

Travaux Pratique

TR-C1 : Traitement du signal Avancée

TP 1 : Signaux Périodiques et FFT

Objectifs :

Le but de ce TP est d'établir :

- d'une part la relation entre le spectre en amplitude d'un signal périodique et son développement en série de Fourier
- d'autre part, prendre conscience de l'approximation apportée par la troncature d'une séquence périodique lorsqu'on ne respecte pas certaines conditions.

Nous utiliserons et compléterons le programme TP1_analyse.m.

Nous appliquerons ensuite notre programme à la constitution d'une note de musique (source : TP1_note.m).

Les objectifs fixés doivent permettre dans un second TP de synthétiser différents signaux périodiques à partir du relevé de leurs spectres.

Manipulations

I) Représentation temporelle

Nous allons à présent rappeler quelques généralités sur l'analyse de Fourier. La décomposition en série de Fourier d'un signal périodique de période T_0 (ou à durée limitée sur un intervalle T_0) s'écrit :

$$x(t) = \sum_{n=-\infty}^{\infty} x_n \exp\left(j2\pi n \frac{t}{T_0}\right)$$

où $x(t)$ est un signal périodique de période T_0 et x_n représente les coefficients de la série de Fourier.

Ces coefficients donnent une représentation en fréquence ou spectrale du signal. Pour passer de la représentation temporelle à la représentation spectrale, on utilise la formule suivante :

$$x_n = \frac{1}{T_0} \int_{T_0} x(t) \exp\left(-j2\pi n \frac{t}{T_0}\right) dt$$

1. Représenter le signal $x(t)=x_0+x_1*\sin(2\pi f_0t)$ en fonction du temps avec $x_0=3V$, $x_1=5V$, $f_0=100$ Hz, numérisé avec une fréquence d'échantillonnage de 1,4kHz pendant une durée d'acquisition de 100 milli-secondes.

La forme classique de programmation pour créer un vecteur est donnée par :

Temps=début : pas : duree-pas ;

La variable temps représente les instants d'échantillonnage (0 à Durée par pas de T_e)

Pour calculer une fonction sinusoïdale de période T_0 , en fonction du temps échantillonné tous les T_e :

```
sig_sin =sin(2*pi*f_0*Temps) ;
```

Pour afficher un signal sous forme de peigne de Diracs

```
stem(Temps, sig_sin)
```

2. Modifier la fréquence d'échantillonnage et représenter sur 4 graphes différents, le signal sinusoïdal à la fréquence d'échantillonnage 300 Hz, 200 Hz, 150Hz et 100Hz, COMMENTER
3. On nommera sig_sin_fe, ou fe vaudra respectivement 1400, 300, 200, 150, 100 le signal sinusoïdal échantillonné aux différentes fréquences 1400 Hz, 300 Hz, 200 Hz, 150 Hz, 100 Hz.
Afficher la taille de chaque signal par la fonction length sur la fenetre de commande Matlab (on tapera length(sin_sin_fe).
Que constatez vous ?
Pourquoi les tailles sont différentes alors que la durée de mesure est identique (100 ms)
4. On souhaite représenter sur un même graphe les signaux sig_sin_300 et sig_sin_150. On peut taper les commandes suivantes :

```
hold on
```

```
stem(t_150,sig_sin_150)
```

```
stem(t_300,sig_sin_300)
```

Cependant, comme la taille des deux fichiers est différente, on peut vouloir sur-échantillonner le signal de taille la plus petite (Fe_150) par 2

Nous allons utiliser la fonction upsample, qui consiste à rajouter des 0 entre deux échantillons et nous tronquerons éventuellement le signal échantillonné : A titre d'exemple, la fonction upsample(sig_sin_150,2) rajoutera un 0 entre deux échantillons de sig_sin_150, permettant ainsi de multiplier sa taille par 2.

Pour tronquer, il suffit de limiter la taille du fichier sur-échantillonné :

```
sig_sin_150_upsample=upsample(sig_sin_150,2);
stem(t_300,sig_sin_150_upsample(1:length(t_300)), 'r')
stem(t_300,sig_sin_300, 'r')
```

5. Modifier $F_0=180$ Hz, et comparer le signal avec celui à $F_e=200$ Hz et $F_e=1400$ Hz. Quelle est la période du signal ainsi générée. Concluez.

II) Représentation fréquentielle

Nous allons maintenant nous intéresser au spectre des signaux générés. Le spectre est obtenu par la fonction `fft`. Nous allons visualiser le module du spectre de chaque signal. Cependant, Matlab trace le spectre entre 0 et F_e , la périodicité du signal étant représenté autour de $F_e/2$. Il faut donc procéder à un décalage en fréquence du signal visualisé sous Matlab par la fonction `fftshift`, et une transposition des axes fréquentiels. Nous appliquerons donc la routine suivante :

```
L=length(t);
f=-L/2:1:L/2-1;
f=f*Fe_ref/L;
TF=fftshift(1/L*abs(fft(sig_sin,L)));
```

La majeure partie des problèmes d'analyse spectrale par calcul de transformée de Fourier discrète provient de la limitation de durée du signal. Cette limitation est évidemment indispensable, on ne peut pas mémoriser une séquence infinie. Limiter la durée de $x(t)$ revient à multiplier $x(t)$ par une fonction porte de largeur égale à T_p . La transformée de Fourier $X(f)$ sera alors convoluée par la transformée de Fourier de cette fonction porte, laquelle est un sinus cardinal $\text{sinc}(\pi f T_p)$. Au lieu d'avoir un spectre de raies pour un signal périodique on aura des fonctions en sinus cardinal autour de ces raies présentant différents lobes (un lobe principal ou central et des lobes secondaires) et qui donnent lieu à l'effet Gibbs. Les problèmes de résolution spectrale en analyse par transformée de Fourier discrète sont liés au recouvrement possible de ces lobes.

Exemple :

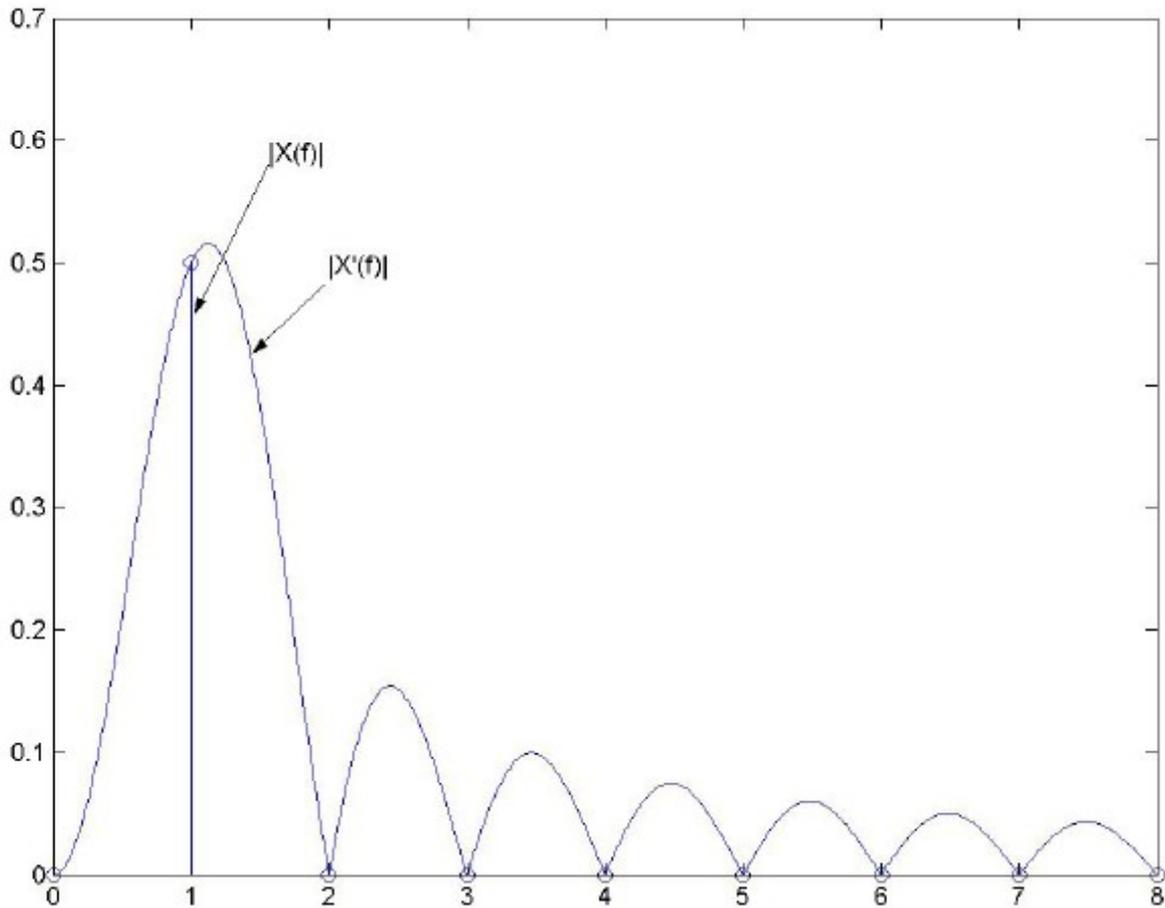
A titre d'exemple, prenons le cas d'une fonction cosinus de fréquence et d'amplitude unité : $x(t)=x_1*\sin(2\pi f_0 t)$ en fonction du temps avec $x_1=5V$, $f_0=100$ Hz, numérisé avec une fréquence d'échantillonnage de 1,4kHz.

Le spectre en amplitude de ce signal périodique est donc formé d'une impulsion de Dirac dans les fréquences positives située en $f_0=100$ Hz.

Pour calculer une TFD sur ce signal, on est amené à limiter le nombre d'échantillons du signal, donc la durée de celui-ci. On note alors $x'(t)$ le signal $x(t)$ limitée à un intervalle de temps allant de 0 à $\text{Durée}=T_p = 100$ ms

$$X'(f) = \frac{T_p}{2} \text{sinc}((f - f_0)T_p) \exp(-i\pi(f - f_0)T_p) + \frac{T_p}{2} \text{sinc}((f + f_0)T_p) \exp(-i\pi(f + f_0)T_p)$$

Le spectre en amplitude du signal limité en durée laisse alors apparaître des pics ou lobes dont le plus important est situé à $f_0=100$ Hz.



Le spectre précédent correspond au signal périodique **continu** sur une durée T_p .

Après échantillonnage de ce signal périodique limité à l'intervalle $[0, T_p]$, le spectre devient aussi périodique de période f_e . Ce spectre est déterminé à partir de la transformée de Fourier du signal échantillonné selon la formule suivante :

$$X_s(f) = f_e \sum_{k=-\infty}^{\infty} X'(f - kf_e)$$

La TF **Discrète** correspond à un ensemble d'échantillons prélevés sur la transformée de Fourier du signal échantillonné selon la formule :

$$X_n = X_s \left(f = \frac{n}{T_p} \right) = \sum_{m=0}^{N-1} x(t_m) \exp(-i2\pi \frac{mn}{N})$$

en notant N le nombre d'échantillon $N=T_p/T_e$

N correspond aussi au nombre d'échantillons prélevés sur $X_s(f)$ entre $[0, f_e]$. La résolution spectrale ainsi obtenue est de $\Delta f=1/T_p$. Nous allons dans la suite fixer le nombre de points N . Ceci impose la durée d'acquisition $T_p=(N-1)/f_e$.

1. Que se passe t'il sur le schéma précédent si la discrétisation du spectre est de pas $\Delta f=1$?
2. Représentation spectrale du sinus cardinal

Soit un signal sinusoïdal de fréquence 100 Hz, échantillonné avec $F_e=800$ Hz, sur une durée égale à 4 périodes. N , le nombre de point est donc égal à 32.

Nous allons prendre comme référence le spectre attendu en sinc. Regarder dans l'aide comment s'exprime la fonction sinc.

Nous tracerons donc :

```
fenetre_rect=sinc(f_fen*T0); avec :

Fe=800;
F0=100;
T0=1/F0;
Duree=4*T0;
N=Duree*Fe;

Pas_freq=Fe/N;
f_fen=0: Pas_freq : Fe-Pas_freq;
```

3. Tracer le spectre du signal sinusoïdal lorsque celui-ci est tronqué sur 4 périodes en utilisant la fonction ff.
4. Tracer le spectre du signal sinusoïdal lorsque celui-ci est tronqué sur 3.2 périodes.
5. Concluez sur les résultats.

III) Jouer une note sous Matlab

III.1) Génération d'une note : Mise en évidence du théorème de SHANNON.

Un la est une note à 440 Hz. Nous allons créer le signal en temporel avec une fréquence d'échantillonnage $f_e=8000$ Hz.

Afin de comparer le signal échantillonné avec le signal analogique, nous allons sur-échantillonner le signal crée par un facteur 15.

Nous compléterons les lignes suivantes

```
% définition de la nouvelle base de temps
t_pseudocontinu=[0:1/(15*fe):duree];

% Génération du signal sinusoidale sur la nouvelle base de temps

% Sur-echantillonnage du la par un facteur 15

% Visualisation des deux courbes
```

% plot pour le signal continu et stem pour le signal échantillonné.

1. Nous diminuerons ensuite la période d'échantillonnage à 2000 Hz, puis par pas de 100 à partir de 1200 Hz jusqu'à 600 Hz. Concluez par rapport à la note entendue et à la représentation spectrale, en expliquant bien la manipulation effectuée.
2. Nous allons maintenant reprendre le programme et le modifier pour garder constant $f_e=8000$ Hz, mais en augmentant la fréquence du signal sinusoïdale par pas de 1kHz entre 1 kHz et 15 kHz. Vous réaliserez une boucle for pour automatiser la construction du signal aux différentes fréquences avec une pause entre chaque itération de la boucle.
3. Réaliser un nouveau programme dans lequel vous générez une octave (do- re- mi – fa –sol – la –si –do)

```
Do 261.63 Hz;  
Re 293.66 Hz;  
Mi 329.63 Hz  
Fa 349.22 Hz  
Sol 392 Hz  
La 440 Hz  
Si 493.88 Hz  
do2 523.25 Hz
```

Jouer un do2+do

4. Représenter le spectre et le signal en temporel pour la note « la » sinusoïdale et pour un « la » carrée (Signal carré de fréquence 440 Hz)
5. Jouer un « la carrée ». Pourquoi l'amplitude du son est elle plus forte que pour un la sinusoïdal ?
6. Si vous trouvez la réponse à la question présente, taper les commandes suivantes

```
load handel;  
plot(y);  
soundsc(y, Fs);
```

Sinon, taper :

```
load handel;  
plot(y);  
soundsc(y, 4000);
```