信号処理とメディア通信　Handout#6　2011/5/24

信号処理とメディア通信　講義レジメ

担当：和田知久　（ファイヤー和田）
所属：琉球大学　工学部　情報工学科
連絡先：wada@ie.u-ryukyu.ac.jp
Home Page: http://www.ie.u-ryukyu.ac.jp/~wada/

１）第７章　高速フーリエ変換（FFT）の考え方
7.1(P127)
　時間間引き法　図 7-2
　周波数間引き法　図 7-11

２）第８章　FFT 高速フーリエ変換　計算アルゴリズム
8.6　周波数間引き FFT を導出する

8 点 FFT の導出（教科書 8.3-8.5 章）
P168 (2)式ただし(1/N)を省く
s(n)のフーリエ変換、

$$G(k)=\sum_{n=0}^{N-1}s(n)W_N^{nk}\cdots(2)$$

$$W_N=e^{-j\frac{2\pi}{N}}=\cos(\frac{2\pi}{N})-j\sin(\frac{2\pi}{N})\cdots(3)$$

N=8 とし、以下の変数変換をする。

$$k=k_0+2k_1+4k_2\quad(k_0,k_1,k_2=0,1)$$
$$n=n_0+2n_1+4n_2\quad(n_0,n_1,n_2=0,1)$$

$$G(k)=\sum_{n_0=0}^{1}\sum_{n_1=0}^{1}\sum_{n_2=0}^{1}s(n_0+2n_1+4n_2)W_N^{(n_0+2n_1+4n_2)(k_0+2k_1+4k_2)}$$

となる。

（ステージ１）

n_2 に関するΣを展開する

$$G(k)=\sum_{n_0=0}^{1}\sum_{n_1=0}^{1}(s(n_0+2n_1)W_8^{(n_0+2n_1)(k_0+2k_1+4k_2)}+s(n_0+2n_1+4)W_8^{(n_0+2n_1+4)(k_0+2k_1+4k_2)})$$

$$=\sum_{n_0=0}^{1}\sum_{n_1=0}^{1}(s(n_0+2n_1)W_8^{(n_0+2n_1)k_0}W_8^{(n_0+2n_1)(2k_1+4k_2)}+s(n_0+2n_1+4)W_8^{(n_0+2n_1)k_0}W_8^{4k_0}W_8^{(n_0+2n_1)(2k_1+4k_2)}W_8^{4(2k_1+4k_2)})$$

$$=\sum_{n_0=0}^{1}\sum_{n_1=0}^{1}(s(n_0+2n_1)W_8^{(n_0+2n_1)k_0}+(-1)^{k_0}s(n_0+2n_1+4)W_8^{(n_0+2n_1)k_0})W_8^{(n_0+2n_1+4)(2k_1+4k_2)}$$

$$=\sum_{n_0=0}^{1}\sum_{n_1=0}^{1}((s(n_0+2n_1)+(-1)^{k_0}s(n_0+2n_1+4))W_8^{(n_0+2n_1)k_0})W_4^{(n_0+2n_1)(k_1+2k_2)}$$

ここで、

$$s1(n_0+2n_1+4k_0)=((s(n_0+2n_1)+(-1)^{k_0}s(n_0+2n_1+4))W_8^{(n_0+2n_1)k_0}$$
$$(k_0=0,1)$$

とすると、

$$G(k)=\sum_{n_0=0}^{1}\sum_{n_1=0}^{1}s1(n_0+2n_1+4k_0)W_4^{(n_0+2n_1)(k_1+2k_2)}$$

となり、これは、s1 を入力とする 4 点 DFT に相当します。ここで、k_0=0,1 なので２つの DFT 計算が必要となります。これが図 8-7 (P177)に相当します。

（ステージ２）

同様に、n_1 に関するΣを展開する

$$G(k)=\sum_{n_0=0}^{1}(s1(n_0+0+4k_0)W_4^{(n_0)(k_1+2k_2)}+s1(n_0+2+4k_0)W_4^{(n_0+2)(k_1+2k_2)})$$

$$=\sum_{n_0=0}^{1}(s1(n_0+0+4k_0)W_4^{n_0k_1}W_4^{n_0 2k_2}+s1(n_0+2+4k_0)W_4^{n_0k_1}W_4^{2k_1}W_4^{n_0 2k_2}W_4^{4k_2})$$

$$=\sum_{n_0=0}^{1}((s1(n_0+0+4k_0)+(-1)^{k_1}s1(n_0+2+4k_0))W_4^{n_0k_1})W_2^{n_0k_2}$$

ここで

$$s2(n_0+2k_1+4k_0)=((s1(n_0+0+4k_0)+(-1)^{k_1}s1(n_0+2+4k_0))W_4^{n_0k_1}$$

$$(k_1=0,1)$$

とすると

$$G(k)=\sum_{n_0=0}^{1}s2(n_0+2k_1+4k_0)W_2^{n_0k_2}$$

となり、これは、s2 を入力とする 2 点 DFT に相当します。ここで、k_0, k_1=0,1 なので 4 つの 2 点 DFT 計算が必要となります。これが図 8-8 (P177)に相当します。

（ステージ３）

同様に、n_0 に関するΣを展開する

$$G(k)=s2(0+2k_1+4k_0)+s2(1+2k_1+4k_0)W_2^{k_2}$$

$$=s2(0+2k_1+4k_0)+(-1)^{k_2}s2(1+2k_1+4k_0)$$

ここで

$$s3(k_2+2k_1+4k_0)=s2(0+2k_1+4k_0)+(-1)^{k_2}s2(1+2k_1+4k_0)$$

とすると、

$$G(k)=G(k_0+2k_1+4k_2)=s3(k_2+2k_1+4k_0)$$

となり、INDEX の値が変わっています。これが図 8-9 (P178)に相当します。

最後に、順番を入れ替える必要があります。

詳細は MATLAB コード myfft8.m により確認しましょう。

３）８点 FFT と８点 DFT の演算量に比較

８点 DFT

（８乗算＋７加算）＊８＝６４乗算＋５６加算

８点 FFT

バラフライ演算 1 個には、加減算 2 個、乗算 1 個とすると

ステージ１：

（1 乗算＋2 加減算）＊4＝4 乗算＋8 加減算

ステージ２：

（1 乗算＋2 加減算）＊4＝4 乗算＋8 加減算

ステージ３

（2 加減算）＊4＝8 加減算

トータル：　8 乗算＋24 加減算

宿題４）

①N=3 の例を参考に、2^N 入力の DFT と FFT の演算量を比較せよ。
FFT に関しては、ステージ数、１ステージあたりのバタフライ演算数を導き、演算量を求めよ。

②myfft8 を改良して、myfft16 を作成せよ。適当な 16 点の入力信号を作成し、MATLAB の fft コマンドと myfft16 の出力値を比較せよ。

以上