Discrete Fourier Transform - DFT

$x[k] \in \mathbb{R}$ ó $\mathbb{C}$; $k \in \{0,1,2,\dots,K-1\}$

$X[n] \in \mathbb{R}$ ó $\mathbb{C}$; $n \in \{0,1,2,\dots,K-1\} \rightarrow w \in [0, 2\pi)$

$w \in [-\pi, +\pi)$

$$X[n] = \underbrace{\sum_{k=0}^{K-1}}_{\text{sumar}} \underbrace{x[k] e^{-jn\frac{2\pi}{K}k}}_{\text{Multiplicación}}$$

Por cada armónico

$w_0 = \frac{2\pi}{K} \rightarrow$ # muestras.

$x[k] = Re\{x[k]\} + j\, Im\{x[k]\}$

$w_n = n\frac{2\pi}{k}$

$$e^{-jn\frac{2\pi}{K}k} = Cos\left[n\frac{2\pi}{k}k\right] - j\, Sen\left[n\frac{2\pi}{k}k\right]$$

$$X[n] = \sum_{k=0}^{K-1} \left(Re\{x[k]\} + j\, Im\{x[k]\}\right)\left(Cos[w_n k] - j\, Sen[w_n k]\right)$$

$$= \sum_{k=0}^{K-1} \Big(Re\{x[k]\} Cos[w_n k] - j\, Re\{x[k]\}\, Sen[w_n k] + j\, Im\{x[k]\} Cos[\cdot] - j^2\, Im\{x[k]\}\, Sen[w_n k]\Big)$$

$$X[n] = \sum_{k=0}^{K-1} \Big(Re\{x[k]\} Cos[w_n k] + Im\{x[k]\}\, Sen[w_n k] + j\left[Im\{x[k]\} Cos[w_n k] - Re\{x[k]\}\, Sen[w_n k]\right]\Big)$$

0 en rFFT

Python FFT ; rFFT

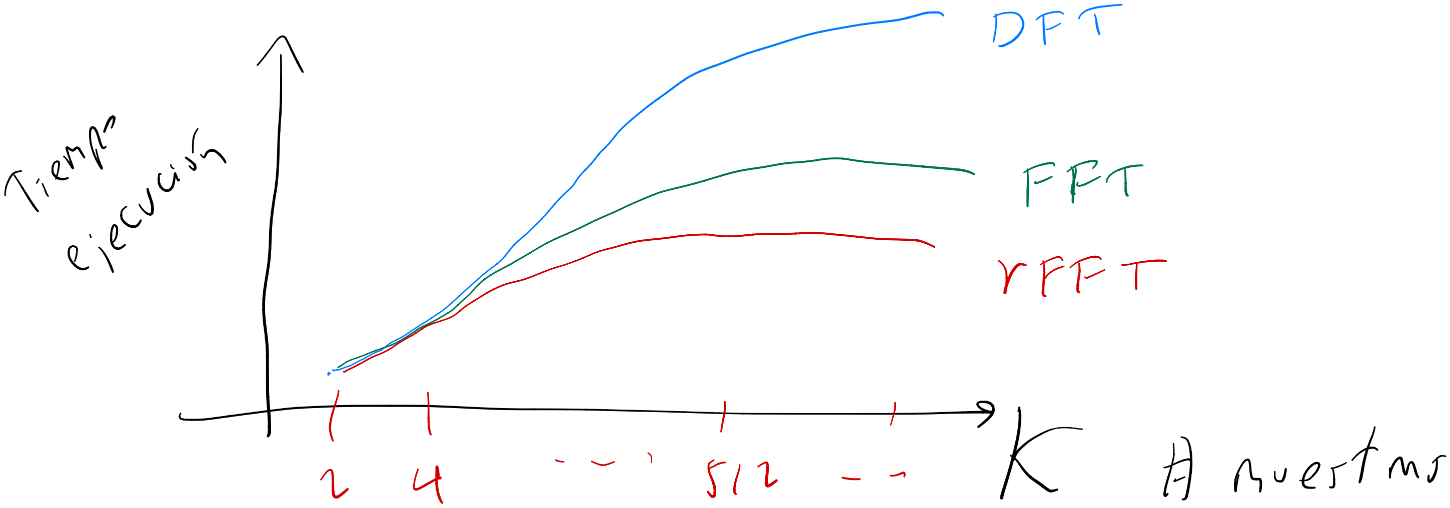
DFT → multiplicar y sumar Todos los contadores $O(N^2)$ complejidad computacional

FFT → aprovecha simetría y ahorra cuentas $O(N \log N)$

$$e^{jn\frac{2\pi}{k}k} = Cos(\theta) + j\, Sen(\theta)$$

rFFT ; $x(k) \in \mathbb{R}$; $x[k] = Re\{x[k]\} + j\, \cancel{Im\{x[k]\}}$ 0

$$X[n] = \sum_{k=0}^{K-1} \Big(Re\{x[k]\} Cos[w_n k] - j\, Re\{x[k]\}\, Sen[w_n k]\Big)$$


Tiempo ejecución
DFT
FFT
rFFT
2
4
512
K
muestras