Traitement Numérique du Signal

TP1

Signaux et Spectres

Première année Département Sciences du Numérique

2019 – 2020

# 1 Introduction

## 1.1 Objectifs

Les objectifs de ce premier TP sont les suivants :

1. Etre capable de générer un signal numérique simple et de le tracer avec une échelle temporelle en secondes.
2. Etre capable d'estimer la représentation fréquentielle d'un signal en numérique (transformée de Fourier discrète et densité spectrale de puissance), de la tracer (avec une échelle fréquentielle en Hz et/ou avec une échelle en fréquences normalisées) et de l'utiliser pour extraire de l'information sur le signal étudié.

Les exercices des deux premiers travaux dirigés de traitement du signal ont permis d'étudier les effets de l'échantillonnage d'un signal et de déterminer sa représentation fréquentielle en considérant différentes modélisations, dans le cas d'un signal simple qu'est le cosinus. Nous allons générer dans ce TP un cosinus numérique, étudier en pratique les effets de son échantillonnage et identifier les modifications apportées à sa représentation fréquentielle théorique du fait que nous travaillons en numérique.
Ces TPs viennent préparer les projets dans lesquels vous aurez à réaliser des analyses utilisant les outils vus ici sur des signaux plus complexes.

## 1.2 Travail à rendre

Le travail sera réalisé de manière individuelle et, devront être rendus le 20 décembre 2019 au plus tard :

1. Une copie manuscrite, ou un fichier pdf, comprenant les calculs demandés et les réponses aux questions posées dans le texte de TP.
2. Les codes réalisés permettant de visualiser les tracés demandés dans le texte de TP.

L'ensemble du travail devra être rendu à l'intervenant de TP. Vous trouverez dans le tableau suivant les adresses mail des différents intervenants.

| | |
|---|---|
| Groupe A : nathalie.thomas@enseeiht.fr | Groupe I : bahaeddine.belmekki@inp-toulouse.fr |
| Groupe B : Mohamad.Hourani@irit.fr | Groupe J : Etienne.Monier@enseeiht.fr |
| Groupe C : Etienne.Monier@enseeiht.fr | Groupe K : charly.poulliat@enseeiht.fr |
| Groupe D : charly.poulliat@enseeiht.fr | Groupe L : bahaeddine.belmekki@inp-toulouse.fr |
| Groupe E : Mohamad.Hourani@irit.fr | Groupe M : Etienne.Monier@enseeiht.fr |
| Groupe F : raoul.prevost@tesa.prd.fr | Groupe N : nathalie.thomas@enseeiht.fr |
| Groupe G : mathieu.dervin@enseeiht.fr | |
| Groupe H : raoul.prevost@tesa.prd.fr | |

### 1.3 Consignes à respecter

1. Tous vos tracés doivent comporter des labels sur les axes (utiliser *xlabel.m* et *ylabel.m* sous matlab) et un titre (utiliser *title.m* sous matlab).
2. Si plusieurs courbes sont tracées sur la même figure, celle-ci devra comporter une légende (utiliser *legend.m* sous matlab).
3. Vos codes doivent être commentés de manière suffisante et claire. Un nouvel utilisateur doit pouvoir comprendre ce que vous avez souhaité implanter.

Remarque : il vous est demandé de réaliser plusieurs tracés. Ces tracés correspondent à ce qu'il est important de visualiser et de savoir analyser. Ces tracés, ainsi que les explications associées, seront évalués pour donner la note du TP, mais, bien entendu, rien ne vous empêche de visualiser d'autres signaux/spectres si cela vous aide dans vos analyses.

## 2 Effet de l'échantillonnage : Génération et tracé d'un cosinus numérique

**Le signal numérique, $x$, sera représenté sous Matlab par un tableau contenant un nombre fini, $N$, de valeurs représentant des échantillons de signal prélevés toutes les $T_e$ secondes (échantillonnage uniforme) : $[x(0)\ x(1)...\ x(N-1)]$, le $k^{\text{ième}}$ élément $x(k)$ représentant en réalité $x(kT_e)$.**

1. Générer 90 échantillons d'un cosinus (fonction *cos.m* sous matlab), d'amplitude 1 (V), de fréquence $f_0 = 1100$ Hz et échantillonné à $F_e = 10000$ Hz.
2. En utilisant la fonction *plot.m* de Matlab, tracer le cosinus généré précédemment avec une échelle temporelle en secondes (figure 1 de votre programme). On doit pouvoir retrouver, à partir du tracé, la fréquence et l'amplitude du cosinus.
3. Générer 90 échantillons d'un cosinus (fonction *cos.m* sous matlab), d'amplitude 1 (V), de fréquence $f_0 = 1100$ Hz et échantillonné à $F_e = 1000$ Hz.
4. En utilisant la fonction *plot.m* de Matlab, tracer le cosinus généré précédemment avec une échelle temporelle en secondes (figure 2 de votre programme). La fréquence mesurée sur le cosinus tracé n'est pas $f_0 = 1100$ Hz. Expliquez pourquoi et d'où vient la valeur de la fréquence observée.

## 3 Transformée de Fourier discrète (TFD) : Génération et tracé de la TFD d'un cosinus numérique

**La transformée de Fourier numérique du signal $x$ devra être estimée à partir du tableau de points représentant le signal numérique.**

1. Etude théorique
   (a) Rappeler l'expression de la transformée de Fourier, $X(f)$, d'un cosinus $x(t)$ d'amplitude A et de fréquence $f_0$. Tracer $|X(f)|$.
   (b) Calculer $X_D(f)$, transformée de Fourier d'un cosinus d'amplitude A et de fréquence $f_0$, échantillonné à $T_e$ (échantillonnage uniforme) et tronqué sur une longueur $[0, ..., (N-1)T_e]$ (tableau de $N$ valeurs représentant le cosinus en numérique).
   (c) Tracer le module de $X_D(f)$ entre $f = -F_e$ et $f = F_e$ et le comparer à celui de $X(f)$. Expliquer les différences constatées.
2. Implantation
   (a) En utilisant la fonction *fft.m* de Matlab (fft pour fast Fourier transform : algorithme de calcul rapide de la TFD), estimer la transformée de Fourier d'un cosinus de fréquence $f_0 = 1100$ Hz, d'amplitude $A = 1$ V et échantillonné à $F_e = 10000$ Hz. Tracer le module (fonction *abs.m* de Matlab) de la TFD estimée, en échelle log grâce à la fonction *semilogy.m* de Matlab (figure 3 de votre programme). L'échelle fréquentielle devra être en Hz. Expliquer le résultat obtenu et en quoi il est identique ou différent par rapport à celui établi de manière théorique précédemment (module de $X_D(f)$ calculé et tracé précédemment).
   (b) En utilisant la fonction *fft.m* de Matlab, estimer la transformée de Fourier d'un cosinus de fréquence $f_0 = 1100$ Hz, d'amplitude $A = 1$ V et échantillonné à $F_e = 1000$ Hz. Tracer le module de la TFD estimée, en échelle log grâce à la fonction *semilogy.m* de Matlab (figure 4 de votre programme). L'échelle fréquentielle devra être en Hz. Ce résultat vous parait-il cohérent avec le tracé du cosinus obtenu précédemment (dernière question de la partie 2) ? Expliquer votre réponse.

**La transformée de Fourier numérique ou discrète (TFD) d'un signal est, elle même, représentée par un tableau, contenant un nombre fini de valeurs, représentant des échantillons de la TFD du signal calculés avec un certain pas. On obtient en effet sous Matlab, en utilisant la fonction *fft.m* sur le signal $x$, un tableau : $[X(0)\ X(1) ...\ X(N-1)]$, le $n^{\text{ième}}$ élément $X(n)$ représentant en réalité $X_D\left(n\Delta f\right)$, si $\Delta f$ désigne le pas de calcul de la TFD. En considèrant que l'on calcule $N$ points pour la TFD entre $0$ et $F_e$, le $n^{\text{ième}}$ élément $X(n)$ du tableau représentant la TFD du signal $x$ est alors $X_D\left(n\frac{F_e}{N}\right)$.**

1. Qu'est-ce qui peut justifier que l'on calcule la transformée de Fourier numérique entre 0 et $F_e$ ?
2. Donner l'expression générale de $X_D(n\frac{F_e}{N})$ en fonction de $x(kT_e)$.
3. Soit le signal numérique $y$ défini par les éléments suivants :

$$y(kT_e) = \begin{cases} x(kT_e) & \text{pour } k = 0, ..., N-1 \\ 0 & \text{pour } k = N, ..., MN-1. \end{cases} \tag{1}$$

   (a) Déterminer l'expression de $Y_D\left(n\frac{F_e}{MN}\right)$ ($n^{\text{ième}}$ élément du tableau représentant la TFD du signal $y$ calculée entre 0 et $F_e$) en fonction de $x(kT_e)$.
   (b) Comparer l'expression de $Y_D\left(n\frac{F_e}{MN}\right)$ en fonction de $x(kT_e)$ à l'expression de $X_D\left(n\frac{F_e}{N}\right)$ en fonction de $x(kT_e)$. En déduire la différence entre les TFDs associées au signal $x$ et au signal $y$ et l'intérêt de définir ce signal $y$ composé du signal $x$ prolongé par des zéros (Zero Padding) avant de calculer la TFD ?
4. En utilisant la fonction *fft.m* de Matlab et la technique du Zero Padding, calculer la transformée de Fourier d'un cosinus de fréquence $f_0 = 1100$, échantillonné à $F_e = 10000$ Hz, sur un nombre de points, $N'$, supérieur au nombre de points de signal : $X = fft(x, N')$. Tracer le module de cette transformée

de Fourier en échelle log (figure 5 de votre programme). Attention afin d'utiliser l'algorithme de calcul rapide de la TFD (FFT) on devra utiliser un nombre de points $N'$ égal à une puissance de 2. On pourra prendre ici un nombre de point $N'$ égal à la puissance de 2 immédiatement supérieure à la longueur du signal ($N' = 2^\wedge$ nextpow2(length(x))). Le tracé obtenu est-il conforme au résultat théorique établi précédemment pour le module de $X_D(f)$ ?

5. Utiliser plusieurs valeurs de $N'$ afin de comparer les résultats obtenus et d'en déduire l'intérêt de la technique dite du Zero Padding. La figure 6 de votre programme devra comporter les tracés superposés du module de plusieurs versions de la TFD, obtenues pour plusieurs valeurs de $N'$ et une légende devra permettre de les identifier (fonction *legend.m* de matlab). L'échelle fréquentielle devra être en Hz. Le tracé superposé de plusieurs courbes sur une même figure se fait grâce à l'instruction *hold on* placée après le premier tracé. En déduire l'intérêt de la technique dite du Zero Padding.

6. Retrouver à partir du tracé de vos TFDs (en expliquant comment), l'amplitude et la fréquence du cosinus généré. L'utilisation de la technique du Zero Padding a t-elle une influence sur la restitution de ces paramètres ? Si oui en quoi ?

## 4 Densité spectrale de puissance (DSP) numérique : Estimation et tracé de la DSP d'un cosinus en numérique

Un autre outil permet de visualiser la représentation fréquentielle d'un signal, notament pour les signaux alétoires. Il s'agit de la Densité Spectrale de Puissance (DSP). Elle peut être estimée en numérique en utilisant deux estimateurs de base que sont le périodogramme et le corrélogramme. On considèrera le signal suivant :

$$x(k) = \cos\left(2\pi f_0 k T_e + \phi\right)$$

où $f_0 = 1100$ Hz, $\phi$ est une phase aléatoire de loi uniforme sur $[0, 2\pi]$ et $F_e = \frac{1}{T_e} = 10000$ Hz.

1. Générer une réalisation du signal $x$. La valeur de $\phi$ pourra être obtenue sous Matlab en utilisant *rand*2*pi*).
2. Estimer, puis tracer, la DSP du signal $x$, en implantant un périodogramme ou un corrélogramme (figure 7 de votre programme). Retrouve t-on, à partir de la DSP, les résultats obtenus précédemment avec la transformée de Fourier (ici fréquence et amplitude du cosinus généré) ? Si oui comment ?
3. Estimer, puis tracer sur une même figure (figure 8 de votre programme), la DSP du signal $x$, en implantant un périodogramme avec un fenêtrage :
   (a) qui utilise une fenêtre de Hamming (fonction *hamming.m* de Matlab pour la générer).
   (b) qui utilise une fenêtre de Blackman (fonction *blackman.m* de Matlab pour la générer).
4. Estimer, puis tracer (figure 9 de votre programme), la DSP du signal $x$ en utilisant la fonction *pwelch.m* de Matlab de la manière suivante *DSPx=pwelch(x,”,”,”,’twosided’) ;* Si $x$ représente votre signal, *DSPx* sera alors sa DSP estimée sur une période $F_e$ en utilisant un périodogramme de Welch.
5. Expliquer les différences observées entre les 4 DSPs précédemment tracées. On pourra éventuellement les tracer sur la même figure (figure 10) pour les comparer plus simplement. En déduire les avantages/inconvénients des différentes méthodes utilisées pour obtenir la DSP du signal traité.