Transformée de Fourier Rapide.

Position du problème.

$(x_n)_{n \in \mathbb{Z}}$ un signal à temps discret.

Evaluer numériquement la transformée de Fourier $\sum_n x_n e^{-2i\pi n f}$.

En pratique, x est observé de $n = 0$ jusqu'à $n = N - 1$.

Evaluer $\sum_{n=0}^{N-1} x_n e^{-2i\pi n f}$ **aux points** $\frac{k}{N}$ **pour** $k = 0, \ldots, N - 1$**.**

Remarque sur l'effet de la troncature.

$y_n = 0$ **si** $n < 0$ **ou** $n \geq N$**, et** $y_n = x_n$ **si** $0 \leq n \leq N-1$**.**

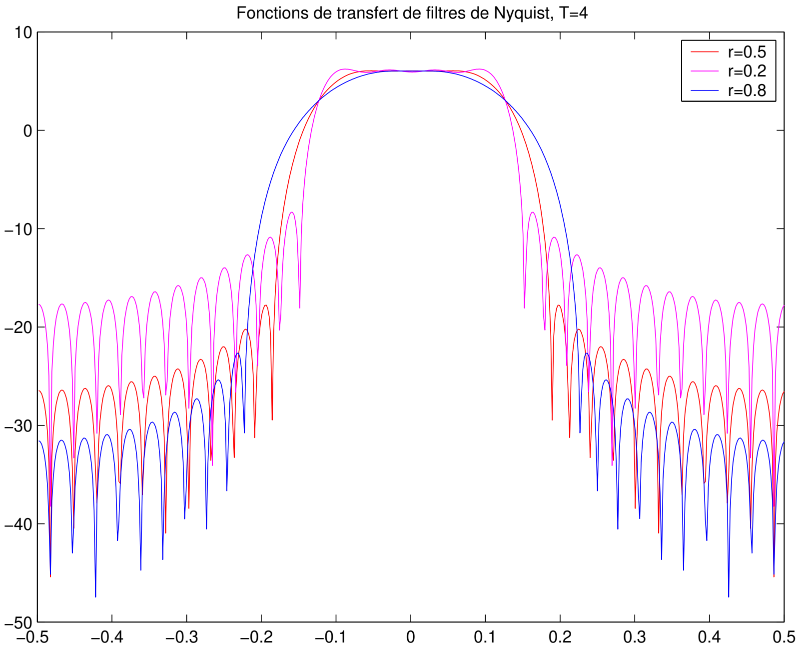
Soit $Y(f)$ **la TF de** y_n **:** $Y(f) = \sum_{n=0}^{N-1} x_n e^{-2i\pi n f}$**. Exprimer** $Y(f)$ **en fonction de** $\sum_{n \in \mathbb{Z}} x_n e^{-2i\pi n f}$**.**

$t_n = 0$ si si $n < 0$ ou $n \geq N$, et $t_n = 1$ si $0 \leq n \leq N-1$.

$y_n = t_n x_n$ **pour tout** $n \Longleftrightarrow Y(f) = X(f) * T(f)$

$T(f) = e^{-2i\pi \frac{(N-1)f}{2}} \frac{\sin(\pi N f/2)}{\sin(\pi f/2)}$.

Visualisation de l'effet de la troncature.



L'algorithme de FFT de Cooley-Tuckey I.

Calculer $X_N(k) = \sum_{n=0}^{N-1} x_n e^{-2i\pi \frac{nk}{N}}$ pour $k = 0, \ldots, \frac{N-1}{N}$.

Apparemment, nécessite N^2 opérations complexes.

Si N est une puissance de 2, nécessite $N \log_2(N)$ opérations.

Résultat très important en pratique.

L'algorithme de FFT de Cooley-Tuckey II.

L'idée de base de l'algorithme.

$X_N(k) = \sum_{n=0}^{N/2-1} x_{2n} e^{-2i\pi \frac{2nk}{N}} + \sum_{n=0}^{N/2-1} x_{2n+1} e^{-2i\pi \frac{(2n+1)k}{N}}$

Ceci s'écrit aussi :

$X_N(k) = \sum_{n=0}^{N/2-1} x_{2n} e^{-2i\pi \frac{nk}{N/2}} + e^{-2i\pi \frac{k}{N}} \sum_{n=0}^{N/2-1} x_{2n+1} e^{-2i\pi \frac{nk}{N/2}}$

Finalement, si $k \leq N/2 - 1$:

$$\begin{aligned} X_N(k) &= X_{N/2,1}(k) + e^{-2i\pi \frac{k}{N}} X_{N/2,2}(k) \\ X_N(k+N/2) &= X_{N/2,1}(k) - e^{-2i\pi \frac{k}{N}} X_{N/2,2}(k) \end{aligned}$$

L'algorithme de FFT de Cooley-Tuckey III.

$$\begin{aligned} X_N(k) &= X_{N/2,1}(k) + e^{-2i\pi\frac{k}{N}} X_{N/2,2}(k) \\ X_N(k+N/2) &= X_{N/2,1}(k) - e^{-2i\pi\frac{k}{N}} X_{N/2,2}(k) \end{aligned}$$

Si on connait $X_{N/2,1}(k)$ et $X_{N/2,2}(k)$ pour $k = 0, \ldots, N/2-1$, il faut N opérations complexes pour calculer $X_N(k =$ pour $k = 0, \ldots, N-1$.

Pour connaitre $X_{N/2,1}(k)$ pour $k = 0, N/2-1$ à partir de $X_{N/4,11}(k)$ et $X_{N/4,12}(k)$, il faut $N/2$ opérations complexes, idem pour connaitre $X_{N/2,2}(k)$ pour $k = 0, N/2-1$ à partir de $X_{N/4,21}(k)$ et $X_{N/4,22}(k)$: N opérations pour calculer les 2 FFT $N/2$ à partir des 4 FFT $N/4$.

Finalement, on obtient $N\log_2(N)$ opérations en itérant le processus.

La FFT inverse.

$X(k) = \sum_{n=0}^{N-1} x_n e^{-2i\pi \frac{kn}{N}}$

équivaut à

$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{2i\pi \frac{kn}{N}}$

FFT inverse : même complexité que la FFT.

FFT et convolution.

$x_0, \ldots, x_{N-1}$ et $y_0, \ldots, y_{N-1}$

$X(0), \ldots, X(N-1)$ et $Y(0), \ldots, Y(N-1)$ leurs FFT.

Quelle suite $z_0, \ldots, z_{N-1}$ admet pour FFT la suite
$Z(0) = X(0)Y(0), \ldots, Z(N-1) = X(N-1)Y(N-1)$?

$z_n = \sum_{l=0}^{N-1} x_{[l]} y_{[n-l]}$ où $[l]$ désigne la valeur de l modulo N.

$z = x *_c y$ produit de convolution circulaire entre x et y.

Application au filtrage rapide I.

Problème, calculer $y_n = \sum_{l=0}^{L-1} h_l x_{n-l}$ pour chaque n quand L est très grand.

De façon stupide, cout calcul de L opérations par échantillon

Calcul des y_n pour $L \leq n \leq 2L-1$ nécessite beaucoup moins que L^2 opérations quand L est une puissance de 2.

Application au filtrage rapide II.

Algorithme basé sur des FFT et IFFT $2L$ points.

$$\left\{ \begin{array}{cccccc} x_0 & \dots & x_{L-1} & x_L & \dots & x_{2L-1} \\ h_0 & \dots & h_{L-1} & 0 & \dots & 0 \end{array} \right\}$$

$z_n = (h *_c x)_n = \sum_{l=0}^{2L-1} h_{[l]} x_{[n-l]} = \sum_{l=0}^{L-1} h_l x_{[n-l]}$

Point important : si et $0 \leq l \leq L-1$ et $L \leq n \leq 2L-1$, $[n-l] = n-l$.

Donc, $z_n = \sum_{l=0}^{L-1} h_l x_{n-l} = y_n$ **si** $L \leq n \leq 2L-1$**.**

Application au filtrage rapide III.

$z_n = (h *_c x)_n = y_n$ si $L \leq n \leq 2L-1$.

Calculer z_n pour $n = 0, \ldots, 2L-1$, et extraire les valeurs $L \leq n \leq 2L-1$ pour récupérer y_n pour $L \leq n \leq 2L-1$.

- Calculer $H(k)$ et $X(k)$ pour $k = 0, 2L-1$: 2 FFT $2L$ points
- Calculer $Z(k) = H(k)X(k)$ pour $k = 0, 2L-1$
- Calculer z_n pour $n = 0, \ldots, 2L-1$: IFFT $2L$ points.

Nombre total d'opérations :

$2 \times 2L \log_2(2L) + 2L + 2L \log_2(2L) = 2L(1 + 3\log_2(2L))$ A comparer avec L^2.