TELECOM SudParis
Cours de traitement du signal SIC3501.

TP 1

Analyse spectrale numérique

Dans la partie 1 sont présentés des rappels de cours. Le travail à effectuer en TP est donné dans la partie 2.

1 Rappels sur les transformées de Fourier

Seules les propriétés indispensables au TP sont rappelées ici et aucun détail technique rigoureux n'est précisé. Vous pouvez consulter le cours de traitement du signal et/ou de mathématiques pour plus de précisions.

1.1 Transformée de Fourier à temps continu

Définition 1 *La transformation de Fourier à temps continu d'un signal $x(t)$ est définie par :*

$$X(f) = \int_{-\infty}^{+\infty} x(t) e^{-\imath 2\pi f t}\, dt$$

et on notera : $X(f) = \mathrm{TF}\,[x(t)]$.

Propriété 1 *La transformée inverse s'écrit alors :*

$$x(t) = \int_{-\infty}^{+\infty} X(f) e^{+\imath 2\pi f t}\, df.$$

Propriété 2 *Une des propriétés importantes de la transformation de Fourier est de transformer un produit de convolution en un produit ordinaire. Ainsi si l'on note :*

$$z(t) = x(t) \star y(t) = \int_{-\infty}^{+\infty} x(\theta) y(t-\theta)\, d\theta$$

alors on a (en notant $Y(f) = \mathrm{TF}\,[y(t)]$ *et* $Z(f) = \mathrm{TF}\,[z(t)]$*) :*

$$Z(f) = X(f)Y(f)$$

1.2 Transformée de Fourier à temps discret (TFTD)

1.2.1 Du temps continu au temps discret

La transformée de Fourier précédente ne concerne que des signaux dits « à temps continu ». Si l'on souhaite réaliser de façon numérique l'analyse spectrale d'un signal, il convient au préalable de l'échantillonner. Nous rappelons les conséquences liées à un échantillonnage du signal.

Echantillonnage Considérons le signal à temps continu $x(t)$. Echantillonner ce signal à la fréquence f_e (correspondant à une période d'échantillonnage $T_e = 1/f_e$) consiste à recueillir la suite de valeurs du signal aux instants $t = kT_e$ ($k \in \mathbb{Z}$). Cette suite de valeurs est notée $x_k = x(kT_e)$. Afin de modéliser le procédé d'échantillonnage, on introduit le peigne de Dirac :

$$\text{Ш}_{T_e}(t) = \sum_{k=-\infty}^{+\infty} \delta(t - kT_e)$$

Définition 2 *Le signal échantillonné de $x(t)$ est par définition :*

$$x_e(t) = x(t)\,\text{Ш}_{T_e}(t) = x(t)\left(\sum_{k=-\infty}^{+\infty} \delta(t - kT_e)\right) \tag{1}$$

Sachant de plus que $x(t)\delta(t - kT_e) = x(kT_e)\delta(t - kT_e)$, on a également :

$$x_e(t) = \sum_{k=-\infty}^{+\infty} x(kT_e)\delta(t - kT_e) \tag{2}$$

Spectre du signal échantillonné Le spectre de $x_e(t)$ peut être facilement lié à celui de $x(t)$ à l'aide de la définition donnée par (1). En effet, on a alors :

$$X_e(f) = X(f) \star \text{TF}\left[\text{Ш}_{T_e}(t)\right] \tag{3}$$

La transformée de Fourier $\text{TF}\left[\text{Ш}_{T_e}(t)\right]$ peut être obtenue à partir de son développement en série de Fourier :

$$\text{TF}\left[\text{Ш}_{T_e}(t)\right] = \frac{1}{T_e}\sum_{k=-\infty}^{+\infty} \delta\left(f - \frac{k}{T_e}\right) \tag{4}$$

En remplaçant (4) dans (3) il vient :

$$X_e(f) = X(f) \star \frac{1}{T_e}\sum_{k=-\infty}^{+\infty} \delta\left(f - \frac{k}{T_e}\right)$$

Or on sait que $X(f) \star \delta\left(f - \frac{k}{T_e}\right) = X\left(f - \frac{k}{T_e}\right)$. Il s'ensuit :

$$X_e(f) = \frac{1}{T_e}\sum_{k=-\infty}^{+\infty} X\left(f - \frac{k}{T_e}\right) \tag{5}$$

On observe ainsi qu'il y a une *périodisation du spectre* : $X_e(f)$ est périodique de période $\frac{1}{T_e}$. On en déduit alors le théorème suivant :

Théorème 1 (Shannon-Nyquist) *Un signal $x(t)$ dont le spectre est à bande limitée $[-F_{\max}, F_{\max}]$ peut être reconstruit sans erreur à partir de son signal échantillonné $x_k = x(kT_e)$ à condition de choisir la fréquence d'échantillonnage telle que :*

$$f_e = \frac{1}{T_e} \geq 2F_{\max}$$

Dans ce cas le signal est reconstitué par un filtrage passe-bas idéal et on a de plus :

$$\forall f \in \left[-\frac{f_e}{2}, \frac{f_e}{2}\right] \qquad X_e(f) = \frac{1}{T_e}X(f)$$

1.2.2 Transformée de Fourier d'un signal à temps discret

D'après l'équation (2), on obtient l'expression suivante de la transformée de Fourier du signal échantillonné :

$$\begin{aligned} X_e(f) &= \int \left(\sum_{k=-\infty}^{+\infty} x(kT_e)\delta(t-kT_e) \right) e^{-\imath 2\pi f t}\, dt \\ &= \sum_{k=-\infty}^{+\infty} x(kT_e) \int \delta(t-kT_e) e^{-\imath 2\pi f t}\, dt \\ &= \sum_{k=-\infty}^{+\infty} x(kT_e) e^{-\imath 2\pi f k T_e} \end{aligned}$$

Sachant par définition du signal échantillonné que $x(kT_e) = x_k$, et en introduisant la fréquence normalisée $\tilde{f} = \frac{f}{f_e} = fT_e$, on peut réécrire l'expression précédente :

$$X_e(f) = \sum_{k=-\infty}^{+\infty} x_k e^{-\imath 2\pi k \tilde{f}}$$

Ceci nous conduit alors à définir pour les signaux à temps discret une transformée de Fourier qui ne dépende pas de la fréquence d'échantillonnage.

Définition 3 *Etant donné x_k un signal à temps discret, sa transformée de Fourier à temps discret (TFTD) est par définition :*

$$X^{(TFTD)}(\tilde{f}) = \sum_{k=-\infty}^{+\infty} x_k e^{-\imath 2\pi k \tilde{f}}$$

$\tilde{f}$ est alors une fréquence normalisée (sans unité). On remarquera que $X^{(TFTD)}(\tilde{f})$ est périodique de période 1.

Propriété 3 *Etant donnés un signal $x(t)$ à temps continu et $x_k = x(kT_e)$ le signal échantillonné correspondant, la transformée de Fourier du signal échantillonné est liée à sa TFTD par :*

$$X_e(f) = X^{(TFTD)}(\tilde{f} = f/f_e)$$

La fréquence normalisée $\tilde{f} = 1$ correspond à la fréquence réelle $f = f_e$. Si de plus la fréquence d'échantillonnage vérifie la condition de Shannon, alors :

$$X(f) = T_e X_e(f) = T_e X^{(TFTD)}(\tilde{f} = f/f_e)$$

D'après ce qui précède, étant donné un signal $x(t)$ à bande limitée $[-F_{\max}, F_{\max}]$, il est possible de calculer sa transformée de Fourier à l'aide d'une somme infinie sur ses échantillons, à condition de choisir correctement la fréquence d'échantillonnage. Nous supposerons dans la suite que cette condition sur la fréquence d'échantillonnage est réalisée.

1.3 La transformée de Fourier discrète (TFD)

Nous avons vu que le calcul de la transformée de Fourier d'un signal à temps continu et à bande limitée se ramène, après échantillonnage, au calcul d'une TFTD :

$$X^{(TFTD)}(\tilde{f}) = \sum_{k=-\infty}^{+\infty} x_k e^{-\imath 2\pi k\tilde{f}}$$

Cette transformée est une fonction continue de $\tilde{f}$. Une possibilité de traitement numérique consiste à évaluer $X^{(TFTD)}(\tilde{f})$ sur un ensemble discret de points. C'est ce qui est effectué par la transformée de Fourier discrète (TFD). On veillera à ne ne pas confondre transformée de Fourier d'un signal à temps discret (TFTD) et transformée de Fourier discrète (TFD).

1.3.1 Définition

Soit $x(t)$ un signal à bande limitée $[-F_{\max}, F_{\max}]$ et de durée infinie. Nous supposerons qu'il a été échantillonné à une fréquence d'échantillonnage $fe \geq 2F_{\max}$, donnant ainsi le signal à temps discret $x_k = x(kT_e)$ où $k \in \mathbb{Z}$. Le calcul numérique de la transformée de Fourier à temps discret de x_k ne peut être effectué que sur un nombre fini d'observations. Nous considérerons par conséquent que nous ne disposons que de N échantillons, $x_k; \quad k \in \{0, 1, \cdots, N-1\}$. La TFTD de ce signal (qui est maintenant échantillonné et tronqué) s'écrit alors :

$$X^{(TFTD)}(\tilde{f}) = \sum_{k=0}^{N-1} x_k e^{-\imath 2\pi k\tilde{f}}$$

Nous n'avons fait que remplacer une somme en théorie infinie par une somme finie et $X^{(TFTD)}(\tilde{f})$ est toujours une fonction continue de $\tilde{f}$. La TFD calcule les valeurs de $X^{(TFTD)}(\tilde{f})$ en $\tilde{f} = \frac{n}{N}; \quad n \in \{0, 1, \cdots, N-1\}$.

Définition 4 *Etant donnée une suite finie d'échantillons $x_k; \quad k \in \{0, 1, \cdots, N-1\}$, on appelle transformée de Fourier discrète (TFD) l'opération qui leur associe les valeurs $X_n; \quad n \in \{0, 1, \cdots, N-1\}$ définies par :*

$$\forall n \in \{0, 1, \cdots, N-1\} \qquad X_n = \sum_{k=0}^{N-1} x_k e^{-\imath \frac{2\pi kn}{N}}$$

Si l'on note $w = e^{-\imath \frac{2\pi}{N}}$, on a aussi :

$$\forall n \in \{0, 1, \cdots, N-1\} \qquad X_n = \sum_{k=0}^{N-1} x_k w^{kn}$$

et par conséquent en notant :

$$\mathbf{W} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & w & w^2 & \cdots & w^{N-1} \\ 1 & w^2 & w^4 & \cdots & w^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & w^{N-1} & w^{2(N-1)} & \cdots & w^{(N-1)^2} \end{pmatrix}$$

la TFD peut se définir comme l'application linéaire suivante :

$$\begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{pmatrix} = \mathbf{W} \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix}$$

Nous disposons ainsi d'un outil permettant d'évaluer la transformée de Fourier sur un ensemble discret de points. Deux conditions doivent être vérifiées pour que s'applique la méthode :

- la fréquence d'échantillonnage doit être convenablement choisie,
- la fenêtre d'observation doit être suffisamment grande pour que les effets indésirables introduits par la troncature du signal puissent être négligés.

1.3.2 La transformée de Fourier rapide (FFT)

Afin d'utiliser ce que nous venons de présenter, il reste à implanter les calculs, qui se doivent d'être aussi rapides que possible. C'est ce que réalise la FFT.

Définition 5 *La transformée de Fourier rapide (ou FFT : Fast Fourier Transform) est un algorithme rapide qui permet de calculer la TFD lorsque le nombre de données est une puissance de 2.*

Dans la mesure du possible, les calculs de TFD se font par conséquent sur un nombre d'échantillons qui soit une puissance de deux.

1.4 Technique du bourrage de zéros

La méthode précédente est limitée par le nombre fini de points utilisés pour le calcul : nous n'obtenons ainsi les valeurs de la TFTD que sur une grille de N points. En effet, la TFD calcule en chacun des points $\tilde{f} = \frac{n}{N}; \quad n \in \{0, 1, \cdots, N-1\}$ les valeurs de la fonction :

$$X^{(TFTD)}(\tilde{f}) = \sum_{k=0}^{N-1} x_k e^{-\imath 2\pi k \tilde{f}}$$

ce qui permet d'obtenir les valeurs $X^{(TFTD)}(\tilde{f} = \frac{n}{N}) = X_n$ pour $n \in \{0, \ldots, N-1\}$.

La technique de *bourrage de zéros* ou *zero-padding* est un artifice qui permet d'évaluer $X^{(TFTD)}(\tilde{f})$ sur une grille plus fine de fréquences : elle consiste simplement à compléter les données $x_k; \quad k \in \{0, 1, \cdots, N-1\}$ par des zéros : $x_k = 0; \quad k \in \{N, N+1, \cdots, M-1\}$. Nous obtenons alors par TFD les valeurs de $X^{(TFTD)}(\tilde{f})$ en M points au lieu de N. On choisira de préférence pour M une puissance de deux compte tenu de la possibilité d'utiliser l'algorithme FFT.

2 Travail à effectuer pendant la séance de TP

On utilisera le langage Python et on chargera le module PyLab à l'aide de :

```
from pylab import * # importe le module pylab
ion() # mode interactif pour les tracés
```

2.1 Etude sur un signal artificiel connu

Pour comprendre les notions de TFTD, TFD, FFT et leur lien avec la TFTC, on commence par étudier le cas d'un simple signal sinusoïdal connu et échantillonné.

2.1.1 Mise en œuvre de la TFD

Dans cette partie, on fixe les paramètres suivants :

$N = 1024$	nombre d'échantillons/ de points de la TFD
$f_e = 1/T_e = 8\text{kHz}$	fréquence d'échantillonnage
$f_0 = 2124\text{Hz}$	fréquence du signal étudié

1– *On échantillonne un signal sinusoïdal $x(t) = \cos(2\pi f_0 t)$ de fréquence f_0 à une fréquence d'échantillonnage f_e. Générer une variable vectorielle x qui contient les N échantillons $x_k = x(kT_e)$ pour $k = 0, \ldots, N-1$. Faire un tracé pour visualiser temporellement le signal. L'allure en «dents de scie» est-elle normale ?*

```
N = 1024
fe, Te = 8000, 1/8000
f0 = 2124
t = arange(N)*Te # Attention Python 2.7: arange(., dtype='float')
x = cos(2*pi*f0*t)
plot(t, x)
```

2– *Calculer la TFD du signal x, puis tracer le module (ou bien le module au carré) de cette TFD.*

```
X = fft(x)
plot(abs(X)) # La graduation de l'axe de abscisses n'est pas significative.
```

3– *Comparer le temps de calcul de la TFD sur un vecteur quelconque avec 2^p et $2^p - 1$ échantillons (choisir p afin d'être illustratif. Attention toutefois au risque de bloquer le PC par surcharge de calcul : sauvegardez vos données et soyez prêts à tuer le processus si nécessaire !). Expliquer la différence observée.*

```
x1 = randn(2**15)
x2 = x1[1::]
timeit fft(x1)
timeit fft(x2)
```

4– *Prévoir sur le tracé de la TFD une graduation correcte de l'axe des fréquences. Le tracé se fera ici pour des fréquences variant de 0 à f_e.*
Vérifier que pour le signal $x(t)$ sinusoïdal pur on obtient bien une raie à la fréquence f_0. Expliquer la deuxième raie.

```
freq = arange(N)/N*fe  # Attention Python 2.7: arange(., dtype='float')
plot(freq, abs(X))
```

5– *Faire un tracé du spectre sur la plage de fréquences allant de $-f_e/2$ à $f_e/2$ (commandes utiles : `fftshift, fftfreq`).*

6– *Mettre les éléments des questions 1 et 4 (ou 1 et 5 si l'on préfère une représentation bilatérale de la TF) dans un même script qui permettra de faire varier les paramètres.*

2.1.2 Phénomène de repliement

7– *Faire varier la fréquence f_0 du signal sinusoïdal étudié jusqu'à dépasser les conditions de Shannon. Observer le phénomène de repliement (on gardera constants les paramètres N et $f_e = 1/T_e$).*

2.1.3 Nombre de points de la TFD

Dans cette partie, on étudie l'influence de N que l'on fera varier tandis que l'on fixe :

$$\begin{aligned} f_e = 1/T_e = 8\text{kHz} \quad & \text{fréquence d'échantillonnage} \\ f_0 = 2124\text{Hz} \quad & \text{fréquence du signal étudié} \end{aligned}$$

8– *Diminuer N en partant de $N = 2^{10}$ jusqu'à $N = 2^3$ et observer. Que vaut la durée d'observation du signal $x(t)$ en fonction de N ? Est-il normal que, pour N faible, on n'observe plus un pic étroit pour la TFTD ? Expliquer ce qui se passe... ou poursuivre pour comprendre les deux phénomènes en jeu !*

2.1.4 Bourrage de zéros

Les tracés précédents n'ont pas fait apparaître des courbes du type sinus-cardinal qui correspondent pourtant à la TF d'un signal sinusoïdal tronqué. Pour obtenir un tracé correct de la TFTD, il convient de la calculer sur un nombre suffisant de points. On prendra ici les paramètres :

$$\begin{aligned} f_e = 1/T_e = 8\text{kHz} \quad & \text{fréquence d'échantillonnage} \\ f_0 = 2124\text{Hz} \quad & \text{fréquence du signal étudié} \\ M = 1024 \quad & \text{nombre de points où la TFTD est calculée} \\ 8 \leq N \leq 1024 \quad & \text{nombre d'échantillons (à faire varier)} \end{aligned}$$

9– *Utiliser la technique de bourrage de zéros (zero-padding) pour calculer en M points la TFD du signal tronqué constitué des N échantillons $x_k = x(kT_e)$ pour $k = 0, \ldots, N-1$.*

```
X_M = fft(x,M)
freq_M = arange(M)/M*fe # Attention Python 2.7: arange(., dtype='float')
plot(freq_M, abs(X_M))
```

10– *Inclure les éléments précédents dans votre script, puis faire varier N. Observer et expliquer le problème de la résolution spectrale.*

11– *Expliquer les deux phénomènes en cause dans la question 8.*

2.1.5 Normalisation

Jusqu'ici, aucun attention n'a été portée à la hauteur des pics des TFTD tracée.

12– *Observer que la hauteur des pics varie selon la durée D de la fenêtre d'observation. Ceci provient du fait que nous réalisons l'analyse d'un signal de puissance finie (une sinusoïde) en considérant son acquisition sur une fenêtre temporelle de durée variable.*

13– *Proposer une normalisation, la mettre en œuvre et vérifier la cohérence des résultats (on rappelle que la TFTC d'un cosinus comprend deux raies d'amplitude 1/2).*

2.2 Etude d'un signal sonore : principe sommaire de compression

14– *A l'aide de la commande* `loadtxt`*, charger le fichier situé à l'adresse :*
`http://www-public.it-sudparis.eu/~castella/SI12/son/`
Ce signal correspond à un chant d'oiseau échantillonné à $F_e = 22\text{KHz}$.

15– *Tracer ce signal en fonction du temps. Quel est le nombre N_e d'échantillons disponibles ? la durée du signal ? Compte tenu de la fréquence d'échantillonnage, quelle est la périodicité du spectre du signal échantillonné ?*

16– *Calculer la TFD de ce signal et représenter le spectre obtenu avec une graduation correcte des fréquences.*

17– *Calculer les indices des 1000 composantes de la TFD de plus grands modules, en employant la commande* `argsort`*.*

18– *Mettre à zéro les composantes de la TFD n'appartenant pas à cet ensemble de valeurs. Observer le spectre obtenu.*

19– *On reconstitue le signal à partir de la TFD ainsi tronquée (utiliser* `ifft`*). Calculer l'énergie de l'erreur de troncature et la comparer à l'énergie du signal original. Cette erreur vous paraît-elle significative ?*

20– *Comment pourrait-on facilement gagner encore un facteur 2 dans la compression de données sommaire qui vient d'être mise en œuvre ?*