TP n° 3

Transformée de Fourier discrète et Filtrage

Introduction

Dans ce TP, on se propose d'étudier la transformée de Fourier discrète telle qu'elle est implémentée sous matlab. On l'utilisera pour analyser des signaux simples ainsi que les signaux sonores enregistrés lors du précédent TP.

On filtrera finalement un signal afin de visualiser chaque composante du signal.

3.1 Transformée de Fourier discrète

Le calcul de la transformée de Fourier par un ordinateur pose le problème de la discrétisation de l'intégrale de Fourier $S(f) = \int_{-\infty}^{+\infty} s(t)e^{-j2\pi ft}dt$. En effet, la variable temporelle est discrète avec $t_n = n \times T_e$ (T_e est le pas de dicrètisation temporelle ou période d'échantillonage du signal $s(t)$). On devrait alors écrire:

$$S(f) = \sum_{n=-\infty}^{+\infty} s\left(nT_e\right) e^{-j2\pi fnT_e}. \tag{3.1}$$

Néanmoins, il est impossible de considérer un nombre infini de points et de plus, le signal $s(t)$ n'est défini que sur une durée limitée (par exemple entre 0 et $T_{max} = N \times T_e$, N est le nombres de point utilisé). On va alors supposer que $s(t)$ est périodique, ce qui revient à remplacer la transformée de Fourier par une série de Fourier.

De même que le temps est discrétisé, la fréquence aussi le sera. On pose $f = k \times \Delta f$ avec $\Delta f = 1/(NT_e)$. Δf est la résolution fréquentielle qui est d'autant meilleur que le produit NT_e est grand c'est à dire que le temps d'acquisition du signal $s(t)$ est grand. C'est l'intervalle de fréquence entre deux échantillons consécutifs du spectre (pas de discrètisation fréquentiel).

On écrira finalement la transformée de Fourier discrète (TFD) du signal $s(t)$:

$$\begin{aligned} S(k\Delta f) &= \sum_{n=0}^{N-1} s\left(nT_e\right) e^{-j2\pi k\Delta f nT_e} \\ &= \sum_{n=0}^{N-1} s\left(nT_e\right) e^{\frac{-j2\pi kn}{N}}. \end{aligned}$$

$S(k\Delta f)$ sera composée de N points avec une résolution fréquentielle de $\Delta f = 1/(NT_e)$. Ainsi, la longuer de l'intervalle dans l'espace de Fourier est l'inverse de la période d'échqntillonage et la résolution frèquentielle est l'inverse de la longuer de l'intervalle temporelle. Il est donc important de bien choisir les paramètre N et T_e en fonction du signal que l'on désire analyser.

La transformée de Fourier inverse s'écrit quant à elle:

$$s(nT_e) = \sum_{n=0}^{N-1} S(k\Delta f) e^{\frac{j2\pi kn}{N}}. \tag{3.2}$$

La transformée de Fourier discète induit une fausse-symétrie. En effet, si on calcule $S([N-k]\Delta f)$, on obtient:

$$\begin{aligned} S([N-k]\Delta f) &= \sum_{n=0}^{N-1} s(nT_e) e^{\frac{-j2\pi[N-k]n}{N}} \\ &= \sum_{n=0}^{N-1} s(nT_e) e^{j2\pi n} e^{\frac{j2\pi kn}{N}}, \end{aligned}$$

où $e^{j2\pi n} = 1$, ce qui nous donne finalement:

$$\begin{aligned} S([N-k]\Delta f) &= \sum_{n=0}^{N-1} s(nT_e) e^{\frac{j2\pi kn}{N}} \\ &= S(-k\Delta f). \end{aligned}$$

Si le signal $s(t)$ est réel (ce qui est très souvent le cas), on aura alors

$$S([N-k]\Delta f) = S(k\Delta f)*.$$

On obtient un spectre où seule la moitié des points est utile, la deuxième moitié n'étant que le complèxe conjugué de la première (symétrique par rapport à $f_e/2$.

Remarque: Le fait que la fréquence manimum observable soit égale à $f_e/2$ et non f_e (f_e est la fréquence d'échantillonage, $f_e = 1/T_e$) semble finalement logique quand on considère le théorème de Shannon.

3.2 Transformée de Fourier discrète sous matlab, FFT

La transformée de Fourier sous matlab est implantée avec la fonction **fft** et la transformée de Fourier inverse avec la fonction **ifft**. L'algorithme calculant la **fft** and la **ifft** est plus rapide si le nombre de points est une puissance de 2.

Remarque: Pour avoir comparer la transformée de Fourier et son inverse produit par matlab et les formules données en cours, il faut diviser la **fft** par N (le nombre de points) et inversement multiplier la **ifft** par N.

Essayer le code suivant et analyser les résultats.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              FFT et IFFT                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
close all
clc

% Creation de signal
% s(t) = Acos(2pif1t + Bcos(2pif2t))

A = 1; % l'amplitude de signal de la porteuse
B = 4; % l'amplitude de signal de l'information
f1 = 1000; % Hz, frequence de signal de la porteuse
f2 = 100; % Hz, frequence de signal de l'information

duree = 0.1; % sec, duree
Te = 1/(10*f1); % sec, le pas d'echantillonage
t = 0:Te:duree; % sec, vecteur de temps

s = A*cos(2*pi*f1*t + B*cos(2*pi*f2*t)); % signal

% Application de la transformee de Fourier
% Pour voir le signal dans le domaine frequentiel

N = length(t); % le nombre de points utilises (il doit etre egal au
               % nombre de points du signal dans le domaine temporel)
df = 1/(N*Te); % Hz, la resolution frequentielle
f = 0:df:(N-1)*df; % Hz, vecteur de frequence

S_c = (fft(s)/N); % transformee de fourier du signal s (complexe)
S = abs(S_c); % reel

% Recuperation de signal a l'aide d'une transformee de Fourier inverse

s1 = ifft(S_c)*N;
```

```
% Les resultats
figure
subplot(3,1,1)
plot(t, s) % signal dans le domaine temporel
xlabel('Temps (sec)')
ylabel('Amplitude')
title('Signal dans le domaine temporel')
subplot(3,1,2)
plot(f, S) % signal dans le domaine frequentiel
xlabel('Frequency, Hz')
ylabel('Amplitude')
title('Signal dans le domaine frequentiel (FFT)')
subplot(3,1,3)
plot(t,s1) % signal recupere
xlabel('Temps (sec)')
ylabel('Amplitude')
title('Signal recupere (IFFT)')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

3.3 Application

3.3.1 Signaux simples

Exercice 1

Prendre une période d'échantillonage $T_e = 0.1$ et un nombre de point égal à 10^4. Synthétiser un signal sinusoïdal de fréquence 1 Hz et d'amplitude 0,5. Calculer sa transformée de Fourier et l'afficher avec en abscisse la bonne échelle fréquentielle.

3.3.2 Spectre des signaux créneau, triangulaire, et dent de scie

---

Exercice 2

Synthétiser les signaux de créneau, de triangle et en dents de scie avec les paramètres suivants: amplitude = 2, frequence = 1 Hz, pas d'échantillonage = 0.001 et nombre de points = 5000. Pour chacun de ces signaux, tracer le spectre d'amplitude à láide de la fonction **fft** de Matlab. Vérifier que l'amplitude et la position des raies obtenues dans le spectre est bien conforme aux valeurs théoriques (les valeurs de a_n et b_n de serie de Fourier).

3.3.3 Filtrage

Exercice 3

Prende une période d'échantillonage $T_e = 1/2000$ et un nombre de points égal à 2^{10}. Prendre au hasard (en utilisant la fonction **rand** de matlab trois amplitude A_1, A_2, et A_3 sans les afficher (exemple: taper $A1 = rand(1,1); \hookleftarrow$). Le signal $s(t)$ sera alors la somme de trois composantes sinusoïdaux de fréquence $f_1 = 20$ Hz, $f_2 = 100$ Hz et $f_3 = 300$ Hz et d'amplitudes A_1, A_2 et A_3 respectivement:

$$s(t) = A_1 \sin(2\pi f_1 t) + A_2 (sin)(2\pi f_2 t) + A_3 \sin(2\pi f_3 t). \tag{3.3}$$

Faire alors l'analyse de Fourier de ce signal et en déduire les amplitudes de chacunes des composantes (on considèrera la résolution fréquentielle).

Construire trois filtres fréquentiels appropriés afin de pouvoir isoler les spectres de chancunes des composantes (fonctions Matlab à utiliser: **butter** et puis **filter**). Appliquer ensuite une transformation de

Fourier inverse aux spectres isolés afin de reconstruire les différentes composantes et les tracer.

3.3.4 Signal bruité

On considère un signal d'amplitude 2 et de fréquence 20 Hz. Syntheétiser et tracer ce signal pour t variant de 0 à 2 par pas de 0.0001.

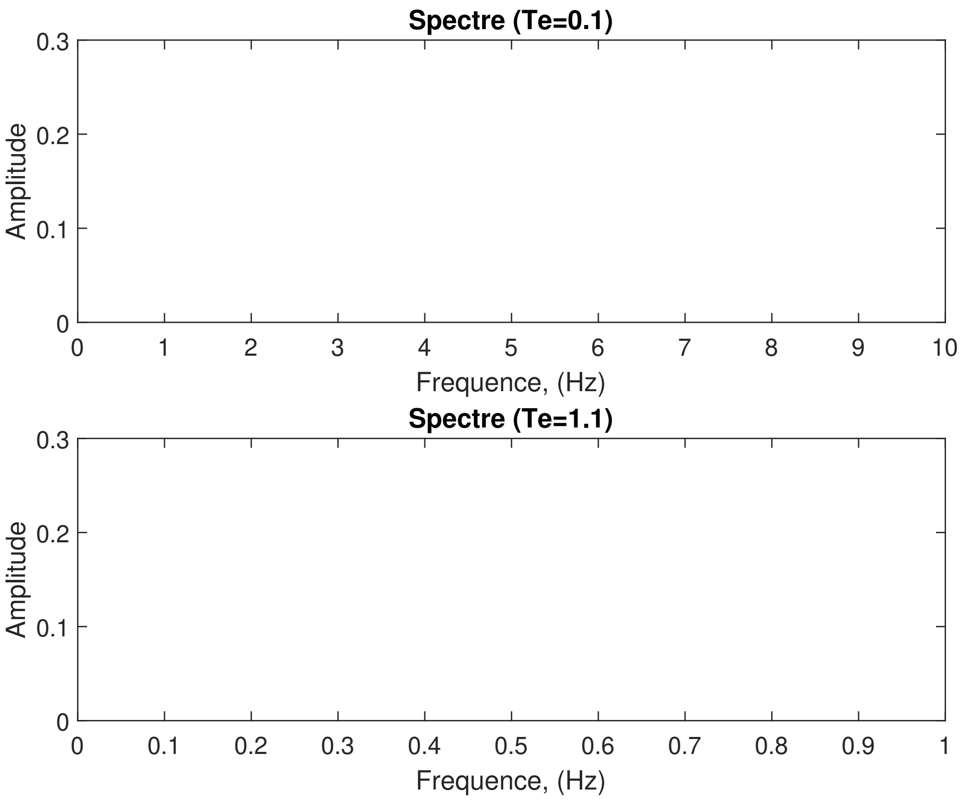
On ajoute un bruit de moyenne nulle par dessus notre signal. Soit Ab l'amplitude RMS de ce bruit (on pourra le faire varier afin d'appécier son effet sur le signal), exprimer ce bruit avec l'aide de la fonction **randn**. Observer l'allure du signal bruité et tracer son spectre.

Avec l'aide d'un filtre approprié, filtre le signal afin s'atténuer l'effet du bruit.

NOM: ________________________ DATE: __________

Traitement du Signal
Travaux Pratiques 3

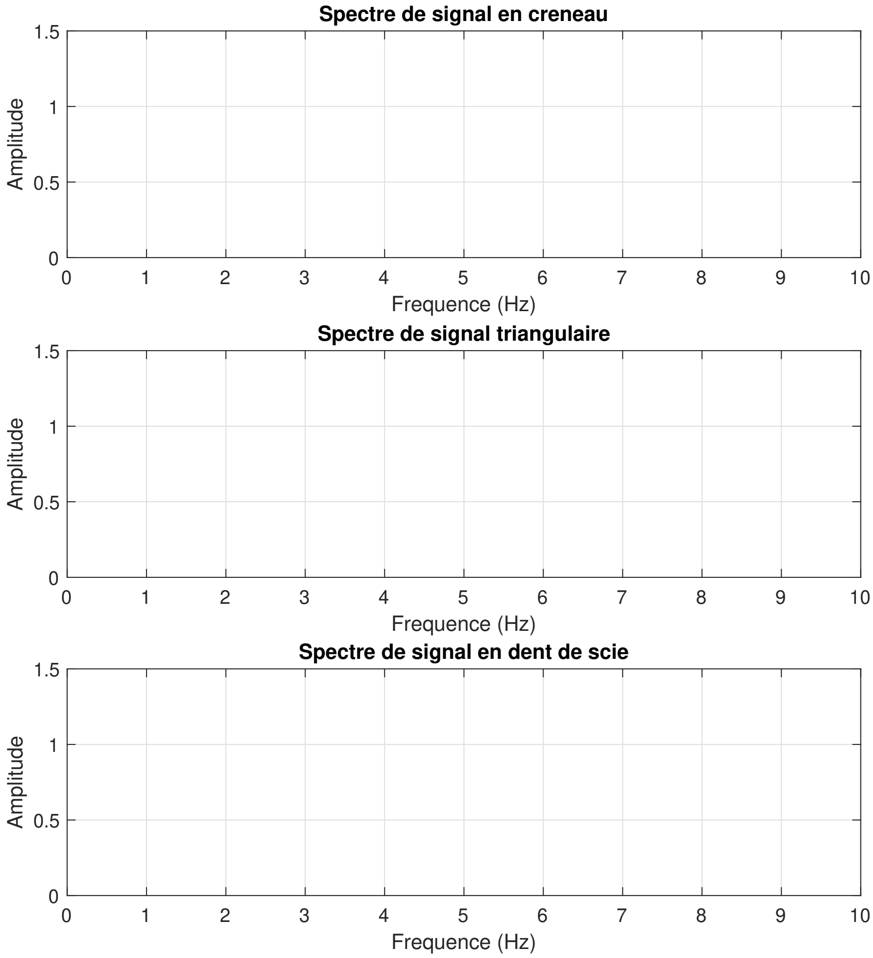
Exercice 1



Vérifier que l'amplitude correspond au résultat attendu. On pourra changer la valeur de l'amplitude du signal A.

Reprendre l'exercice avec une période d'échantillonage de 1.1. Qu'observez vous? Expliquez pourquoi cela s'est produit.

Exercice 2



Comparer les trois spectre.

Exercice 3

Créer quatre figures. La figure 1 contient le signal syntétisé et le spectre correspondant. Les figures 2 à 4 contiennent le spectre filtré et les signaux récupérés pour chaque fréquence.

Appeler l'enseignante pour vérifier.

Exercice 4

Tracer le signal bruité, son spectre, and le signal filtré sur la même figure mais différent graphiques.

Appeler l'enseignante pour vérifier.