Procesamiento de Señales
Fourier – Algoritmo FFT

Profesor Jesús Vega

Escuela de Ingeniería, Ciencia y Tecnología

Viernes 5 de Mayo de 2023

Relaciones de síntesis y análisis - Fourier

Señal Tiempo continuo

Señal periódica x(t)
$w_0 = {}^{2\pi}/_{T_0} = 2\pi f_0$

Transformada de Fourier

$$\begin{cases} X(j\omega) = \sum_{k=-\infty}^{k=\infty} 2\pi a_k \delta(\omega - k\omega_0) \\ x(t) = \sum_{k=-\infty}^{k=\infty} a_k e^{jk\omega_0 t} \end{cases}$$

Depende de $\boldsymbol{a_k}$

Serie de Fourier

$$\begin{cases} a_k = \frac{1}{T}\int_{\langle T\rangle} x(t)e^{-jk\omega_0 t} \\ x(t) = \sum_{k=-\infty}^{k=\infty} a_k e^{jk\omega_0 t} \end{cases}$$

Involucra una integral que dificulta crear un algoritmo eficiente…
Sería necesario aproximar **numéricamente** esta integral

Señal aperiódica y(t)

Transformada de Fourier

$$\begin{cases} Y(j\omega) = \int_{-\infty}^{\infty} y(t)e^{-j\omega t}dt \\ y(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} Y(j\omega)e^{j\omega t}dt \end{cases}$$

Son integrales que deben ser aproximadas

Relaciones de síntesis y análisis - Fourier

Señal Tiempo continuo

Señal periódica x[n]
$\Omega_0 = 2\pi/_N$

Transformada de Fourier

$$\begin{cases} X\left(e^{j\Omega}\right) = \sum_{k=-\infty}^{k=\infty} 2\pi a_k \delta(\Omega - k\Omega_0) \\ x[n] = \sum_{k=-\infty}^{k=\infty} a_k e^{jk\Omega_0 n} \end{cases}$$

Depende de $\boldsymbol{a_k}$

Serie de Fourier

$$\begin{cases} b_k = \frac{1}{N} \sum_{n=\langle N \rangle} y[n] e^{-jk\Omega_0 n} \\ y[n] = \sum_{k=\langle N \rangle} b_k e^{jk\Omega_0 n} \end{cases}$$

Tenemos dos sumatorias que pueden ser utilizadas para construir con un algoritmo eficiente

Señal aperiódica
$y[n]$

Transformada de Fourier

$$\begin{cases} M\left(e^{j\Omega}\right) = \sum_{n=-\infty}^{\infty} y[n] e^{-j\Omega n} \\ y[n] = \frac{1}{2\pi} \int_{2\pi} M\left(e^{j\Omega}\right) e^{j\Omega n} d\Omega \end{cases}$$

Tiene una integral, pero existe una simetría para $M\left(e^{j\Omega}\right)$

Como $\Omega \epsilon \mathbb{R}$ tendríamos que calcular $M\left(e^{j\Omega}\right)$ en diferente valores de Ω_i

Relaciones de síntesis y análisis - Fourier

Nos interesa obtener un algoritmo para la transformada de Fourier de una señal o secuencia de datos x[n]

Partiendo de $x[n]$ → Periódica con período **N**

Dos posibles formulas ...
Para construir un algoritmo eficiente

Serie de Fourier

$$\begin{cases} b_k = \dfrac{1}{N} \displaystyle\sum_{n=\langle N \rangle} y[n] e^{-jk\Omega_0 n} \\ y[n] = \displaystyle\sum_{k=\langle N \rangle} b_k e^{jk\Omega_0 n} \end{cases}$$

Donde $\Omega_0 = {}^{2\pi}/_{N}$

$$x[n] \underset{\text{S.F.T.D.}}{\rightarrow} b_k$$

Transformada de Fourier

$$\begin{cases} M(e^{j\Omega}) = \displaystyle\sum_{n=-\infty}^{\infty} y[n] e^{-j\Omega n} \\ y[n] = \dfrac{1}{2\pi} \displaystyle\int_{2\pi} M(e^{j\Omega}) e^{j\Omega n} d\Omega \end{cases}$$

Donde $\Omega_0 = {}^{2\pi}/_{N}$

$$x[n] \underset{\text{T.F.}}{\rightarrow} M(e^{j\Omega})$$

¿Qué nos interesa?

1. Construir un algoritmo que (no gaste mucho tiempo) posea un número de operaciones internas reducido.
2. Que así como x[n] es una secuencia de datos, su T.F. sea otra secuencia de datos.

por esto → Utilizamos la **S.F.T.D.** para construir un algoritmo que posea un número de operaciones reducido.

Relaciones de síntesis y análisis - Fourier

Dos aspectos:

1. Existen funciones que se encuentran en el dominio del tiempo y de la frecuencia.
2. Con la S.F. ……podemos expresar una señal como una combinación de funciones sinusoidales.

$$\underset{\textbf{Periódica}}{x(t)} \xleftrightarrow{\text{S.F.}} \begin{cases} a_k = \dfrac{1}{T}\displaystyle\int_{\langle T\rangle} x(t)e^{-jk\omega_0 t} \\ x(t) = \displaystyle\sum_{k=-\infty}^{k=\infty} a_k e^{jk\omega_0 t} \end{cases}$$

Se requiere:

1. Que x(t) sea periódica.
2. Que k relacione las diferentes frecuencias… $\mathrm{k} \, \epsilon \, \mathbb{Z}^{\mp}$
3. Conocer el período de la señal x(t)

Las funciones que componen x(t) son sinusoidales de la forma

$$y(t) = C_1\, cos\left(\omega t + {}^{\pi}/_{2}\right)$$

Esta función tiene una fase.
Con el período T, calculamos frecuencia de y(t)….$f_0 = {}^{1}/_{T}$

$$y(t) = C_1\, cos\left(2\,\pi\, f_0\, t + {}^{\pi}/_{2}\right)$$

Var. dependiente

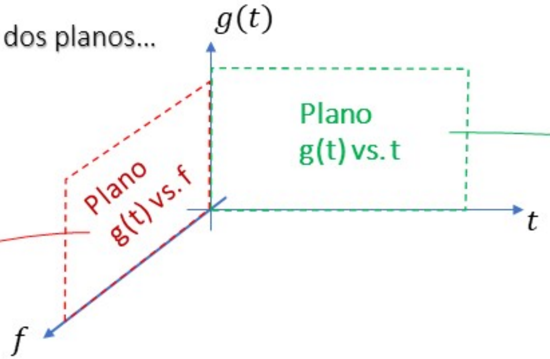
Var. Independiente

Var. Independiente

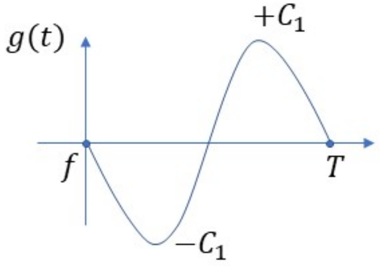
Desfase

Relaciones de síntesis y análisis - Fourier

Con estas variables podemos construir dos planos…



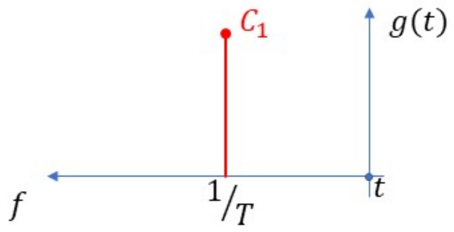
Es la proyección de la sinusoidal sobre el plano g(t) vs. t

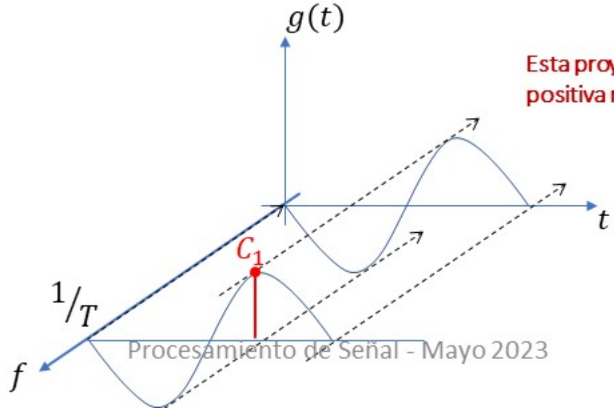


En este plano, el eje t es un punto, tenemos funciones sinusoidales en cada punto del eje f

Para nuestro caso…

$$\mathrm{g}(t) = C_1\, cos(2\pi f_0 t + {}^{\pi}/_{2})$$





Esta proyección es una medida de la amplitud, positiva máxima de g(t)

Relaciones de síntesis y análisis - Fourier

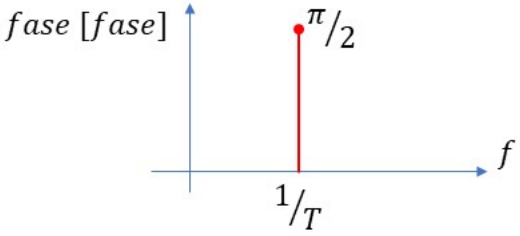
De acuerdo con la anterior…

¿Para construir $y(t) = C_1\ cos(\omega t +\ \varphi)$ necesito solamente información de frecuencia (f), período (T) y magnitud?

No

También se requiere la fase que es el punto que ubica el valor de y(t) en t=0

La fase en función de la frecuencia



La fase puede ser
Positiva o Negativa

Conclusión:

Para construir $y(t) = C_1\ cos(\omega t +\ \varphi)$ se requiere ***frecuencia*, *magnitud*** y ***fase***

Esta conclusión permite entender que un par de transformadas de Fourier requieren frecuencia, magnitud y fase

Relaciones de síntesis y análisis - Fourier

De acuerdo con la anterior…

$$b_k = \frac{1}{N}\sum_{n=\langle N\rangle} y[n]e^{-jk\Omega_0 n} = \frac{1}{N}\left[\sum_{n=0}^{N-1} y[n]\left(e^{-jk\Omega_0 n}\right)\right]$$

Donde…
N: número de muestras de la señal
Ω_0 : Frecuencia (rad/muestra) de la señal de tiempo discreto

Ejemplo:
Tenemos una señal con cuatro (4) muestras y con los valores de señal y[0], y[1], y[2], y[3]

Entonces,

$$b_k = \frac{1}{N}\sum_{n=0}^{N-1} y[n]\left(e^{-jk\Omega_0 n}\right) = \sum_{n=0}^{3} y[n]\left(e^{-jk\Omega_0 n}\right)$$

$$b_k = \underbrace{y[0]\left(e^{-jk\Omega_0 (0)}\right)}_{\text{Primera Multiplicación}} + \underbrace{y[1]\left(e^{-jk\Omega_0 (1)}\right)}_{\text{Segunda Mult.}} + \underbrace{y[2]\left(e^{-jk\Omega_0 (2)}\right)}_{\text{Tercera Mult.}} + \underbrace{y[3]\left(e^{-jk\Omega_0 (3)}\right)}_{\text{Cuarta Mult.}}$$

N Multiplicaciones ← **Primera** Multiplicación

N-1 Suma ← Primera Suma

Segunda Suma

Tercera Suma

Estas **N multiplicaciones** y **N-1 sumas**….
permiten calcular la **k-ésima** muestra de la DTFT.

↓

Entre mas grande es N, mas tiempo consume en PC

Relaciones de síntesis y análisis - Fourier

El algoritmo de la FFT:

1. Puede ser analizado desde dos puntos de vista: **diezmado de tiempo** y diezmado de frecuencia.
2. Es simplificado si tenemos el número de muestras N de la señal como una potencia de 2.

Sabemos que...
$$\begin{cases} b_k = \dfrac{1}{N}\displaystyle\sum_{n=\langle N\rangle} y[n]e^{-jk\Omega_0 n} \\ y[n] = \displaystyle\sum_{k=\langle N\rangle} b_k e^{jk\Omega_0 n} \end{cases}$$

Definimos...
$$W_N = e^{-j\Omega_0}$$

entonces...
$$\begin{cases} b_k = \dfrac{1}{N}\displaystyle\sum_{n=\langle N\rangle} y[n]\, W_N{}^{kn} & 0 \le k \le N-1 \\ y[n] = \displaystyle\sum_{k=\langle N\rangle} b_k\, W_N{}^{-kn} & 0 \le n \le N-1 \end{cases}$$

Diezmado de tiempo

Dividimos la secuencia y[n] de N puntos en dos secuencias de (N/2) puntos cada una, par e impar.

Ejemplo,

Tenemos y[0], y[1], y[2], y[3].......lo cual nos da N=4 puntos

Con N=4 puntos, sacamos dos secuencias,

y[0] , y[2] .... secuencia de datos con N/2 puntos – secuencia par
y[1] , y[3] .... secuencia de datos con N/2 puntos – secuencia impar

Dos secuencias, par e impar

Generalizando...

Para una señal con **N** puntos, es decir, una secuencia con **N** muestras.

$$\underbrace{y_0\,, y_2\,, y_4\,, \dots y_{N-2}} \qquad \underbrace{y_1\,, y_3\,, y_5\,, \dots y_{N-1}}$$

Impares	Pares	Para **N impar**
Pares	Impares	Para **N par**
Son N/2 muestras	Son N/2 muestras	

Relaciones de síntesis y análisis - Fourier

¿Qué sucede si **N** es impar? ¿Como dividimos las muestras?

$$y_n = \{ y_0 , y_1 , y_2 , y_3 , y_4 , y_5 \} \rightarrow \frac{N}{2} = \frac{5}{2} = 2.5$$

Una de las secuencias es mas grande ….en una muestra

Entonces

$$\underbrace{y_0 , y_2 , y_4 \rightarrow \boxed{y_{N-1}}}_{\text{Pares}} \qquad \underbrace{y_1 , y_3 \rightarrow \boxed{y_{N-2}}}_{\text{Impares}}$$

Esto sucede cuando **N** es impar

Siguiendo con el caso N par,

Estos dos conjuntos hace que separemos la sumatoria en dos…

$$b_k = \underbrace{\frac{1}{N} \sum_{n=0}^{\frac{N}{2}-1} y[2n] \, {W_N}^{2kn}}_{\text{Pares}} + \underbrace{\frac{1}{N} \sum_{n=0}^{\frac{N}{2}-1} y[2n+1] \, {W_N}^{(2n+1)k}}_{\text{Impares}}$$

como

$$\underbrace{W_N = e^{-j\Omega_0} = e^{-j\frac{2\pi}{N}}}_{\text{Si } \mathbf{N=N/2}}$$

Entonces

$$W_{N/_2} = e^{-j\frac{2\pi}{N/_2}} = e^{-j2\left(\frac{2\pi}{N}\right)} = \left(e^{-j\Omega_0}\right)^2 = {W_N}^2$$

Relaciones de síntesis y análisis - Fourier

Por consiguiente...

$$b_k = \frac{1}{N}\sum_{n=0}^{\frac{N}{2}-1} y[2n]\, W_N{}^{2kn} + \frac{1}{N}\sum_{n=0}^{\frac{N}{2}-1} y[2n+1]\, W_N{}^{(2n+1)k}$$

$$\boxed{W_N{}^{2kn} = \left(W_N{}^2\right)^{nk} = \left(W_{N/2}\right)^{nk} = \boldsymbol{W_{N/2}{}^{nk}}}$$

$$\boxed{W_N{}^{(2n+1)k} = W_N{}^{2kn}\, W_N{}^k = \left(W_N{}^2\right)^{nk} W_N{}^k = \left(\boldsymbol{W_{N/2}}\right)^{\boldsymbol{nk}} \boldsymbol{W_N{}^k}}$$

$$b_k = \frac{1}{N}\sum_{n=0}^{\frac{N}{2}-1} y[2n]\ \boldsymbol{W_{N/2}{}^{nk}} + \frac{1}{N}\sum_{n=0}^{\frac{N}{2}-1} y[2n+1]\ \left(\boldsymbol{W_{N/2}}\right)^{\boldsymbol{nk}} \boldsymbol{W_N{}^k}$$

Sale de la sumatoria porque no depende de n.

$$b_k = \underbrace{\frac{1}{N}\sum_{n=0}^{\frac{N}{2}-1} y[2n]\ \boldsymbol{W_{N/2}{}^{nk}}}_{G_k} + \frac{1}{N}\boldsymbol{W_N{}^k}\underbrace{\sum_{n=0}^{\frac{N}{2}-1} y[2n+1]\ \left(\boldsymbol{W_{N/2}}\right)^{\boldsymbol{nk}}}_{H_k}$$

$$b_k = \frac{1}{N}\boldsymbol{G_k} + \frac{1}{N}\boldsymbol{W_N{}^k}\boldsymbol{H_k}$$

Son las Transformadas Discretas de Fourier de cada secuencia (par e impar)

Son DFTs de N/2 puntos y son periódicas, con **período N/2**

Por consiguiente...

$$\begin{cases} G_{k+\frac{N}{2}} = G_k \\ H_{k+\frac{N}{2}} = H_k \end{cases}$$

Relaciones de síntesis y análisis - Fourier

Además…
para

$$W_N{}^k \Longrightarrow W_N{}^{k+\left(\frac{N}{2}\right)} = W_N{}^k \; W_N{}^{\frac{N}{2}} = W_N{}^k e^{-j\frac{2\pi}{N}\left(\frac{N}{2}\right)} = W_N{}^k e^{-j\pi} = -W_N{}^k$$

entonces

$$W_N{}^k \Longrightarrow W_N{}^{k+\left(\frac{N}{2}\right)} = -W_N{}^k$$

Por consiguiente…

$$b_{k+\left(\frac{N}{2}\right)} = \frac{1}{N}\boldsymbol{G}_{\boldsymbol{k}+\left(\frac{N}{2}\right)} + \frac{1}{N}\boldsymbol{W_N}^{\boldsymbol{k}+\left(\frac{N}{2}\right)}\boldsymbol{H}_{\boldsymbol{k}+\left(\frac{N}{2}\right)}$$

$$b_{k+\left(\frac{N}{2}\right)} = \frac{1}{N}\boldsymbol{G_k} + \frac{1}{N}\left(-W_N{}^k\right)\boldsymbol{H_k}$$

$$b_{k+\left(\frac{N}{2}\right)} = \frac{1}{N}\boldsymbol{G_k} - \frac{1}{N}\left(W_N{}^k\right)\boldsymbol{H_k}$$

Podemos utilizar esta propiedad para reducir el # de cálculos (operaciones)

Los primeros **N/2** puntos	Los últimos **N/2** puntos

$$0 \le n \le \frac{N}{2} - 1$$

utilizando…

$$b_k = \frac{1}{N}\boldsymbol{G_k} + \frac{1}{N}\left(W_N{}^k\right)\boldsymbol{H_k}$$

$$0 \le k \le \frac{N}{2} - 1$$

utilizando…

$$b_{k+\left(\frac{N}{2}\right)} = \frac{1}{N}\boldsymbol{G_k} - \frac{1}{N}\left(W_N{}^k\right)\boldsymbol{H_k}$$

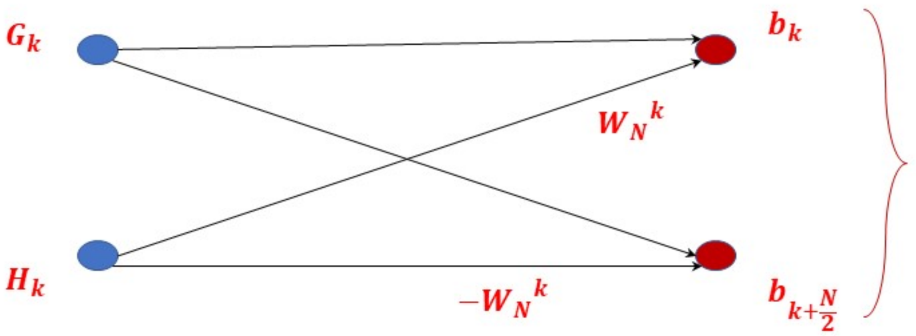
$$0 \le k \le \frac{N}{2} - 1$$

entonces…

Una DFT de N puntos → Puede ser,
Calculada combinando dos DFTs de N/2 puntos

Relaciones de síntesis y análisis - Fourier

Lo anterior puede ser dibujado como…



$b_k \quad + b_{k+\frac{N}{2}}$

Algoritmo Mariposa