

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed BOUDIAF - M'sila



Faculté de Technologie
Département de Génie Electrique

***MODULE :** TP Traitement du signal & TP Optimisation*

***ANNEE D'ETUDE :** 1^{er} Année Master Automatique et systèmes*

***ENSEIGNANT :** Dr. HERIZI Abdelghafour*

TP 3 : Transformée de Fourier Discrète

I. Objectif du TP :

L'objectif de ce TP est d'aborder les notions de base de la transformée de Fourier discrète. Puis, effectuer l'analyse spectrale des signaux non périodiques à l'aide des outils disponibles sous Matlab.

II. Rappel :

II.1 Echantillonnages et quantification :

La plupart des signaux (du moins ceux qui ont une origine naturelle) sont intrinsèquement analogiques. Il est donc logique que les machines permettant de créer ou de modifier ces signaux aient longtemps été elles-mêmes exclusivement analogiques. Le téléphone en est un bel exemple. Ce sont ces signaux analogiques qui nous intéressent dans ce cours. Cependant, depuis les années 70, on dispose de systèmes électroniques (CAN : convertisseur analogique numérique, ou ADC : analog-digital converter) permettant d'échantillonner et de quantifier les signaux analogiques, les transformant ainsi en signaux numériques.

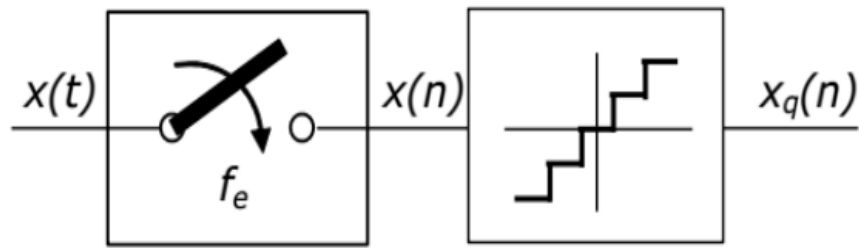
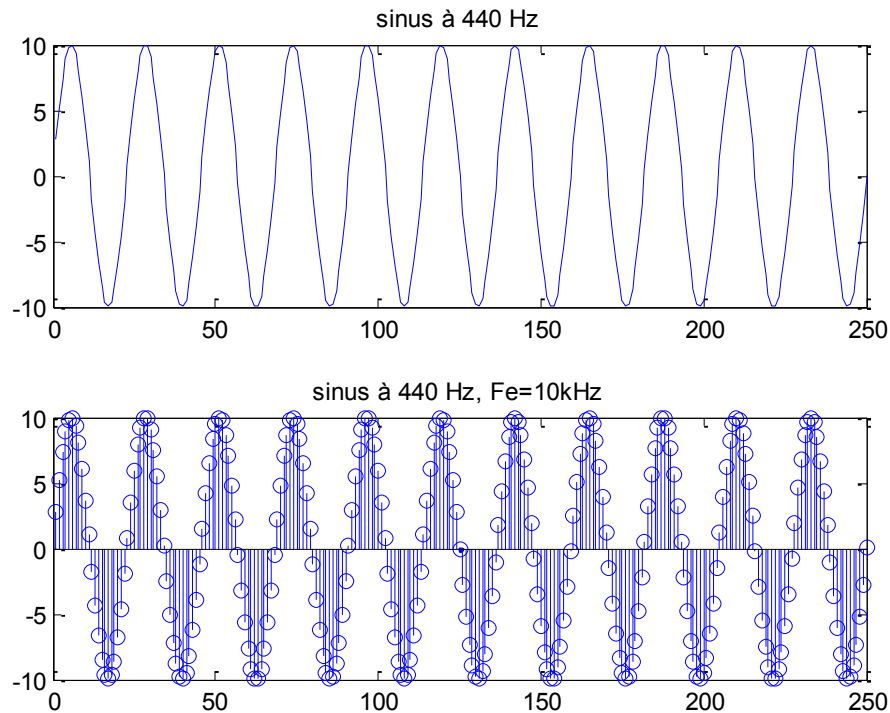


Figure 1 : Représentation schématique de l'échantillonnage et de la quantification d'un signal analogique.

Exemple de l'échantillonnage:

```
A=10; f=440; Fe=10000;
t=(1:20000)/Fe;
signal=A*sin(2*pi*f*t);
soundsc(signal,10000);
subplot(2,1,1);plot(signal(1:250));title('sinus à 440 Hz');
subplot(2,1,2);stem(signal(1:250));title('sinus à 440 Hz, Fe=10kHz');
```

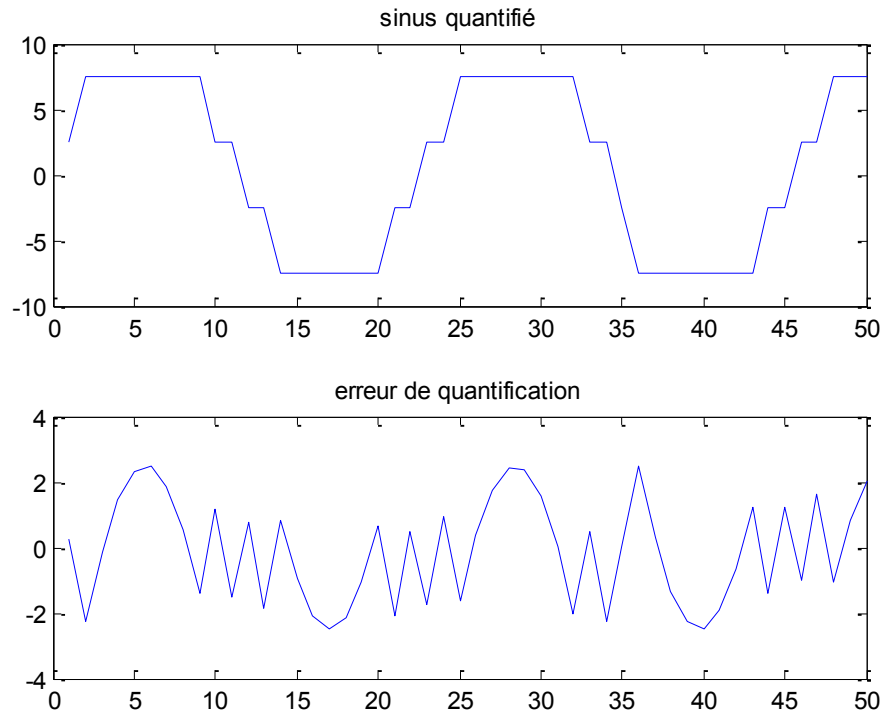


Exemple de quantification:

```
A=10; f=440; Fe=10000;
t=(1:20000)/Fe;
signal=A*sin(2*pi*f*t);

b=2;
quantizer_max=max(signal);
q=(2*quantizer_max)/2^b;
signal_quantized=floor(signal/q)*q+q/2;
error=signal-signal_quantized;

soundsc(signal_quantized,10000);
plot(signal(1:50));
hold on;
subplot(2,1,1);plot(signal_quantized(1:50));title('sinus quantifié');
subplot(2,1,2);plot(error(1:50));title('erreur de quantification');
```



II.2 Transformée de Fourier Discrète :

La transformée de Fourier à temps discret est conceptuellement plus simple que la transformée de Fourier d'un signal analogique, il n'en reste pas moins que le calcul de la TFTD nécessite en principe une charge de calcul infinie, puisque la série qui la définit comporte un nombre infini de termes, et que l'estimation doit en être faite pour toutes les valeurs de la fréquence normalisée F entre 0 et 1.

C'est la raison pour laquelle la transformée de Fourier Discrète ou TFD (en anglais Discrete Fourier Transform, ou DFT) a été introduite. Son calcul est en effet limité à un nombre fini de valeurs de n et pour un nombre fini de valeurs de F .

On appelle transformée de Fourier discrète d'une suite de N termes $x(0), x(1), \dots, x(N-1)$, la suite de N termes $X(0), X(1), \dots, X(N-1)$, définis par :

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi \frac{nk}{N}}$$

En pratique, les N termes $x(n)$ peuvent être N échantillons d'un signal analogique échantillonné: $x_n = x(nT_e)$, et les N termes $X(k)$ correspondre à une approximation de la transformée de Fourier de ce signal aux N points de fréquence $f_k = kf_e/N$, avec k entre 0 et $N-1$, c'est à dire f entre 0 et f_e .

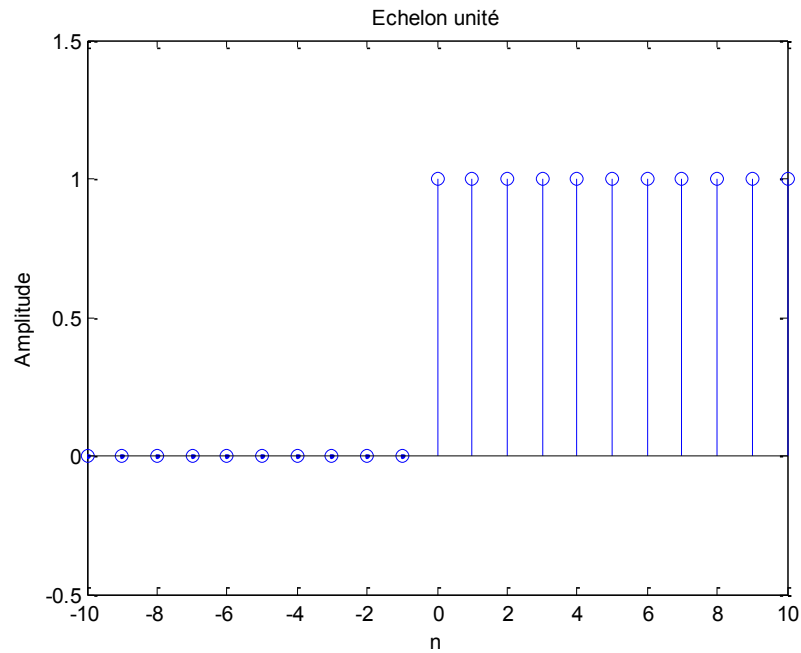
La transformée inverse de Fourier discrète défini par :

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi \frac{nk}{N}}$$

II.3 Signaux numériques :

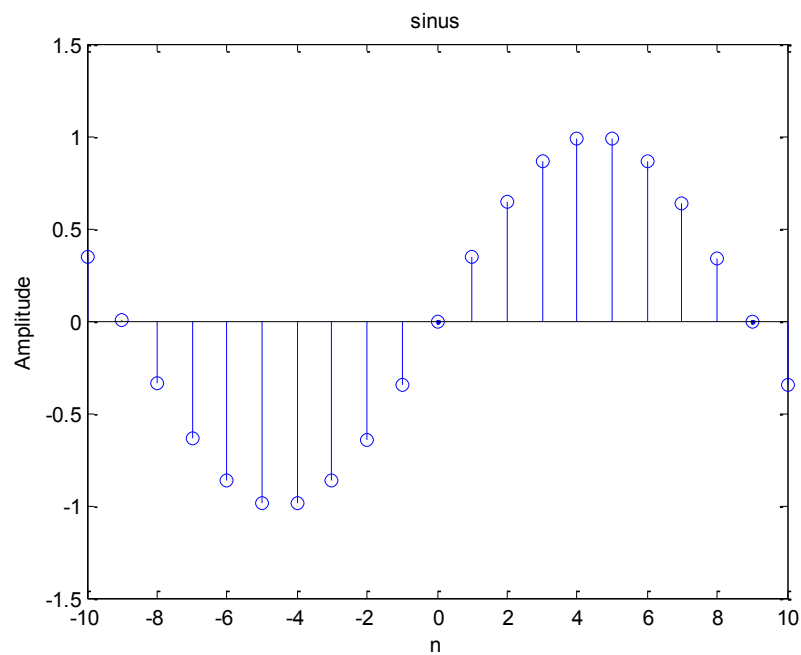
II.3.1 L'échelon unité :

```
t=-10:10;  
x=[zeros(1,10),ones(1,11)];  
stem(t,x);  
axis([-10 10 -0.5 1.5]);  
title('Echelon unité');  
xlabel('n');  
ylabel('Amplitude');
```



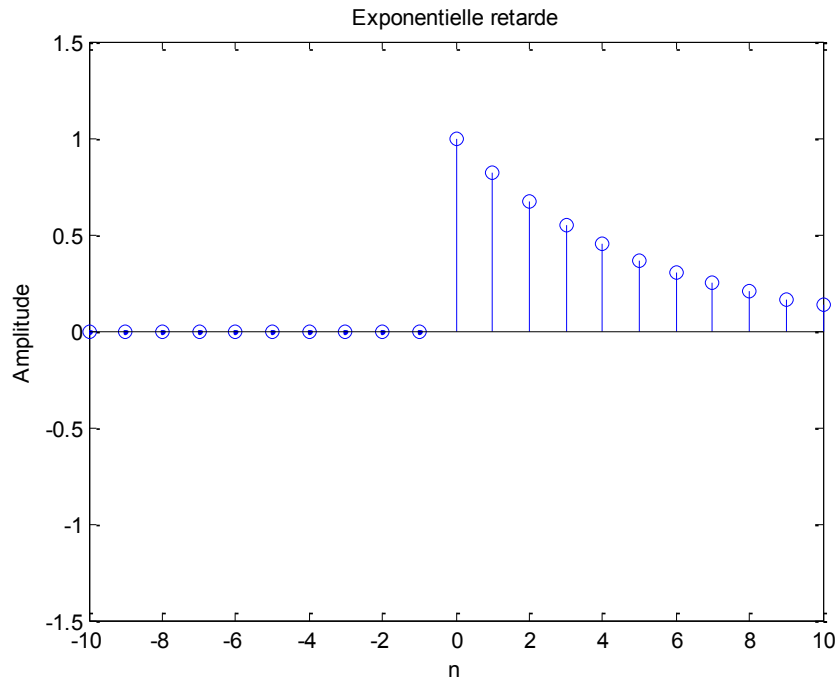
II.3.2 Sinus :

```
t=-10:10;  
x=sin(0.35*t);  
stem(t,x);  
axis([-10 10 -1.5 1.5]);  
title('sinus'); xlabel('n');  
ylabel('Amplitude');
```



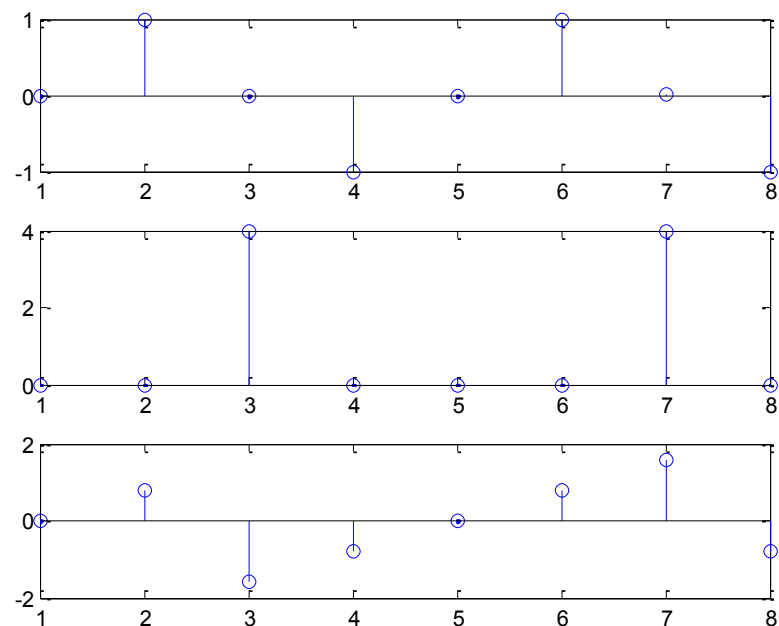
II.3.3 Exponentielle :

```
t=-10:10;  
u=[zeros(1,10),ones(1,11)];  
x=exp(-0.2*t).*u;  
stem(t,x);  
axis([-10 10 -1.5 1.5]);  
title('Exponentielle retarde');  
xlabel('n');  
ylabel('Amplitude');
```



II.3.4 Exemple de TFD :

```
x=sin(2*pi/8*2*(0:7));  
X=fft(x,8);  
subplot(3,1,1);stem(x);  
subplot(3,1,2);stem(abs(X));  
subplot(3,1,3);stem(angle(X));
```



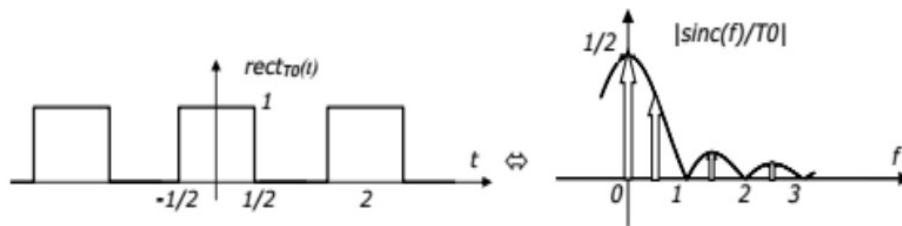
II.4 Transformée de Fourier Rapide :

En 1965, Cooley et Tukey proposèrent une méthode qui permet de réduire considérablement le temps de calcul de la TFD d'une suite dont le nombre d'échantillons N est décomposable en facteurs (typiquement, une puissance de 2). Par la suite, de nombreux algorithmes ont été publiés; ils sont connus sous le terme général de transformation de Fourier rapide (TFR ; ou FFT : Fast Fourier Transform). Tous ces algorithmes sont basés sur un même principe qui consiste à décomposer le calcul de la TFD en plusieurs TFD de longueur plus petite. La mise en œuvre de ce principe conduit à différentes méthodes dont les performances sont comparables. Nous décrirons ici plus particulièrement l'algorithme dit radix 2 avec entrelacement dans le temps

II.4.1 TFD Signaux périodique :

Rendons le signal rectangulaire $\text{rect}(t)$ périodique de période égale à 2s. sa transformée de Fourier fera apparaître des raies en $f=0, 1/2, 1, 3/2, \text{etc}$, de valeur $\text{sinc}(f)/2$

On constate que la transformée fait apparaître une raie en $f=0$ (composante continue= $1/2$), puis uniquement des harmoniques F^* impaires, d'amplitudes décroissant en $1/k$.

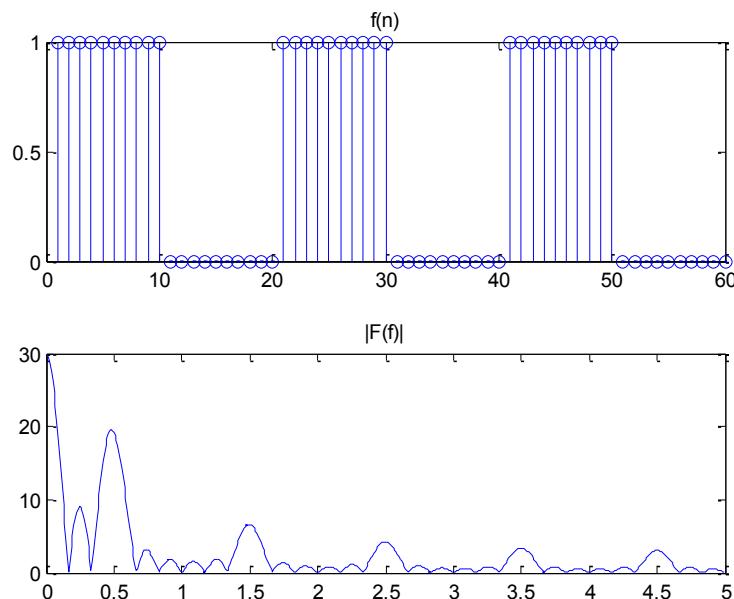


```
signal=[rectwin(10);zeros(10,1)]; %rectangle width : 1s =10samples
signal=[signal;signal;signal];
[fourier,frequencies]=freqz(signal); % computes the FT in f=[0,Fe/2]
```

```
subplot(2,1,1);stem(signal);title('f(n)')
```

```
Fe=10;
```

```
subplot(2,1,2);plot(frequencies/pi*(Fe/2),abs(fourier));title('|F(f)|');
```

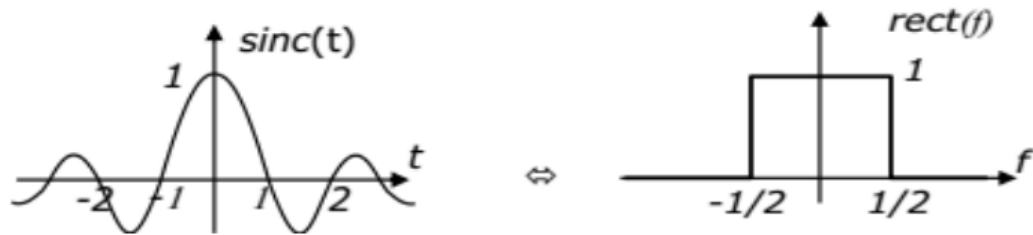


II.4.2 TFD Signaux non périodique :

Si le signal dont on cherche à observer le spectre est de durée finie, et que le nombre N d'échantillons dont on dispose couvre une plage temporelle supérieure ou égale à cette durée, l'effet de N est évidemment transparent : la TFD affiche alors N valeurs de la TFD originale.

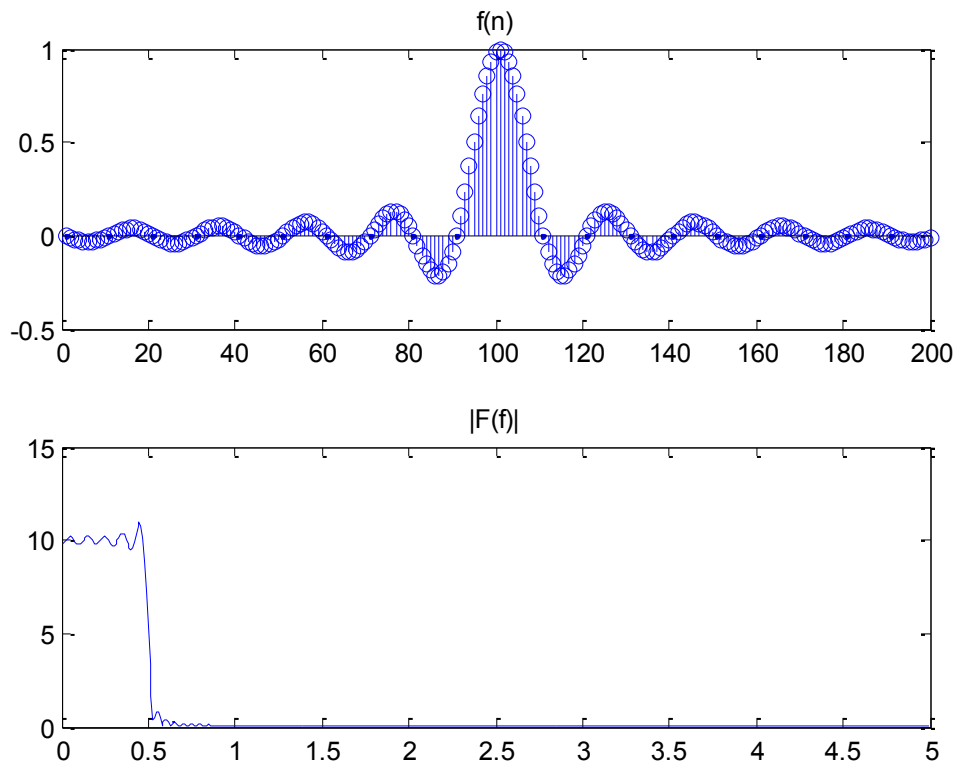
Transformée du sinus cardinal : Par application de la propriété de dualité de la transformée de Fourier, il vient :

$$\text{sinc}(t) \stackrel{\mathcal{F}}{\Leftrightarrow} \text{rect}(-f) = \text{rect}(f)$$



```
Fe=10; t=(-100:99)*1/Fe; %Fe=10; 200 samples ; 20 oscillations
signal=sinc(t);
[fourier,frequencies]=freqz(signal); % computes the FT in f=[0,Fe/2]

subplot(2,1,1);stem(signal);title('f(n)')
Fe=10;
subplot(2,1,2);plot(frequencies/pi*(Fe/2),abs(fourier));title('|F(f)|');
```



III. Manipulations :

III.1 Manipulation 1 :

On échantillonne une sinusoïde $f(t) = \sin(2\pi f_0 t)$ à la fréquence d'échantillonnage 10000hz. Dessinons l'allure des échantillons (c'est-à-dire l'allure de la fonction $f^*(t)$ correspondante) pour des valeurs de f_0 égales à : 1000hz, 5000hz, 7500hz, 9000hz et 31000hz. Pour que les graphiques possèdent des axes temporels identiques, choisissons de montrer les 10 premières ms de signaux.

Donner le programme ?

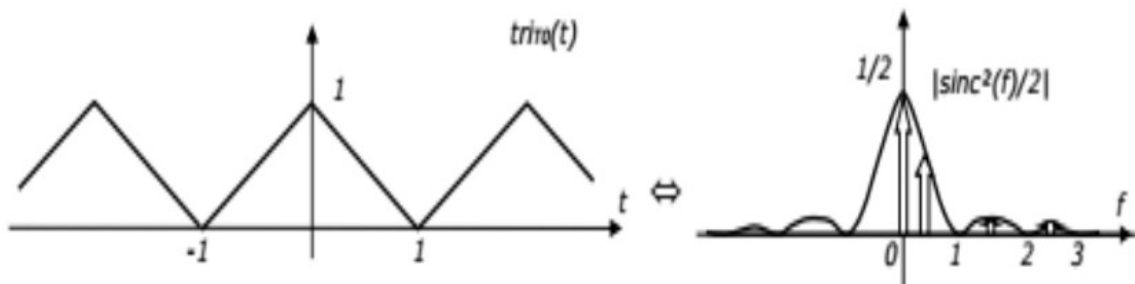
III.2 Manipulation 2 :

Considérons la TFD de la partie imaginaire (le sinus) d'un des vecteurs de base, par exemple w_2 pour $N=8$. Puisque $\text{Im}(w_2) = -j/2 (w_2 - w_2^*) = -j/2 (w_2 - w_{N-2})$, toutes les valeurs $X(k)$ de la TFD seront nulles, sauf $X(2)$ et $X(6)$ qui vaudront $-4j$ et $+4j$, ce qui correspond bien à une suite conjuguée modulo N .

Donner sous Matlab TFD de sin ?

III.3 Manipulation 3 :

Rendons le signal rectangulaire $\text{tri}(t)$ périodique de période égale à 2s. Sa transformée de Fourier apparaitra des raies en $f=0, 1/2, 1, 3/2$, etc, de valeur $\text{sinc}^2(f)/2$.



Donner le programme qui représente sa transformée de Fourier de ce signal ?

III.4 Manipulation 4 :

Calculons la TFD d'une fenêtre rectangulaire de 16 points, par FFT sur 16, 8, et 4 points.