ref-2)