

Discreate Fourier Transform Menjadi Fast Fourier Transform

Santi Gita Radiana, 32853-TE
Jurusan Teknik Elektro FT UGM,
Yogyakarta

1. PENDAHULUAN

Tranformasi Fourier digunakan untuk mentransformasi sinyal waktu kontinyu ke dalam kawasan frekuensi. Transformasi Fourier mendikripsikan spektrum kontinyu dari sinyal nonperiodik. Transformasi Fourier $X(f)$ dari waktu kontinyu $x(t)$ adalah sebagai berikut :

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-i2\pi ft} dt$$

Invers Transformasi :

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{i2\pi ft} df$$

Agar Transformasi Fourier dapat digunakan dalam operasi digital, maka diperlukan sampel-sampel pada kawasan frekuensi dan waktu. Sampel-sampel sinyal kontinyu pada kawasan waktu akan merepresentasikan keseluruhan sinyal kontinyu tersebut. Sampel-sampel ini akan mengubah sinyal kontinyu menjadi sinyal diskrit, maka dibutuhkan Discreate Fourier Transform (DFT- transformasi Fourier sinyal diskrit). Transformasi Fourier sinyal diksrit adalah sebagai berikut :

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn}$$

Dengan

$$W_N = e^{-j(2\pi/N)}$$

Invers transformasi :

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-kn}$$

Pengambilan sampel untuk analisis DFT dari sinyal kontinyu perlu diperhatikan agar tidak terjadi kesalahan. Analisis DFT dari sinyal kontinyu menggunakan perkiraan berupa sampel-sampel, maka perlu dipahami adanya keterbatasan sampel-sampel terhadap bentuk sinyal kontinyu yang sebenarnya. Ada tiga yang bisa terjadi akibat kesalahan perkiraan sinyal kontinyu, yaitu :

- a. Aliasing, karena *sample rate* tidak cukup tinggi untuk menghindari overlap spektrum
- b. Leakage, timbul efek distorsi spektrum karena pengabaian sinyal frekuensi pada waktu yang tak terhingga
- c. Picket-fence effect, timbul karena ketidakmampuan DFT mengobservasi sinyal sebagai sinyal kontinyu, karena perhitungan spektrum yang terbatas

2. DFT MENJADI FFT

Fast Fourier Transform (FFT) adalah algoritma untuk menghitung DFT dengan cepat dan efisien. Perhitungan DFT secara langsung akan membutuhkan operasi aritmatika sebanyak $O(N^2)$, sedangkan perhitungan dengan FFT akan membutuhkan operasi sebanyak $O(N \log N)$.

Kebutuhan kalkulasi DFT

$$X_k = \sum_{n=0}^{N-1} x(n) W_N^{kn}$$

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

Karena $x(n) = x_r(n) + jx_i(n)$ bisa bernilai kompleks, maka

$$X(k) = X_R(k) + jX_I(k)$$

$$1. X_R(k) = \sum_{n=0}^{N-1} \left[x_r(n) \cos 2\pi \frac{k}{N} n + x_i(n) \sin 2\pi \frac{k}{N} n \right]$$

$$2. X_I(k) = \sum_{n=0}^{N-1} \left[x_r(n) \sin 2\pi \frac{k}{N} n - x_i(n) \cos 2\pi \frac{k}{N} n \right]$$

Perhitungan di atas memerlukan

- $2N^2$ evaluasi trigonometri
- $4N^2$ perkalian real
- $4N(N-1)$ penjumlahan real

Sering disebut $O(N^2)$.

Salah satu perhitungan FFT adalah dengan Radix-2 FFT. Sinyal diskrit $x(n)$ didesimalkan dengan cara dipangkatkan 2 dan menghasilkan $f_1(n)$ dan $f_2(n)$ yang disebut algoritma *decimation-in-time*. N-point DFT didesimalkan dengan cara :

$$\begin{aligned} X(k) &= \sum_{n=0}^{N-1} x(n) W_N^{kn}, \quad k = 0, 1, 2, \dots, N-1 \\ &= \sum_{n \text{ even}} x(n) W_N^{kn} + \sum_{n \text{ odd}} x(n) W_N^{kn} \\ &= \sum_{n=0}^{\left(\frac{N}{2}-1\right)} x(2m) W_N^{2mk} + \sum_{n=0}^{\left(\frac{N}{2}-1\right)} x(2m+1) W_N^{k(2m+1)} \end{aligned}$$

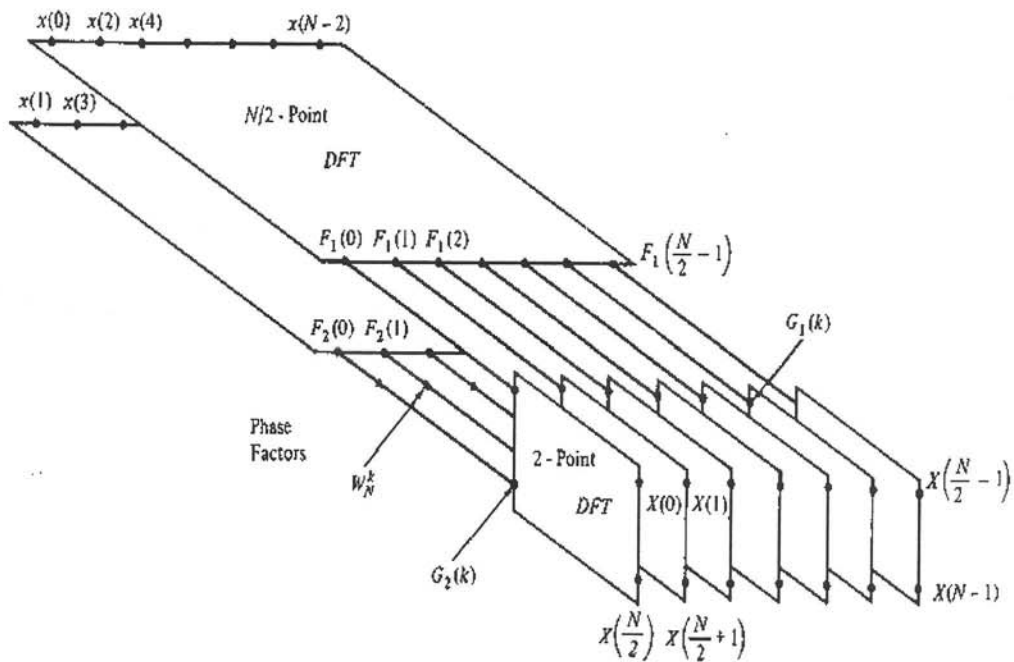
Substitusi $W_N^2 = W_{N/2}$

$$\begin{aligned} X(k) &= \sum_{n=0}^{\left(\frac{N}{2}-1\right)} f_1(m) W_{N/2}^{km} + W_N^k \sum_{n=0}^{\left(\frac{N}{2}-1\right)} f_2(m) W_{N/2}^{km} \\ &= F_1(k) + W_N^k F_2(k), \quad k = 0, 1, \dots, N-1 \end{aligned}$$

Dengan F_1 dan F_2 adalah titik ke- $N/2$ dari $f_1(m)$ dan $f_2(m)$, periodik dengan periode $N/2$, maka $F_1(k+N/2) = F_1(k)$ dan $F_2(k+N/2) = F_2(k)$. Sehingga $W_N^{k+N/2} = -W_N^k$, maka persamaannya

$$X(k) = F_1(k) + W_N^k F_2(k), \quad k = 0, 1, \dots, \frac{N}{2} - 1$$

$$X\left(k + \frac{N}{2}\right) = F_1(k) - W_N^k F_2(k), \quad k = 0, 1, \dots, \frac{N}{2} - 1$$



Gambar 1. First step in the decimation-in-time algorithm

Dengan menghitung DFT $N/4$ -point, akan didapatkan $F_1(k)$ dan $F_2(k)$ dari DFT $N/2$ -point dengan cara

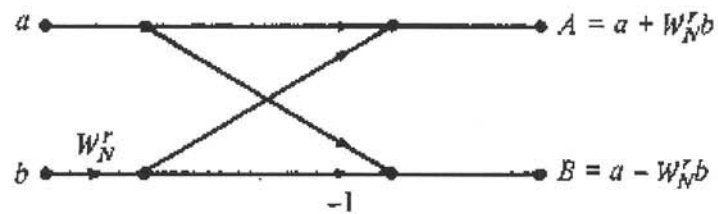
$$F_1 = F\{f_1(2n)\} + W_{N/2}^k F\{f_1(2n+1)\}, \quad k = 0, 1, \dots, \frac{N}{4} - 1; \quad n = 0, 1, \dots, \frac{N}{4} - 1$$

$$F_1\left(k + \frac{N}{4}\right) = F\{f_1(2n)\} - W_{N/2}^k F\{f_1(2n+1)\}$$

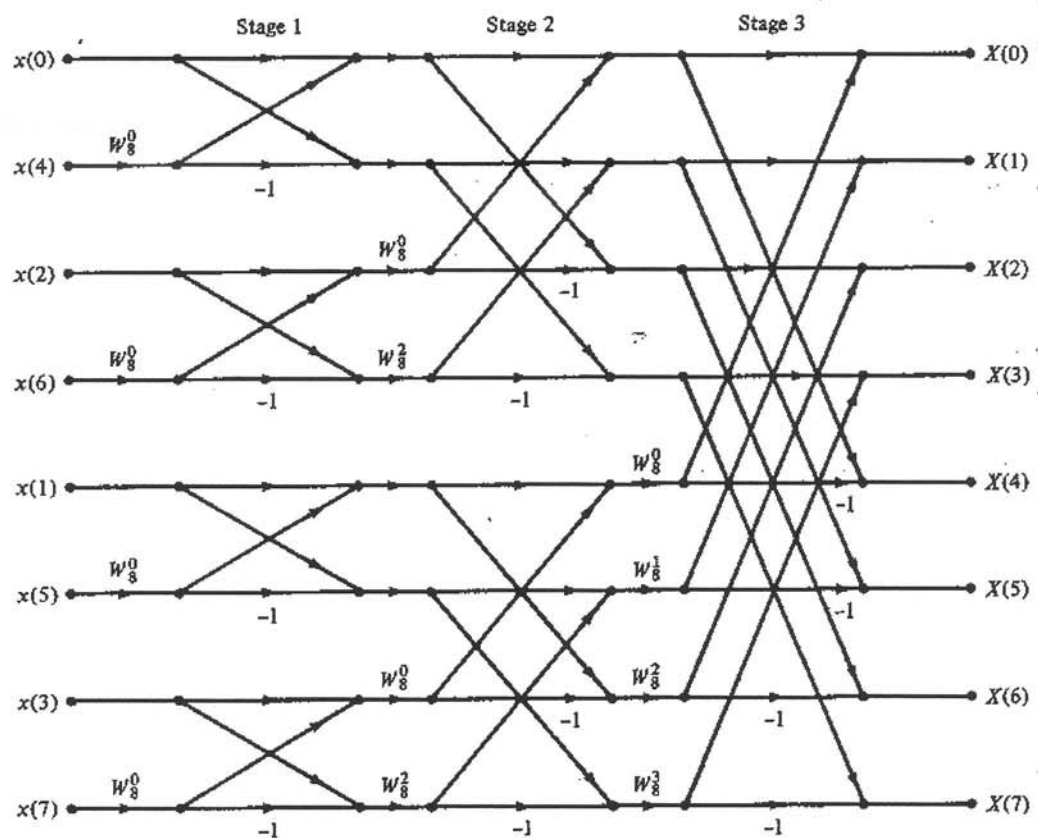
$$F_2 = F\{f_2(2n)\} + W_{N/2}^k F\{f_2(2n+1)\}, \quad k = 0, 1, \dots, \frac{N}{4} - 1; \quad n = 0, 1, \dots, \frac{N}{4} - 1$$

$$F_2\left(k + \frac{N}{4}\right) = F\{f_2(2n)\} - W_{N/2}^k F\{f_2(2n+1)\}$$

Ilustrasi perhitungan decimation-in-time DFT dapat digambarkan dengan perhitungan butterfly sebagai berikut :



Gambar 2. Basic butterfly computation in the decimation-in-time FFT algorithm



Gambar 3. Eighth-point decimation-in-time FFT algorithm

3. KESIMPULAN

Transformasi sinyal diskrit dengan Transformasi Fourier membutuhkan sampel-sampel dari sinyal kontinyu sehingga didapatkan sinyal diskrit yang diinginkan. Penagmbilan sampel sinyal kontinyu perlu diperhatikan agar hasil transformasi yang diharapkan sesuai dengan gambaran sinyal tersebut dalam bentuk sinyal kontinyu. Transformasi Fourier sinyal diskrit juga akan lebih cepat dan efisiein jika menggunakan algoritma Fast Fourier Transform terutama untuk bentuk sinyal diskrit dalam bilangan kompleks.

4. REFERENSI

- [1] Hsu, Hwei P. 1976. *Schaum's Outline Of Theory And Problems Of Signals And Systems*. McGraw Hill.