

DSP 課題 1－7

平成	30	年	11	月	27	日
クラス	4J	番号	02			
基本取組時間				22	時間	
自主課題取組時間				3	時間	

1. DFT と FFT の比較調査

課題 1－5 で作成した DFT プログラムと今回作成した FFT プログラムを実行し，その結果から DFT と FFT の違いを調査する。

調査の方法を以下に示す。

- ・ 2^{14} 個のデータを用意する。データは課題 1－6 で利用した音声の波形データの一部を利用した。
- ・ $N = 2^7$ の時の DFT と FFT の振幅スペクトルを調査する。(調査 A)
- ・ $N = 2^7$ の時と $N = 2^{14}$ の時の DFT と FFT の計算時間を比較する。(調査 B)

調査 A

データ 27 個すべてはスペースの関係上乗せることはできないので，最初から 5 個の結果を記載し，後のデータは図 1 にグラフとして表す。なお，グラフは，データの比較のみに利用するため，データ

DFT 90.598868[dB], 69.810638[dB], 59.821865[dB], 60.152667[dB], 59.186553[dB]

FFT 90.598868[dB], 69.810638[dB], 59.821865[dB], 60.152667[dB], 59.186553[dB]

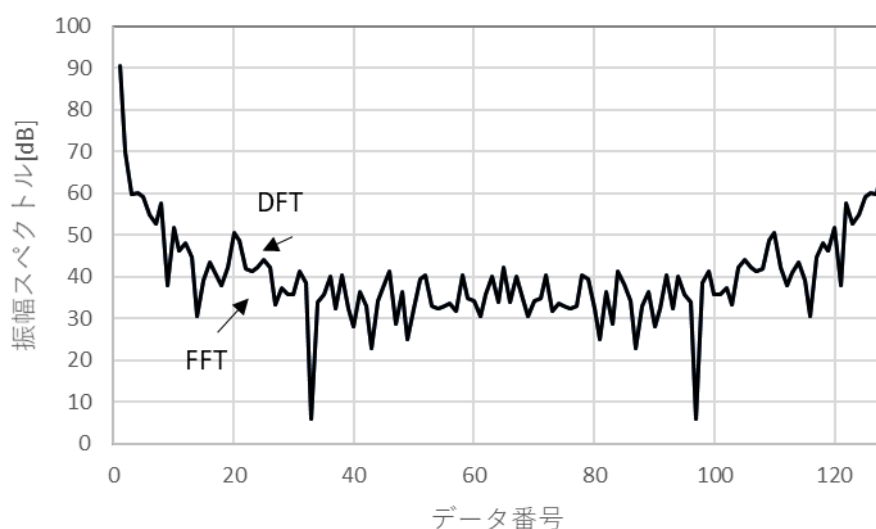


図 1 DFT,FFT 処理後の振幅スペクトル

図 1 では，グラフが完全に一致し，ひとつの線になっている。よって，DFT，FFT において，計算結果は全く同じになることが分かった。

調査 B

C 言語の関数 clock0 を利用し，データ数 $N = 2^7$ また， $N = 2^{14}$ にて DFT，FFT における処理時間をそれぞれ計測した。表 1 に結果を示す。なお，関数 clock0 の最小計測可能時間は ms 単位である。

表 1 DFT,FFT の処理時間

	$N=2^7$	$N=2^{14}$
DFT	4ms	64803ms
FFT	0ms	8ms

表 1 にて、 $N = 2^7$ の FFT の時、0ms となっているが、これは関数 clock0 で計測できる最小単位に満たない微小時間で処理が完了したことを表す。従って、FFT は DFT よりもかなり高速であることが分かる。

2. 考察

・DFT では冗長な計算が多く、バタフライ演算を利用することで処理の高速化が可能であり、反対にデータ数が増えるごとに計算数が膨大になっていくため、DFT は実用的ではない。

・FFT も DFT も計算方法の一種であり、両方とも離散フーリエ展開をしていることに変わりはないため、元の信号が周期的であるか必ず調査し、出力された結果が信頼できるものであるか注意する必要がある。

3. 自主課題

DFT と FFT の処理時間の差について、より詳しく調査する。DFT、FFT で演算するデータ数を変化させ、それぞれの実行時間をまとめたものを表 2、図 2,3 に示す。なお、本調査では処理時間のみに着目するため、データが周期的であるか等の議論は無視する。

表 2 DFT,FFT の処理時間（拡張）

	DFT	FFT
128	4ms	0ms
256	14ms	0ms
512	63ms	0ms
1024	237ms	1ms
2048	939ms	1ms
4096	3928ms	2ms
8192	15346ms	4ms
16384	64803ms	8ms

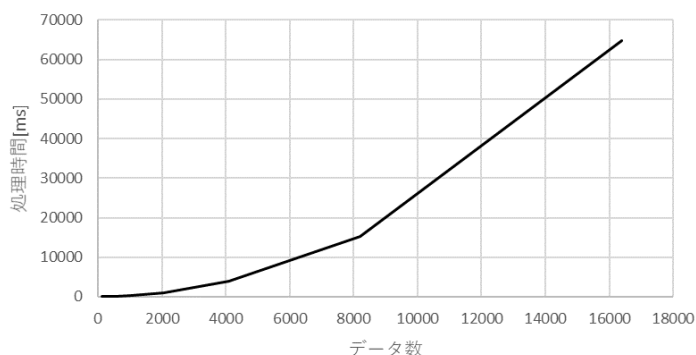


図 2 DFT の処理時間

図 3 にて FFT の処理時間が 0 付近で急増しているように見えるが、図 2 と図 3 ではメモリのスケールが大幅に違うため、急増したように見える。これは、考察でも述べた通り、関数 clock0 の測定限界のためでもある。

図 2,3 より、DFT の場合、指数的に処理時間も増加しているが、FFT の場合、処理時間の増加は一次的な変化であり、データ数が増えるごとに FFT と DFT の速度の差が顕著になることが言える。

よって、FFT の方が実用性が高く、また、変化が一次的なため、処理にかかる時間もある程度予想が可能であると考えられる。

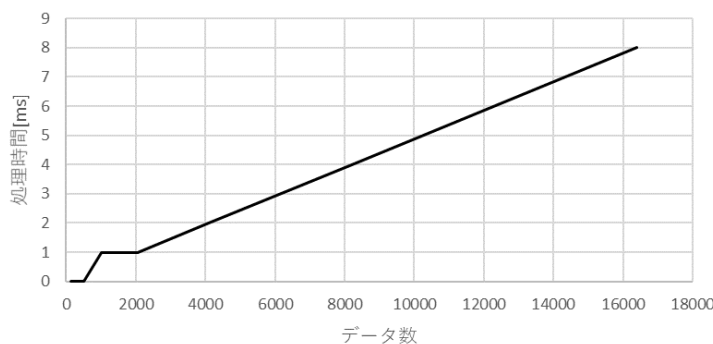


図 3 FFT の処理時間