UNIVERSITÉ DE RENNES 1

SIG : Initiation Traitement Signal - TP1

Rappels

Transformée de Fourier et Convolution

Soit un signal $x(t)$ à temps continu. On définit la Transformée de Fourier de x :

$$TF[x(t)] = X(\nu) = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi\nu t}dt$$

La TF inverse est donnée par :

$$TF^{-1}[X(\nu)] = x(t) = \int_{-\infty}^{+\infty} X(\nu)e^{+j2\pi\nu t}d\nu$$

Et son spectre par :

$$S(\nu) = |X(\nu)|$$

Rappelons aussi que le produit de convolution $y(t)$ entre $s(t)$ et $h(t)$ est donné par :

$$y(t) = s(t) * h(t) = \int_{-\infty}^{+\infty} s(\tau)h(t-\tau)d\tau$$

Et que le lien entre TF et produit de convolution est :

$$y(t) = s(t) * h(t) \overset{TF}{\leftrightarrow} Y(\nu) = S(\nu)H(\nu)$$

Définition et syntaxe des Transformées de Fourier Discrète et Rapide

Il n'est pas possible de travailler avec des signaux à temps continu sur un ordinateur. Ainsi, dans ce TP, on travaillera avec des signaux discrets définis de la façon suivante :

$$x_n = x[n] = \{x_0, x_1, \dots, x_{N-1}\}$$

x_n est égal à la valeur d'un signal continu $x(t)$ au temps $t = nT$ avec $n = 0, 1, \dots, N-1$ et T la période d'échantillonnage ($f_e = \frac{1}{T}$). C'est-à-dire : $x_n = x(t = nT), \forall\, n \in]-\infty;\ +\infty[$.
Ainsi, la Transformée de Fourier Discrète (DFT = Discrete Fourier Transform) est donnée par :

$$DFT[x_n] = X_k = \sum_{n=0}^{N-1} x_n e^{\frac{-i2\pi}{N}kn}$$

Il en résulte un signal discret X_k dans le domaine fréquentiel. Sa réciproque est donnée par :

$$DFT^{-1}[X_k] = x_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k e^{\frac{+i2\pi}{N}kn}$$

Matlab utilise la Transformée de Fourier Rapide (FFT = Fast Fourier Transform) qui est un algorithme de calcul de la DFT. Elle s'utilise avec les instructions *X = fft(x)* et *x = ifft(X).*

Utilisation correcte de la fonction fftshift

y = fftshift(x) décale le premier élément d'un vecteur vers son centre en échangeant la première avec la deuxième moitié. Exemple :

x = 1 :10 ;
y = fftshift(x)
y = 6 7 8 9 10 1 2 3 4 5

Partie 1 : Convolution et Transformée de Fourier (TF d'une porte)

Générer un signal 'porte' s d'une durée de 1 seconde avec pour fréquence d'échantillonnage $f_e = 1024\ Hz$, comme suit :

$$s(t) = \left\| \begin{array}{ll} 1 & pour - T \leq t \leq T \\ 0 & sinon \end{array} \right.$$

Avec $T = 25$ échantillons.

1/ Tracer $s(t)$ avec la fonction *plot*. Toutes les figures doivent avoir un titre et les axes labellisés.

2/ L'axe des fréquences étant défini par : *[-fe/2 : fe/length(s) : (fe/2 – fe/length(s))],* calculer le module (*abs*) et la phase (*angle*) de la FFT de s (que l'on notera S) sur l'axe des fréquences défini précédemment. Utiliser *subplot* pour tracer les différentes courbes sur une même figure. Commenter le résultat. Que doit-on faire pour une visualisation correcte du spectre ?

3/ Tracer *Y = fftshift(S).* Commenter la figure.

4/ Calculer les FFT inverses de Y et de S, les tracer sur la même figure et sur l'axe temporel. Que remarquez-vous ?

5/ Calculer le produit de convolution $x = s * s$. Tracer x.

6/ Calculer la FFT de x et comparer le résultat avec le carré de la FFT de s. Les résultats sont-ils cohérents ?

Partie 2 : TF d'un signal carré

1/ Créer un signal carré $s(t)$ d'une fréquence de 100 Hz à l'aide de la fonction square. On prend $f_e = 1\ kHz$. Afficher le signal ainsi créé sur 10 périodes.

2/ Calculer sa FFT sur 1024 points et afficher son module. Que doit-on conclure ?

Partie 3 : Débruitage d'un signal par FFT

TF et TF inverse

Générer un signal sinusoïdal s de fréquence $f = 10\ Hz$, d'une durée de 1 seconde, avec une fréquence d'échantillonnage $f_e = 1kHz$ et d'amplitude 3. Le tracer.

1/ Calculer et tracer le module de sa FFT.

2/ Calculer la FFT inverse (*ifft*) et la tracer. Commenter les résultats.

Bruit blanc

Un bruit blanc est un signal aléatoire qui a la même énergie pour toutes les fréquences.

3/ A l'aide de la fonction rand, générer un bruit blanc de même taille que le signal précédent. Tracer sur la même figure le signal et son spectre.

Bruitage du signal

4/ Ajouter au signal initial un bruit blanc avec un ratio d'amplitude de 50% inférieur à celui du signal de départ. Représenter ce signal bruité en temps et en fréquence (module) sur la même figure.

Filtrage par FFT

Repérer l'amplitude maximale *M* de la TF du signal bruité. Définir un seuil *S* (10% par exemple). Créer un filtre *H* selon les caractéristiques suivantes :

- *H* a la même taille que le signal (ou sa Transformée de Fourier).
- Lorsque l'amplitude de la FFT est plus petite que *S* x *M*, le filtre vaut 0. Sinon, il vaut 1.

5/ Appliquer le filtre à la FFT du signal bruité dans le domaine fréquentiel. Retrouve-t-on le signal de départ, sans le bruit additionnel ?

Etude paramétrique

6/ Recommencer les simulations précédentes en faisant varier les paramètres (seuil S, ratio d'amplitude entre le bruit et le signal).

7/ Recommencer les simulations avec un signal constitué de la somme de sinusoïdes de différentes fréquences.

$$s(t) = \cos(2f_1 t) + 3\cos(2f_2 t) - 6\cos(2f_3 t)$$

Où f_1, f_2 et f_3 sont trois fréquences différentes (par exemple : $f_1 = 10\ Hz$, $f_2 = 55\ Hz$ et $f_3 = 122\ Hz$). Commenter les résultats.