

Traitement du signal avec Scilab : la transformée de Fourier discrète

L'objectif de cette séance est de valider l'expression de la transformée de Fourier Discrète (TFD), telle que peut la déterminer un calculateur numérique (ordinateur, DSP etc...) et de mettre en évidence ses principales propriétés. Les applications sont nombreuses et ne se limitent pas à la compréhension de la fonction « fft » de Scilab, une bonne connaissance de la transformation de Fourier discrète permettant par exemple une utilisation plus rationnelle de la FFT sur un oscilloscope numérique.

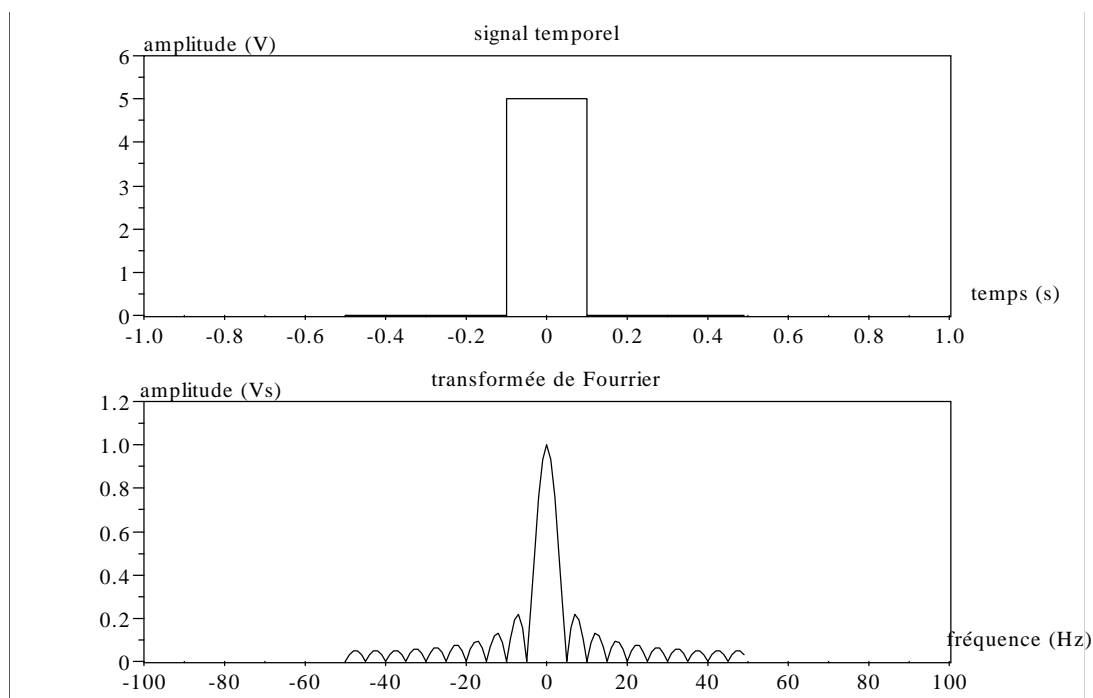
Dans une première partie, nous effectuerons une approche qualitative de la DFT, puis à partir de l'expression de la transformée analogique, nous établirons celle de la DFT. Suivra une démonstration plus rigoureuse, qui nous permettra de mettre en évidence certaines propriétés que nous vérifierons expérimentalement.

Remarque : Certaines questions proposent des démonstrations faisant intervenir sommations de séries et intégration de fonctions. Ceux que rebutent les calculs un peu compliqués pourront passer les questions, la réponse étant donnée dans l'énoncé de la forme « démontrer que ... ». L'important n'est pas tellement la démonstration, bien qu'elle permette une bonne approche du phénomène, ni même le résultat lui-même, mais ces conséquences.

1 Approche qualitative de la transformée de Fourier discrète

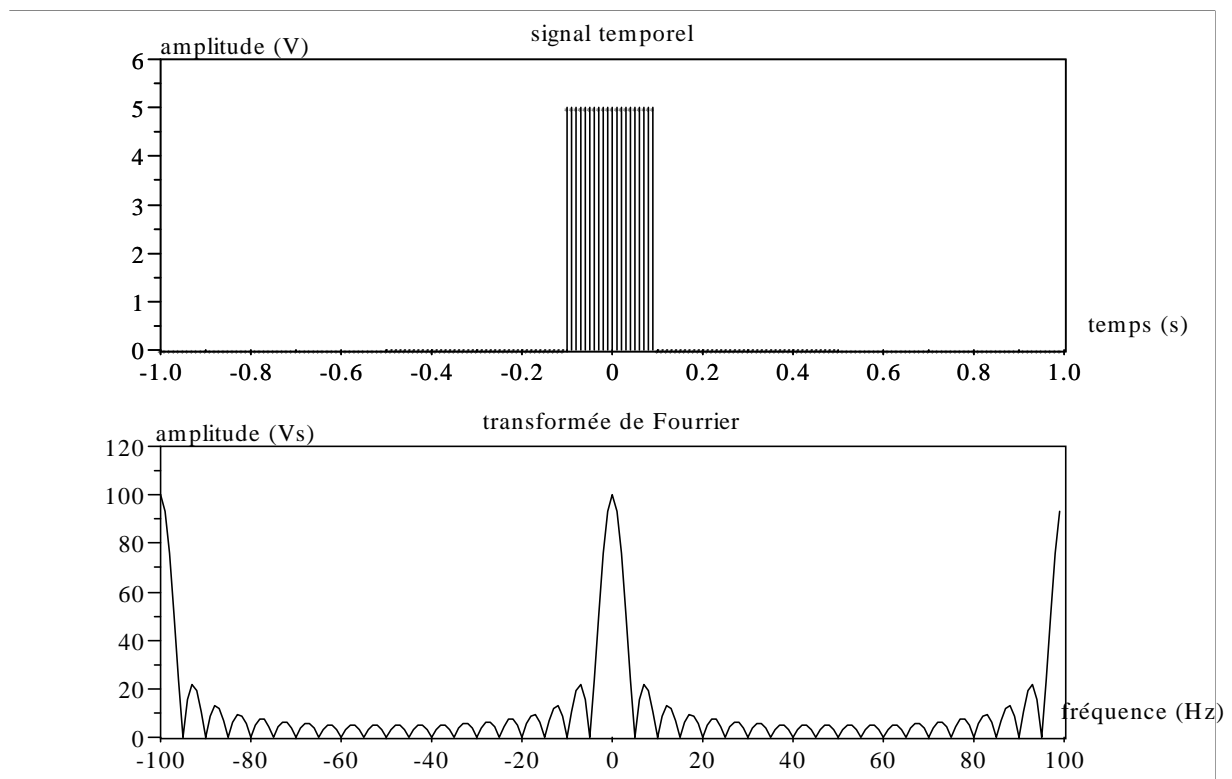
La transformée de Fourier discrète a des propriétés bien spécifiques qui la différencie de la transformée de Fourier « analogique ». Dans les figures ci-après, on peut observer, la transformée de Fourier (au sens analogique), attendue pour un signal porte d'amplitude A et de largeur T ; on retrouve pour le module de la transformée l'expression :

$$A.T.|\text{sinc}(\pi T)|$$



Cette fonction a une amplitude $A.T$ à la fréquence nulle et le premier lobe s'annule à $f=1/T$ comme on peut le constater sur la première figure.
 Dans notre cas l'amplitude du signal temporel vaut 5 V et sa largeur 0,2 s, l'amplitude à la fréquence nulle de la transformée vaut donc 1.

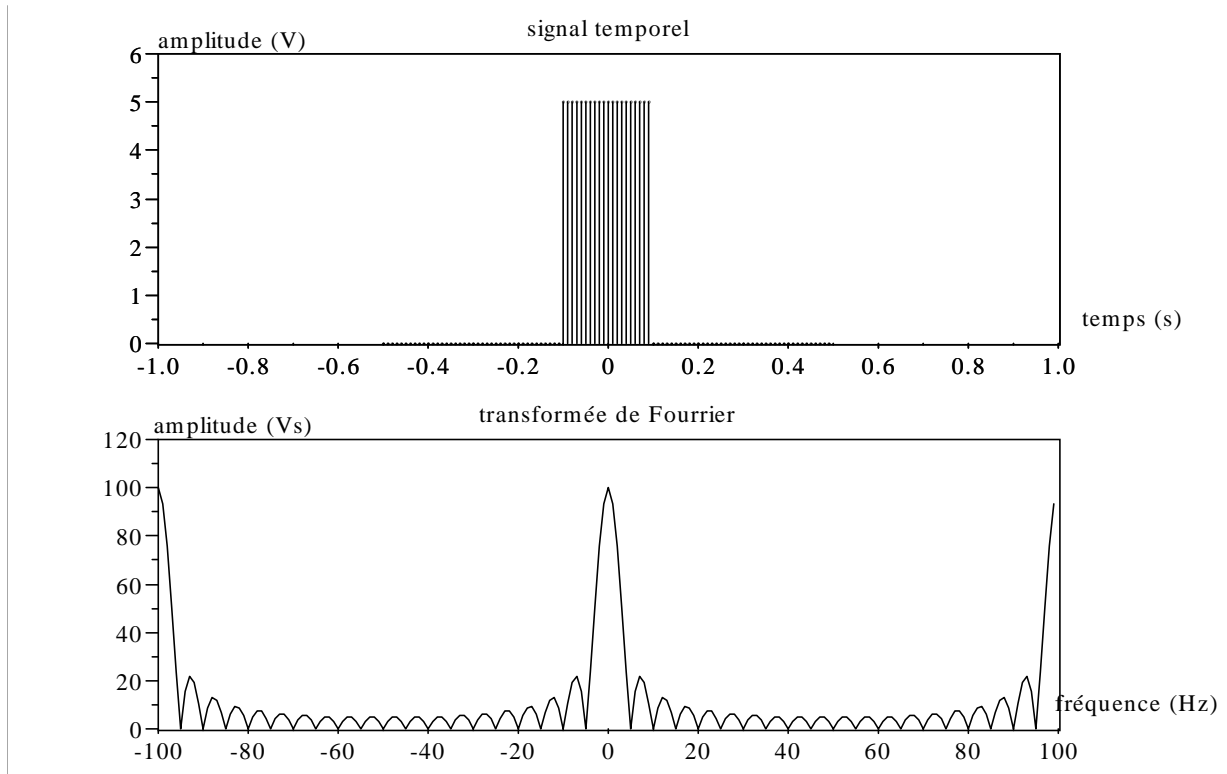
La transformation de Fourier n'est pas calculée sur $s(t)$, mais sur l'échantillonnage $s_e(t)$ de celui-ci. Si T_E est la période d'échantillonnage et F_E la fréquence correspondante (ici 100 Hz), échantillonner revient à multiplier dans le domaine temporel par un peigne de Dirac de période T_E et d'amplitude unitaire ; l'opération effectuée dans le domaine fréquentielle est donc une convolution avec un peigne de Dirac d'amplitude $1/T_E$ et de « période » F_E . Cette opération a pour effet de périodiser le spectre et de multiplier son amplitude par F_E , comme le montre la figure suivante :



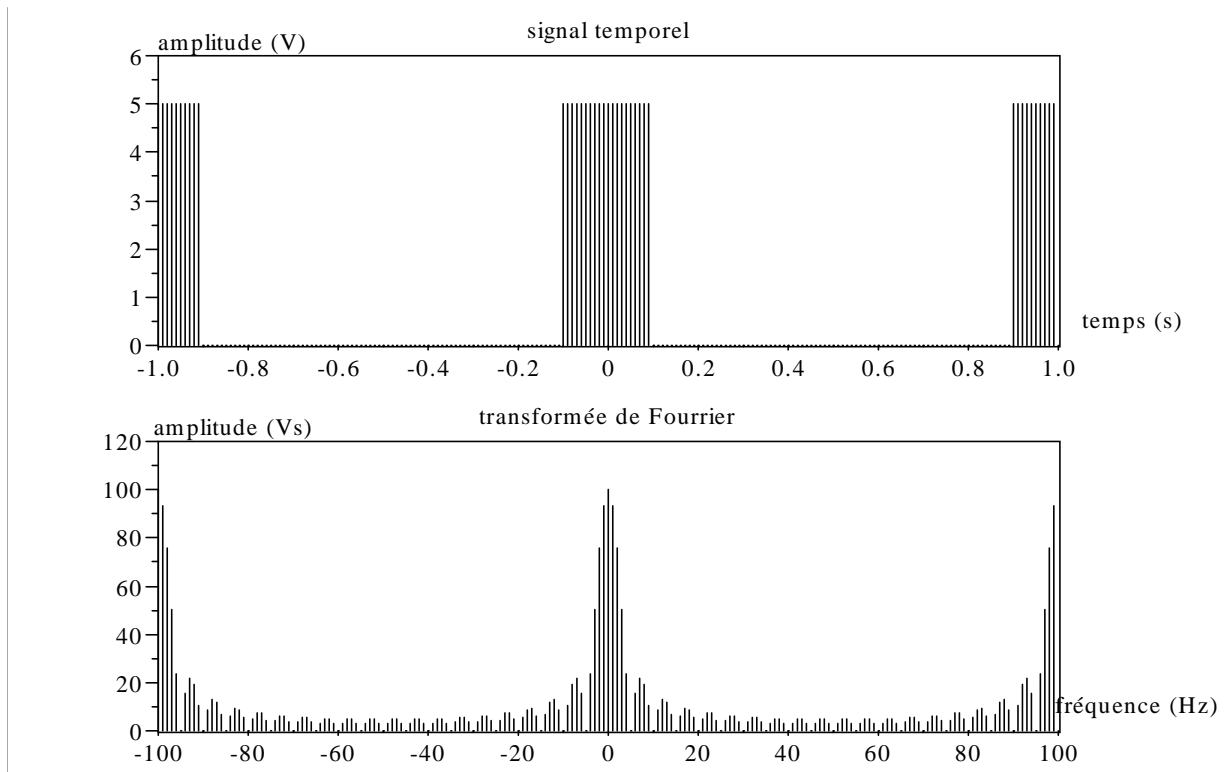
Le nombre d'échantillons acquis n'étant pas infini, le signal à partir duquel le calcul est fait n'est pas $s_e(t)$, mais $s_{et}(t)$, le signal échantillonné tronqué. L'opération dans le domaine temporel est la multiplication par une porte unitaire de largeur $N.T_E$, si N est le nombre d'échantillons acquis. Dans le domaine fréquentiel, cette opération se traduit par une convolution par la transformée de Fourier de la fonction porte, à savoir :

$$N.T_E.\text{sinc}(\pi.N.T_E)$$

Si la fenêtre d'acquisition est suffisamment large, cette fonction est presque une impulsion d'amplitude $N.T_E$. Seule alors l'amplitude du spectre de $s_e(t)$ est affectée, en étant multipliée par $N.T_E$, valeur que nous avons fixé à « 1 » pour des raisons justifiées par la suite.
 Par rapport au spectre d'origine, qui a été multiplié par F_E lors de l'opération précédente, nous n'avons plus qu'un facteur N .



Le calculateur, en raison de son mode de fonctionnement, ne peut donner que des échantillons du spectre et non une fonction continue de la fréquence. Tout ce passe comme si le spectre analogique était échantillonné, c'est à dire multiplié dans le domaine fréquentielle par un peigne de Dirac, de « période » F_E/N . Du côté temporel, cela se traduit par une convolution par un peigne de Dirac de période $N.T_E$ et d'amplitude $N.T_E$. Le signal temporel est donc équivalent pour le calculateur, à un signal périodique, de période égale à la fenêtre d'acquisition $N.T_E$, et d'amplitude $N.T_E$ fois son amplitude réelle.



Dans notre exemple, $N.T_E$, a été choisi unitaire, car c'est le seul moyen avec cette forme de signal d'obtenir par la fonction « fft », à N près la valeur souhaitée :

- nous venons en effet de voir que le signal temporel vu par le calculateur est périodique, échantillonné et d'amplitude $A.N.T_E$. Le signal d'origine était non périodique et non échantillonné, même si on peut faire correspondre les formes des deux spectres par des artifices d'affichage (afficher l'enveloppe du premier de $-F_E/2$ à $F_E/2$), pour les ordonnées, il va falloir imposer certaines conditions. Raisonons sur la valeur à $f=0$: pour le signal périodique, elle vaut $A.T$, ce qui devient $A.T/T_E$ pour le signal périodique échantillonné ; puis $A.T/(N.T_E)$ après avoir divisé l'opération « fft » par N (voir programme en annexe). Pour obtenir l'égalité avec le spectre souhaité d'amplitude $A.T$ à $f=0$, il faut donc $N.T_E=1$.

Prendre sur un graphique ces différentes étapes pour un signal quelconque, en indiquant l'opération subie côté temporel et fréquentiel par les signaux, ainsi que la forme des différents termes de l'opération. On précisera toutes les valeurs remarquables sur les chronogrammes et spectrogrammes.

2 Etablissement de l'expression de la TFD

Commençons, avant de déterminer l'expression de la TFD, par chercher celle de la transformation de Fourier (au sens analogique) d'un signal temps discret (TFtd). Elle se déduit de celle de la transformée Fourier d'un signal à temps continu (1) :

$$S(f) = \int_{-\infty}^{+\infty} s(t) e^{-2\pi j f t} dt \quad (1)$$

ce qui donne en remplaçant t par nT_E (T_E période d'échantillonnage), ainsi que l'intégrale par une sommation l'expression de la TFtd (2) :

$$S(f) = \sum_{n=-\infty}^{+\infty} s(nT_E) e^{-2\pi j f n T_E} \quad (2)$$

Dans le cas d'un système physique, l'acquisition et le calcul ne peuvent se faire avec un nombre infini d'échantillons.

D'autre part, pour le calculateur la variable f devient également une variable discrète $k F_0$ (le calcul ne peut donner qu'un nombre fini de valeurs), d'où :

$$S(kF_0) = \sum_{n=0}^{N-1} s(nT_E) e^{-2\pi j k F_0 n T_E} \quad (3)$$

Pour des raisons pratiques justifiées par la suite le nombre d'échantillons temporels N , et le nombre d'échantillons calculés sur la transformée de Fourier sont identiques (n et k varie de 0 à $N-1$).

D'autre part, si l'échantillonnage de $s(t)$ se fait avec une période de T_E , l'échantillonnage (dû au pas de calcul) de $S(f)$ se fait avec une « période » F_E/N , d'où :

$$N F_0 = F_E = 1/T_E$$

On arrive alors à la notion de transformée de Fourier discrète (TFD) définie par :

$$S(k) = \sum_{n=0}^{N-1} s(nT_E) e^{-2\pi j n k/N} \quad (4)$$

On montre que l'expression de la transformation inverse est :

$$s(n) = \frac{1}{N} \sum_{k=0}^{N-1} S(k) e^{+2\pi j n k/N} \quad (5)$$

Remarque : L'expression (4) correspond au spectre du signal $s(t)$ échantillonné, soit $s(nT_E)$. Si c'est réellement le signal $s(t)$ qui nous intéresse et non $s(nT_E)$, il est alors préférable d'utiliser l'expression suivante :

$$S(k) = \frac{1}{N} \sum_{n=0}^{N-1} s(nT_E) e^{-2\pi j n k/N} \quad (6)$$

3 Validation de l'expression de la transformée de Fourier discrète

Nous allons maintenant démontrer la validité de l'expression (4) en détaillant le calcul étape par étape.

Remarque : pour ceux qui ne sont pas à l'aise avec ces calculs, on pourra remarquer que les grandeurs étudiées dépendent à chaque fois du temps ou de la fréquence. Un moyen efficace de se familiariser avec la manipulation des séries et intégrales est de faire une représentation graphique du signal représenté.

Rappeler l'expression mathématique (sous forme d'une somme) de P_{gnT_E} qui représente un peigne de Dirac de période T_E .

Considérons un signal $s_e(t)$ résultat de l'échantillonnage de $s(t)$ à une période T_E . Donner l'expression mathématique de ce signal en fonction de P_{gnT_E} et de $s(t)$, puis mettre sous forme d'une sommation. Vérifier que l'on peut remplacer $s(t)$ par $s(n)$ qui ne dépend plus du temps mais de l'échantillon considéré.

Le signal $s(t)$ étant échantillonné sur un nombre fini N d'échantillons, modifier l'expression précédente pour obtenir l'expression tronquée $s_{et}(t)$ de $s_e(t)$.

Appliquer la formule de transformation de Fourier d'un signal continu sur $s_{et}(t)$ pour obtenir $S_{et}(f)$. Inverser la sommation et l'intégration en sortant de l'intégrale tous les termes possibles ($s(t)$ aura été remplacé par $s(n)$).

Calculer l'expression à l'intérieur de l'intégrale, remarquer qu'elle ne dépend pas du temps, calculer l'intégrale.

Cette transformée de Fourier est à son tour échantillonnée à « période » F_0 par le mode de fonctionnement du calculateur. En reprenant les mêmes outils que pour le signal temporel, déduire l'expression de $S_{etD}(f)$ de la transformée de Fourier discrète de $s_{et}(t)$ à partir de $S_{et}(f)$.

Arranger l'expression pour supprimer les Dirac dans les calculs.

Déduire l'expression $S_{etD}(k)$ d'un échantillon k de $S_{etD}(f)$ et vérifier qu'il s'agit de la même formulation que (4).

4 Effet du fenêtrage

Le fait d'observer un signal pendant un temps fini et de n'acquérir qu'un nombre N fini d'échantillons, introduit une erreur dans le calcul de son spectre, erreur due au fenêtrage que nous allons essayer d'évaluer maintenant.

Prenons un signal exponentiel $s(t) = A e^{-j 2 \pi F_x t}$ comme signal test. Ce signal présente l'avantage par rapport à un signal sinusoïdal de ne présenter qu'une seule raie sur son spectre, les calculs en seront simplifiés, sans rien enlever à la généralité de la démonstration.

4.1 Approche qualitative par la transformation de Fourier continue

Rappeler la représentation spectrale (en module) de ce signal (spectre bilatéral) ainsi que son expression mathématique.

Dans le cas d'un échantillonnage idéal à fréquence $F_E=1/T_E$ par un peigne de Dirac, donner la représentation spectrale (en module) du signal échantillonné, ainsi que son expression mathématique.

On rappelle que la transformation de Fourier d'un peigne de Dirac $P_{gn_{TE}}$ a pour expression $F_E \cdot P_{gn_{FE}}$

$$\text{avec } P_{gn_{TE}} = \sum_{n=-\infty}^{+\infty} \delta(t - nT_E) \text{ et } P_{gn_{FE}} = \sum_{n=-\infty}^{+\infty} \delta(t - nF_E).$$

Déduire de la représentation graphique, l'expression mathématique du module du spectre du signal échantillonné.

Le fait de ne prendre qu'un nombre réduit d'échantillons équivaut à multiplier le signal temporel par la fonction porte $P_{t_T}(t)$ définie par :

$$P_{t_T}(t) = 1 \text{ pour } -T/2 < t < T/2 \\ P_{t_T}(t) = 0 \text{ ailleurs}$$

Le calcul de la transformée de Fourier d'une telle fonction donne :

$$\|TF(P_{t_T}(t))\| = T |\text{sinc}(\pi f T)| = T \left| \frac{\sin(\pi f T)}{\pi f T} \right|$$

Rappeler la représentation graphique du module du spectre d'une telle fonction.

Dans le cas, généralement vérifié, où la durée T est supérieure à la période T_x du signal numérisé, représenter le spectre du signal échantillonné et tronqué.

Quelle représentation retrouvera-t-on en sortie d'une fonction de calcul du type "fft" si le calcul est fait pour une fréquence F_X faible (c'est à dire une bonne précision fréquentielle).

Comment évolue ce spectre si la largeur d'observation varie ?

Quel est l'effet d'un décalage temporel de la fonction porte vers les bornes 0 et T ?

4.2 Expression mathématique de la transformée de Fourier discrète

Les résultats précédents, mettant en évidence pour le signal échantillonné et tronqué une superposition de sinus cardinaux, ont été établis en supposant le calcul fait avec la transformation pour signaux continus (expression (1)). Hors le calcul est réellement fait avec l'expression (4) qui correspond à une transformée discrète.

Montrer qu'avec la transformation discrète, pour le signal qui nous intéresse, on arrive à :

$$S(k) = A \frac{\sin(\pi N (\frac{k}{N} - \frac{F_X}{F_E}))}{\sin(\pi (\frac{k}{N} - \frac{F_X}{F_E}))} e^{j\pi (\frac{k}{N} - \frac{F_X}{F_E})(N-1)}$$

$$\text{d'où } \|S(k)\| = A \frac{\sin(\pi N (\frac{k}{N} - \frac{F_X}{F_E}))}{\sin(\pi (\frac{k}{N} - \frac{F_X}{F_E}))}$$

on rappelle que : $\sum_{x=0}^N a^x = \frac{1-a^{N+1}}{1-a}$

Nous allons maintenant vérifier expérimentalement ce résultat en utilisant Scilab ; mais avant toute chose, examinons la manière de calculer une DFT, et l'intérêt d'utiliser un algorithme dit de « transformée de Fourier rapide (TFR ou FFT) ».

5 Efficacité de l'algorithme de la fonction « fft »

La fonction « fft » de Scilab utilise l'algorithme de Cooley Tukey pour effectuer le calcul. Celui-ci permet de réduire de manière conséquente le nombre d'opérations et donc le temps de calcul, en particulier si le nombre d'échantillons est une puissance de 2.

Pour s'en convaincre, nous allons décrire l'opération de transformation nous même par une ligne de programme, puis comparer, sur un signal test, le temps de calcul avec la fonction « fft » ; nous utiliserons pour cela la fonction « timer() » de Scilab, qui donne le temps écoulé depuis l'instruction « timer() » précédente (taper « help timer » pour plus d'information).

Attention : le temps mesuré entre deux instructions « timer() » est le temps réellement écoulé, qui prend en compte la durée de saisie des instructions ; pour des mesures correctes, il est donc conseillé de taper le programme qui va suivre avec un éditeur de texte quelconque et de le rentrer par « copier coller ».

D'après l'expression (4), le calcul d'une TFD conduit, pour chacun des N échantillons de S(k), à une sommation sur N éléments des produits de chaque échantillon de s(n) par le terme exponentiel correspondant ; soit 2N multiplications et additions (les nombres sont complexes) pour un échantillon de S(k), cela répété N fois.

Scilab étant un logiciel optimisé pour le calcul matriciel, on n'utilisera pas de boucle « for » pour effectuer cette opération, mais l'instruction « sum(a,'r') » qui renvoie un vecteur ligne ('r' pour row), dont chaque élément est la somme des composantes de la colonne correspondante de la matrice « a » (on pourrait faire de même avec les lignes et l'argument 'c' -pour column- voir l'annexe de la séance d'introduction).

Etudier et expérimenter alors le programme suivant ; vérifier que l'algorithme proposé est bien conforme à l'expression donnée en (4).

Dans un premier temps on imposera 4 échantillons et on supprimera les « ; » à la fin de chaque instruction pour observer la forme de la matrice obtenue et comparer notre calcul à celui de la fonction « fft » :

```
clear ;
//
// nombre d'échantillons, 4 puis 512, puis1024 pour comparer les résultats
N=4 ;
//
// vecteur temps et signal
t=(0 :N-1) ; s=sin(t) ;
//
// début de la temporisation
timer() ;
//
// notre algorithme
Sf1=sum(s * ones(1,N) .* exp( -2*%pi * %i * t' *t /N), 'r') ;
timer()
// lire la valeur retournée
//
// nouvelle temporisation pour l'algorithme fft cette fois
timer() ;
Sf2=fft(s,-1) ;
timer()
// lire la valeur et comparer à la précédente
```

6 Vérification expérimentale

Afin de vérifier les calculs des paragraphes précédents, nous allons comparer le résultat donné par la fonction « fft » de Scilab, avec le calcul théorique de la DFT.

Dans le premier cas l'échantillonnage se fait à $F_e=64$ kHz avec une acquisition de $N=32$ points.

Dans le second cas, toujours avec une fréquence d'échantillonnage de 64 kHz, on calcul des points intermédiaire de manière à obtenir $N_a=1024$ points, ce qui est équivalent à un échantillonnage de $1024.64/32=2048$ kHz.

L'affichage se fera dans le domaine temporel avec une classique échelle en secondes, tandis que dans le domaine fréquentiel, nous afficherons en fonction du rang de l'échantillon, en prenant la valeur « 0 » comme première indice (contrairement à Scilab qui prend la valeur « 1 »). Avec cet échelle l'échantillon « 0 » correspond au continu, l'échantillon « N-1 » à $F_e.(N-1)/N$ et par une simple règle de 3, on en déduit que l'échantillon « x » correspond à $x.F_e/N$.

Le signal temporel choisi est un signal exponentiel, ce qui permet de n'avoir qu'une seule raie sur le spectre et correspond au calcul effectué. Dans le domaine temporel, c'est sa partie réelle qui est affichée (son module est constant). On rappelle que si l'amplitude d'un tel signal est « A », son spectre « analogique » est composé d'une seule raie d'amplitude « A ». Avec la DFT, il faut donc s'attendre à une seule raie d'amplitude $A.N$.

L'indétermination sur l'expression théorique de la DFT est supprimée, comme nous l'avons déjà fait en utilisant la condition « $x==0$ » qui renvoie « 1 » ou « 0 » suivant qu'elle est vraie ou fautive.

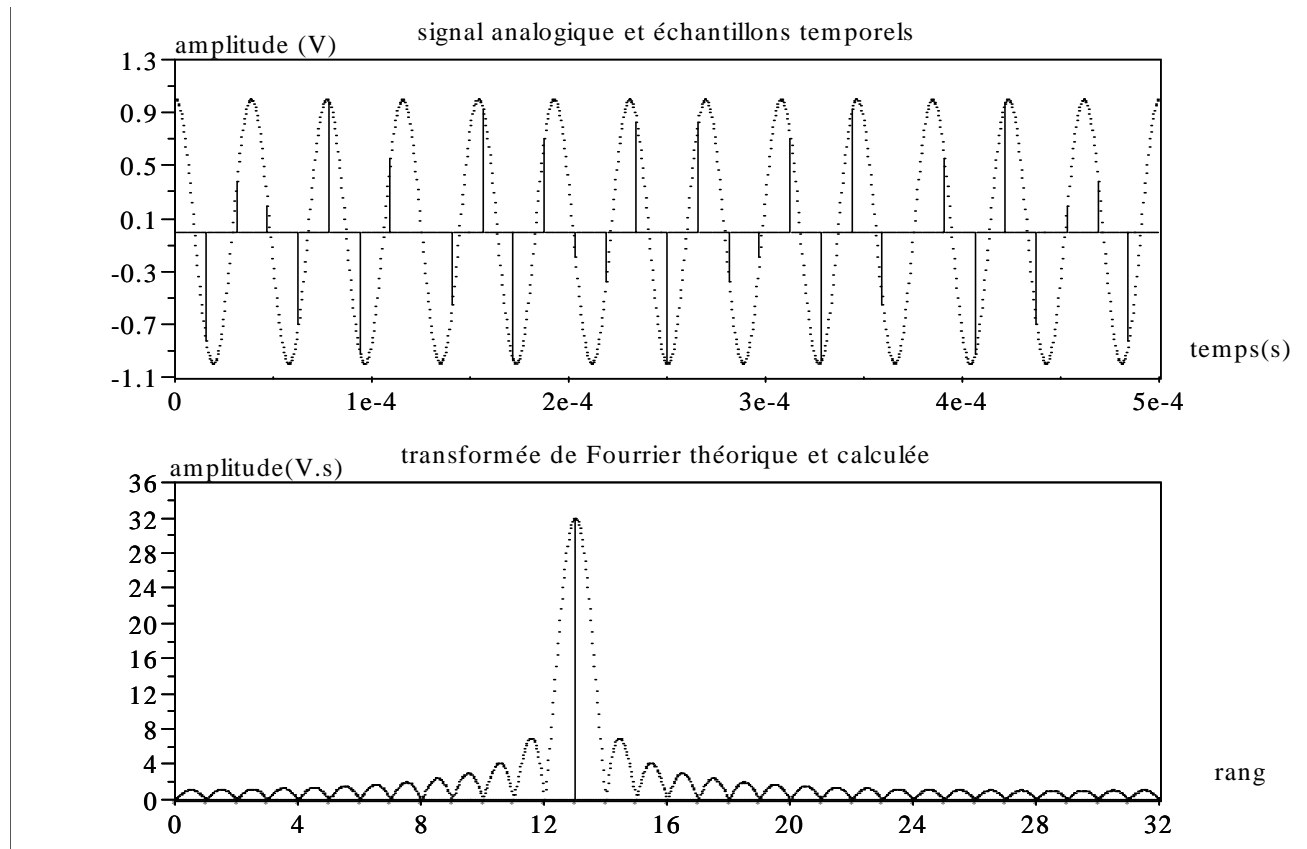
```

clear
//
// paramètres de l'acquisition
// nombre de points, fréquence du signal, amplitude, fréquence d'échantillonnage
N=32; Fx=26e3 ; A=1 ; Fe=64e3 ;
//
// paramètre de l'échelle analogique : nombre de points
Na=1024 ;
//
// définition du vecteur temps et du vecteur fréquence normalisé à N
ta=N/(Na*Fe)*(0 :Na-1) ; ; k=N/Na*(0 : Na-1) ;
//
// définition du signal temporel
sa= A*exp(2*i*pi*Fx*ta) ;
//
// échelle des temps et fréquence numérique
tn=1/Fe*(0:N-1) ; fn=(0 :N-1) ;
//
// échantillonnage du signal temporel, calcul de la fft

sn=A*exp(2*i*pi*Fx*tn) ;
Sf=abs(fft(sn,-1));
//
// expression théorique de la DFT d'une exponentielle
x=k/N-Fx/Fe;
S= A*sin(N*x*pi) ./ (sin(x*pi)+(x==0)) + (x==0)*N;
//
// précision sur l'échelle des fréquences des points affichables
pts=zeros(1, N) ;
//
// affichage
xset("window", 0) ; xbas(0) ; xset ("font size", 4) ;
xsetech([ 0, 0, 1, 1/2]);
plot2d(ta, real(sa), rect=[0, -1.1*A, N/Fe, 1.1*A], style=0) ;
plot2d3(tn,real(sn)) ;
xtitle("signal analogique et échantillons temporels", "temps(s)", "amplitude (V)");
//
xsetech([ 0, 1/2, 1, 1/2]) ;
plot2d(k, abs(S), style=0, rect=[0, 0, N, 1.1*A*N]) ;
plot2d3(fn, Sf) ;
plot2d(fn, pts, style=-3) ;
xtitle("transformée de Fourier théorique et calculée", "rang", "amplitude(V.s)");

```


Voici le résultat obtenu pour une fréquence de 26 kHz :

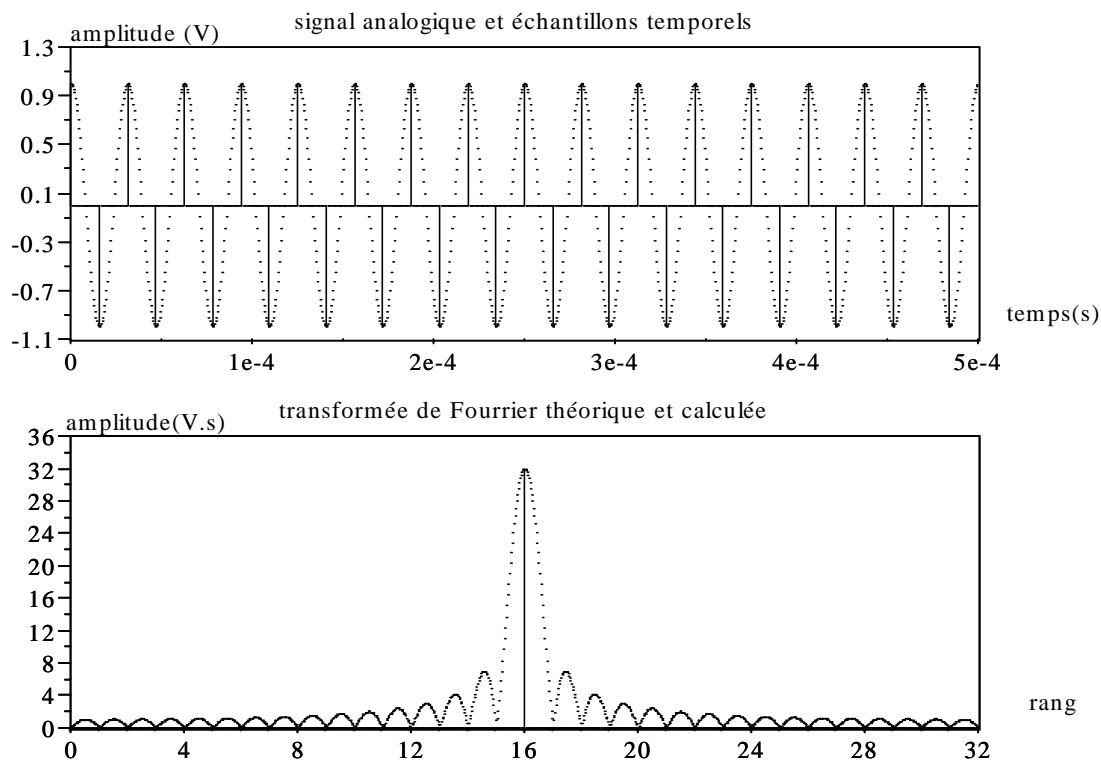


On peut constater que la raie obtenue correspond au cas théorique d'un signal exponentiel, les autres points affichés (marqués par des croix sur les valeurs entières du rang d'échantillon) correspondant aux zéros de la fonction en sinus.

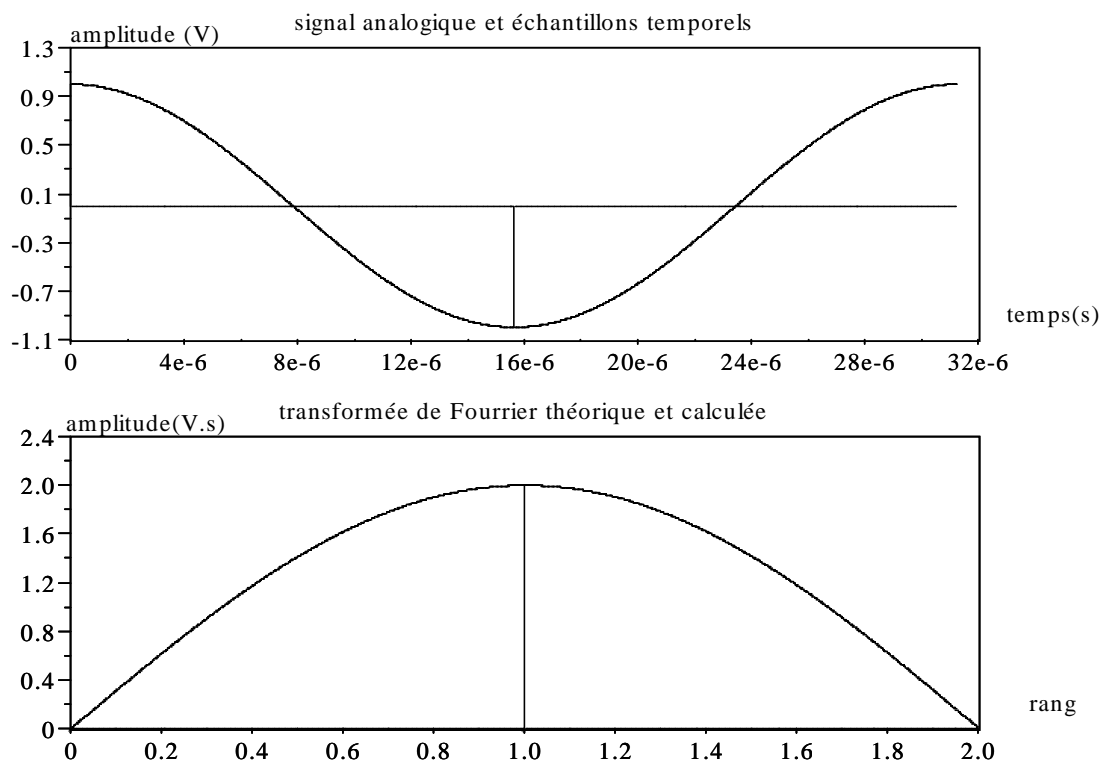
La fréquence de rang 13 correspond bien à 26 kHz pour 32 échantillons à 64 kHz, soit : $26 = 13 \times 64 / 32$
L'amplitude correspond bien à N.

Observons maintenant ce qui se passe pour un signal d'entrée à la limite de celle définie par le théorème de Shannon, soit 32 kHz ici.

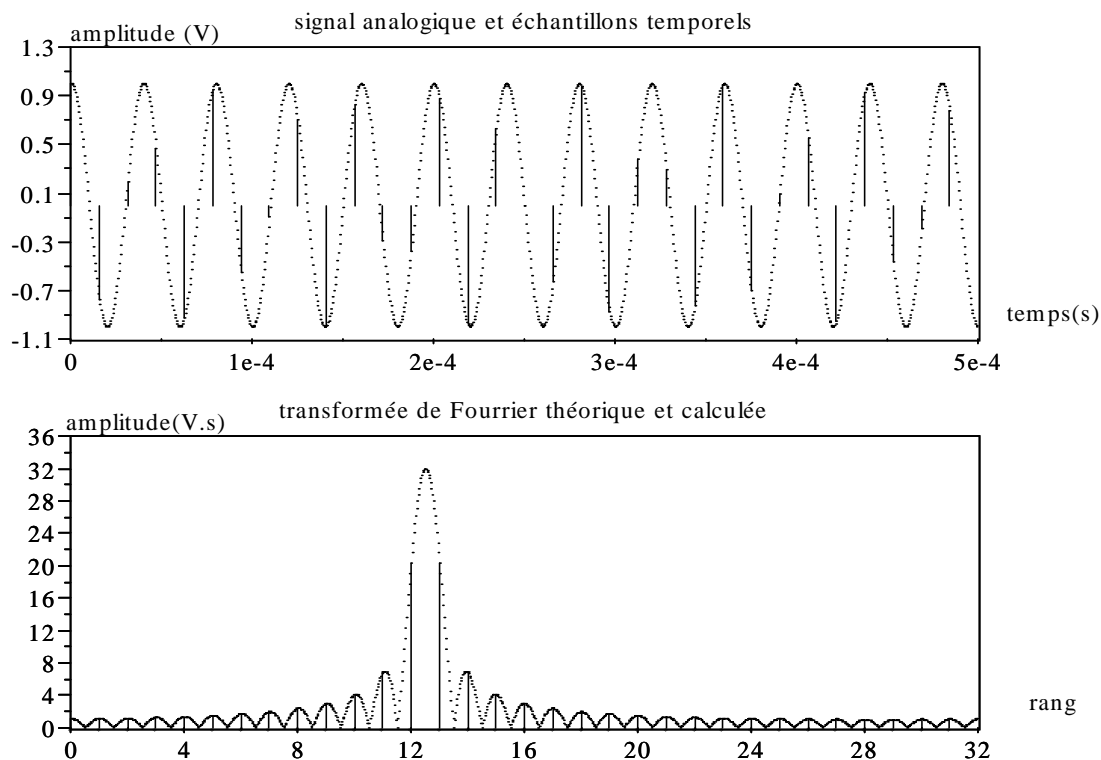
Si les échantillons sont pris au maximum de l'amplitude, deux points par période sont suffisant pour une analyse correcte.



Il es même possible de limiter à deux le nombre d'échantillons (le premier se trouve en $t=0$) :



Par contre, si les points affichés ne tombent pas sur le maximum et les zéros de la fonction en sinus, l'affichage peut présenter des erreurs importantes, malgré une fréquence très en dessous de la limite de Shannon et un nombre de points importants. Voici un exemple pour une fréquence de 25 kHz, toujours sur 32 échantillons :



On constate la disparition de la raie principale au profit de deux raies d'amplitude plus faible, ainsi que l'apparition d'un ensemble de raies, d'amplitudes non négligeables, à des fréquences qui n'existent pas sur le signal d'origine.

On peut donc établir à partir de ces exemples :

Seules les fréquences multiples de F_E/N seront donc affichées correctement.

Reprendre les programmes précédents et essayer différentes valeurs de fréquence d'échantillonnage, de nombre de points et de fréquences du signal temporel.

6.1 Résolution en fréquence

Après avoir abordé la précision, voyons maintenant l'aptitude de la fonction à distinguer deux fréquences voisines. On parle alors de résolution.

La résolution est souvent définie par l'écart minimum en fréquence qu'il faut mettre entre deux signaux sinusoïdaux, d'amplitudes différentes, pour observer sur le spectre de leur somme un creux de plus de 3 dB entre les deux maxima (attention : une bonne résolution est une résolution faible).

Sur le tracé du spectre calculé avec la TFD pour un signal exponentiel, nous pouvons observer un lobe principal dont la largeur vaut $2F_E/N$ (le cas d'une sinusoïde est identique, au détail près qu'il y a aussi un lobe principal en zone négative). On ne peut donc espérer séparer deux sinusoïdes dont la différence de fréquence est inférieure à $F_E/N=1/(T_E N)$, à plus forte raison si leurs amplitudes sont très différentes.

La résolution fréquentielle est sensiblement inversement proportionnelle au temps d'analyse $N T_E$.

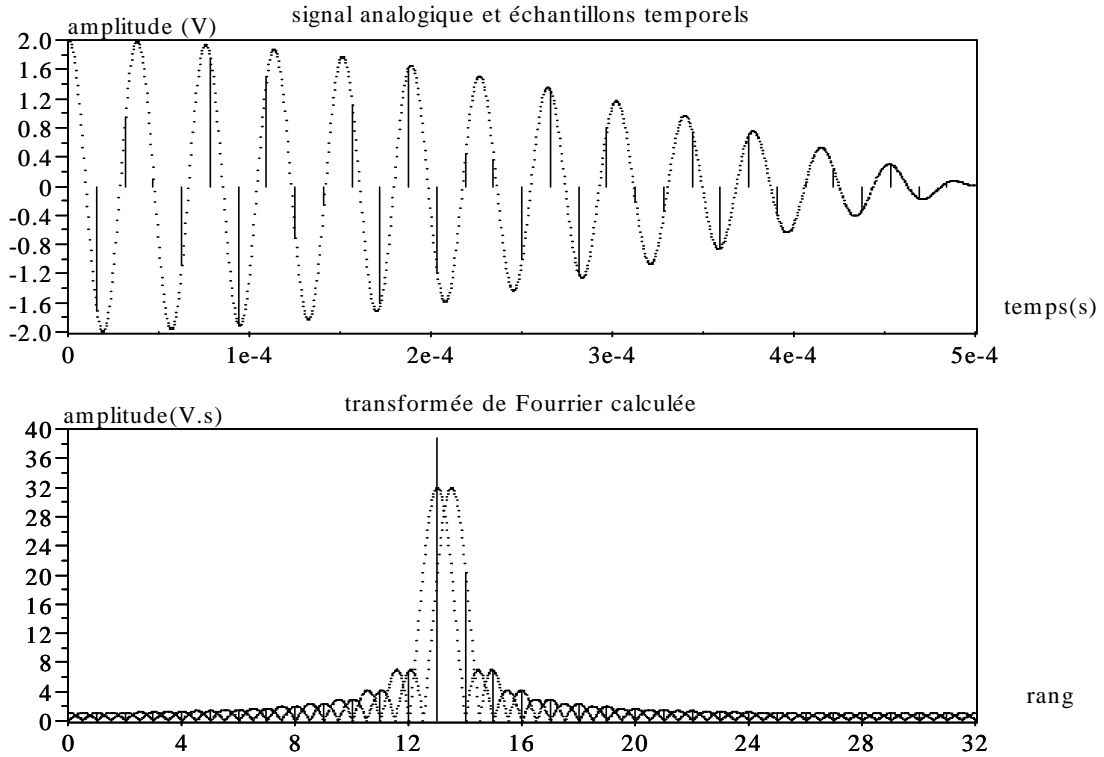
Pour illustrer, reprenons le programme précédent en modifiant légèrement de manière à ce que le signal « s » soit la somme de deux exponentielles.

```

clear
// paramètre de l'acquisition
// nombre de points, fréquence du signal, amplitude
N=128; Fx=26e3 ; A=1 ; Fx2=27e3 ;
//
// paramètre de l'échelle analogique
// nombre de points, fréquence d'échantillonnage
Na=1024 ; Fe=64e3 ;
//
// définition du vecteur temps et du vecteur fréquence normalisé à N
ta=N/(Na*Fe)*(0 :Na-1) ; k=N/Na*(0 : Na-1) ;
//
// définition du signal temporel
sa= A*(exp(2*i*pi*Fx*ta)+ exp(2*i*pi*Fx2*ta)) ;
//
// échantillonnage du signal temporel, calcul de la fft
tn=1/Fe*(0:N-1) ; fn=(0 :N-1) ;
sn=A*(exp(2*i*pi*Fx*tn)+ exp(2*i*pi*Fx2*tn)) ;
Sf=abs(fft(sn,-1));
//
// expression théorique de la DFT d'une exponentielle
x=k/N-Fx/Fe;
x2=k/N-Fx2/Fe;
S= A*sin(N*x*pi) ./ (sin(x*pi)+(x==0)) + (x==0)*N;
S2= A*sin(N*x2*pi) ./ (sin(x2*pi)+(x2==0)) + (x2==0)*N;
// précision sur l'échelle des fréquences des points affichables
pts=zeros(1, N) ;
//
// affichage
xset("window", 0) ; xbas(0) ; xset ("font size", 4) ;
xsetech([ 0, 1/2, 1, 1/2]) ;
plot2d(k, abs(S), style=0, rect=[0, 0, N, 1.1*A*N]) ;
plot2d(k, abs(S2), style=0) ;
plot2d3(fn, Sf) ;
plot2d(fn, pts, style=-3) ;
xtitle("transformée de Fourier calculée","rang", "amplitude(V.s)") ;
xsetech([ 0, 0, 1, 1/2]);
plot2d(ta, real(sa), style=0) ;
plot2d3(tn,real(sn)) ;
xtitle("signal analogique et échantillons temporels", "temps(s)", "amplitude (V)") ;
//
xset("window", 1) ; xbas(1) ; xset ("font size", 4) ;
plot2d(k(0.33*Na :0.5*Na), abs(S(0.33*Na :0.5*Na)), rect=[0.33*Na/N, 0, 0.5*Na/N, 1.1*A*N] , style=0) ;
plot2d(k(0.33*Na :0.5*Na), abs(S2(0.33*Na :0.5*Na)) ,style=0) ;
plot2d3(fn(0.33*N :0.5*N), Sf(0.33*N :0.5*N)) ;
plot2d(fn(0.33*N :0.5*N), pts(0.33*N :0.5*N), style=-3) ;
xtitle("transformée de Fourier calculée","rang", "amplitude(V.s)") ;

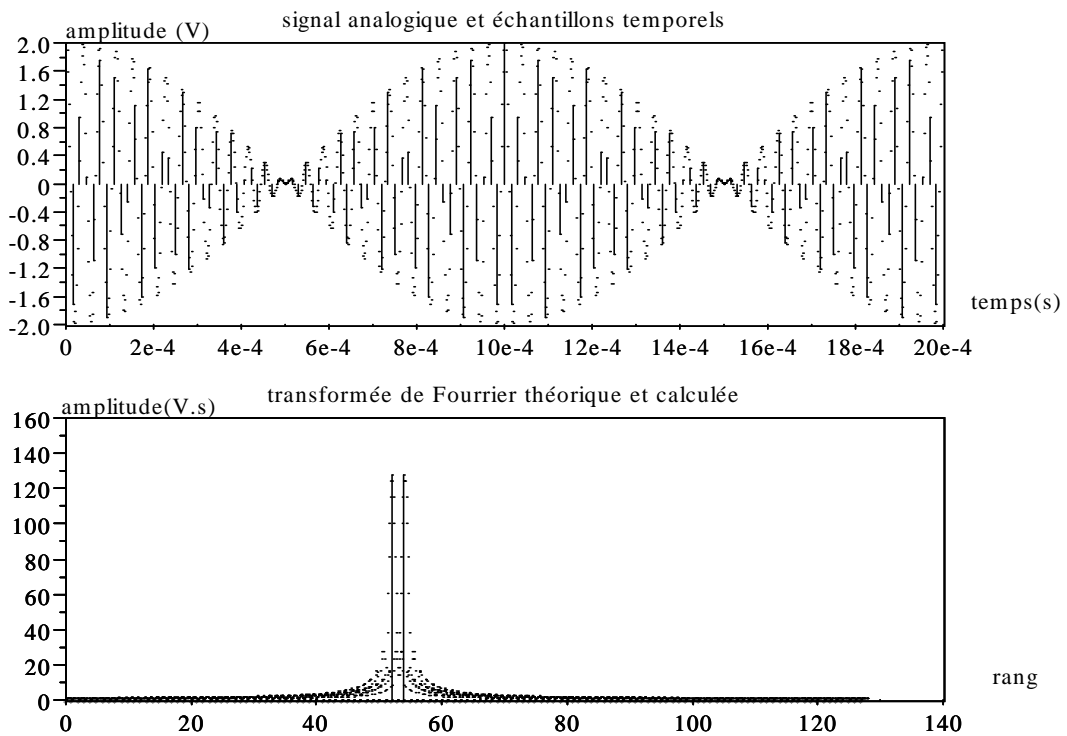
```

Voici les courbes obtenues pour des fréquences de 26 et 27 kHz pour 32 points acquis :

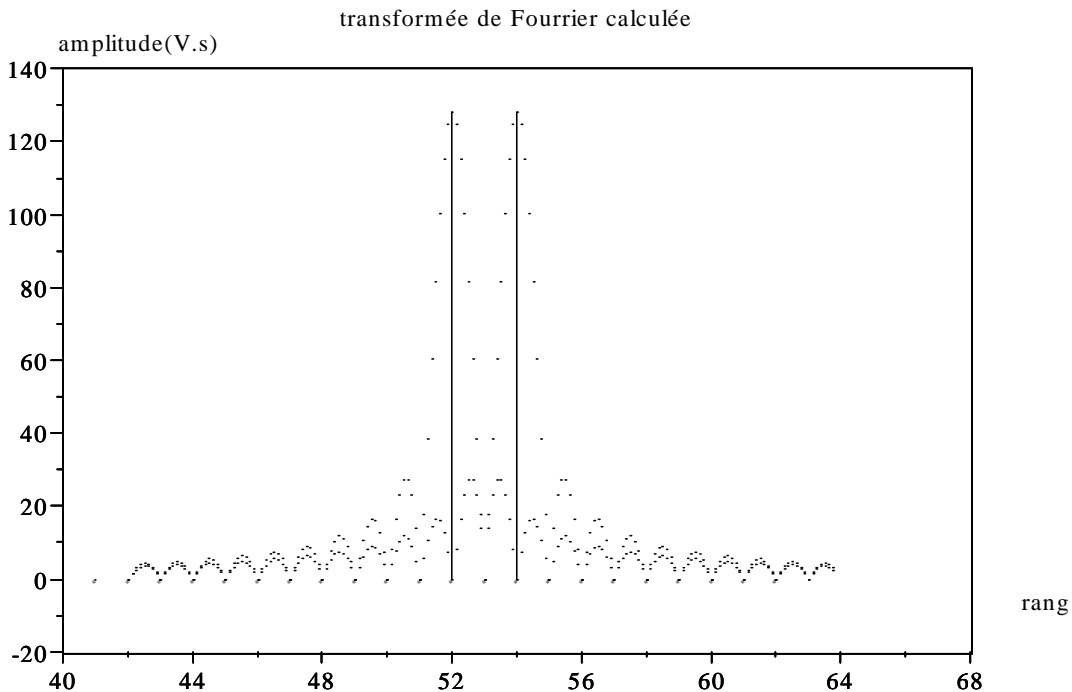


On peut voir que les deux raies sont présentes, mais n'ont pas les valeurs attendues ; on remarque que l'une est trop faible, effet du aux phénomène précédent, tandis que l'autre est trop importante, effet du à la superposition des spectres des deux signaux. Dans le cas d'un affichage continu, du style de ceux que l'on trouve sur les analyseurs de spectre, ces deux raies seraient confondues.

Observons maintenant ce qui se passe si on augmente la durée d'acquisition en passant N à 128 :



On constate cette fois que les deux raies ont la bonne amplitude (128), avec un point à « 0 » entre les deux comme le montre le zoom suivant :



Ce phénomène peut s'expliquer assez simplement en observant la représentation temporelle du signal : dans le premier cas, nous ne faisons pas l'acquisition d'une période complète, alors que dans le second cas, deux périodes sont acquises ; une seule aurait suffi (soit 64 points) mais nous n'aurions pas eu le « 0 » entre les deux raies.

6.2 Choix d'une fenêtre

Nous n'avons utilisé jusqu'à présent qu'une fenêtre d'observation rectangulaire, où tous les échantillons ont même valeur ; il existe d'autres fenêtres permettant d'obtenir une atténuation plus importante du premier lobe secondaire et/ou une décroissance plus rapide des autres lobes. Cet avantage se traduit cependant par un lobe primaire plus large, réduisant la résolution.

Citons pour les principales fenêtres l'expression par laquelle le signal acquis est multiplié :

Fenêtre de Hanning :

$$0,5 \left(1 - \cos\left(\frac{2\pi t}{N T_E}\right)\right)$$

Fenêtre de Hamming :

$$0,54 - 0,46 \cos\left(\frac{2\pi t}{N T_E}\right)$$

Fenêtre de Blackman :

$$0,42 - 0,5 \cos\left(\frac{2\pi t}{N T_E}\right) + 0,08 \cos\left(\frac{4\pi t}{N T_E}\right)$$

7 Illustration des propriétés de la transformation de Fourier

Proposer des programmes permettant d'illustrer les propriétés de « similitude » et de « retard temporel » de la transformée de Fourier.

La fonction « rand(nl,nc,'n') » génère une matrice de nombres aléatoires de « nl » lignes pour « nc » colonnes, où les valeurs comprises entre 0 et 1 sont équiprobables (loi normale).

Utiliser cette fonction pour générer un signal de 64 échantillons et vérifier que pour un signal pair, la transformée de Fourier présente :

- un module pair
- une phase impaire
- une partie réelle paire (fonction « real »)
- une partie imaginaire impaire (fonction « imag »)

Bibliographie

Traitement numérique du signal, simulation sous Matlab par G. Blanchet M. Charbit chez Hermes

Traitement du signal sous Matlab par A. Quinquis chez Hermes

Traitement du signal et acquisition de données par F. Cottet chez Dunod

Signal Processing with Scilab par Scilab Group sur le site INRIA : <http://www-roq.inria.fr/scilab/>

Annexe : programme des illustration

```
clear;
//
// définition des constantes, rapport cyclique, nombre de points affichés
// période et fréquence d'échantillonnage, amplitude du signal
// largeur de l'affichage (en nombre paire de périodes), constantes d'affichage
// vecteur temps
a=20 ; N=100 ; Te=10e-3 ; Fe=1/Te ; Amp=5 ;
P=2; At=N*P*Te/2; Af=P*Fe/2;
t=Te*(-N/2: N/2-1);
//
// description du signal porte
s=Amp*[zeros(1,(N-a)/2), ones(1, a), zeros(1, (N-a)/2)] ;
//
// transformé de Fourier sous forme bi-lattérale et vecteur fréquence
S1=1/N*fft(s,-1);
S=[S1(N/2+1:N), S1(1:N/2)];
f=Fe/N*[-N/2 : N/2-1];
//
// affichage du signal et de sa transformée de Fourier
xset("window", 0) ; xbas(0) ; xset ("font size", 4) ;
xsetech([0,0,1,1/2]);
plot2d2(t, s, rect=[-At, 0, At, 1.2*Amp]);
xtitle("signal temporel","temps (s)","amplitude (V)");
xsetech([0, 1/2,1,1/2]);
plot2d(f,abs(S), rect=[-Af, 0,Af, 1.2*a *Te*Amp]);
xtitle("transformée de Fourier"," fréquence (Hz)","amplitude (Vs)");
//
// périodisation de la transformée de Fourier
// définition d'une nouvelle échelle de fréquence
S1=[S(N/2+1:N), S(1:N/2)];
S2=Fe*matrix(S1' *ones(1, P), 1 , N*P);
f2=Fe/N*[-P*N/2 : P*N/2-1];
//
```

```

//
//
// signal échantillonné, définition d'une nouvelle échelle de temps
se= Amp*[zeros(1,P*N/4+(N-a)/2), ones(1, a), zeros(1,P*N/4+(N-a)/2)] ;
t2=Te*(-P*N/2 : P*N/2-1);
//
// affichage du signal échantillonné et de sa transformée
xset("window", 1) ; xbas(1) ; xset ("font size", 4) ;
xsetech([0,0,1,1/2]);
plot2d3(t, s, rect=[-At, 0, At, 1.2*Amp]);
plot2d (t2, se, style=-9);
xtitle("signal temporel", "temps (s)", "amplitude (V)");
xsetech([0, 1/2,1,1/2]);
plot2d(f2,abs(S2), rect=[-Af, 0, Af, 1.2*a*Te*Fe*Amp]);
xtitle("transformée de Fourier", " fréquence (Hz)", "amplitude (Vs)");
//
// effet de la troncature, le spectre d'origine est multiplié par N
S2=N*Te*S2;
//
// affichage du signal tronqué
xset("window", 2) ; xbas(2) ; xset ("font size", 4) ;
xsetech([0,0,1,1/2]);
plot2d3(t, s, rect=[-At, 0, At, 1.2*Amp]);
plot2d (t, s, style=-9);
xtitle("signal temporel", "temps (s)", "amplitude (V)");
xsetech([0, 1/2,1,1/2]);
plot2d(f2,abs(S2), rect=[-Af, 0, Af, 1.2*a*Te*N*Amp]);
xtitle("transformée de Fourier", " fréquence (Hz)", "amplitude (Vs)");
//
// signal temporel périodisé
s1=[s(N/2+1:N), s(1:N/2)];
s2=N*Te*matrix(s1*ones(1,P), 1, N*P);
//
// affichage du signal temporel périodique
xset("window", 3) ; xbas(3) ; xset ("font size", 4) ;
xsetech([0,0,1,1/2]);
plot2d3(t2, N/Fe*s2, rect=[-At, 0, At, 1.2*N/Fe*Amp]);
xtitle("signal temporel", "temps (s)", "amplitude (V)");
xsetech([0, 1/2,1,1/2]);
plot2d3(f2,abs(S2), rect=[-Af, 0, Af, 1.2*a*Te*N*Amp]);
xtitle("transformée de Fourier", " fréquence (Hz)", "amplitude (Vs)");

```