Diskret Fourier Transform dan FFT

1. DFT

   Untuk melakukan analisis frekuensi dari sinyal waktu diskret x(n) maka perlu mendapatkan representasi domain frekuensi dari sinyal yang biasanya dinyatakan dalam domain waktu. DFT digunakan untuk melakukan analisa freuensi dari sinyal waktu diskret.

$$x(n) \xleftrightarrow{N Po\text{int } DFT} X(k) \quad \text{dimana n = 0, …N-1} \quad \text{dan k = 0, …N-1}$$

DFT dihitung menggunakan persamaan:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad \text{dimana} \quad W_N = e^{-j\frac{2p}{N}}$$

Sehingga

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2p\left(\frac{k}{N}\right)n}$$

Inverse DFT (IDFT) menghitung kembali representasi sinyal waktu diskret x(n) dari sinyal yang dinyatakan dalam domain frekuensi X(ω).

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2p\left(\frac{k}{N}\right)n}$$
$$= \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}$$

Di mana

$$W_N = e^{-j\frac{2p}{N}} \rightarrow \text{akar ke N dari unity}$$

DFT dan IDFT dapat juga dipandang sebagai transformasi linear antara x(n) dan X(k), jadi

$$\overline{x_N} \leftrightarrow \overline{X_N}$$

Di mana x_N dan X_N masing-masing adalah vektor dengan n buah elemen

$$\overline{x_N} = \begin{bmatrix} x(0) \\ \vdots \\ x(N-1) \end{bmatrix} \qquad \overline{X_N} = \begin{bmatrix} X(0) \\ \vdots \\ X(N-1) \end{bmatrix}$$

Jika dinyatakan matriks W_N

$$\overline{W_N} = \left[w_{ij} = W_N^{(i)(j)} \right]$$

Maka, N point DFT dapat dinyatakan dalam bentuk

$$\overline{X_N} = \overline{W_N}\,\overline{x_N}$$

Sedangkan IDFT dapat dihitung jika terdapat inverse dari W_N.

$$\overline{x_N} = \overline{W_N^{-1}}\,\overline{X_N} \quad \text{bila } W_N^{-1} \quad \text{exist}$$

Contoh:

Hitung 4 point DFT dari sinyal x(n) = (0 1 2 3)

Penyelesaian:

$$\overline{W_4} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_4^1 & W_4^2 & W_4^3 \\ 1 & W_4^2 & W_4^4 & W_4^6 \\ 1 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix}$$

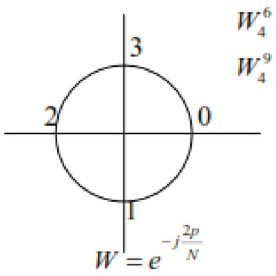
$$\text{ingat } W_N^{k+\frac{N}{2}} = -W_N^k$$

Ingat $W_N^{k+\frac{N}{2}} = -W_N^k$

$W_4^4 = 1$

$W_4^6 = -1$

$W_4^9 = j$



$W = e^{-j\frac{2p}{N}}$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

$$\rightarrow \overline{X_4} = W_4 \bar{x} = \begin{bmatrix} 6 \\ -2+2j \\ -2 \\ -2-2j \end{bmatrix}$$

2. Sifat DFT

   Sifat Linear:

   Jika

   $X_1(n) \leftarrow$ N-DFT $\rightarrow X_1(k)$

   Dan

   $X_2(n) \leftarrow$ N-DFT $\rightarrow X_2(k)$

   Maka untuk sembarang konstanta a_1 dan a_2 real atau kompleks

   $a_1.x_1(n) + a_2.x_2(n) \leftarrow$ N-DFT $\rightarrow a_1.X_1(k) + a_2.X_2(k)$

   sifat periodik:

   jika x(n) $\leftarrow$ N-DFT $\rightarrow$X(k)

   maka

   x(n + N) = x(n) untuk semua n

   X(k+N) = X(k) untuk semua k

3. Filter Menggunakan DFT

   $x(n) \leftarrow\rightarrow X(\omega)$

   $h(n) \leftarrow\rightarrow H(\omega)$

   $y(n) \leftarrow\rightarrow Y(\omega)$

   $X(\omega) \rightarrow H(\omega) \rightarrow Y(\omega) = H(\omega)X(\omega)$

   Asumsikan FIR dan Finite duration

   Jika: x(n) = 0, n<0 dan n≥L →Durasi L

   h(n) = 0, n<0 dan n≥M →durasi M

   $Y(\omega) = H(\omega)\, X(\omega)$ durasi: L+M-1

   Bila $Y(\omega)$ dicuplik maka pencuplikan harus N≥L+M-1

   Agar

   $$y\left(\frac{2pk}{N}\right) \xleftrightarrow{IDFT} y(n)$$

   Maka

   $$Y(k) = Y(w)\Big|_{w=\frac{2pk}{N}} \qquad \text{k = 0, …, N-1}$$

   $$\rightarrow Y(k) = \underset{\substack{\downarrow \\ \text{zero padding}}}{X(k)}\underset{\downarrow}{H(k)}, \quad k = 0,\ldots,N-1 \quad N \geq L+M+1$$

   $$\rightarrow Y(k) \xleftrightarrow{IDFT} y(n)$$

   Contoh:

   FIR: h(n) = {1,2,3} dan X(n) = {1,2,2,1}

   Carilah keluaran menggunakan DFT dan IDFT

   Penyelesaian:

   L = 4, M= 3→ N = L+M-1 = 6

   Pilih N= L+M+1=8 (agar sesuai dngan FFT)

$$H(k)=\sum_{n=0}^{7} k(n)e^{-j2p\left(\frac{k}{8}\right)n}$$

$$H(k)=1+2e^{-j2p\frac{k}{8}}+3e^{-j2p\frac{k}{4}}+2e^{-j2p\frac{3k}{8}},\quad k=0,\ldots,7$$

$$X(k)=\sum_{n=0}^{7} h(n)e^{-j2p\left(\frac{k}{8}\right)n}$$

$$=1+2e^{-jp\frac{k}{8}}+2e^{-jp\frac{k}{4}}+2e^{-jp\frac{3k}{8}},\quad k=0,\ldots,7$$

$$X(0)=6 \qquad X(1)=\frac{2+\sqrt{2}}{2}+j\left(\frac{4+3\sqrt{2}}{2}\right)$$

$$X(2)=-1-j \qquad X(3)=\frac{2-\sqrt{2}}{2}+j\left(\frac{4-3\sqrt{2}}{2}\right)$$

$$X(4)=0 \qquad X(5)=\frac{2-\sqrt{2}}{2}+j\left(\frac{4-3\sqrt{2}}{2}\right)$$

$$X(6)=-1+j \qquad X(7)=\frac{2+\sqrt{2}}{2}+j\left(\frac{4+3\sqrt{2}}{2}\right)$$

$$H(0)=6 \qquad H(1)=\left(1+\sqrt{2}\right)-j\left(3+\sqrt{2}\right)$$

$$H(2)=-2-j2 \qquad H(3)=\left(1-\sqrt{2}\right)+j\left(3-\sqrt{2}\right)$$

$$H(4)=2 \qquad H(5)=\left(1-\sqrt{2}\right)-j\left(3-\sqrt{2}\right)$$

$$H(6)=-2+j2 \qquad H(7)=\left(1+\sqrt{2}\right)+j\left(3+\sqrt{2}\right)$$

Y(k) = H(k) X(k)

Y(0) = 36	Y(1) = -14.07 - j17.48
Y(2) = j4	Y(3) = 0.07 + j0.515
Y(4) = 0	Y(5) = 0.07 - j0.515
Y(6) = -j4	Y(7) = -14.07 + j17.48

➔ IDFT

$$y(n)=\sum_{k=0}^{7} Y(k)e^{j2p\left(\frac{k}{8}\right)n} \qquad n = 0, 1, \ldots, 7$$

$\rightarrow$ y(n) = {1, 4, 9, 11, 8, 3, 0, 0}

$\downarrow\downarrow$

zeropad akibat 8 point

➔ aliasing akan terjadi bila N<M+L-1

4. Fast Fourier Transform (FFT)

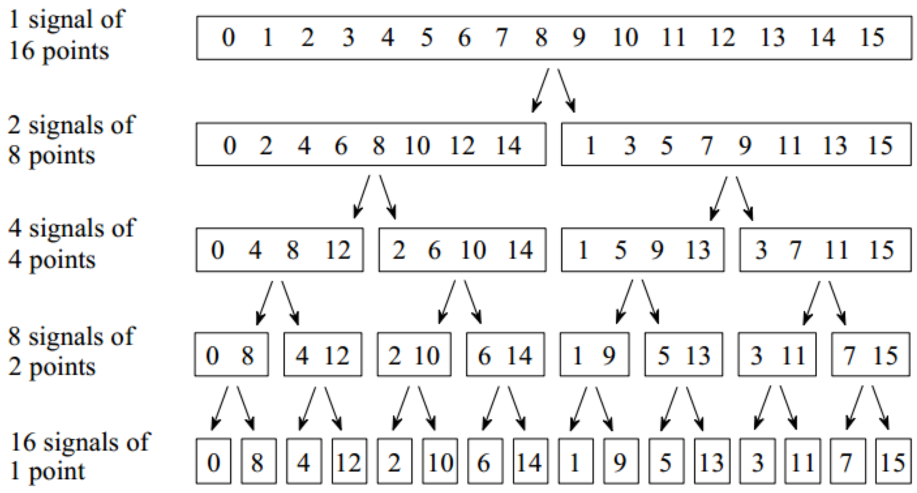
Alternatif lain untuk menghitung DFT (selain menggunakan rumusan di atas) adalah menggunakan algoritma cepat yang dikenal dengan Fast Fourier Transform (FFT). Dengan FFT maka waktu yang dibutuhkan untuk menghitung DFT menjadi lebih cepat. Metode FFT bekerja secara rekursif dengan membagi vektor asli menjadi dua bagian, menghitung FFT masing-masing bagian, dan kemudian menggabungkannya. Hal ini mengindikasikan bahwa FFT akan menjadi sangat efisien jika panjang vektor merupakan bilangan pangkat 2. Tabel berikut memuat perbandingan jumlah operasi perkalian dalam perhitungan DFT menggunakan rumus DFT secara langsung dan menggunakan algoritma FFT.

2^N	Direct arithmetic	FFT	Increase in speed
4	16	8	2.0
8	84	24	2.67
16	256	64	4.0
32	1024	160	6.4
64	4096	384	10.67
128	16384	896	18.3
256	65536	2048	32.0
512	262144	4608	56.9
1024	1048576	10240	102.4

Kerja FFT

Operasi FFT mendekomposisi N point sinyal domain waktu ke dalam N sinyal domain waktu menjadi point tunggal. Langkah ke dua menghitung spektrum frekuensi yang terkait dalam N sinyal domain waktu hingga N spektrum disintesis ke dalam spektrum frekuensi tunggal. Berikut adalah contoh dekomposisi domain waktu dalam FFT. Pada contoh ini, digunakan 16 point yang dikomposisi melalui empat kondisi terpisah.

Kondisi pertama 16 point menjadi dua sinyal yang terdiri dari 8 poin, seterusnya hingga menjadi satu poin. Secara matematis dapat dirumuskan ada $\mathrm{Log}_2 N$ kondisi. Contoh 16 poin (2^4)=4 kondisi, 512 poin (2^7) = 7 kondisi, 4096 poin (2^{12}) = 12 kondisi, dst



Kebutuhan kalkulasi DFT

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \qquad W_N = e^{-j\frac{2p}{N}} = \cos\frac{2p}{N} - j\sin\frac{2p}{N}$$

$x(n) = x_r(n) + j\, x_I(n)$ maka $X(k) = X_R(k) + jX_I(k)$

1. $X_R(k) = \sum_{n=0}^{N-1}\left[x_r(n)\cos 2p\,\frac{k}{N}n + x_I(n)\sin 2p\,\frac{k}{N}n\right]$
2. $X_I(k) = -\sum_{n=0}^{N-1}\left[x_R(n)\sin 2p\,\frac{k}{N}n - x_I(n)\cos 2p\,\frac{k}{N}n\right]$

$W_N^{k+N/2} = -W_N^k\ (simetri) W_N^{k+N/2} = W_N^k$